

Introduction

This reference manual targets application developers.

It provides complete information on how to use the SPC1169 microcontroller, memories and peripherals. For device mechanical and electrical characteristics please refer to the SPC1169 datasheet document.

Contents

Revision history	39
1 Memory and bus architecture	41
1.1 System architecture	41
1.2 Clock tree	43
1.3 Memory map	44
1.3.1 Embedded Flash Memory	46
1.3.2 Embedded SRAM	46
1.3.3 Boot ROM	46
1.4 Boot configuration	49
2 Power	50
2.1 Power supply description	50
2.2 Power mode management	51
2.3 Power-up sequence	52
2.3.1 VCAP12 LDO	52
2.4 Over-voltage/under-voltage detectors	53
2.5 System wake up	55
2.6 Registers	56
2.6.1 MCU Power register map	56
2.6.2 MCU Power registers	57
3 Clocks and reset	62
3.1 Reset architecture	62
3.2 Pin reset	62
3.3 POR reset	63
3.4 Clock architecture	63
3.5 High frequency RC oscillator	64
3.6 On-chip crystal oscillator and external clock	64
3.7 PLL clock	66
3.7.1 Reference input and divider	67
3.7.2 Output divider	67
3.7.3 Feedback divider	67
3.7.4 PLL unlock detection	67
3.8 Clock safety	68
3.9 Registers	70
3.9.1 Clock register map	70
3.9.2 Clock registers	71

4	General-purpose I/O (GPIO)	86
4.1	Overview	86
4.2	Pin multiplexing (PINMUX)	87
4.2.1	Deglitch control	88
4.2.2	I/O alternate functions	89
4.2.3	Loopback path for communication	90
4.2.4	Debug interface	90
4.3	GPIO control module (GPIO)	92
4.3.1	Direction and level control	92
4.3.2	Edge-triggered interrupt	93
4.3.3	Level-triggered interrupt	94
4.4	Registers	97
4.4.1	PINMUX register map	97
4.4.2	PINMUX registers	99
4.4.3	GPIO register map	142
4.4.4	GPIO registers	143
5	Interrupts and events	162
5.1	Nested vectored interrupt controller (NVIC)	162
5.1.1	Features	162
5.1.2	Interrupt and exception vectors	162
5.1.3	Non-maskable interrupt	164
5.1.4	NVIC interrupt configuration	165
5.2	SysTick Timer	165
6	General-purpose timers	166
6.1	Timer overview	166
6.2	Functional description	166
6.2.1	General timer mode	167
6.2.2	Gated timer mode	167
6.2.3	Event counter mode	167
6.2.4	Event capture mode	168
6.3	Initialization sequence	168
6.4	Registers	169
6.4.1	TIMER register map	169
6.4.2	TIMER registers	170
7	Watchdog timers	175
7.1	Watchdog overview	175
7.2	Functional description	175
7.3	Initialization sequence	176

7.4	Registers.....	177
7.4.1	WDT register map	177
7.4.2	WDT registers.....	178
8	PWM	183
8.1	PWM overview.....	183
8.2	Time-Base (TB) sub-module	185
8.3	Counter Compare (CC) sub-module	188
8.4	Action-Qualifier (AQ) sub-module	189
8.5	Dead-Band (DB) sub-module	191
8.6	Trip-Zone (TZ) sub-module	194
8.7	Digital Compare (DC) sub-module	197
8.8	Event-Trigger (ET) sub-module	200
8.9	Global reload.....	200
8.10	Simultaneous write	201
8.11	Registers.....	202
8.11.1	PWM register map.....	202
8.11.2	PWM registers	204
8.11.3	PWMCFG register map	282
8.11.4	PWMCFG registers.....	283
9	ECAP.....	295
9.1	Introduction	295
9.2	Description.....	295
9.3	Capture and APWM operating mode.....	296
9.4	Capture mode description	296
9.4.1	Event prescaler.....	297
9.4.2	Edge polarity select and qualifier.....	298
9.4.3	Dual pin capture.....	298
9.4.4	Continuous/one-shot control.....	298
9.4.5	32-bit counter and phase control.....	299
9.4.6	CAP0-CAP3 registers.....	299
9.4.7	Interrupt control	300
9.5	APWM mode operation	301
9.6	Application of the ECAP module	303
9.6.1	Absolute time-stamp for rising edge.....	303
9.6.2	Absolute time-stamp operation for duty and period.....	304
9.6.3	Time difference (Delta) operation with counter reset.....	305
9.6.4	Time difference operation for duty and period	306
9.6.5	Dual pin capture with counter reset	307

9.7	Registers.....	308
9.7.1	ECAP register map	308
9.7.2	ECAP registers	309
10	13-bit ADC	324
10.1	13-bit ADC overview	324
10.2	13-bit ADC architecture	325
10.3	Principle of channel based operation	326
10.3.1	SOC trigger source	326
10.3.2	Input selection	327
10.3.3	Sample and convert window	328
10.3.4	Result average	328
10.3.5	Gain and offset correction.....	329
10.3.6	Timing diagram.....	329
10.4	ADC conversion priority	330
10.5	EOC and interrupt operation	334
10.6	Post-processing units	334
10.6.1	Reference comparison.....	334
10.6.2	Window comparison and zero-cross detection.....	335
10.6.3	Sample delay measurement.....	335
10.7	Calculate the expected code.....	336
10.8	ADC safety feature	336
10.8.1	Input floating detection.....	336
10.8.2	Input short detection.....	338
10.9	Registers.....	339
10.9.1	ADC register map.....	339
10.9.2	ADC registers	341
11	13-bit Monitoring ADC.....	382
11.1	Overview	382
11.2	Architecture and operation	382
11.3	Registers.....	384
12	SPGA	385
12.1	SPGA overview	385
12.2	SPGA architecture	385
12.3	SPGA channel selection.....	385
12.4	SPGA gain selection.....	386
12.5	Registers.....	386
13	DPGA.....	387

13.1	DPGA overview	387
13.2	DPGA gain and input range	387
13.3	DPGA application	387
13.4	DPGA offset calibration.....	389
13.5	Registers.....	390
13.5.1	PGA register map.....	390
13.5.2	PGA registers	391
14	Analog Comparator.....	394
14.1	Comparator overview	394
14.2	Comparator architecture	394
14.3	Comparator MUX selection.....	394
14.4	Digital filter	395
14.5	Registers.....	396
14.5.1	Comparator register map	396
14.5.2	Comparator registers.....	397
15	Phase Comparator	417
15.1	Overview	417
15.2	Architecture	417
15.3	Post processing	418
15.4	Registers.....	421
15.4.1	Phase Comparator register map.....	421
15.4.2	Phase Comparator registers	422
16	Buffered 10-bit DAC.....	432
16.1	Buffered DAC overview.....	432
16.2	Buffered DAC architecture	432
16.3	Ramp generator	433
16.4	Registers.....	435
17	D2S buffer	436
17.1	Overview	436
17.2	Function description	436
18	Temperature sensor.....	437
18.1	T-sensor overview	437
18.2	Transfer function	437
18.3	Registers.....	437
19	UART	438
19.1	UART overview.....	438

19.2	UART features	439
19.3	UART signal descriptions.....	440
19.3.1	UART device connection.....	440
19.3.2	LIN device connection	440
19.4	UART operation.....	441
19.4.1	Reset.....	442
19.4.2	FIFO operation	442
19.4.3	Timeout and removing trailing bytes	443
19.4.4	Programmable baud-rate generator	443
19.4.5	Auto-baud-rate detection	444
19.4.6	Serial infrared asynchronous interface	445
19.5	LIN operation	447
19.5.1	LIN protocol	447
19.5.2	Frame format.....	447
19.5.3	Protected ID.....	449
19.5.4	Checksum calculation	449
19.6	Interrupt.....	449
19.6.1	Receive interrupt.....	450
19.6.2	Receive timeout interrupt	450
19.6.3	Receive auto baud counter lock interrupt.....	450
19.6.4	Receive FIFO interrupt.....	450
19.6.5	Transmit FIFO interrupt.....	450
19.6.6	LIN error interrupt	450
19.6.7	DMA interrupt	451
19.7	Registers.....	452
19.7.1	UART register map.....	452
19.7.2	UART registers	453
20	SPI	475
20.1	SPI overview.....	475
20.2	SPI features	476
20.3	SPI signal description	476
20.4	SPI function description	476
20.4.1	Baud rate generation.....	477
20.4.2	FIFO operation	477
20.4.3	SPI interrupts	479
20.5	SPI (Motorola Serial Peripheral Interface)	480
20.5.1	Device connection	480
20.5.2	Protocol description	481
20.6	SSP (TI Synchronous Serial Protocol)	487

20.6.1	Device connection	487
20.6.2	Protocol description	487
20.7	PSP (Programmable Serial Protocol)	488
20.7.1	Device connection	488
20.7.2	Protocol description	489
20.8	Registers.....	491
20.8.1	SPI register map.....	491
20.8.2	SPI registers	492
21	I2C.....	507
21.1	I2C overview.....	507
21.2	I2C features	507
21.3	I2C signal description	507
21.4	I2C operation	508
21.4.1	START and STOP generation	509
21.4.2	Combined formats.....	509
21.5	I2C protocols	509
21.5.1	START and STOP conditions.....	509
21.5.2	Address formats	510
21.5.3	Transmitting and receiving protocol	510
21.6	Registers.....	512
21.6.1	I2C register map	512
21.6.2	I2C registers.....	513
22	CAN	538
22.1	Features	538
22.2	CAN I/O signal	540
22.2.1	CAN device connection.....	540
22.3	CAN frame format.....	541
22.3.1	ISO classical CAN.....	541
22.3.2	ISO FDCAN	541
22.3.3	CAN frame based on CAN2.0B.....	541
22.3.4	NONISO FDCAN	542
22.3.5	Arbitration phase and Data phase.....	542
22.4	CAN error state	543
22.4.1	Error record	543
22.4.2	Error active	543
22.4.3	Error passive	543
22.4.4	Bus-Off recovery.....	543
22.5	CAN operation.....	544

22.5.1	Bit rate configuration	544
22.5.2	Monitor mode	545
22.5.3	Restricted mode	545
22.5.4	Test mode	545
22.6	Mailbox control	547
22.6.1	Mailbox information	547
22.6.2	Message transmission without remote frame auto response	548
22.6.3	Message transmission with remote frame auto response	549
22.6.4	Message reception without request data from remote node	549
22.6.5	Message reception with request data from remote node	550
22.7	Registers	551
22.7.1	CAN register map	551
22.7.2	CAN registers	552
22.7.3	CANMBOX register map	573
22.7.4	CANMBOX registers	573
23	CRC	578
23.1	CRC overview	578
23.2	Main features	578
23.3	CRC operation flow	579
23.4	Registers	580
23.4.1	CRC register map	580
23.4.2	CRC registers	581
24	Flash memory	588
24.1	Features	588
24.2	Flash module organization	588
24.3	Read operation	590
24.3.1	Read through XIP interface	590
24.3.2	Read through controller interface	590
24.4	Program and erase operation	591
24.4.1	Enable flash controller interface	591
24.4.2	Flash memory programming	591
24.4.3	Flash memory erase	593
24.5	Protections	595
24.5.1	Read-out protection	595
24.5.2	Write protection	595
24.6	Setup Flash timing	595
24.7	Configuration words description	596
24.8	Registers	597

24.8.1	Flash register map	597
24.8.2	Flash registers	598
25	System control.....	607
25.1	Overview	607
25.2	Unique device ID register (64 bits).....	607
25.3	System monitor control registers	607
25.4	GPIO wake-up control register.....	607
25.5	Memory ECC error address registers	607
25.6	NMI event control registers	607
25.7	Reset event control registers	608
25.8	System information register	608
25.9	Boot information register	608
25.10	Registers.....	609
25.10.1	System register map	609
25.10.2	System registers	611
26	Direct memory access controller (DMA)	666
26.1	DMA Introduction	666
26.2	DMA features	666
26.3	DMA operation	668
26.3.1	Basic definitions.....	668
26.3.2	Hardware handshaking interface	668
26.3.3	Memory-to-memory transfer operation	669
26.3.4	Memory-to-peripheral and Peripheral-to-memory transfer operation	670
26.3.5	Peripheral-to-peripheral transfer operation	672
26.4	AHB Transfer Error Handling.....	673
26.5	Interrupt handling.....	673
26.5.1	Risk mitigation for interrupt loss and blocking.....	674
26.6	Registers.....	676
26.6.1	DMACH register map.....	676
26.6.2	DMACH registers	677
26.6.3	DMAC register map	685
26.6.4	DMAC registers.....	687

List of Figures

Figure 1-1: Block diagram of SPC1169	41
Figure 1-2: SPC1169 clock tree	43
Figure 1-3: SPC1169 memory map	44
Figure 1-4: Boot ROM memory map.....	47
Figure 2-1: Power supply overview.....	50
Figure 2-2: Power mode transition	51
Figure 2-3: Power up sequence	52
Figure 2-4: VCAP12 LDO.....	52
Figure 2-5: 3.3V domain brown-out detector diagram	53
Figure 2-6: 1.2V domain brown-out detector diagram	53
Figure 2-7: DVDD33 thresholds	54
Figure 2-8: VCAP12 thresholds	55
Figure 3-1: Active-low POR reset signal diagram.....	62
Figure 3-2: Active-low POR reset signal timing diagram	63
Figure 3-3: Clock overview	64
Figure 3-4: On-chip crystal oscillator	65
Figure 3-5: External clock.....	66
Figure 3-6: PLL clock overview	66
Figure 3-7: PLL unlock detection timing diagram example ($F_{REF} > F_{DIV}$)	68
Figure 3-8: Clock cross-detection.....	69
Figure 4-1: Block diagram of I/O control	87
Figure 4-2: PINMUX block diagram.....	88
Figure 4-3: Input deglitching.....	88
Figure 4-4: Select source of COMP_MONx	90
Figure 4-5: GPIO block diagram	92
Figure 4-6: GPIO edge detect.....	93
Figure 4-7: GPIO edge-triggered interrupt	94
Figure 4-8: GPIO level detect	95
Figure 4-9: GPIO level-triggered interrupt.....	95
Figure 6-1: General-purpose timer topology	166
Figure 6-2: Timer waveform in general timer mode ($TMRLOAD=100$)	167
Figure 6-3: Timer waveform in gated timer mode ($TMRLOAD=100$)	167
Figure 6-4: Timer waveform in event counter mode ($TMRLOAD=100$)	168
Figure 6-5: Timer waveform in event capture mode ($TMRLOAD=100$).....	168
Figure 6-6: GPT run flow diagram	168
Figure 7-1: Watchdog operation flow.....	175
Figure 8-1: PWM overview	183
Figure 8-2: PWM architecture	184
Figure 8-3: Time-Base sub-module signals	185
Figure 8-4: Time-base counter waveform	186
Figure 8-5: Time-base phase synchronization waveform.....	187
Figure 8-6: Counter-Compare sub-module structure	188
Figure 8-7: Counter-Compare sub-module waveform	189
Figure 8-8: Action-Qualifier sub-module inputs and outputs.....	189
Figure 8-9: Up-Down-Count, dual edge symmetric waveform of Action-Qualifier.....	191

Figure 8-10: Up-Down-Count, dual edge asymmetric waveform of Action-Qualifier	191
Figure 8-11: Dead-Band sub-module structure	192
Figure 8-12: Typical Dead-Band waveform	193
Figure 8-13: Dual-edge delay Dead-Band waveform	193
Figure 8-14: Trip-Zone sub-module logic	194
Figure 8-15: Trip-Zone interrupt generation	196
Figure 8-16: Digital Compare sub-module structure	197
Figure 8-17: Event filtering logic	198
Figure 8-18: Blank window timing diagram	198
Figure 8-19: Event processing	199
Figure 8-20: Event-Trigger sub-module structure	200
Figure 8-21: Global reload structure	201
Figure 9-1: Basic operation of enhanced capture	296
Figure 9-2: Capture and APWM modes of operation	296
Figure 9-3: Capture function diagram	297
Figure 9-4: Event prescale control	297
Figure 9-5: Prescale function waveforms	298
Figure 9-6: Details of the continuous/one-shot block	299
Figure 9-7: Details of the counter and synchronization block	299
Figure 9-8: Interrupts in ECAP module	300
Figure 9-9: Active and shadow register in APWM mode	301
Figure 9-10: ECAP module APWM mode output waveform (active high)	302
Figure 9-11: Capture sequence for absolute time-stamp and rising edge detect	303
Figure 9-12: Capture sequence for period and duty without counter reset	304
Figure 9-13: Capture period with counter reset	305
Figure 9-14: Capture duty and period with counter reset	306
Figure 9-15: Dual pin capture with counter reset	307
Figure 10-1: ADC block diagram	325
Figure 10-2: ADC Channel SOC generation	327
Figure 10-3: Analog input pin model	328
Figure 10-4: Averaging control for ADC result	329
Figure 10-5: Timing example	329
Figure 10-6: Example for round robin arbitration	331
Figure 10-7: Example for group with higher priority	333
Figure 10-8: ADC Interrupt	334
Figure 10-9: Post-processing unit topology	335
Figure 10-10: Detect the pin floating of ADC input (fault injection by discharging)	336
Figure 10-11: Detect the pin floating of ADC input (fault injection by precharging)	337
Figure 11-1: 13-bit Monitoring ADC block diagram	382
Figure 12-1: SPGA block architecture	385
Figure 13-1: DPGA application diagram	388
Figure 13-2: DPGA input range requirement	388
Figure 13-3: DPGA offset calibration path	389
Figure 14-1: Comparator block diagram	394
Figure 14-2: Digital filter behavior	395
Figure 15-1: Phase Comparator block diagram	417
Figure 15-2: Phase comparator output post-process	418
Figure 15-3: Digital filter behavior	419

Figure 15-4: Phase comparator interrupt generation	420
Figure 16-1: Buffered DAC block diagram	432
Figure 16-2: Ramp generator diagram.....	434
Figure 16-3: Ramp generator waveform	434
Figure 17-1: D2S buffer application diagram.....	436
Figure 18-1: Temperature sensor transfer function	437
Figure 19-1: UART block diagram	438
Figure 19-2: UART device connection diagram.....	440
Figure 19-3: UART LIN device connection diagram	441
Figure 19-4: UART frame.....	441
Figure 19-5: Example NRZ bit Encoding - 0b0100_1011.....	442
Figure 19-6: Example of auto baud.....	444
Figure 19-7: IR transmit and receive example	445
Figure 19-8: XMODE example	446
Figure 19-9: LIN bus topology	447
Figure 19-10: LIN byte field.....	447
Figure 19-11: LIN frame	448
Figure 19-12: LIN transfer	448
Figure 19-13: LIN protected ID byte field	449
Figure 20-1: SPI block diagram.....	475
Figure 20-2: FIFO packed and non-packed mode	478
Figure 20-3: SPI master-slave connection.....	480
Figure 20-4: SPI protocol (PHS = 0)	484
Figure 20-5: SPI protocol (PHS = 1)	486
Figure 20-6: SPI protocol (back to back transfer)	486
Figure 20-7: SSP protocol (single transfer)	487
Figure 20-8: SSP protocol (back to back transfer)	488
Figure 20-9: PSP protocol (single transfer)	489
Figure 20-10: PSP protocol (Consecutive Transfers).....	490
Figure 21-1: Data transfer on the I2C bus.....	508
Figure 21-2: START and STOP condition	509
Figure 21-3: 7-bit address format	510
Figure 21-4: 10-bit address format.....	510
Figure 21-5: Master-transmitter protocol	511
Figure 21-6: Master-receiver protocol.....	511
Figure 22-1: CAN Block Diagram	539
Figure 22-2: CAN device connection.....	540
Figure 22-3: ISO classical CAN frame format	541
Figure 22-4: ISO FDCAN frame format.....	541
Figure 22-5: CAN2.0B CAN frame format	541
Figure 22-6: NONISO FDCAN frame format	542
Figure 22-7: Arbitration phase and data phase	542
Figure 22-8: CAN node error state conversion	543
Figure 22-9: Bit timing.....	544
Figure 22-10: Transmitter delay compensation	545
Figure 22-11: Monitor mode.....	545
Figure 22-12: Loopback mode	546
Figure 22-13: External loopback mode	546

Figure 23-1: CRC block diagram	578
Figure 24-1: Flash memory access interfaces	588
Figure 24-2: Logical structure of the Flash module	589
Figure 24-3: Flash memory reading procedure	591
Figure 24-4: Flash memory programming procedure.....	592
Figure 24-5: Flash memory Sector Erase procedure.....	593
Figure 24-6: Flash memory Chip Erase procedure.....	594
Figure 26-1: DMA block diagram	667
Figure 26-2: Breakdown of DMA transfer for memory	669
Figure 26-3: Breakdown of DMA transfer for non-memory peripherals.....	670
Figure 26-4: DMA interrupt generation	673
Figure 26-5: Pseduo-code for DMA ISR.....	674
Figure 26-6: Scenario DMA interrupt events for multiple channels.....	674

List of Tables

Table 1-1: Peripheral Module Base Address.....	45
Table 1-2: Boot ROM hardware library entry table	47
Table 1-3: Boot modes.....	49
Table 2-1: POWER Module Base Address	56
Table 2-2: POWER Register Map.....	56
Table 2-3: Power Control Register (LDOCTL) Layout	57
Table 2-4: Power Control Register (LDOCTL) Description.....	57
Table 2-5: 3.3V BOD Control Register (BOD33CTL) Layout.....	58
Table 2-6: 3.3V BOD Control Register (BOD33CTL) Description	58
Table 2-7: 1.2V BOD Control Register (BOD12CTL) Layout.....	59
Table 2-8: 1.2V BOD Control Register (BOD12CTL) Description	59
Table 2-9: Power Mode Command Register (PWRMODECMD) Layout	60
Table 2-10: Power Mode Command Register (PWRMODECMD) Description.....	61
Table 2-11: Power Register Write-Allow Key Register (PWRREGKEY) Layout.....	61
Table 2-12: Power Register Write-Allow Key Register (PWRREGKEY) Description	61
Table 3-1: Input dividing ratio configuration	67
Table 3-2: Output dividing ratio and clock frequency	67
Table 3-3: CLOCK Module Base Address.....	70
Table 3-4: CLOCK Register Map	70
Table 3-5: Clock Status Register (CLKSTS) Layout	71
Table 3-6: Clock Status Register (CLKSTS) Description	71
Table 3-7: RCO Control Register (RCOCTL) Layout	72
Table 3-8: RCO Control Register (RCOCTL) Description.....	72
Table 3-9: Crystal Oscillator Control Register (XOCTL) Layout	73
Table 3-10: Crystal Oscillator Control Register (XOCTL) Description.....	73
Table 3-11: PLL Control Register 0 (PLLCTL0) Layout.....	74
Table 3-12: PLL Control Register 0 (PLLCTL0) Description	74
Table 3-13: PLL Control Register 1 (PLLCTL1) Layout.....	75
Table 3-14: PLL Control Register 1 (PLLCTL1) Description	75
Table 3-15: Clock Detection Control Register (CLKDETCTL) Layout.....	76
Table 3-16: Clock Detection Control Register (CLKDETCTL) Description	76
Table 3-17: Clock Detection Counter Threshold Register (CLKDETCTH) Layout.....	77
Table 3-18: Clock Detection Counter Threshold Register (CLKDETCTH) Description	78
Table 3-19: Clock Detection Counter Register (CLKDETCNT) Layout.....	78
Table 3-20: Clock Detection Counter Register (CLKDETCNT) Description	78
Table 3-21: Clock Trip-zone Event Enable Register (CLKTZE) Layout.....	78
Table 3-22: Clock Trip-zone Event Enable Register (CLKTZE) Description	79
Table 3-23: System Clock Control Register (SYSCLKCTL) Layout.....	79
Table 3-24: System Clock Control Register (SYSCLKCTL) Description	79
Table 3-25: Deglitch Clock Control Register (DGCLKCTL) Layout.....	80
Table 3-26: Deglitch Clock Control Register (DGCLKCTL) Description	80
Table 3-27: WDT0 Clock Control Register (WDT0CLKCTL) Layout	80
Table 3-28: WDT0 Clock Control Register (WDT0CLKCTL) Description	81
Table 3-29: WDT1 Clock Control Register (WDT1CLKCTL) Layout	81
Table 3-30: WDT1 Clock Control Register (WDT1CLKCTL) Description	81
Table 3-31: Peripheral Clock Enable Register (PERICKEN) Layout	82

Table 3-32: Peripheral Clock Enable Register (PERICKEN) Description.....	82
Table 3-33: Peripheral Clock Status Register (PERICKSTS) Layout	83
Table 3-34: Peripheral Clock Status Register (PERICKSTS) Description	83
Table 3-35: Clock Register Write-Allow Key Register (CLKREGKEY) Layout.....	85
Table 3-36: Clock Register Write-Allow Key Register (CLKREGKEY) Description	85
Table 4-1: I/O Pin function channel definition.....	89
Table 4-2: Debug interface definition	91
Table 4-3: PINMUX Module Base Address	97
Table 4-4: PINMUX Register Map	97
Table 4-5: GPIO0 Pin Control Register (GPIO0) Layout.....	99
Table 4-6: GPIO0 Pin Control Register (GPIO0) Description	99
Table 4-7: GPIO1 Pin Control Register (GPIO1) Layout.....	100
Table 4-8: GPIO1 Pin Control Register (GPIO1) Description	100
Table 4-9: GPIO2 Pin Control Register (GPIO2) Layout.....	101
Table 4-10: GPIO2 Pin Control Register (GPIO2) Description	101
Table 4-11: GPIO3 Pin Control Register (GPIO3) Layout.....	102
Table 4-12: GPIO3 Pin Control Register (GPIO3) Description	102
Table 4-13: GPIO4 Pin Control Register (GPIO4) Layout.....	103
Table 4-14: GPIO4 Pin Control Register (GPIO4) Description	103
Table 4-15: GPIO5 Pin Control Register (GPIO5) Layout.....	104
Table 4-16: GPIO5 Pin Control Register (GPIO5) Description	104
Table 4-17: GPIO6 Pin Control Register (GPIO6) Layout.....	105
Table 4-18: GPIO6 Pin Control Register (GPIO6) Description	105
Table 4-19: GPIO7 Pin Control Register (GPIO7) Layout.....	106
Table 4-20: GPIO7 Pin Control Register (GPIO7) Description	106
Table 4-21: GPIO8 Pin Control Register (GPIO8) Layout.....	107
Table 4-22: GPIO8 Pin Control Register (GPIO8) Description	107
Table 4-23: GPIO9 Pin Control Register (GPIO9) Layout.....	108
Table 4-24: GPIO9 Pin Control Register (GPIO9) Description	108
Table 4-25: GPIO10 Pin Control Register (GPIO10) Layout.....	109
Table 4-26: GPIO10 Pin Control Register (GPIO10) Description	109
Table 4-27: GPIO11 Pin Control Register (GPIO11) Layout.....	110
Table 4-28: GPIO11 Pin Control Register (GPIO11) Description	110
Table 4-29: GPIO12 Pin Control Register (GPIO12) Layout.....	111
Table 4-30: GPIO12 Pin Control Register (GPIO12) Description	111
Table 4-31: GPIO13 Pin Control Register (GPIO13) Layout.....	112
Table 4-32: GPIO13 Pin Control Register (GPIO13) Description	112
Table 4-33: GPIO14 Pin Control Register (GPIO14) Layout.....	113
Table 4-34: GPIO14 Pin Control Register (GPIO14) Description	113
Table 4-35: GPIO15 Pin Control Register (GPIO15) Layout.....	114
Table 4-36: GPIO15 Pin Control Register (GPIO15) Description	114
Table 4-37: GPIO16 Pin Control Register (GPIO16) Layout.....	115
Table 4-38: GPIO16 Pin Control Register (GPIO16) Description	115
Table 4-39: GPIO17 Pin Control Register (GPIO17) Layout.....	116
Table 4-40: GPIO17 Pin Control Register (GPIO17) Description	116
Table 4-41: GPIO18 Pin Control Register (GPIO18) Layout.....	117
Table 4-42: GPIO18 Pin Control Register (GPIO18) Description	117
Table 4-43: GPIO19 Pin Control Register (GPIO19) Layout.....	118

Table 4-44: GPIO19 Pin Control Register (GPIO19) Description	118
Table 4-45: GPIO20 Pin Control Register (GPIO20) Layout.....	119
Table 4-46: GPIO20 Pin Control Register (GPIO20) Description	119
Table 4-47: GPIO21 Pin Control Register (GPIO21) Layout.....	120
Table 4-48: GPIO21 Pin Control Register (GPIO21) Description	120
Table 4-49: GPIO22 Pin Control Register (GPIO22) Layout.....	121
Table 4-50: GPIO22 Pin Control Register (GPIO22) Description	121
Table 4-51: GPIO23 Pin Control Register (GPIO23) Layout.....	122
Table 4-52: GPIO23 Pin Control Register (GPIO23) Description	122
Table 4-53: GPIO24 Pin Control Register (GPIO24) Layout.....	123
Table 4-54: GPIO24 Pin Control Register (GPIO24) Description	123
Table 4-55: GPIO25 Pin Control Register (GPIO25) Layout.....	124
Table 4-56: GPIO25 Pin Control Register (GPIO25) Description	124
Table 4-57: GPIO26 Pin Control Register (GPIO26) Layout.....	125
Table 4-58: GPIO26 Pin Control Register (GPIO26) Description	125
Table 4-59: GPIO27 Pin Control Register (GPIO27) Layout.....	126
Table 4-60: GPIO27 Pin Control Register (GPIO27) Description	126
Table 4-61: GPIO28 Pin Control Register (GPIO28) Layout.....	127
Table 4-62: GPIO28 Pin Control Register (GPIO28) Description	127
Table 4-63: GPIO29 Pin Control Register (GPIO29) Layout.....	128
Table 4-64: GPIO29 Pin Control Register (GPIO29) Description	128
Table 4-65: GPIO30 Pin Control Register (GPIO30) Layout.....	129
Table 4-66: GPIO30 Pin Control Register (GPIO30) Description	129
Table 4-67: GPIO31 Pin Control Register (GPIO31) Layout.....	130
Table 4-68: GPIO31 Pin Control Register (GPIO31) Description	130
Table 4-69: GPIO32 Pin Control Register (GPIO32) Layout.....	131
Table 4-70: GPIO32 Pin Control Register (GPIO32) Description	131
Table 4-71: GPIO33 Pin Control Register (GPIO33) Layout.....	132
Table 4-72: GPIO33 Pin Control Register (GPIO33) Description	132
Table 4-73: GPIO34 Pin Control Register (GPIO34) Layout.....	133
Table 4-74: GPIO34 Pin Control Register (GPIO34) Description	133
Table 4-75: GPIO35 Pin Control Register (GPIO35) Layout.....	134
Table 4-76: GPIO35 Pin Control Register (GPIO35) Description	134
Table 4-77: GPIO36 Pin Control Register (GPIO36) Layout.....	135
Table 4-78: GPIO36 Pin Control Register (GPIO36) Description	135
Table 4-79: GPIO37 Pin Control Register (GPIO37) Layout.....	136
Table 4-80: GPIO37 Pin Control Register (GPIO37) Description	136
Table 4-81: GPIO38 Pin Control Register (GPIO38) Layout.....	137
Table 4-82: GPIO38 Pin Control Register (GPIO38) Description	137
Table 4-83: GPIO39 Pin Control Register (GPIO39) Layout.....	138
Table 4-84: GPIO39 Pin Control Register (GPIO39) Description	138
Table 4-85: GPIO40 Pin Control Register (GPIO40) Layout.....	139
Table 4-86: GPIO40 Pin Control Register (GPIO40) Description	139
Table 4-87: Comparator Output Monitor Control Register (COMPMONCTL) Layout	140
Table 4-88: Comparator Output Monitor Control Register (COMPMONCTL) Description	140
Table 4-89: PINMUX Register Write-Allow Key Register (PINMUXREGKEY) Layout.....	142
Table 4-90: PINMUX Register Write-Allow Key Register (PINMUXREGKEY) Description	142
Table 4-91: GPIO Module Base Address	142

Table 4-92: GPIO Register Map.....	142
Table 4-93: GPIO Pin Level Register n ([0:1]) (GPLR) Layout	143
Table 4-94: GPIO Pin Level Register n ([0:1]) (GPLR) Description.....	144
Table 4-95: GPIO Pin Output Set Register n ([0:1]) (GSLR) Layout	144
Table 4-96: GPIO Pin Output Set Register n ([0:1]) (GSLR) Description.....	144
Table 4-97: GPIO Pin Output Clear Register n ([0:1]) (GCLR) Layout.....	144
Table 4-98: GPIO Pin Output Clear Register n ([0:1]) (GCLR) Description	145
Table 4-99: GPIO Pin Direction Register n ([0:1]) (GPDR) Layout	145
Table 4-100: GPIO Pin Direction Register n ([0:1]) (GPDR) Description	145
Table 4-101: GPIO Pin Bitwise Set Direction Register n ([0:1]) (GSDR) Layout	145
Table 4-102: GPIO Pin Bitwise Set Direction Register n ([0:1]) (GSDR) Description	146
Table 4-103: GPIO Pin Bitwise Clear Direction Register n ([0:1]) (GCDR) Layout.....	146
Table 4-104: GPIO Pin Bitwise Clear Direction Register n ([0:1]) (GCDR) Description	146
Table 4-105: GPIO Pin Rising Edge Detect Enable Register n ([0:1]) (GRER) Layout	146
Table 4-106: GPIO Pin Rising Edge Detect Enable Register n ([0:1]) (GRER) Description.....	147
Table 4-107: GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1]) (GSRE) Layout.....	147
Table 4-108: GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1]) (GSRE) Description.....	147
Table 4-109: GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1]) (GCRE) Layout ...	147
Table 4-110: GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1]) (GCRE) Description	148
Table 4-111: GPIO Pin Falling Edge Detect Enable Register n ([0:1]) (GFER) Layout.....	148
Table 4-112: GPIO Pin Falling Edge Detect Enable Register n ([0:1]) (GFER) Description	148
Table 4-113: GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1]) (GSFE) Layout	148
Table 4-114: GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1]) (GSFE) Description	149
Table 4-115: GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1]) (GCFE) Layout...	149
Table 4-116: GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1]) (GCFE) Description	149
Table 4-117: GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1]) (GRERAWIF) Layout	149
Table 4-118: GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1]) (GRERAWIF) Description	150
Table 4-119: GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1]) (GFERAWIF) Layout	150
Table 4-120: GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1]) (GFERAWIF) Description	150
Table 4-121: GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1]) (GREIF) Layout.....	150
Table 4-122: GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1]) (GREIF) Description	151
Table 4-123: GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1]) (GFEIF) Layout	151
Table 4-124: GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1]) (GFEIF) Description	151
Table 4-125: GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1]) (GREIE) Layout	151
Table 4-126: GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1]) (GREIE) Description	152
Table 4-127: GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1]) (GFEIE) Layout ...	152
Table 4-128: GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1]) (GFEIE) Description	152
Table 4-129: GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1]) (GREIC) Layout.....	152

Table 4-130: GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1]) (GREIC) Description	153
Table 4-131: GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1]) (GFEIC) Layout	153
Table 4-132: GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1]) (GFEIC) Description	153
Table 4-133: GPIO Pin Rising Edge Event Force Register n ([0:1]) (GREIFRC) Layout	153
Table 4-134: GPIO Pin Rising Edge Event Force Register n ([0:1]) (GREIFRC) Description	154
Table 4-135: GPIO Pin Falling Edge Event Force Register n ([0:1]) (GFEIFRC) Layout	154
Table 4-136: GPIO Pin Falling Edge Event Force Register n ([0:1]) (GFEIFRC) Description	154
Table 4-137: GPIO Pin Level Detect Enable Register n ([0:1]) (GLDE) Layout	154
Table 4-138: GPIO Pin Level Detect Enable Register n ([0:1]) (GLDE) Description	155
Table 4-139: GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1]) (GSLDE) Layout	155
Table 4-140: GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1]) (GSLDE) Description	155
Table 4-141: GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1]) (GCLDE) Layout	155
Table 4-142: GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1]) (GCLDE) Description	156
Table 4-143: GPIO Pin Level Detect Polarity Register n ([0:1]) (GLPOL) Layout	156
Table 4-144: GPIO Pin Level Detect Polarity Register n ([0:1]) (GLPOL) Description	156
Table 4-145: GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1]) (GLRAWIF) Layout	156
Table 4-146: GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1]) (GLRAWIF) Description ..	157
Table 4-147: GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1]) (GLIF) Layout	157
Table 4-148: GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1]) (GLIF) Description	157
Table 4-149: GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1]) (GLIE) Layout	157
Table 4-150: GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1]) (GLIE) Description	158
Table 4-151: GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1]) (GLIC) Layout	158
Table 4-152: GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1]) (GLIC) Description	158
Table 4-153: GPIO Pin Level-Triggered Event Force Register n ([0:1]) (GLIFRC) Layout	158
Table 4-154: GPIO Pin Level-Triggered Event Force Register n ([0:1]) (GLIFRC) Description	159
Table 4-155: GPIO Global Interrupt Raw Flag Register (GGRAWIF) Layout	159
Table 4-156: GPIO Global Interrupt Raw Flag Register (GGRAWIF) Description	159
Table 4-157: GPIO Global Interrupt Flag Register (GGIF) Layout	160
Table 4-158: GPIO Global Interrupt Flag Register (GGIF) Description	160
Table 4-159: GPIO Global Interrupt Clear Register (GGIC) Layout	160
Table 4-160: GPIO Global Interrupt Clear Register (GGIC) Description	161
Table 4-161: GPIO Register Write-Allow Key Register (GPIOREGKEY) Layout	161
Table 4-162: GPIO Register Write-Allow Key Register (GPIOREGKEY) Description	161
Table 5-1: Vector table for SPC1169	162
Table 6-1: Timer Module Base Address	169
Table 6-2: TIMER Register Map	169
Table 6-3: Timer Load Register (TMRLOAD) Layout	170
Table 6-4: Timer Load Register (TMRLOAD) Description	170
Table 6-5: Timer Counter Value Register (TMRCNT) Layout	170
Table 6-6: Timer Counter Value Register (TMRCNT) Description	170
Table 6-7: Timer Control Register (TMRCTL) Layout	171
Table 6-8: Timer Control Register (TMRCTL) Description	171
Table 6-9: Timer Interrupt Raw Flag Register (TMRRAWIF) Layout	172
Table 6-10: Timer Interrupt Raw Flag Register (TMRRAWIF) Description	172
Table 6-11: Timer Interrupt Flag Register (TMRIF) Layout	173
Table 6-12: Timer Interrupt Flag Register (TMRIF) Description	173

Table 6-13: Timer Interrupt Clear Register (TMRIC) Layout	173
Table 6-14: Timer Interrupt Clear Register (TMRIC) Description	173
Table 6-15: Timer Interrupt Enable Register (TMRIE) Layout.....	174
Table 6-16: Timer Interrupt Enable Register (TMRIE) Description	174
Table 6-17: Timer Register Write-Allow Key Register (TMRREGKEY) Layout	174
Table 6-18: Timer Register Write-Allow Key Register (TMRREGKEY) Description.....	174
Table 7-1: WDT Module Base Address.....	177
Table 7-2: WDT Register Map	177
Table 7-3: Watchdog Timer Load Register (WDTLOAD) Layout	178
Table 7-4: Watchdog Timer Load Register (WDTLOAD) Description.....	178
Table 7-5: Watchdog Timer Current Value Register (WDTCNT) Layout	178
Table 7-6: Watchdog Timer Current Value Register (WDTCNT) Description	178
Table 7-7: Watchdog Timer Control Register (WDTCTL) Layout	179
Table 7-8: Watchdog Timer Control Register (WDTCTL) Description.....	179
Table 7-9: Watchdog Timer Interrupt Raw Flag Register (WDTRAWIF) Layout	179
Table 7-10: Watchdog Timer Interrupt Raw Flag Register (WDTRAWIF) Description.....	180
Table 7-11: Watchdog Timer Interrupt Flag Register (WDTIF) Layout	180
Table 7-12: Watchdog Timer Interrupt Flag Register (WDTIF) Description.....	180
Table 7-13: Watchdog Timer Interrupt Clear Register (WDTIC) Layout	180
Table 7-14: Watchdog Timer Interrupt Clear Register (WDTIC) Description	181
Table 7-15: Watchdog Timer Interrupt Enable Register (WDTIE) Layout.....	181
Table 7-16: Watchdog Timer Interrupt Enable Register (WDTIE) Description	181
Table 7-17: Watchdog Windowed-Feed Low Threshold Register (WDTTHL) Layout.....	181
Table 7-18: Watchdog Windowed-Feed Low Threshold Register (WDTTHL) Description	182
Table 7-19: Watchdog Windowed-Feed High Threshold Register (WDTTHH) Layout	182
Table 7-20: Watchdog Windowed-Feed High Threshold Register (WDTTHH) Description.....	182
Table 7-21: Watchdog Register Write-Allow Key Register (WDTREGKEY) Layout	182
Table 7-22: Watchdog Register Write-Allow Key Register (WDTREGKEY) Description	182
Table 8-1: Action-Qualifier event priority for Up-Count mode	190
Table 8-2: Action-Qualifier Event Priority for Down-Count Mode	190
Table 8-3: Action-Qualifier Event Priority for Up-Down-Count Mode	190
Table 8-4: Typical Dead-Band operating modes.....	192
Table 8-5: Priority of events that effect the final PWM outputs.....	196
Table 8-6: PWM Module Base Address	202
Table 8-7: PWM Register Map	202
Table 8-8: Shadow Status Register (SHADOWSTS) Layout	204
Table 8-9: Shadow Status Register (SHADOWSTS) Description.....	204
Table 8-10: Global Shadow to Active Load Control Register 0 (GLDCTL0) Layout	205
Table 8-11: Global Shadow to Active Load Control Register 0 (GLDCTL0) Description	206
Table 8-12: Global Shadow to Active Load Control Register 1 (GLDCTL1) Layout	207
Table 8-13: Global Shadow to Active Load Control Register 1 (GLDCTL1) Description	207
Table 8-14: Global Shadow to Active Load Select Register (GLDSEL) Layout	207
Table 8-15: Global Shadow to Active Load Select Register (GLDSEL) Description	208
Table 8-16: PWM Link Control Register (PWMLINK) Layout	209
Table 8-17: PWM Link Control Register (PWMLINK) Description.....	209
Table 8-18: Time-Base Control Register (TBCTL) Layout	211
Table 8-19: Time-Base Control Register (TBCTL) Description.....	211
Table 8-20: Time-Base Period Register (TBPRD) Layout.....	213

Table 8-21: Time-Base Period Register (TBPRD) Description	213
Table 8-22: Time-Base Period Active Register (TBPRDA) Layout.....	213
Table 8-23: Time-Base Period Active Register (TBPRDA) Description	213
Table 8-24: Time-Base Phase Register (TBPHS) Layout	214
Table 8-25: Time-Base Phase Register (TBPHS) Description.....	214
Table 8-26: Time-Base Counter Register (TBCNT) Layout	214
Table 8-27: Time-Base Counter Register (TBCNT) Description.....	214
Table 8-28: Time-Base Status Register (TBFLG) Layout	215
Table 8-29: Time-Base Status Register (TBFLG) Description	215
Table 8-30: Time-Base Status Clear Register (TBFLGCLR) Layout.....	216
Table 8-31: Time-Base Status Clear Register (TBFLGCLR) Description	216
Table 8-32: Counter-Compare Control Register (CMPCTL) Layout.....	217
Table 8-33: Counter-Compare Control Register (CMPCTL) Description	217
Table 8-34: Counter-Compare A Threshold Register (CMPA) Layout.....	219
Table 8-35: Counter-Compare A Threshold Register (CMPA) Description	219
Table 8-36: Counter-Compare A Threshold Active Register (CMPAA) Layout	220
Table 8-37: Counter-Compare A Threshold Active Register (CMPAA) Description.....	220
Table 8-38: Counter-Compare B Threshold Register (CMPB) Layout.....	220
Table 8-39: Counter-Compare B Threshold Register (CMPB) Description	220
Table 8-40: Counter-Compare B Threshold Active Register (CMPBA) Layout.....	221
Table 8-41: Counter-Compare B Threshold Active Register (CMPBA) Description	221
Table 8-42: Counter-Compare C Threshold Register (CMPC) Layout.....	221
Table 8-43: Counter-Compare C Threshold Register (CMPC) Description	221
Table 8-44: Counter-Compare C Threshold Active Register (CMPCA) Layout.....	222
Table 8-45: Counter-Compare C Threshold Active Register (CMPCA) Description	222
Table 8-46: Counter-Compare D Threshold Register (CMPD) Layout	222
Table 8-47: Counter-Compare D Threshold Register (CMPD) Description.....	222
Table 8-48: Counter-Compare D Threshold Active Register (CMPDA) Layout	223
Table 8-49: Counter-Compare D Threshold Active Register (CMPDA) Description	223
Table 8-50: Action-Qualifier Control Register (AQCTL) Layout.....	223
Table 8-51: Action-Qualifier Control Register (AQCTL) Description	224
Table 8-52: Action-Qualifier Output A Control Register (AQCTLA) Layout.....	225
Table 8-53: Action-Qualifier Output A Control Register (AQCTLA) Description	225
Table 8-54: Action-Qualifier Output A Control Active Register (AQCTLAA) Layout	227
Table 8-55: Action-Qualifier Output A Control Active Register (AQCTLAA) Description.....	227
Table 8-56: Action-Qualifier Output B Control Register (AQCTLB) Layout.....	229
Table 8-57: Action-Qualifier Output B Control Register (AQCTLB) Description	229
Table 8-58: Action-Qualifier Output B Control Active Register (AQCTLBA) Layout.....	231
Table 8-59: Action-Qualifier Output B Control Active Register (AQCTLBA) Description	231
Table 8-60: Action-Qualifier Software Force Register (AQSFRC) Layout	232
Table 8-61: Action-Qualifier Software Force Register (AQSFRC) Description	233
Table 8-62: Action-Qualifier Continuous Software Force Register (AQCSFRC) Layout	234
Table 8-63: Action-Qualifier Continuous Software Force Register (AQCSFRC) Description.....	234
Table 8-64: Action-Qualifier Continuous Software Force Active Register (AQCSFRCA) Layout	234
Table 8-65: Action-Qualifier Continuous Software Force Active Register (AQCSFRCA) Description	235
Table 8-66: Dead-Band Generator Control Register (DBCTL) Layout	235
Table 8-67: Dead-Band Generator Control Register (DBCTL) Description	235
Table 8-68: Dead-Band Generator Control Active Register (DBCTLA) Layout.....	237

Table 8-69: Dead-Band Generator Control Active Register (DBCTLA) Description	238
Table 8-70: Dead-Band Generator Rising Edge Delay Register (DBRED) Layout	239
Table 8-71: Dead-Band Generator Rising Edge Delay Register (DBRED) Description	239
Table 8-72: Dead-Band Generator Rising Edge Delay Active Register (DBREDA) Layout.....	239
Table 8-73: Dead-Band Generator Rising Edge Delay Active Register (DBREDA) Description	239
Table 8-74: Dead-Band Generator Falling Edge Delay Register (DBFED) Layout	240
Table 8-75: Dead-Band Generator Falling Edge Delay Register (DBFED) Description.....	240
Table 8-76: Dead-Band Generator Falling Edge Delay Active Register (DBFEDA) Layout	240
Table 8-77: Dead-Band Generator Falling Edge Delay Active Register (DBFEDA) Description	240
Table 8-78: Trip-Zone Event Select Register (TZSEL) Layout	241
Table 8-79: Trip-Zone Event Select Register (TZSEL) Description.....	241
Table 8-80: Trip-Zone Flag Register (TZFLG) Layout	243
Table 8-81: Trip-Zone Flag Register (TZFLG) Description.....	243
Table 8-82: Trip-Zone Flag Clear Register (TZFLGCLR) Layout	245
Table 8-83: Trip-Zone Flag Clear Register (TZFLGCLR) Description	245
Table 8-84: Trip-Zone Digital Compare Event Select Register (TZDCSEL) Layout.....	247
Table 8-85: Trip-Zone Digital Compare Event Select Register (TZDCSEL) Description	247
Table 8-86: Trip-Zone Output A Control Register (TZACTL) Layout.....	248
Table 8-87: Trip-Zone Output A Control Register (TZACTL) Description	248
Table 8-88: Trip-Zone Output B Control Register (TZBCTL) Layout	250
Table 8-89: Trip-Zone Output B Control Register (TZBCTL) Description.....	250
Table 8-90: Trip-Zone Interrupt Flag Register (TZIF) Layout.....	251
Table 8-91: Trip-Zone Interrupt Flag Register (TZIF) Description	251
Table 8-92: Trip-Zone Clear Register (TZIC) Layout	253
Table 8-93: Trip-Zone Clear Register (TZIC) Description.....	253
Table 8-94: Trip-Zone Interrupt Enable Register (TZIE) Layout	254
Table 8-95: Trip-Zone Interrupt Enable Register (TZIE) Description	254
Table 8-96: Trip-Zone Force Register (TZFRC) Layout.....	255
Table 8-97: Trip-Zone Force Register (TZFRC) Description	255
Table 8-98: Digital Compare AL Trip Select Register (DCALTRIPSEL) Layout	257
Table 8-99: Digital Compare AL Trip Select Register (DCALTRIPSEL) Description	257
Table 8-100: Digital Compare AH Trip Select Register (DCAHTRIPSEL) Layout	258
Table 8-101: Digital Compare AH Trip Select Register (DCAHTRIPSEL) Description.....	258
Table 8-102: Digital Compare BL Trip Select Register (DCBLTRIPSEL) Layout	259
Table 8-103: Digital Compare BL Trip Select Register (DCBLTRIPSEL) Description.....	259
Table 8-104: Digital Compare BH Trip Select Register (DCBHTRIPSEL) Layout.....	260
Table 8-105: Digital Compare BH Trip Select Register (DCBHTRIPSEL) Description	261
Table 8-106: Digital Compare A Control Register (DCACTL) Layout	262
Table 8-107: Digital Compare A Control Register (DCACTL) Description	262
Table 8-108: Digital Compare B Control Register (DCBCTL) Layout	263
Table 8-109: Digital Compare B Control Register (DCBCTL) Description.....	263
Table 8-110: Digital Compare Filter Register (DCFCTL) Layout.....	264
Table 8-111: Digital Compare Filter Register (DCFCTL) Description.....	264
Table 8-112: Digital Compare Filter Offset Register (DCFOFFSET) Layout	265
Table 8-113: Digital Compare Filter Offset Register (DCFOFFSET) Description.....	266
Table 8-114: Digital Compare Filter Offset Counter Register (DCFOFFSETCNT) Layout.....	266
Table 8-115: Digital Compare Filter Offset Counter Register (DCFOFFSETCNT) Description	266
Table 8-116: Digital Compare Filter Window Register (DCFWINDOW) Layout.....	266

Table 8-117: Digital Compare Filter Window Register (DCFWINDOW) Description	267
Table 8-118: Digital Compare Filter Window Counter Register (DCFWINDOWCNT) Layout	267
Table 8-119: Digital Compare Filter Window Counter Register (DCFWINDOWCNT) Description.....	267
Table 8-120: Digital Compare Capture Control Register (DCCAPCTL) Layout	268
Table 8-121: Digital Compare Capture Control Register (DCCAPCTL) Description.....	268
Table 8-122: Digital Compare Counter Capture Register (DCCAP) Layout	269
Table 8-123: Digital Compare Counter Capture Register (DCCAP) Description	269
Table 8-124: Event-Trigger Control Register (ETCTL) Layout.....	270
Table 8-125: Event-Trigger Control Register (ETCTL) Description	270
Table 8-126: Event-Trigger Prescale Register (ETPS) Layout	272
Table 8-127: Event-Trigger Prescale Register (ETPS) Description	273
Table 8-128: Event-Trigger Flag Register (ETFLG) Layout.....	278
Table 8-129: Event-Trigger Flag Register (ETFLG) Description	278
Table 8-130: Event-Trigger Clear Register (ETCLR) Layout	279
Table 8-131: Event-Trigger Clear Register (ETCLR) Description.....	279
Table 8-132: Event-Trigger Force Register (ETFRC) Layout	279
Table 8-133: Event-Trigger Force Register (ETFRC) Description.....	280
Table 8-134: PWM Register Write-Allow Key Register (PWMREGKEY) Layout	281
Table 8-135: PWM Register Write-Allow Key Register (PWMREGKEY) Description	281
Table 8-136: PWMCFG Module Base Address	282
Table 8-137: PWMCFG Register Map	282
Table 8-138: TZ0 Source Control Register (TZ0SRCCTL) Layout	283
Table 8-139: TZ0 Source Control Register (TZ0SRCCTL) Description	283
Table 8-140: TZ1 Source Control Register (TZ1SRCCTL) Layout	283
Table 8-141: TZ1 Source Control Register (TZ1SRCCTL) Description	283
Table 8-142: TZ2 Source Control Register (TZ2SRCCTL) Layout	284
Table 8-143: TZ2 Source Control Register (TZ2SRCCTL) Description	284
Table 8-144: TZ3 Source Control Register (TZ3SRCCTL) Layout	284
Table 8-145: TZ3 Source Control Register (TZ3SRCCTL) Description	284
Table 8-146: TZ4 Source Control Register (TZ4SRCCTL) Layout	285
Table 8-147: TZ4 Source Control Register (TZ4SRCCTL) Description	285
Table 8-148: Global PWM Force Synchronization Register (FRCSYNC) Layout	285
Table 8-149: Global PWM Force Synchronization Register (FRCSYNC) Description.....	285
Table 8-150: GPIO PWMSYNCI Control Register (GPiOSYNCICTL) Layout.....	286
Table 8-151: GPIO PWMSYNCI Control Register (GPiOSYNCICTL) Description	286
Table 8-152: GPIO Force PWMSYNCI Enable Register (GPiOSYNCIEN) Layout	287
Table 8-153: GPIO Force PWMSYNCI Enable Register (GPiOSYNCIEN) Description.....	287
Table 8-154: Timer 0 Force PWMSYNCI Enable Register (TMR0SYNCIEN) Layout.....	287
Table 8-155: Timer 0 Force PWMSYNCI Enable Register (TMR0SYNCIEN) Description	288
Table 8-156: Timer 1 Force PWMSYNCI Enable Register (TMR1SYNCIEN) Layout.....	288
Table 8-157: Timer 1 Force PWMSYNCI Enable Register (TMR1SYNCIEN) Description	288
Table 8-158: Timer 2 Force PWMSYNCI Enable Register (TMR2SYNCIEN) Layout.....	289
Table 8-159: Timer 2 Force PWMSYNCI Enable Register (TMR2SYNCIEN) Description	289
Table 8-160: PWMSYNCO Control Register (SYNCOCTL) Layout	289
Table 8-161: PWMSYNCO Control Register (SYNCOCTL) Description	290
Table 8-162: PWMSOCA Output Control Register (SOCAOCTL) Layout	290
Table 8-163: PWMSOCA Output Control Register (SOCAOCTL) Description.....	291
Table 8-164: PWMSOCB Output Control Register (SOCBOCTL) Layout.....	291

Table 8-165: PWMSOCB Output Control Register (SOCBOCTL) Description	292
Table 8-166: PWMSOCC Output Control Register (SOCCOCTL) Layout.....	292
Table 8-167: PWMSOCC Output Control Register (SOCCOCTL) Description	293
Table 8-168: PWMCFG Register Write-Allow Key Register (PWMCFGREGKEY) Layout	293
Table 8-169: PWMCFG Register Write-Allow Key Register (PWMCFGREGKEY) Description	294
Table 9-1: ECAP Module Base Address	308
Table 9-2: ECAP Register Map.....	308
Table 9-3: Capture Source Input Control Register (CAPSRCCTL) Layout	309
Table 9-4: Capture Source Input Control Register (CAPSRCCTL) Description.....	309
Table 9-5: Capture Synchronization Source Input Control Register (CAPSYNCICTL) Layout	310
Table 9-6: Capture Synchronization Source Input Control Register (CAPSYNCICTL) Description	310
Table 9-7: Time-Stamp Counter Register (CAPTSCNT) Layout	310
Table 9-8: Time-Stamp Counter Register (CAPTSCNT) Description	310
Table 9-9: Counter Phase Offset Value Register (CAPCNTPHS) Layout	311
Table 9-10: Counter Phase Offset Value Register (CAPCNTPHS) Description	311
Table 9-11: Capture Register 0 (CAP0) Layout.....	311
Table 9-12: Capture Register 0 (CAP0) Description	311
Table 9-13: Capture Register 1 (CAP1) Layout.....	312
Table 9-14: Capture Register 1 (CAP1) Description	312
Table 9-15: Capture Register 2 (CAP2) Layout.....	312
Table 9-16: Capture Register 2 (CAP2) Description	313
Table 9-17: Capture Register 3 (CAP3) Layout.....	313
Table 9-18: Capture Register 3 (CAP3) Description	313
Table 9-19: Capture Control Register (CAPCTL) Layout	314
Table 9-20: Capture Control Register (CAPCTL) Description	314
Table 9-21: Capture Interrupt Raw Flag Register (CAPRAWIF) Layout.....	317
Table 9-22: Capture Interrupt Raw Flag Register (CAPRAWIF) Description	317
Table 9-23: Capture Interrupt Flag Register (CAPIF) Layout.....	318
Table 9-24: Capture Interrupt Flag Register (CAPIF) Description	318
Table 9-25: Capture Interrupt Enable Register (CAPIE) Layout	320
Table 9-26: Capture Interrupt Enable Register (CAPIE) Description.....	320
Table 9-27: Capture Interrupt Clear Register (CAPIC) Layout.....	321
Table 9-28: Capture Interrupt Clear Register (CAPIC) Description	321
Table 9-29: Capture Interrupt Event Force Register (CAPIFRC) Layout	322
Table 9-30: Capture Interrupt Event Force Register (CAPIFRC) Description	322
Table 9-31: Capture Register Write-Allow Key Register (CAPREGKEY) Layout.....	323
Table 9-32: Capture Register Write-Allow Key Register (CAPREGKEY) Description	323
Table 10-1: ADC channel input selection.....	327
Table 10-2: ADC Module Base Address.....	339
Table 10-3: ADC Register Map	339
Table 10-4: ADC Interrupt Flag Register (ADCIF) Layout	341
Table 10-5: ADC Interrupt Flag Register (ADCIF) Description.....	341
Table 10-6: ADC Interrupt Flag Clear Register (ADCIC) Layout.....	342
Table 10-7: ADC Interrupt Flag Clear Register (ADCIC) Description	342
Table 10-8: ADC Interrupt Overflow Flag Register (ADCIOVF) Layout.....	345
Table 10-9: ADC Interrupt Overflow Flag Register (ADCIOVF) Description	345
Table 10-10: ADC Interrupt Overflow Flag Clear Register (ADCIOVFC) Layout	346
Table 10-11: ADC Interrupt Overflow Flag Clear Register (ADCIOVFC) Description.....	346

Table 10-12: ADC Interrupt Enable Register (ADCIE) Layout	348
Table 10-13: ADC Interrupt Enable Register (ADCIE) Description	348
Table 10-14: ADC Start of Conversion Priority Control Register (ADCSOCPRICTL) Layout.....	350
Table 10-15: ADC Start of Conversion Priority Control Register (ADCSOCPRICTL) Description	350
Table 10-16: ADC SOC Flag Register (ADCSOCFLG) Layout.....	352
Table 10-17: ADC SOC Flag Register (ADCSOCFLG) Description	352
Table 10-18: ADC SOC Force Register (ADCSOCFRC) Layout	354
Table 10-19: ADC SOC Force Register (ADCSOCFRC) Description	354
Table 10-20: ADC SOC Overflow Register (ADCSOCOVF) Layout.....	355
Table 10-21: ADC SOC Overflow Register (ADCSOCOVF) Description	355
Table 10-22: ADC SOC Overflow Clear Register (ADCSOCOVFC) Layout	359
Table 10-23: ADC SOC Overflow Clear Register (ADCSOCOVFC) Description.....	359
Table 10-24: ADC Interrupt Trigger SOC Enable Register (ADCINTSOCEN) Layout	363
Table 10-25: ADC Interrupt Trigger SOC Enable Register (ADCINTSOCEN) Description.....	363
Table 10-26: ADC Interrupt Trigger SOC Select Register 0 (ADCINTSOCSEL) Layout.....	365
Table 10-27: ADC Interrupt Trigger SOC Select Register 0 (ADCINTSOCSEL) Description	365
Table 10-28: ADC External SOC Input Control Register (ADCEXTSOCCTL) Layout.....	365
Table 10-29: ADC External SOC Input Control Register (ADCEXTSOCCTL) Description	365
Table 10-30: ADC Channel n ([0:15]) Control Register (ADCCHCTL) Layout.....	366
Table 10-31: ADC Channel n ([0:15]) Control Register (ADCCHCTL) Description	366
Table 10-32: ADC Offset Trim Register n ([0:15]) (ADCOFFSET) Layout	368
Table 10-33: ADC Offset Trim Register n ([0:15]) (ADCOFFSET) Description	368
Table 10-34: ADC Gain Trim Register n ([0:15]) (ADCGAIN) Layout	368
Table 10-35: ADC Gain Trim Register n ([0:15]) (ADCGAIN) Description.....	369
Table 10-36: ADC SH0 Offset Trim Register (SHOFFSET) Layout	369
Table 10-37: ADC SH0 Offset Trim Register (SHOFFSET) Description.....	369
Table 10-38: ADC SH0 Gain Trim Register (SHGAIN) Layout	369
Table 10-39: ADC SH0 Gain Trim Register (SHGAIN) Description.....	369
Table 10-40: ADC Sample Hold Channel EOC Flag Register (SHFLG) Layout	370
Table 10-41: ADC Sample Hold Channel EOC Flag Register (SHFLG) Description.....	370
Table 10-42: ADC Sample Hold Channel EOC Flag Clear Register (SHFLGCLR) Layout.....	370
Table 10-43: ADC Sample Hold Channel EOC Flag Clear Register (SHFLGCLR) Description	370
Table 10-44: ADC Control Register (ADCCTL) Layout.....	371
Table 10-45: ADC Control Register (ADCCTL) Description	371
Table 10-46: ADC SH0 Raw Code Register (SHRAWCODE) Layout	372
Table 10-47: ADC SH0 Raw Code Register (SHRAWCODE) Description.....	372
Table 10-48: ADC Result Register n ([0:15]) (ADCRESULT) Layout	373
Table 10-49: ADC Result Register n ([0:15]) (ADCRESULT) Description.....	373
Table 10-50: PPU Comparison Result Register n ([0:2]) (PPURESULT) Layout	373
Table 10-51: PPU Comparison Result Register n ([0:2]) (PPURESULT) Description.....	373
Table 10-52: PPU SOC Delay Register n ([0:2]) (PPUSOCDLY) Layout.....	374
Table 10-53: PPU SOC Delay Register n ([0:2]) (PPUSOCDLY) Description	374
Table 10-54: PPU Interrupt Raw Flag Register n ([0:2]) (PPURAWIF) Layout.....	374
Table 10-55: PPU Interrupt Raw Flag Register n ([0:2]) (PPURAWIF) Description	374
Table 10-56: PPU Interrupt Flag Register n ([0:2]) (PPUIF) Layout.....	375
Table 10-57: PPU Interrupt Flag Register n ([0:2]) (PPUIF) Description	375
Table 10-58: PPU Interrupt Clear Register n ([0:2]) (PPUIC) Layout.....	376
Table 10-59: PPU Interrupt Clear Register n ([0:2]) (PPUIC) Description	376

Table 10-60: PPU Interrupt Enable Register n ([0:2]) (PPUIE) Layout	376
Table 10-61: PPU Interrupt Enable Register n ([0:2]) (PPUIE) Description.....	377
Table 10-62: PPU Trip-Zone Event Enable Register n ([0:2]) (PPUTZE) Layout	377
Table 10-63: PPU Trip-Zone Event Enable Register n ([0:2]) (PPUTZE) Description.....	377
Table 10-64: PPU Control Register n ([0:2]) (PPUCTL) Layout	378
Table 10-65: PPU Control Register n ([0:2]) (PPUCTL) Description	378
Table 10-66: PPU Reference Register n ([0:2]) (PPUREF) Layout	378
Table 10-67: PPU Reference Register n ([0:2]) (PPUREF) Description.....	379
Table 10-68: PPU Trip-Zone Low-Side Threshold Register n ([0:2]) (PPUTHL) Layout	379
Table 10-69: PPU Trip-Zone Low-Side Threshold Register n ([0:2]) (PPUTHL) Description.....	379
Table 10-70: PPU Trip-Zone High-Side Threshold Register n ([0:2]) (PPUTHH) Layout.....	379
Table 10-71: PPU Trip-Zone High-Side Threshold Register n ([0:2]) (PPUTHH) Description	379
Table 10-72: Temperature Sensor Control Register (TSENSCTL) Layout.....	380
Table 10-73: Temperature Sensor Control Register (TSENSCTL) Description	380
Table 10-74: ADC Register Write-Allow Key Register (ADCREGKEY) Layout	380
Table 10-75: ADC Register Write-Allow Key Register (ADCREGKEY) Description.....	381
Table 11-1: 13-bit Monitoring ADC monitor items	383
Table 11-2: Parameters to derive the measured temperature	384
Table 12-1: SPGA MUX selection	385
Table 12-2: SPGA gain selection	386
Table 12-3: PGA input range in different gain for single-ended mode	386
Table 13-1: DPGA gain and differential input range.....	387
Table 13-2: PGA Module Base Address.....	390
Table 13-3: PGA Register Map	390
Table 13-4: Differential PGA Control Register (DPGACTL) Layout.....	391
Table 13-5: Differential PGA Control Register (DPGACTL) Description	391
Table 13-6: Single-ended PGA Control Register (SPGACTL) Layout.....	392
Table 13-7: Single-ended PGA Control Register (SPGACTL) Description	392
Table 13-8: PGA Register Write-Allow Key Register (PGAREGKEY) Layout	393
Table 13-9: PGA Register Write-Allow Key Register (PGAREGKEY) Description.....	393
Table 14-1: Comparator MUX selection	395
Table 14-2: Comparator Module Base Address.....	396
Table 14-3: COMP Register Map.....	396
Table 14-4: Comparator Filter Output Register (COMPFLTOUT) Layout.....	397
Table 14-5: Comparator Filter Output Register (COMPFLTOUT) Description	397
Table 14-6: Comparator Flag Register (COMPFLG) Layout.....	397
Table 14-7: Comparator Flag Register (COMPFLG) Description	398
Table 14-8: Comparator Flag Clear Register (COMPFLGCLR) Layout.....	398
Table 14-9: Comparator Flag Clear Register (COMPFLGCLR) Description	398
Table 14-10: Comparator 0 Control Register (COMPCTL) Layout.....	399
Table 14-11: Comparator 0 Control Register (COMPCTL) Description	399
Table 14-12: COMP0L Control Register (COMPLCTL) Layout	400
Table 14-13: COMP0L Control Register (COMPLCTL) Description	400
Table 14-14: COMP0H Control Register (COMPHCTL) Layout.....	401
Table 14-15: COMP0H Control Register (COMPHCTL) Description	401
Table 14-16: DAC0 Control Register (DAC0CTL) Layout	403
Table 14-17: DAC0 Control Register (DAC0CTL) Description	403
Table 14-18: DAC0 Code Register (DAC0CODE) Layout.....	404

Table 14-19: DAC0 Code Register (DAC0CODE) Description	404
Table 14-20: DAC0 Active Code Register (DAC0CODEA) Layout.....	404
Table 14-21: DAC0 Active Code Register (DAC0CODEA) Description	404
Table 14-22: RAMP0 Delay Shadow Register (RAMP0DLY) Layout	405
Table 14-23: RAMP0 Delay Shadow Register (RAMP0DLY) Description.....	405
Table 14-24: RAMP0 Delay Active Register (RAMP0DLYA) Layout.....	405
Table 14-25: RAMP0 Delay Active Register (RAMP0DLYA) Description	405
Table 14-26: RAMP0 Decrement Shadow Register (RAMP0DEC) Layout.....	406
Table 14-27: RAMP0 Decrement Shadow Register (RAMP0DEC) Description	406
Table 14-28: RAMP0 Decrement Active Register (RAMP0DECA) Layout	406
Table 14-29: RAMP0 Decrement Active Register (RAMP0DECA) Description.....	406
Table 14-30: RAMP0 Reload Value Shadow Register (RAMP0MAX) Layout	407
Table 14-31: RAMP0 Reload Value Shadow Register (RAMP0MAX) Description.....	407
Table 14-32: RAMP0 Reload Value Active Register (RAMP0MAXA) Layout.....	407
Table 14-33: RAMP0 Reload Value Active Register (RAMP0MAXA) Description	407
Table 14-34: RAMP0 Count Register (RAMP0CNT) Layout.....	408
Table 14-35: RAMP0 Count Register (RAMP0CNT) Description	408
Table 14-36: DAC1 Control Register (DAC1CTL) Layout	408
Table 14-37: DAC1 Control Register (DAC1CTL) Description	408
Table 14-38: DAC1 Code Register (DAC1CODE) Layout.....	409
Table 14-39: DAC1 Code Register (DAC1CODE) Description	409
Table 14-40: DAC1 Active Code Register (DAC1CODEA) Layout.....	410
Table 14-41: DAC1 Active Code Register (DAC1CODEA) Description	410
Table 14-42: RAMP1 Delay Shadow Register (RAMP1DLY) Layout	410
Table 14-43: RAMP1 Delay Shadow Register (RAMP1DLY) Description.....	410
Table 14-44: RAMP1 Delay Active Register (RAMP1DLYA) Layout.....	411
Table 14-45: RAMP1 Delay Active Register (RAMP1DLYA) Description	411
Table 14-46: RAMP1 Decrement Shadow Register (RAMP1DEC) Layout.....	411
Table 14-47: RAMP1 Decrement Shadow Register (RAMP1DEC) Description	411
Table 14-48: RAMP1 Decrement Active Register (RAMP1DECA) Layout	412
Table 14-49: RAMP1 Decrement Active Register (RAMP1DECA) Description.....	412
Table 14-50: RAMP1 Reload Value Shadow Register (RAMP1MAX) Layout	412
Table 14-51: RAMP1 Reload Value Shadow Register (RAMP1MAX) Description.....	412
Table 14-52: RAMP1 Reload Value Active Register (RAMP1MAXA) Layout.....	413
Table 14-53: RAMP1 Reload Value Active Register (RAMP1MAXA) Description	413
Table 14-54: RAMP1 Count Register (RAMP1CNT) Layout.....	413
Table 14-55: RAMP1 Count Register (RAMP1CNT) Description	413
Table 14-56: DAC2 Control Register (DAC2CTL) Layout	414
Table 14-57: DAC2 Control Register (DAC2CTL) Description	414
Table 14-58: DAC2 Code Register (DAC2CODE) Layout	414
Table 14-59: DAC2 Code Register (DAC2CODE) Description	415
Table 14-60: DAC2 Active Code Register (DAC2CODEA) Layout.....	415
Table 14-61: DAC2 Active Code Register (DAC2CODEA) Description	415
Table 14-62: DAC Buffer Control Register (DACBUFCTL) Layout.....	415
Table 14-63: DAC Buffer Control Register (DACBUFCTL) Description	416
Table 14-64: COMP Register Write-Allow Key Register (COMPREGKEY) Layout.....	416
Table 14-65: COMP Register Write-Allow Key Register (COMPREGKEY) Description	416
Table 15-1: Phase Comparator Module Base Address	421

Table 15-2: PHCOMP Register Map	421
Table 15-3: Phase Comparison Status Register (PHCOMPSTS) Layout.....	422
Table 15-4: Phase Comparison Status Register (PHCOMPSTS) Description	422
Table 15-5: Phase Comparison Interrupt Raw Flag Register (PHCOMPRAWIF) Layout	423
Table 15-6: Phase Comparison Interrupt Raw Flag Register (PHCOMPRAWIF) Description.....	423
Table 15-7: Phase Comparison Interrupt Flag Register (PHCOMPIF) Layout	424
Table 15-8: Phase Comparison Interrupt Flag Register (PHCOMPIF) Description.....	424
Table 15-9: Phase Comparison Interrupt Clear Register (PHCOMPIC) Layout	425
Table 15-10: Phase Comparison Interrupt Clear Register (PHCOMPIC) Description.....	425
Table 15-11: Phase Comparison Interrupt Enable Register (PHCOMPIE) Layout.....	426
Table 15-12: Phase Comparison Interrupt Enable Register (PHCOMPIE) Description	426
Table 15-13: Phase Comparison Control Register (PHCOMPCTL) Layout	427
Table 15-14: Phase Comparison Control Register (PHCOMPCTL) Description.....	427
Table 15-15: Phase Comparison Blanking Control (PHCOMPBLANK) Layout.....	429
Table 15-16: Phase Comparison Blanking Control (PHCOMPBLANK) Description	429
Table 15-17: Phase Comparison Filter Control (PHCOMPFILTER) Layout	429
Table 15-18: Phase Comparison Filter Control (PHCOMPFILTER) Description.....	429
Table 15-19: Phase Comparison FSM Reset Register (PHCOMPRST) Layout	430
Table 15-20: Phase Comparison FSM Reset Register (PHCOMPRST) Description.....	430
Table 15-21: Phase Comparison Register Write-Allow Key Register (PHCOMPREGKEY) Layout	431
Table 15-22: Phase Comparison Register Write-Allow Key Register (PHCOMPREGKEY) Description	431
Table 16-1: DAC buffer MUX.....	433
Table 19-1: UART signal descriptions.....	440
Table 19-2: UART Module Base Address	452
Table 19-3: UART Register Map	452
Table 19-4: UART Data Register (UARTDAT) Layout.....	453
Table 19-5: UART Data Register (UARTDAT) Description	453
Table 19-6: UART Control Register (UARTCTL) Layout	453
Table 19-7: UART Control Register (UARTCTL) Description.....	454
Table 19-8: LIN Control Register (LINCTL) Layout.....	456
Table 19-9: LIN Control Register (LINCTL) Description	456
Table 19-10: LIN ID Register (LINID) Layout.....	458
Table 19-11: LIN ID Register (LINID) Description	458
Table 19-12: LIN ID Filter Register (LINIDFILT) Layout.....	459
Table 19-13: LIN ID Filter Register (LINIDFILT) Description	459
Table 19-14: Bus Idle Threshold Register (BUSIDLETH) Layout	459
Table 19-15: Bus Idle Threshold Register (BUSIDLETH) Description	459
Table 19-16: UART Baud Count Register (UARTBDCNT) Layout.....	460
Table 19-17: UART Baud Count Register (UARTBDCNT) Description	460
Table 19-18: UART Auto-Baud Count Register (UARTABCNT) Layout.....	460
Table 19-19: UART Auto-Baud Count Register (UARTABCNT) Description	460
Table 19-20: UART Timeout Threshold Register (UARTTOTH) Layout	461
Table 19-21: UART Timeout Threshold Register (UARTTOTH) Description.....	461
Table 19-22: UART Receive FIFO Level Register (UARTRXLVL) Layout	461
Table 19-23: UART Receive FIFO Level Register (UARTRXLVL) Description.....	462
Table 19-24: UART Transmit FIFO Level Register (UARTTXLVL) Layout.....	462
Table 19-25: UART Transmit FIFO Level Register (UARTTXLVL) Description	462

Table 19-26: UART Receive FIFO Threshold Register (UARTRXTH) Layout.....	462
Table 19-27: UART Receive FIFO Threshold Register (UARTRXTH) Description	462
Table 19-28: UART Transmit FIFO Threshold Register (UARTTXTH) Layout	463
Table 19-29: UART Transmit FIFO Threshold Register (UARTTXTH) Description	463
Table 19-30: UART Infrared Control Register (UARTSIRCTL) Layout	463
Table 19-31: UART Infrared Control Register (UARTSIRCTL) Description.....	463
Table 19-32: UART Status Register (UARTSTS) Layout	464
Table 19-33: UART Status Register (UARTSTS) Description	465
Table 19-34: UART Interrupt Raw Flag Register (UARTRAWIF) Layout	466
Table 19-35: UART Interrupt Raw Flag Register (UARTRAWIF) Description.....	466
Table 19-36: UART Interrupt Flag Register (UARTIF) Layout	468
Table 19-37: UART Interrupt Flag Register (UARTIF) Description	468
Table 19-38: UART Interrupt Clear Register (UARTIC) Layout	471
Table 19-39: UART Interrupt Clear Register (UARTIC) Description	471
Table 19-40: UART Interrupt Enable Register (UARTIE) Layout.....	472
Table 19-41: UART Interrupt Enable Register (UARTIE) Description	472
Table 19-42: UART RXD Deglitch Control (UARTDGCTL) Layout.....	474
Table 19-43: UART RXD Deglitch Control (UARTDGCTL) Description	474
Table 19-44: Uart Register Write-Allow Key Register (UARTREGKEY) Layout	474
Table 19-45: Uart Register Write-Allow Key Register (UARTREGKEY) Description	474
Table 20-1: SPI signal description	476
Table 20-2: SPI Module Base Address.....	491
Table 20-3: SPI Register Map	491
Table 20-4: SPI Data Register (SPIDATA) Layout.....	492
Table 20-5: SPI Data Register (SPIDATA) Description	492
Table 20-6: SPI Control Register 0 (SPICTL0) Layout.....	492
Table 20-7: SPI Control Register 0 (SPICTL0) Description	492
Table 20-8: SPI Control Register 1 (SPICTL1) Layout.....	494
Table 20-9: SPI Control Register 1 (SPICTL1) Description	495
Table 20-10: SPI Status Register (SPISTS) Layout.....	495
Table 20-11: SPI Status Register (SPISTS) Description	495
Table 20-12: SPI Receive FIFO Level Register (SPIRFLVL) Layout.....	497
Table 20-13: SPI Receive FIFO Level Register (SPIRFLVL) Description	497
Table 20-14: SPI Transmit FIFO Level Register (SPITFLVL) Layout	497
Table 20-15: SPI Transmit FIFO Level Register (SPITFLVL) Description	498
Table 20-16: SPI Receive FIFO Threshold Register (SPIRFTH) Layout	498
Table 20-17: SPI Receive FIFO Threshold Register (SPIRFTH) Description	498
Table 20-18: SPI Transmit FIFO Threshold Register (SPITFTH) Layout	498
Table 20-19: SPI Transmit FIFO Threshold Register (SPITFTH) Description	498
Table 20-20: SPI Interrupt Raw Flag Register (SPIRAWIF) Layout	499
Table 20-21: SPI Interrupt Raw Flag Register (SPIRAWIF) Description	499
Table 20-22: SPI Interrupt Flag Register (SPIIF) Layout	500
Table 20-23: SPI Interrupt Flag Register (SPIIF) Description.....	500
Table 20-24: SPI Interrupt Clear Register (SPIIC) Layout	501
Table 20-25: SPI Interrupt Clear Register (SPIIC) Description.....	501
Table 20-26: SPI Interrupt Enable Register (SPIIE) Layout.....	502
Table 20-27: SPI Interrupt Enable Register (SPIIE) Description	502
Table 20-28: SPI Time Out Register (SPITO) Layout.....	503

Table 20-29: SPI Time Out Register (SPITO) Description	503
Table 20-30: SPI Programmable Serial Protocol Register (SPIPSP) Layout.....	503
Table 20-31: SPI Programmable Serial Protocol Register (SPIPSP) Description	503
Table 20-32: SPI Test Control Register (SPITCTL) Layout.....	505
Table 20-33: SPI Test Control Register (SPITCTL) Description	505
Table 20-34: SPI Baud Count Register (SPIBDCNT) Layout	505
Table 20-35: SPI Baud Count Register (SPIBDCNT) Description	506
Table 20-36: SPI Register Write-Allow Key Register (SPIREGKEY) Layout	506
Table 20-37: SPI Register Write-Allow Key Register (SPIREGKEY) Description.....	506
Table 21-1: I2C signal description	507
Table 21-2: I2C Module Base Address	512
Table 21-3: I2C Register Map.....	512
Table 21-4: I2C Control Register (I2CCTL) Layout	513
Table 21-5: I2C Control Register (I2CCTL) Description	513
Table 21-6: I2C Master Address Register (I2CMADDR) Layout.....	515
Table 21-7: I2C Master Address Register (I2CMADDR) Description	515
Table 21-8: I2C Slave Address Register (I2CSADDR) Layout	516
Table 21-9: I2C Slave Address Register (I2CSADDR) Description.....	516
Table 21-10: I2C Data and Command Register (I2CDATACMD) Layout	516
Table 21-11: I2C Data and Command Register (I2CDATACMD) Description.....	517
Table 21-12: Standard Speed I2C Clock SCL Low Count Register (I2CSSLCNT) Layout.....	517
Table 21-13: Standard Speed I2C Clock SCL Low Count Register (I2CSSLCNT) Description	517
Table 21-14: Standard Speed I2C Clock SCL High Count Register (I2CSSHCNT) Layout	518
Table 21-15: Standard Speed I2C Clock SCL High Count Register (I2CSSHCNT) Description.....	518
Table 21-16: Fast Speed I2C Clock SCL Low Count Register (I2CFSLCNT) Layout.....	518
Table 21-17: Fast Speed I2C Clock SCL Low Count Register (I2CFSLCNT) Description	519
Table 21-18: Fast Speed I2C Clock SCL High Count Register (I2CFSHCNT) Layout	519
Table 21-19: Fast Speed I2C Clock SCL High Count Register (I2CFSHCNT) Description.....	519
Table 21-20: High Speed I2C Clock SCL Low Count Register (I2CHSLCNT) Layout	520
Table 21-21: High Speed I2C Clock SCL Low Count Register (I2CHSLCNT) Description	520
Table 21-22: High Speed I2C Clock SCL High Count Register (I2CHSHCNT) Layout.....	520
Table 21-23: High Speed I2C Clock SCL High Count Register (I2CHSHCNT) Description	521
Table 21-24: Fast Speed I2C Spike Suppresion Limit Register (I2CFSSPKLEN) Layout.....	521
Table 21-25: Fast Speed I2C Spike Suppresion Limit Register (I2CFSSPKLEN) Description	521
Table 21-26: High Speed I2C Spike Suppresion Limit Register (I2CHSSPKLEN) Layout	522
Table 21-27: High Speed I2C Spike Suppresion Limit Register (I2CHSSPKLEN) Description.....	522
Table 21-28: I2C SDA Setup Time Register (I2CSDASETUP) Layout	522
Table 21-29: I2C SDA Setup Time Register (I2CSDASETUP) Description	523
Table 21-30: I2C SDA Hold Time Register (I2CSDAHOLD) Layout.....	523
Table 21-31: I2C SDA Hold Time Register (I2CSDAHOLD) Description	523
Table 21-32: I2C Transmit FIFO Level Register (I2CTXLVL) Layout	524
Table 21-33: I2C Transmit FIFO Level Register (I2CTXLVL) Description.....	524
Table 21-34: I2C Receive FIFO Level Register (I2CRXLVL) Layout.....	524
Table 21-35: I2C Receive FIFO Level Register (I2CRXLVL) Description	524
Table 21-36: I2C Transmit FIFO Threshold Register (I2CTXTH) Layout.....	525
Table 21-37: I2C Transmit FIFO Threshold Register (I2CTXTH) Description	525
Table 21-38: I2C Receive FIFO Threshold Register (I2CRXTH) Layout	525
Table 21-39: I2C Receive FIFO Threshold Register (I2CRXTH) Description.....	525

Table 21-40: I2C Status Register (I2CSTS) Layout	526
Table 21-41: I2C Status Register (I2CSTS) Description.....	526
Table 21-42: I2C Transmit Abort Source Register (I2CTXABRTSRC) Layout.....	527
Table 21-43: I2C Transmit Abort Source Register (I2CTXABRTSRC) Description	527
Table 21-44: I2C Interrupt Raw Flag Register (I2CRAWIF) Layout.....	530
Table 21-45: I2C Interrupt Raw Flag Register (I2CRAWIF) Description	530
Table 21-46: I2C Interrupt Flag Register (I2CIF) Layout.....	532
Table 21-47: I2C Interrupt Flag Register (I2CIF) Description	532
Table 21-48: I2C Interrupt Enable Register (I2CIE) Layout	534
Table 21-49: I2C Interrupt Enable Register (I2CIE) Description.....	534
Table 21-50: I2C Interrupt Enable Register (I2CIC) Layout	535
Table 21-51: I2C Interrupt Enable Register (I2CIC) Description	535
Table 21-52: I2C Register Write-Allow Key Register (I2CREGKEY) Layout.....	537
Table 21-53: I2C Register Write-Allow Key Register (I2CREGKEY) Description	537
Table 22-1: CAN I/O signal	540
Table 22-2: Mailbox control bits	547
Table 22-3: Mailbox control bits setting for message transmission.....	548
Table 22-4: Mailbox control bits setting for message transmission with remote frame response ..	549
Table 22-5: Mailbox control bits setting to receive message	549
Table 22-6: Mailbox control bits setting to receive message with remote frame response.....	550
Table 22-7: CAN Module Base Address	551
Table 22-8: CAN Register Map	551
Table 22-9: CAN Control Register (CANCTL) Layout	552
Table 22-10: CAN Control Register (CANCTL) Description.....	552
Table 22-11: CAN Nominal Bit Timing Quantum Register (CANNBTQ) Layout	554
Table 22-12: CAN Nominal Bit Timing Quantum Register (CANNBTQ) Description.....	554
Table 22-13: CAN Data Bit Timing Quantum Register (CANDBTQ) Layout.....	555
Table 22-14: CAN Data Bit Timing Quantum Register (CANDBTQ) Description	555
Table 22-15: CAN Transmitter Delay Compensation Register (CANTDC) Layout	556
Table 22-16: CAN Transmitter Delay Compensation Register (CANTDC) Description	556
Table 22-17: CAN Bus-Off Recovery Delay Register (CANBRDLY) Layout.....	556
Table 22-18: CAN Bus-Off Recovery Delay Register (CANBRDLY) Description	556
Table 22-19: CAN Status Register (CANSTS) Layout	557
Table 22-20: CAN Status Register (CANSTS) Description.....	557
Table 22-21: CAN Last Error Code Register (CANLEC) Layout.....	558
Table 22-22: CAN Last Error Code Register (CANLEC) Description	558
Table 22-23: CAN Miscellaneous Interrupt Enable Register (CANMIE) Layout	559
Table 22-24: CAN Miscellaneous Interrupt Enable Register (CANMIE) Description	559
Table 22-25: CAN Miscellaneous Interrupt Raw Flag Register (CANMRAWIF) Layout.....	560
Table 22-26: CAN Miscellaneous Interrupt Raw Flag Register (CANMRAWIF) Description	560
Table 22-27: CAN Miscellaneous Interrupt Flag Register (CANMIF) Layout.....	561
Table 22-28: CAN Miscellaneous Interrupt Flag Register (CANMIF) Description	562
Table 22-29: CAN Miscellaneous Interrupt Flag Clear Register (CANMIC) Layout.....	563
Table 22-30: CAN Miscellaneous Interrupt Flag Clear Register (CANMIC) Description	563
Table 22-31: CAN Mailbox Group Transfer Done Interrupt Flag Register (CANTGIF) Layout	564
Table 22-32: CAN Mailbox Group Transfer Done Interrupt Flag Register (CANTGIF) Description....	564
Table 22-33: CAN Mailbox Group Transfer Done Interrupt Clear Register (CANTGIC) Layout	566
Table 22-34: CAN Mailbox Group Transfer Done Interrupt Clear Register (CANTGIC) Description..	566

Table 22-35: CAN Memory Error Address Register (CANMEMERRADDR) Layout.....	567
Table 22-36: CAN Memory Error Address Register (CANMEMERRADDR) Description	567
Table 22-37: CAN Test Control Register (CANTEST) Layout.....	567
Table 22-38: CAN Test Control Register (CANTEST) Description	567
Table 22-39: CAN Error Counter Control Register (CANECCTL) Layout	568
Table 22-40: CAN Error Counter Control Register (CANECCTL) Description	568
Table 22-41: CAN Transmission Request Register (CANTXREQ) Layout.....	569
Table 22-42: CAN Transmission Request Register (CANTXREQ) Description	569
Table 22-43: CAN New Message Register (CANMSGNEW) Layout.....	569
Table 22-44: CAN New Message Register (CANMSGNEW) Description	569
Table 22-45: CAN Mailbox Transfer Done Interrupt Flag Register (CANMSGIF) Layout	570
Table 22-46: CAN Mailbox Transfer Done Interrupt Flag Register (CANMSGIF) Description.....	570
Table 22-47: CAN Mailbox Enable Register (CANMBEN) Layout	570
Table 22-48: CAN Mailbox Enable Register (CANMBEN) Description	570
Table 22-49: CAN Timestamp Control Register (CANTSCTL) Layout	571
Table 22-50: CAN Timestamp Control Register (CANTSCTL) Description.....	571
Table 22-51: CAN Timestamp Counter Register (CANTSCNT) Layout	571
Table 22-52: CAN Timestamp Counter Register (CANTSCNT) Description.....	571
Table 22-53: CAN Register Write-Allow Key Register (CANREGKEY) Layout	572
Table 22-54: CAN Register Write-Allow Key Register (CANREGKEY) Description	572
Table 22-55: CANMBOX Module Base Address	573
Table 22-56: CANMBOX Register Map.....	573
Table 22-57: CAN Mailbox Data Word Register n (CANMBOXFDW) Layout	573
Table 22-58: CAN Mailbox Data Word Register n (CANMBOXFDW) Description.....	573
Table 22-59: CAN Mailbox Frame Mask Register (CANMBOXFMSK) Layout.....	574
Table 22-60: CAN Mailbox Frame Mask Register (CANMBOXFMSK) Description	574
Table 22-61: CAN Mailbox Frame Identifier Register (CANMBOXFID) Layout	575
Table 22-62: CAN Mailbox Frame Identifier Register (CANMBOXFID) Description.....	575
Table 22-63: CAN Mailbox Message Timestamp Register (CANMBOXMTS) Layout	575
Table 22-64: CAN Mailbox Message Timestamp Register (CANMBOXMTS) Description.....	575
Table 22-65: CAN Mailbox Message Control Register (CANMBOXMCTL) Layout	576
Table 22-66: CAN Mailbox Message Control Register (CANMBOXMCTL) Description.....	576
Table 23-1: CRC Module Base Address	580
Table 23-2: CRC Register Map.....	580
Table 23-3: CRC Control Register (CRCCTL) Layout.....	581
Table 23-4: CRC Control Register (CRCCTL) Description	581
Table 23-5: CRC Stream Length Register (CRCSTRLEN) Layout.....	582
Table 23-6: CRC Stream Length Register (CRCSTRLEN) Description	582
Table 23-7: CRC Seed Register (CRCSEED) Layout	583
Table 23-8: CRC Seed Register (CRCSEED) Description.....	583
Table 23-9: CRC Stream Input Register (CRCSTRIN) Layout.....	583
Table 23-10: CRC Stream Input Register (CRCSTRIN) Description	583
Table 23-11: CRC Result Register (CRCRESULT) Layout	584
Table 23-12: CRC Result Register (CRCRESULT) Description	584
Table 23-13: CRC Interrupt Raw Flag Register (CRCRAWIF) Layout	584
Table 23-14: CRC Interrupt Raw Flag Register (CRCRAWIF) Description.....	584
Table 23-15: CRC Interrupt Flag Register (CRCIF) Layout	585
Table 23-16: CRC Interrupt Flag Register (CRCIF) Description.....	585

Table 23-17: CRC Interrupt Clear Register (CRCIC) Layout	586
Table 23-18: CRC Interrupt Clear Register (CRCIC) Description.....	586
Table 23-19: CRC Interrupt Enable Register (CRCIE) Layout.....	587
Table 23-20: CRC Interrupt Enable Register (CRCIE) Description	587
Table 23-21: CRC Register Write-Allow Key Register (CRCREGKEY) Layout	587
Table 23-22: CRC Register Write-Allow Key Register (CRCREGKEY) Description.....	587
Table 24-1: Flash module organization.....	589
Table 24-2: Flash memory read access latency	590
Table 24-3: Description of the configuration words.....	596
Table 24-4: Flash Module Base Address	597
Table 24-5: FLASHC Register Map.....	597
Table 24-6: Flash Control Register (FLASHCTL) Layout.....	598
Table 24-7: Flash Control Register (FLASHCTL) Description	598
Table 24-8: Flash Address Register (FLASHADDR) Layout	599
Table 24-9: Flash Address Register (FLASHADDR) Description.....	599
Table 24-10: Flash Data Input Low Word Register (FLASHDINL) Layout	599
Table 24-11: Flash Data Input Low Word Register (FLASHDINL) Description.....	599
Table 24-12: Flash Data Input High Word Register (FLASHDINH) Layout.....	600
Table 24-13: Flash Data Input High Word Register (FLASHDINH) Description	600
Table 24-14: Flash Data Input ECC Byte Register (FLASHDINE) Layout	600
Table 24-15: Flash Data Input ECC Byte Register (FLASHDINE) Description.....	600
Table 24-16: Flash Data Input Pattern Register (FLASHDINP) Layout	601
Table 24-17: Flash Data Input Pattern Register (FLASHDINP) Description.....	601
Table 24-18: Flash Data Output Low Word Register (FLASHDOUTL) Layout.....	601
Table 24-19: Flash Data Output Low Word Register (FLASHDOUTL) Description	601
Table 24-20: Flash Data Output High Word Register (FLASHDOUTH) Layout	602
Table 24-21: Flash Data Output High Word Register (FLASHDOUTH) Description	602
Table 24-22: Flash Data Output ECC Byte Register (FLASHDOUTE) Layout.....	602
Table 24-23: Flash Data Output ECC Byte Register (FLASHDOUTE) Description	602
Table 24-24: Flash Data Output Pattern Register (FLASHDOUTP) Layout.....	603
Table 24-25: Flash Data Output Pattern Register (FLASHDOUTP) Description	603
Table 24-26: Flash Write Protect Register (FLASHWP) Layout	603
Table 24-27: Flash Write Protect Register (FLASHWP) Description	603
Table 24-28: Flash Controller Register Write-Allow Key Register (FLASHCREGKEY) Layout	606
Table 24-29: Flash Controller Register Write-Allow Key Register (FLASHCREGKEY) Description.....	606
Table 25-1: SYSTEM Module Base Address.....	609
Table 25-2: SYSTEM Register Map	609
Table 25-3: Chip ID Register 0 (CID0) Layout	611
Table 25-4: Chip ID Register 0 (CID0) Description	611
Table 25-5: Chip ID Register 1 (CID1) Layout	611
Table 25-6: Chip ID Register 1 (CID1) Description	611
Table 25-7: Unique ID Register 0 (UID0) Layout	612
Table 25-8: Unique ID Register 0 (UID0) Description	612
Table 25-9: Unique ID Register 1 (UID1) Layout	612
Table 25-10: Unique ID Register 1 (UID1) Description	612
Table 25-11: Random Number Register 0 (RND0) Layout	613
Table 25-12: Random Number Register 0 (RND0) Description	613
Table 25-13: Random Number Register 1 (RND1) Layout	613

Table 25-14: Random Number Register 1 (RND1) Description	613
Table 25-15: Revision Information Register 0 (REV0) Layout	614
Table 25-16: Revision Information Register 0 (REV0) Description	614
Table 25-17: Revision Information Register 1 (REV1) Layout	614
Table 25-18: Revision Information Register 1 (REV1) Description	614
Table 25-19: System Monitor Control Register (SYSMONCTL) Layout	615
Table 25-20: System Monitor Control Register (SYSMONCTL) Description	615
Table 25-21: VBAT Monitor Threshold Register (VBATTH) Layout	616
Table 25-22: VBAT Monitor Threshold Register (VBATTH) Description	616
Table 25-23: VCP Monitor Threshold Register (VCPTH) Layout	617
Table 25-24: VCP Monitor Threshold Register (VCPTH) Description	617
Table 25-25: VMON Monitor Threshold Register (VMONTH) Layout	617
Table 25-26: VMON Monitor Threshold Register (VMONTH) Description	617
Table 25-27: VDD5 Monitor Threshold Register (VDD5TH) Layout	618
Table 25-28: VDD5 Monitor Threshold Register (VDD5TH) Description	618
Table 25-29: Power Bandgap Monitor Threshold Register (VBGTH) Layout	618
Table 25-30: Power Bandgap Monitor Threshold Register (VBGTH) Description	618
Table 25-31: Power Module Temperature Monitor Threshold Register (TPMUTH) Layout	619
Table 25-32: Power Module Temperature Monitor Threshold Register (TPMUTH) Description	619
Table 25-33: LIN PHY Temperature Monitor Threshold Register (TLINTH) Layout	619
Table 25-34: LIN PHY Temperature Monitor Threshold Register (TLINTH) Description	619
Table 25-35: Controller Temperature Monitor Threshold Register (TMCUTH) Layout	620
Table 25-36: Controller Temperature Monitor Threshold Register (TMCUTH) Description	620
Table 25-37: VDD33 Monitor Threshold Register (VDD33TH) Layout	620
Table 25-38: VDD33 Monitor Threshold Register (VDD33TH) Description	620
Table 25-39: VDD12 Monitor Threshold Register (VDD12TH) Layout	621
Table 25-40: VDD12 Monitor Threshold Register (VDD12TH) Description	621
Table 25-41: 13-bit ADC Reference Voltage Monitor Threshold Register (VADCREFTH) Layout	621
Table 25-42: 13-bit ADC Reference Voltage Monitor Threshold Register (VADCREFTH) Description	621
Table 25-43: VBAT Monitor Code Register (VBATCODE) Layout	622
Table 25-44: VBAT Monitor Code Register (VBATCODE) Description	622
Table 25-45: VCP Monitor Code Register (VCPCODE) Layout	622
Table 25-46: VCP Monitor Code Register (VCPCODE) Description	622
Table 25-47: VMON Monitor Code Register (VMONCODE) Layout	623
Table 25-48: VMON Monitor Code Register (VMONCODE) Description	623
Table 25-49: VDD5 Monitor Code Register (VDD5CODE) Layout	623
Table 25-50: VDD5 Monitor Code Register (VDD5CODE) Description	623
Table 25-51: Power Bandgap Monitor Code Register (VBGCODE) Layout	624
Table 25-52: Power Bandgap Monitor Code Register (VBGCODE) Description	624
Table 25-53: Power Module Temperature Monitor Code Register (TPMUCODE) Layout	624
Table 25-54: Power Module Temperature Monitor Code Register (TPMUCODE) Description	624
Table 25-55: LIN PHY Temperature Monitor Code Register (TLINCODE) Layout	625
Table 25-56: LIN PHY Temperature Monitor Code Register (TLINCODE) Description	625
Table 25-57: Controller Temperature Monitor Code Register (TMCUCODE) Layout	625
Table 25-58: Controller Temperature Monitor Code Register (TMCUCODE) Description	625
Table 25-59: VDD33 Monitor Code Register (VDD33CODE) Layout	626
Table 25-60: VDD33 Monitor Code Register (VDD33CODE) Description	626

Table 25-61: VDD12 Monitor Threshold Register (VDD12CODE) Layout	626
Table 25-62: VDD12 Monitor Threshold Register (VDD12CODE) Description	626
Table 25-63: 13-bit ADC Reference Voltage Monitor Code Register (VADCREFCODE) Layout	627
Table 25-64: 13-bit ADC Reference Voltage Monitor Code Register (VADCREFCODE) Description	627
Table 25-65: GPIO Wake-Up Control Register (GPIOWKUPCTL) Layout	627
Table 25-66: GPIO Wake-Up Control Register (GPIOWKUPCTL) Description	627
Table 25-67: GPIO Non-Maskable Interrupt Control Register (GPIONMCTL) Layout	628
Table 25-68: GPIO Non-Maskable Interrupt Control Register (GPIONMCTL) Description	628
Table 25-69: Non-Maskable Interrupt Raw Flag Register (NMRAWIF) Layout	629
Table 25-70: Non-Maskable Interrupt Raw Flag Register (NMRAWIF) Description	629
Table 25-71: Non-Maskable Interrupt Flag Register (NMIF) Layout	631
Table 25-72: Non-Maskable Interrupt Flag Register (NMIF) Description	631
Table 25-73: Non-Maskable Interrupt Clear Register (NMIC) Layout	634
Table 25-74: Non-Maskable Interrupt Clear Register (NMIC) Description	634
Table 25-75: Non-Maskable Interrupt Enable Register (NMIE) Layout	637
Table 25-76: Non-Maskable Interrupt Enable Register (NMIE) Description	637
Table 25-77: Memory Interrupt Raw Flag Register (MEMRAWIF) Layout	639
Table 25-78: Memory Interrupt Raw Flag Register (MEMRAWIF) Description	639
Table 25-79: Memory Interrupt Flag Register (MEMIF) Layout	640
Table 25-80: Memory Interrupt Flag Register (MEMIF) Description	640
Table 25-81: Memory Interrupt Clear Register (MEMIC) Layout	641
Table 25-82: Memory Interrupt Clear Register (MEMIC) Description	641
Table 25-83: Memory Interrupt Enable Register (MEMIE) Layout	641
Table 25-84: Memory Interrupt Enable Register (MEMIE) Description	642
Table 25-85: Reset Event Flag Register (RSTEVTF LG) Layout	642
Table 25-86: Reset Event Flag Register (RSTEVTF LG) Description	642
Table 25-87: Reset Event Flag Clear Register (RSTEVTC LR) Layout	644
Table 25-88: Reset Event Flag Clear Register (RSTEVTC LR) Description	644
Table 25-89: Reset Event Enable Register (RSTEVTE N) Layout	645
Table 25-90: Reset Event Enable Register (RSTEVTE N) Description	645
Table 25-91: System Information Register (SYSINFO) Layout	646
Table 25-92: System Information Register (SYSINFO) Description	647
Table 25-93: ROM ECC Error Address (ROMERRADDR) Layout	647
Table 25-94: ROM ECC Error Address (ROMERRADDR) Description	648
Table 25-95: FLASH ECC Error Address (FLASHERRADDR) Layout	648
Table 25-96: FLASH ECC Error Address (FLASHERRADDR) Description	648
Table 25-97: RAM0 ECC Error Address (RAM0ERRADDR) Layout	648
Table 25-98: RAM0 ECC Error Address (RAM0ERRADDR) Description	648
Table 25-99: RAM1 ECC Error Address (RAM1ERRADDR) Layout	649
Table 25-100: RAM1 ECC Error Address (RAM1ERRADDR) Description	649
Table 25-101: Debug Interface Control Register (DBGIFCTL) Layout	649
Table 25-102: Debug Interface Control Register (DBGIFCTL) Description	649
Table 25-103: Boot Information Register (BOOTINFO) Layout	650
Table 25-104: Boot Information Register (BOOTINFO) Description	650
Table 25-105: AHB Multi-Bit ECC Error Event Flag Register (AHBMBEFLG) Layout	650
Table 25-106: AHB Multi-Bit ECC Error Event Flag Register (AHBMBEFLG) Description	651
Table 25-107: AHB Multi-bit ECC Error Event Flag Register (AHBMBECLR) Layout	653
Table 25-108: AHB Multi-bit ECC Error Event Flag Register (AHBMBECLR) Description	653

Table 25-109: AHB Single-Bit ECC Error Event Flag Register (AHBSBEFLG) Layout	655
Table 25-110: AHB Single-Bit ECC Error Event Flag Register (AHBSBEFLG) Description	656
Table 25-111: AHB Single-bit ECC Error Event Flag Register (AHBSBECLR) Layout	658
Table 25-112: AHB Single-bit ECC Error Event Flag Register (AHBSBECLR) Description.....	658
Table 25-113: Peripheral Reset Event Flag Register (PERIRSTFLG) Layout.....	660
Table 25-114: Peripheral Reset Event Flag Register (PERIRSTFLG) Description	661
Table 25-115: Peripheral Reset Event Clear Register (PERIRSTCLR) Layout.....	662
Table 25-116: Peripheral Reset Event Clear Register (PERIRSTCLR) Description	662
Table 25-117: Non-Volatile Register 0 (NVREG0) Layout	663
Table 25-118: Non-Volatile Register 0 (NVREG0) Description.....	664
Table 25-119: Non-Volatile Register 1 (NVREG1) Layout	664
Table 25-120: Non-Volatile Register 1 (NVREG1) Description.....	664
Table 25-121: Non-Volatile Register 2 (NVREG2) Layout	664
Table 25-122: Non-Volatile Register 2 (NVREG2) Description.....	664
Table 25-123: System Register Write-Allow Key Register (SYSREGKEY) Layout.....	665
Table 25-124: System Register Write-Allow Key Register (SYSREGKEY) Description	665
Table 26-1: DMA hardware handshaking interface for UART, SPI and ADC.....	668
Table 26-2: DMACH Module Base Address.....	676
Table 26-3: DMACH Register Map	676
Table 26-4: DMA Channel Source Address Register (DMACHSA) Layout.....	677
Table 26-5: DMA Channel Source Address Register (DMACHSA) Description	677
Table 26-6: DMA Channel Destination Address Register (DMACHDA) Layout.....	677
Table 26-7: DMA Channel Destination Address Register (DMACHDA) Description	677
Table 26-8: DMA Channel Control Register 0 (DMACHCTL0) Layout	678
Table 26-9: DMA Channel Control Register 0 (DMACHCTL0) Description.....	678
Table 26-10: DMA Channel Control Register 1 (DMACHCTL1) Layout	680
Table 26-11: DMA Channel Control Register 1 (DMACHCTL1) Description.....	681
Table 26-12: DMA Channel Configure Register 0 (DMACHCFG0) Layout.....	681
Table 26-13: DMA Channel Configure Register 0 (DMACHCFG0) Description	681
Table 26-14: DMA Channel Configure Register 1 (DMACHCFG1) Layout.....	683
Table 26-15: DMA Channel Configure Register 1 (DMACHCFG1) Description	683
Table 26-16: DMAC Module Base Address	685
Table 26-17: DMAC Register Map.....	685
Table 26-18: DMA Transfer Complete Interrupt Raw Flag Register (DMATCRAWIF) Layout.....	687
Table 26-19: DMA Transfer Complete Interrupt Raw Flag Register (DMATCRAWIF) Description	687
Table 26-20: DMA Block Transfer Complete Interrupt Raw Flag Register (DMABTCRAWIF) Layout	688
Table 26-21: DMA Block Transfer Complete Interrupt Raw Flag Register (DMABTCRAWIF) Description	688
Table 26-22: DMA Source Transfer Complete Interrupt Raw Flag Register (DMASTCRAWIF) Layout	689
Table 26-23: DMA Source Transfer Complete Interrupt Raw Flag Register (DMASTCRAWIF) Description	690
Table 26-24: DMA Destination Transfer Complete Interrupt Raw Flag Register (DMADTCRAWIF) Layout.....	691
Table 26-25: DMA Destination Transfer Complete Interrupt Raw Flag Register (DMADTCRAWIF) Description	691
Table 26-26: DMA Error Interrupt Raw Flag Register (DMAERRRAWIF) Layout	692
Table 26-27: DMA Error Interrupt Raw Flag Register (DMAERRRAWIF) Description.....	692

Table 26-28: DMA Transfer Complete Interrupt Flag Register (DMATCIF) Layout.....	693
Table 26-29: DMA Transfer Complete Interrupt Flag Register (DMATCIF) Description	694
Table 26-30: DMA Block Transfer Complete Interrupt Flag Register (DMABTCIF) Layout.....	694
Table 26-31: DMA Block Transfer Complete Interrupt Flag Register (DMABTCIF) Description	695
Table 26-32: DMA Source Transfer Complete Interrupt Flag Register (DMASTCIF) Layout.....	695
Table 26-33: DMA Source Transfer Complete Interrupt Flag Register (DMASTCIF) Description	696
Table 26-34: DMA Destination Transfer Complete Interrupt Flag Register (DMADTCIF) Layout	696
Table 26-35: DMA Destination Transfer Complete Interrupt Flag Register (DMADTCIF) Description	697
Table 26-36: DMA Error Interrupt Flag Register (DMAERRIF) Layout	697
Table 26-37: DMA Error Interrupt Flag Register (DMAERRIF) Description.....	698
Table 26-38: DMA Transfer Complete Interrupt Enable Register (DMATCIE) Layout	698
Table 26-39: DMA Transfer Complete Interrupt Enable Register (DMATCIE) Description	699
Table 26-40: DMA Block Transfer Complete Interrupt Enable Register (DMABTCIE) Layout	700
Table 26-41: DMA Block Transfer Complete Interrupt Enable Register (DMABTCIE) Description....	700
Table 26-42: DMA Source Transfer Complete Interrupt Enable Register (DMASTCIE) Layout	702
Table 26-43: DMA Source Transfer Complete Interrupt Enable Register (DMASTCIE) Description..	702
Table 26-44: DMA Destination Transfer Complete Interrupt Enable Register (DMADTCIE) Layout .	703
Table 26-45: DMA Destination Transfer Complete Interrupt Enable Register (DMADTCIE) Description	704
Table 26-46: DMA Error Interrupt Enable Register (DMAERRIE) Layout.....	705
Table 26-47: DMA Error Interrupt Enable Register (DMAERRIE) Description	705
Table 26-48: DMA Transfer Complete Interrupt Clear Register (DMATCIC) Layout	707
Table 26-49: DMA Transfer Complete Interrupt Clear Register (DMATCIC) Description.....	707
Table 26-50: DMA Block Transfer Complete Interrupt Clear Register (DMABTCIC) Layout.....	708
Table 26-51: DMA Block Transfer Complete Interrupt Clear Register (DMABTCIC) Description	708
Table 26-52: DMA Source Transfer Complete Interrupt Clear Register (DMASTCIC) Layout.....	709
Table 26-53: DMA Source Transfer Complete Interrupt Clear Register (DMASTCIC) Description	710
Table 26-54: DMA Destination Transfer Complete Interrupt Clear Register (DMADTCIC) Layout ...	711
Table 26-55: DMA Destination Transfer Complete Interrupt Clear Register (DMADTCIC) Description	711
Table 26-56: DMA Error Interrupt Clear Register (DMAERRIC) Layout	712
Table 26-57: DMA Error Interrupt Clear Register (DMAERRIC) Description.....	712
Table 26-58: DMA Combined Interrupt Flag Register (DMAIF) Layout	713
Table 26-59: DMA Combined Interrupt Flag Register (DMAIF) Description.....	714
Table 26-60: DMA Source Software Transaction Request Register (DMASSREQ) Layout.....	714
Table 26-61: DMA Source Software Transaction Request Register (DMASSREQ) Description	715
Table 26-62: DMA Destination Software Transaction Request Register (DMADSREQ) Layout	716
Table 26-63: DMA Destination Software Transaction Request Register (DMADSREQ) Description .	716
Table 26-64: DMA Source Software Transaction Single Request Register (DMASSSREQ) Layout	718
Table 26-65: DMA Source Software Transaction Single Request Register (DMASSSREQ) Description	718
Table 26-66: DMA Destination Software Transaction Single Request Register (DMADSSREQ) Layout	719
Table 26-67: DMA Destination Software Transaction Single Request Register (DMADSSREQ) Description	720
Table 26-68: DMA Source Software Transaction Last Request Register (DMASSLREQ) Layout	721
Table 26-69: DMA Source Software Transaction Last Request Register (DMASSLREQ) Description	721

Table 26-70: DMA Destination Software Transaction Last Request Register (DMADSLREQ) Layout	723
Table 26-71: DMA Destination Software Transaction Last Request Register (DMADSLREQ) Description	723
Table 26-72: DMA Enable Register (DMAEN) Layout	724
Table 26-73: DMA Enable Register (DMAEN) Description.....	725
Table 26-74: DMA Channel Enable Register (DMACHEN) Layout	725
Table 26-75: DMA Channel Enable Register (DMACHEN) Description	725

SPINTROL

Revision history

Revision	Date	Author	Changes
A/0	2023-4-14	C. Hu	1. Initial Release
A/1	2023-05-12	C. Hu	2. Update Section 3.6 and Table 3-10 . 3. Update Table 14-3 . 4. Update Figure 3-6 . 5. Update Section 1.3.2 . 6. Update Section 24.7 and Table 24-3 . 7. Update Figure 20-1 . 8. Update Section 0 . 9. Update Table 20-7 . 10. Update Figure 10-10 .
A/2	2023-08-20	C.Chai	11. Modify Section 1.3.2 . 12. Update Figure 10-10 . 13. Update Table 20-29 . 14. Modify Figure 8-2 and Figure 8-6 . 15. Modify Table 4-1 . 16. Modify Table 5-1 . 17. Update Section 8.11.2 . 18. Update Section 4.4.2 .
A/3	2023-11-08	C.Chai	19. Modify Table 20-6 、 Table 20-7 、 Table 19-7 、 Table 4-2 . 20. Modify Section 20.4 . 21. Modify Figure 8-1 . 22. Modify Section 4.2.2 .
A/4	2023-12-05	C.Chai	23. Modify Section 2.2 . 24. Modify Section 11.2 .
A/5	2024-03-08	Jiali Zhou	25. Modify Figure 4-1 : Block diagram of I/O control. 26. Modify Section 19 27. Delete EEPROM 28. Modify Figure 8-14 : Trip-Zone sub-module logic 29. Modify Section 4.3.2,0
C/0	2025-02-08	Jiali Zhou	30. Modify Table 25-2 , Table 25-80 MEMIF description. 31. Modify ADCSTS to SHFLG. 32. Delete invalid bit in register BOOTINFO. 33. Modify register UARTIF, UARTIC, UARTIE. 34. Modify PWMCFG register. 35. Modify Section 4.2.3 . 36. Modify Table 6-2 .

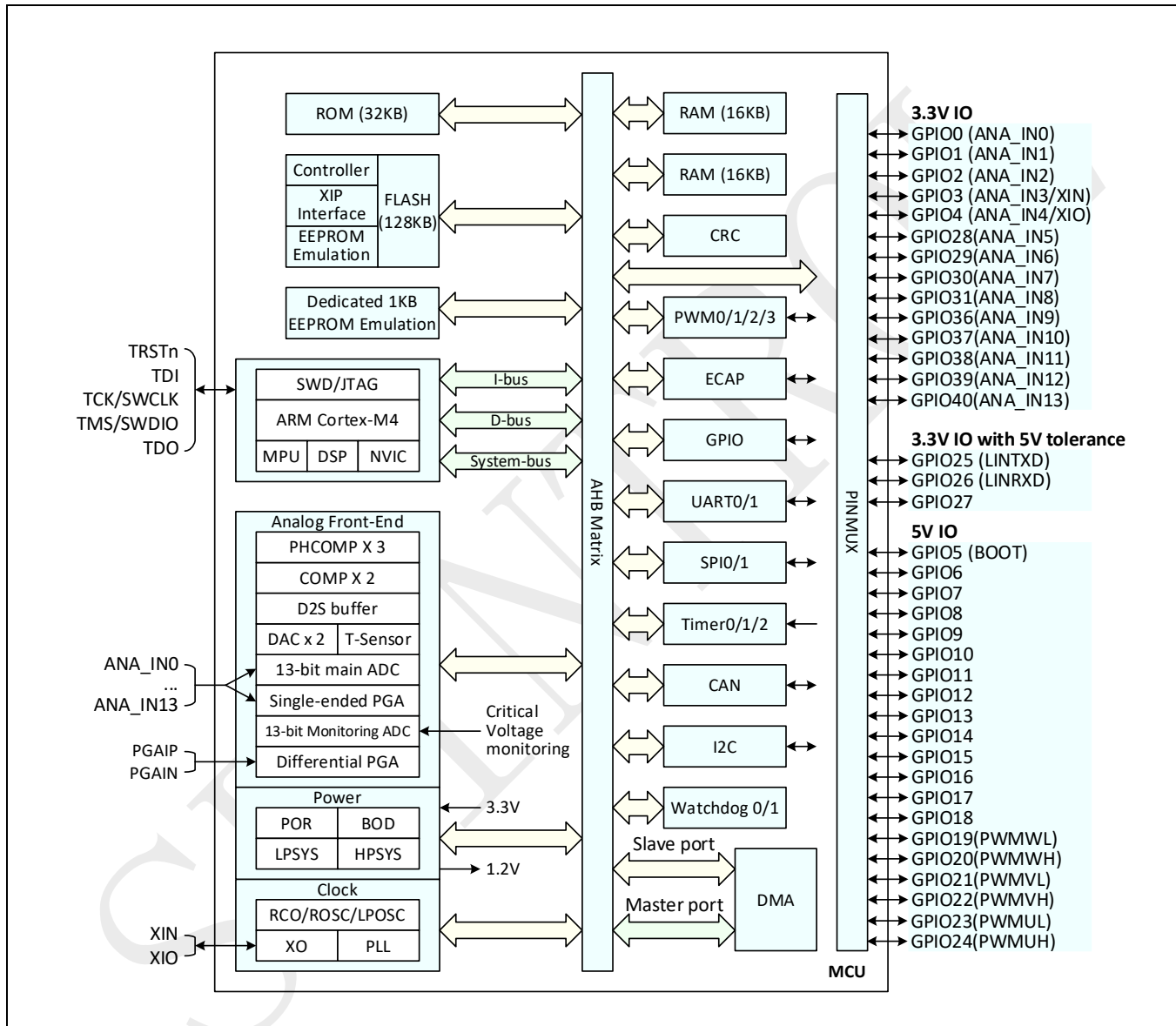
Terms or abbreviations

Terms or abbreviations	Description
MCU	Microcontroller Unit
AHB	Advanced High Performance Bus
DMA	Direct Memory Access
XIP	Execution In Place
SWD	Serial Wire Debug
PLL	Phase Locked Loop
BOD	Brownout Detector
NMI	Non Maskable Interrupt
ISP	In System Programming
PFD	Phase Frequency Detector
NVIC	Nested Vectored Interrupt Controller
UART	Universal Asynchronous Receiver-Transmitter
LIN	Local Interconnect Network
CAN	Controller Area Network
ADC	Analog-to-Digital Converter
DAC	Digital-to-Analog Converter
PGA	Programmable-Gain Amplifier
CRC	Cyclic Redundancy Check
AES	Advanced Encryption Standard
SAR ADC	Successive Approximation A/D Converter

1 Memory and bus architecture

1.1 System architecture

Figure 1-1: Block diagram of SPC1169



As shown in in [Figure 1-1](#), in SPC1169, the main system consists of micro-controller sub-system and peripheral sub-system.

The micro-controller sub-system consists of:

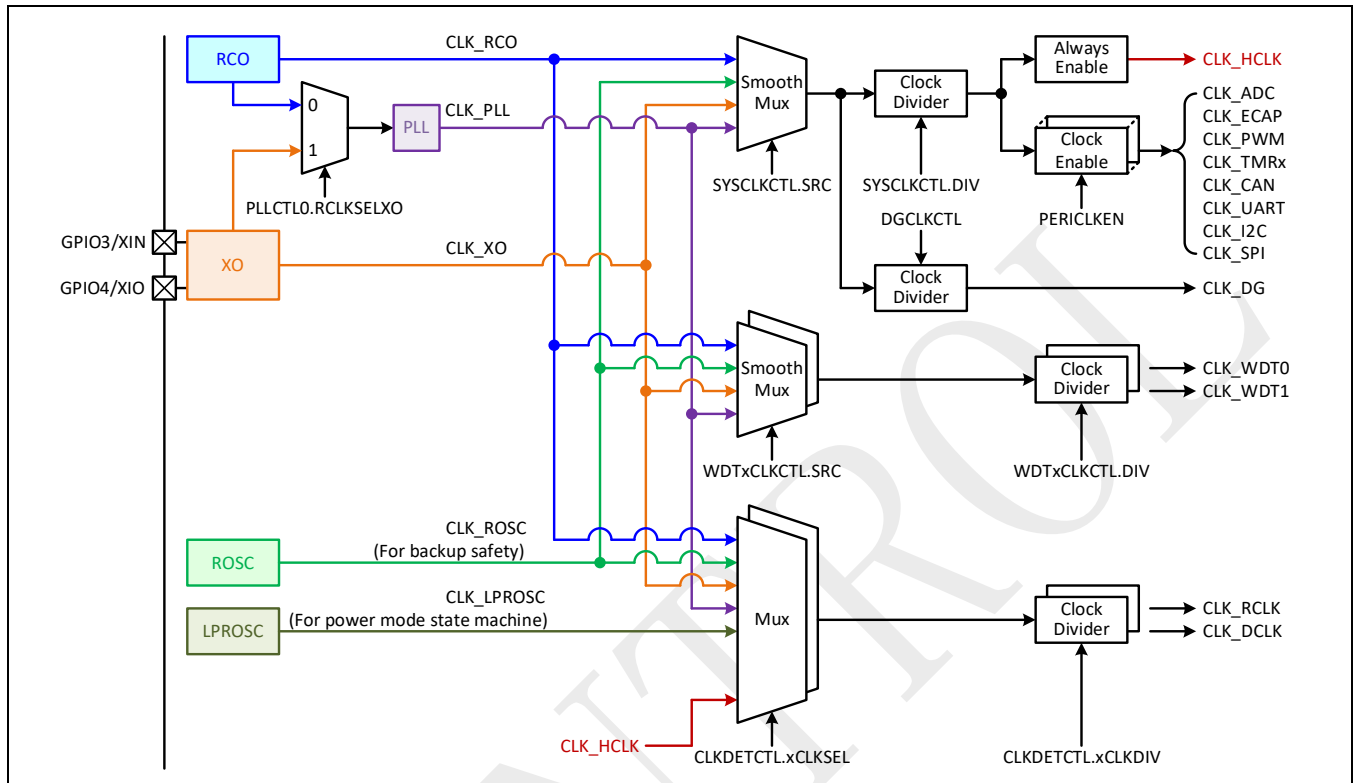
- Power management unit:
 - LDO to generate 1.2V power supply
 - Sleep mode entry and wakeup handling
- 4 masters:

- Cortex-M4 I-bus, D-bus, S-bus, and DMA master port
- 23 slaves:
 - ROM
 - Two RAMs
 - Flash controller and XIP interface
 - Power and Clock registers
 - AHB peripherals: AFE, PWM, ECAP, WDT0/1, TIMER0/1/2, GPIO, SPI0/1, UART0/1, I2C, CAN, CRC and DMA configuration slave port

These are interconnected using a low latency AHB bus architecture as shown in [Figure 1-1](#).

1.2 Clock tree

Figure 1-2: SPC1169 clock tree



As shown in Figure 1-2, there are 5 root clocks in SPC1169:

- RCO: Trimmed at 32MHz
- ROSC: Free running ring oscillator with nominal frequency of 32MHz, which serves as backup safety clock in case RCO is missed.
- XO: External crystal with on-chip active resonator or external clock input (via XIN)
- PLL: Provide flexible on-chip clock up to 100MHz. The reference clock can be selected as RCO or XO clock
- LPROSC: For power mode state machine

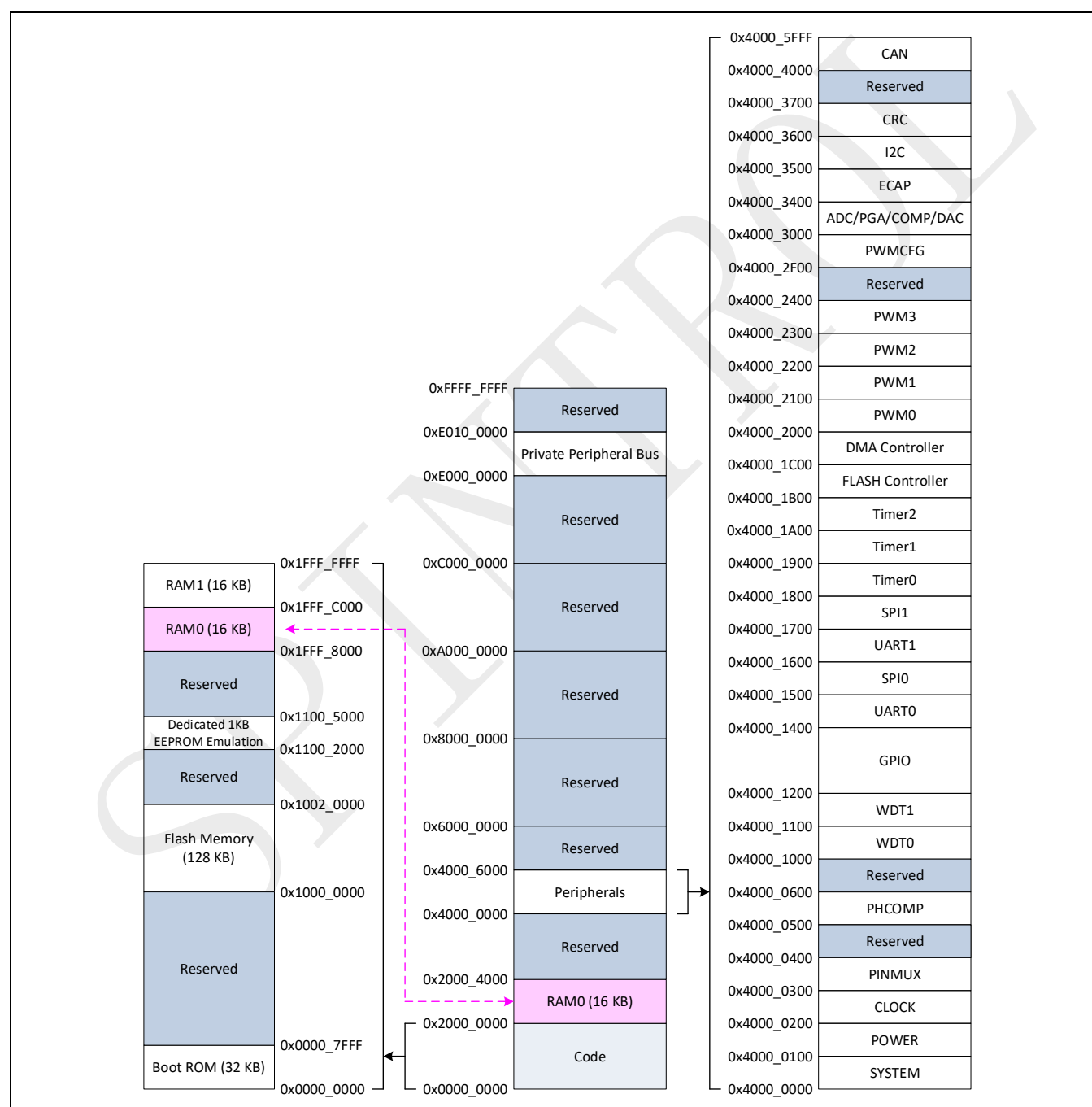
The succeeding smooth MUX provides glitch-free switching. Dedicated clocks are generated for each function building block with independent clock gate and divider.

1.3 Memory map

In SPC1169 device, ROM, RAMs, registers and I/O ports are organized within the same linear 4GB address space. The bytes are coded in memory in Little Endian format.

All the memory areas that are not allocated to on-chip memories and peripherals are considered “Not Used”. Access to these address results in bus fault. The memory map of SPC1169 is shown in [Figure 1-3](#).

Figure 1-3: SPC1169 memory map



Note: The SRAM0 can be mapped to the address range 0x1FFF 8000 ~ 0x1FFF BFFF (accessible to I-bus and D-bus) or 0x2000 0000 ~ 0x2000 3FFF (accessible to S-bus).

Table 1-1 gives the base address of the peripherals available in SPC1169 device.

Table 1-1: Peripheral Module Base Address

Peripheral Module	Base Address	Register Map
SYSTEM	0x4000 0000	Section 25.10.1
POWER	0x4000 0100	Section 2.6.1
CLOCK	0x4000 0200	Section 3.9.1
PINMUX	0x4100 0300	Section 4.4.3
GPIO	0x4000 1200	Section 4.4.1
TIMER0	0x4000 1800	Section 6.4.1
TIMER1	0x4000 1900	Section 6.4.1
TIMER2	0x4000 1A00	Section 6.4.1
WDT0	0x4000 1000	Section 7.4.1
WDT1	0x4000 1100	Section 7.4.1
PWM0	0x4000 2000	Section 8.11.1
PWM1	0x4000 2100	Section 8.11.1
PWM2	0x4000 2200	Section 8.11.1
PWM3	0x4000 2300	Section 8.11.1
PWMCFG	0x4000 2F00	Section 8.11.3
ECAP	0x4000 3400	Section 9.7.1
ADC	0x4000 3000	Section 10.9.1
PGA	0x4000 3000	Section 13.5.1
COMP	0x4000 3000	Section 14.5.1
PHCOMP	0x4000 0500	Section 15.4.1
UART0	0x4000 1400	Section 19.7.1
UART1	0x4000 1600	Section 19.7.1
SPI0	0x4000 1500	Section 20.8.1
SPI1	0x4000 1700	Section 20.8.1
FLASH	0x4000 1B00	Section 24.8.1
DMAC	0x4000 1C00	Section 26.6.3
I2C	0x4000 3500	Section 21.6.1
CRC	0x4000 3600	Section 23.4.1
CAN	0x4000 5400	Section 22.7.1

1.3.1 Embedded Flash Memory

The SPC1169 is integrated with an embedded flash memory up to 128kB for storing program code and data. An XIP (Execution-in-Place) interface is provided to read flash memory as shown in [Figure 1-1](#), while another AHB controller interface is used to erase and program the Flash memory. After the SPC1169 is powered up, any code in the Flash memory can be directly executed, or loaded into RAM memory before running from the RAM.

For the data safety concern, ECC feature is supported for the embedded Flash, which is capable to correct 1-bit error and detect 2-bit errors (SECEDED). When the event is enabled, a non-makable interrupt will be issued to the CPU upon FLASH ECC multi-bit error, and a makable interrupt will be issued to the CPU upon FLASH ECC single-bit error. And the address where the bit error happened is recorded in FLASHERRADDR register for diagnosis.

The Flash memory is organized into 32 programmable sectors with 4096 bytes per sector. The Flash memory can be programmed in 64-bit (double-word) at a time. Data can be erased via 4KB sector erase, or entire chip erase.

1.3.2 Embedded SRAM

The SPC1169 has implemented two SRAM memories for running code and data, which is named as RAM0 and RAM1. Both of them can be accessed in byte, half-word (16 bits) or full word (32 bits) with 0 wait state.

The 16kB RAM1 can be accessed by the I-bus and D-bus of the CPU via 0x1FFF C000 ~ 0x1FFF FFFF.

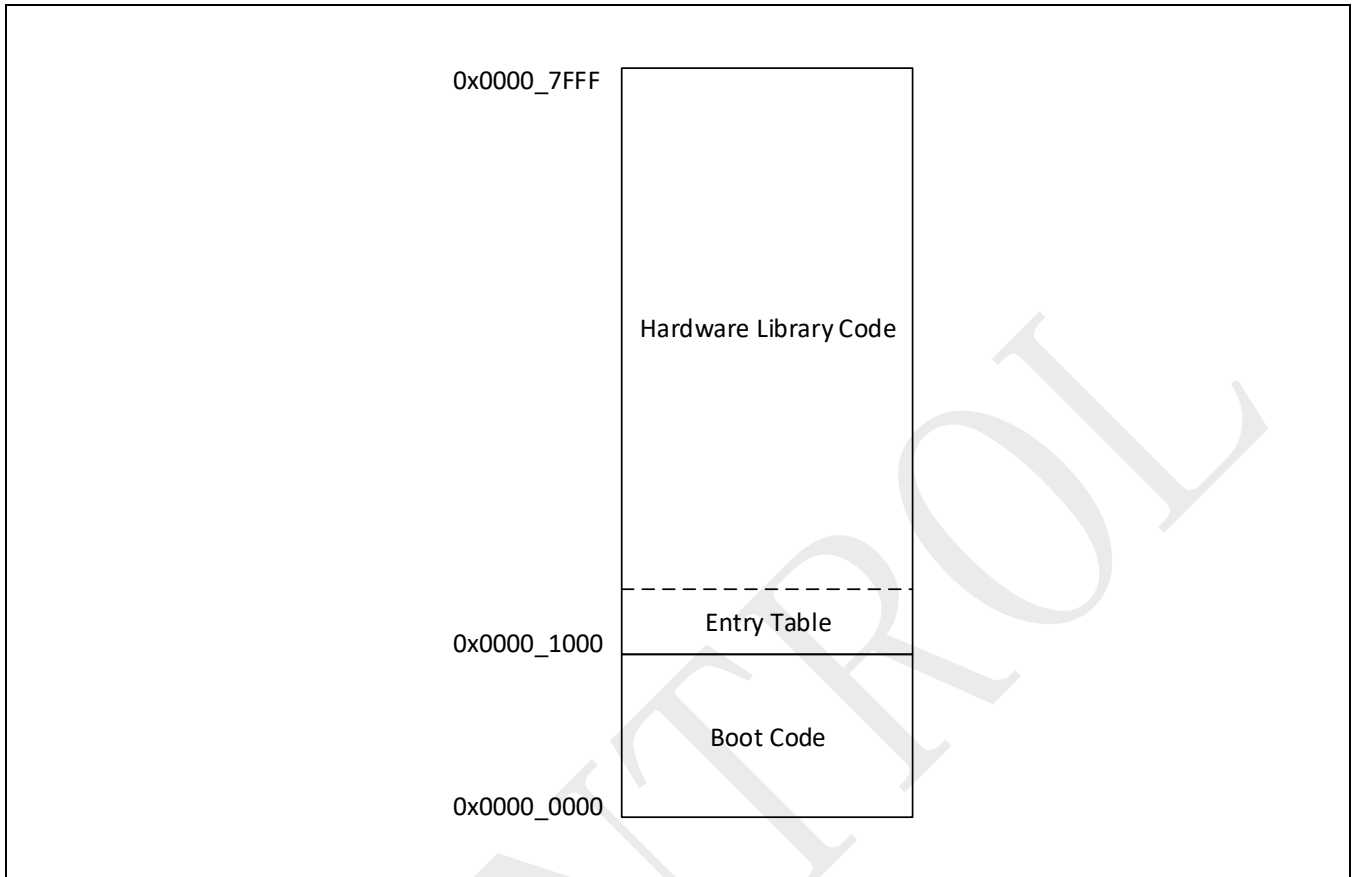
The RAM0 size is up to 16kB and it can be accessed by the I-bus and D-bus of the CPU via 0x1FFF 8000 ~ 0x1FFF BFFF, or by the S-bus of the CPU via 0x2000 0000 ~ 0x2000 3FFF. Different sub products have different RAM0 sizes, such as, 0kB, 4kB (mapped to 0x1FFF B000 ~ 0x1FFF BFFF or 0x2000 3000 ~ 0x2000 3FFF), 8kB (mapped to 0x1FFF A000 ~ 0x1FFF BFFF or 0x2000 2000 ~ 0x2000 3FFF), 12kB (mapped to 0x1FFF 9000 ~ 0x1FFF BFFF or 0x2000 1000 ~ 0x2000 3FFF), or full 16kB.

For the data safety concern, Both RAM0 and RAM1 support SECEDED ECC. When the event is enabled, a non-makable interrupt will be issued to the CPU upon RAM ECC multi-bit error, and a maskable interrupt will be issued to the CPU upon RAM ECC single-bit error. And the address where the bit error happened is recorded in RAM0ERRADDR and RAM1ERRADDR registers for diagnosis.

1.3.3 Boot ROM

The SPC1169 has implemented a 32KB ROM with SECEDED ECC support. By default, the system will reset upon ROM ECC multi-bit error and the address where the bit error happened is recorded in ROMERRADDR register for diagnosis. When enabled, the ROM ECC multi-bit error can also trigger a non-maskable interrupt to the CPU and the ROM ECC single-bit error can trigger a maskable interrupt.

Boot ROM also provides a hardware library entry table located at the address 0x1000. The hardware library entry table consists of function entry points. The function entry point contains the information of function entry address. Through the entry table, user can call the specified function easily. The memory map of Boot ROM is shown in [Figure 1-4](#). The detail information of hardware library entry table is listed in [Table 1-2](#).

Figure 1-4: Boot ROM memory map

Table 1-2: Boot ROM hardware library entry table

Function Entry Point	Address	Description
FLASHC_Init	0x0000 1000	Initialize Flash module configurations
FLASHC_Read	0x0000 1004	Read word
FLASHC_VerifyRead	0x0000 1008	Read to verify whether the content is correct after erase or program. It will not trigger ECC error.
FLASHC_ProgramDWord	0x0000 100C	Program a double word data
FLASHC_Program	0x0000 1010	Program word data
FLASHC_EraseSector	0x0000 1014	Erase flash sector
FLASHC_EraseChip	0x0000 1018	Erase flash chip
FLASHC_VerifyErase	0x0000 101C	Verify whether the content is all 1's after erase. It will not trigger ECC error.
FLASHC_PowerUp	0x0000 1020	Power up the Flash
FLASHC_PowerDown	0x0000 1024	Power down the Flash
FLASHC_WriteProtect	0x0000 1028	Enable write protect for the specified area
FLASHC_WriteUnProtect	0x0000 102C	Disable write protect for the specified area
FLASHC_EnableCache	0x0000 1030	Enable double word cache for Flash XIP
FLASHC_DisableCache	0x0000 1034	Disable double word cache for Flash XIP
FLASHC_SetTiming	0x0000 1038	Set Flash timing parameters

Function Entry Point	Address	Description
-	0x0000 103C	Reserved
-	0x0000 1040	Reserved
-	0x0000 1044	Reserved
-	0x0000 1048	Reserved
-	0x0000 104C	Reserved
-	0x0000 1050	Reserved
-	0x0000 1054	Reserved
-	0x0000 1058	Reserved
-	0x0000 105C	Reserved
SYSTEM_Sleep	0x0000 1060	Whole system sleep
SYSTEM_Stop	0x0000 1064	Whole system stop
MCU_Sleep	0x0000 1068	Only the micro-controller sleep
MCU_Stop	0x0000 106C	Only the micro-controller stop
SYSTEM_CalculateMemCRC	0x0000 1070	Calculate the CRC value of the specified memory area
SYSTEM_CheckCriticalRegister	0x0000 1074	Check critical register reset value
-	0x0000 1078	Reserved
Sin	0x0000 107C	Reserved
Cos	0x0000 1080	Reserved
Sin2	0x0000 1084	Reserved
Cos2	0x0000 1088	Reserved
Atan2	0x0000 108C	Reserved
AtanF32	0x0000 1090	Reserved
FastSqrt	0x0000 1094	Reserved
Sin512	0x0000 1098	Reserved
Cos512	0x0000 109C	Reserved
Atan512	0x0000 10A0	Reserved
ThetaRadWrap	0x0000 10A4	Reserved
SAT0	0x0000 10A8	Reserved
SVPWM	0x0000 10AC	Reserved
ClarkeF32	0x0000 10B0	Reserved
InvParkF32	0x0000 10B4	Reserved
ParkF32	0x0000 10B8	Reserved
BubbleSortForCurrent	0x0000 10BC	Reserved
Motor_1PhaseCurrentReconstructInit	0x0000 10C0	Reserved
Motor_1PhaseCurrentReconstruct	0x0000 10C4	Reserved
Motor_1PhaseCurrentSamplingShift	0x0000 10C8	Reserved

1.4 Boot configuration

In SPC1169, two different boot modes can be selected through BOOT pin (GPIO5):

- Boot from the Flash memory (BOOT): After a reset, the boot loader jumps to the Flash memory and runs from the address 0x1000 0000
- ISP mode (BOOT = High): the boot loader reprograms the Flash memory by using UART or LIN interface. During the process, if UART interface is used, the GPIO10 is configured as UART_TXD and the GPIO11 is configured as UART_RXD.

Table 1-3: Boot modes

BOOT (GPIO5)	Boot Mode
1	ISP mode (If UART interface is used, GPIO10/GPIO11 are configured as UART_TXD/UART_RXD)
0	Normal start mode, boot from Flash memory Note that when TRSTn is high, the debug interface is in active.

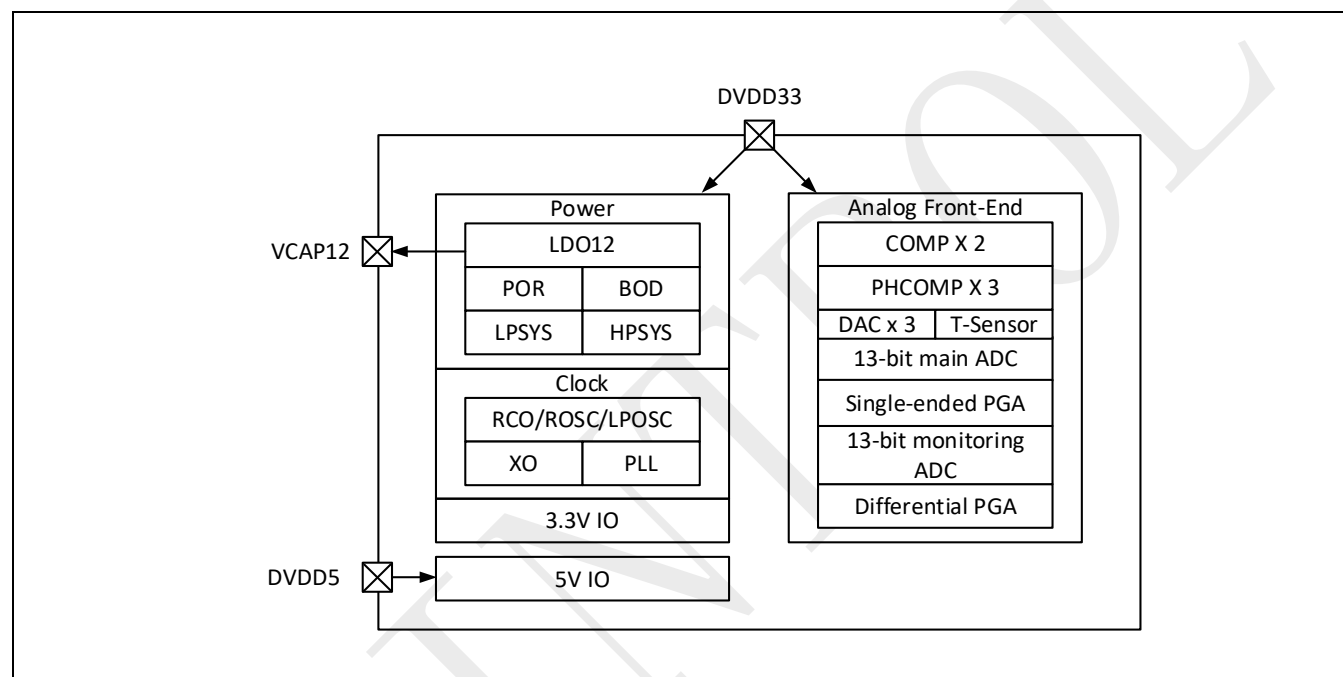
Note: The ISP mode can be entered only when BOOT (GPIO5) is high after power-on reset due to whole chip power re-cycle or pin reset due to X_RSTn pin is asserted low for sufficient time. For other event triggered reset (such as watchdog timeout or memory multi-bit ECC error), the system enters normal start mode and boots from flash memory regardless of the BOOT (GPIO5) pin status.

2 Power

2.1 Power supply description

As shown in [Figure 2-1](#), the SPC1169 implementation requires two power supply(DVDD33 and DVDD5). DVDD5 is only used for 5V IO. DVDD33 is used for internal circuit and 3.3V IO. The on-chip regulator fed by DVDD33 generates 1.2V for MCU and it need the separated pin VCAP12 for outside decoupling cap.

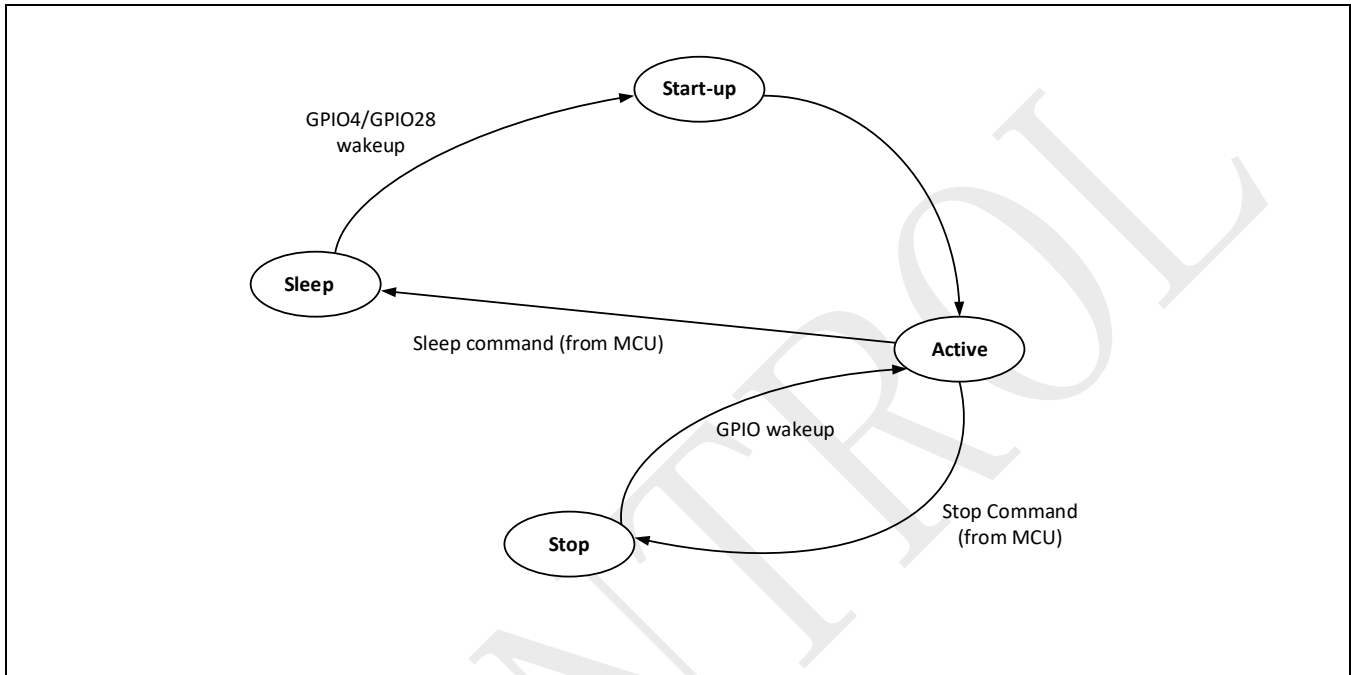
Figure 2-1: Power supply overview



2.2 Power mode management

The power management unit is responsible for generating all the required voltage supply for the embedded MCU(VCAP12). The PMU has a well-defined state machine, to control power mode transition as shown in Figure 2-2.

Figure 2-2: Power mode transition



The 3 power modes provided in SPC1169 are defined as below:

Active mode

The voltage regulators (VCAP12) are supplied by on-chip LDO with full current driving capability. All clocks are enabled per user's setup in register control bits.

Stop mode

All clocks are stopped except for the 128 KHz clock for power mode state machine. VCAP12 are supplied by low-power LDO with limited current driving capability. VCAP12 can be configured to be below 1.2 V to further reduce the static current. All registers and memories retain the content. The system can be woken up by any level on any GPIO pin depending on the configuration. The CPU will continue to run the instructions when it exits stop mode.

Sleep mode

The voltage regulators (VCAP12) are totally shutdown. It can be waken up by GPIO and the whole system will go through a cold start-up.

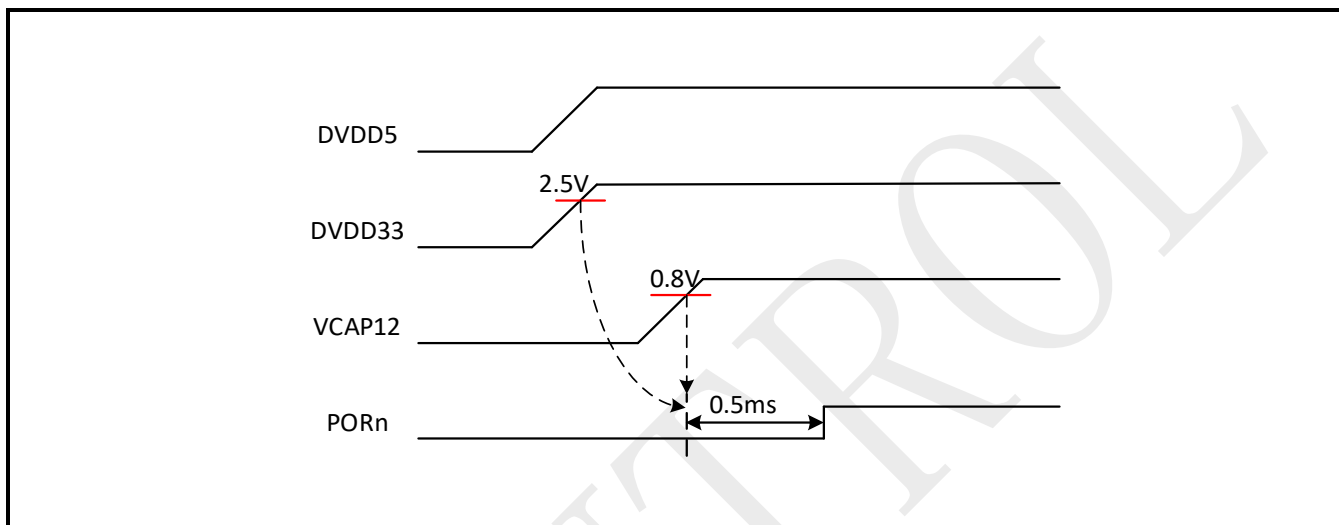
- Note:
1. User should safely disable the unused analog modules before issuing the stop command.
 2. The leakage in stop mode increases dramatically at high temperature, and may exceed the loading capability of the low power LDO. User needs to make sure the junction temperature is below 100°C before issuing the stop command.

2.3 Power-up sequence

SPC1169 power up sequence is shown in [Figure 2-3](#). When DVDD33 rises up to 2.5V and VCAP12 rises up to 0.8V both, after 0.5ms, MCU core logic PORn is released. SPC1169 enters active mode.

There is not any power-up sequence between DVDD5 and DVDD33. Because DVDD5 is only used for 5V IO and it didn't take part in for system power-up sequence.

Figure 2-3: Power up sequence



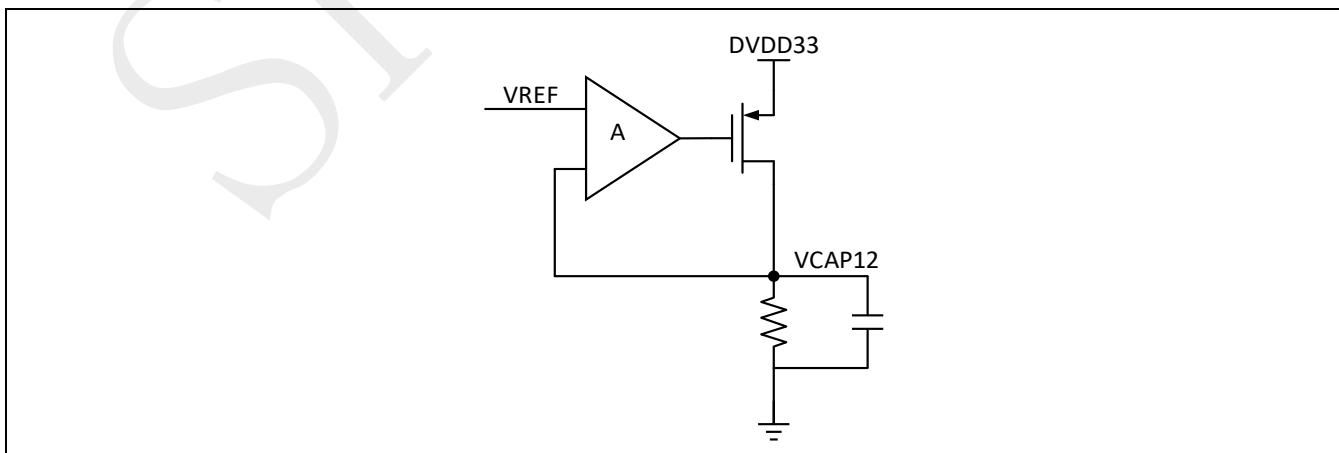
2.3.1 VCAP12 LDO

The VCAP12 LDO provides power for the embedded MCU. The SPC1169 has implemented a VCAP12 LDO module with following features:

- 1.2V LDO with 40mA driving capability

VCAP12 LDO schematic is shown in [Figure 2-4](#).

Figure 2-4: VCAP12 LDO



2.4 Over-voltage/under-voltage detectors

The over-voltage/under-voltage detectors are implemented via brown-out detectors (BOD). The BOD can be used to monitor the various power levels by comparing them to the pre-set thresholds. This section describes low-voltage BOD for DVDD33 and VCAP12.

The VDD33OV/VDD12OV and VDD33UV/VDD12UV flags are available in the NMIF register, to indicate whether the 3.3V/1.2V domain is higher or lower than the pre-set thresholds. The VDD33OV/VDD12OV and VDD33UV/VDD12UV events can generate an interrupt if enabled.

The BOD33CTL and BOD12CTL registers determine how to set the over-voltage and under-voltage thresholds. The NMIE register enables the 3.3V/1.2V domain over-voltage and under-voltage events to trigger NMI interrupts. The RSTEV TEN register enables reset upon the 3.3V/1.2V domain over-voltage and under-voltage events.

Figure 2-5 and Figure 2-6 shows the 3.3V and 1.2V domain brown-out detector structure.

Figure 2-5: 3.3V domain brown-out detector diagram

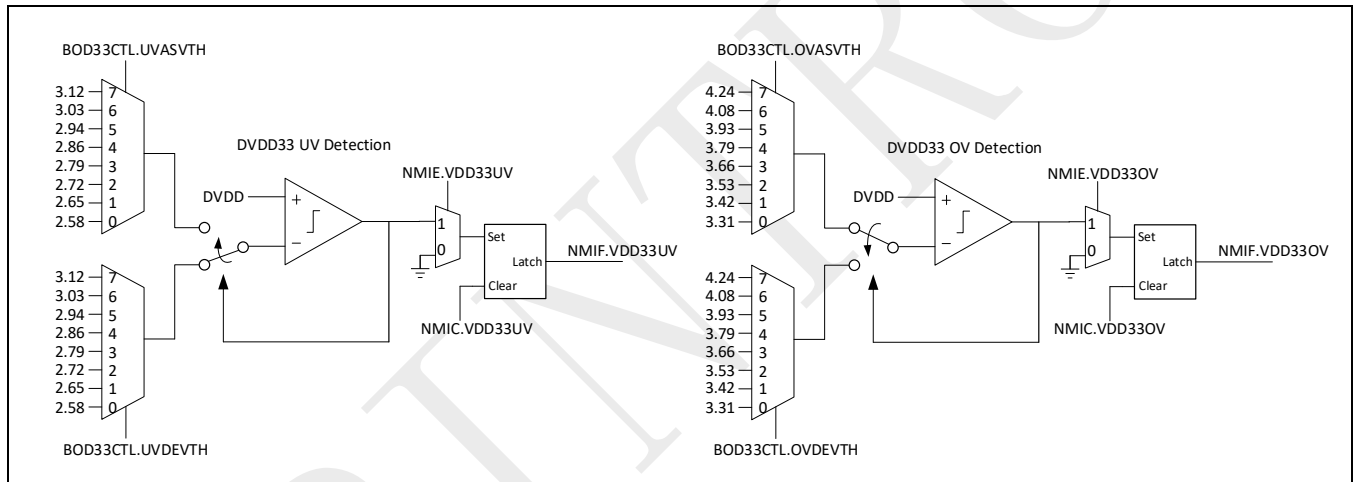
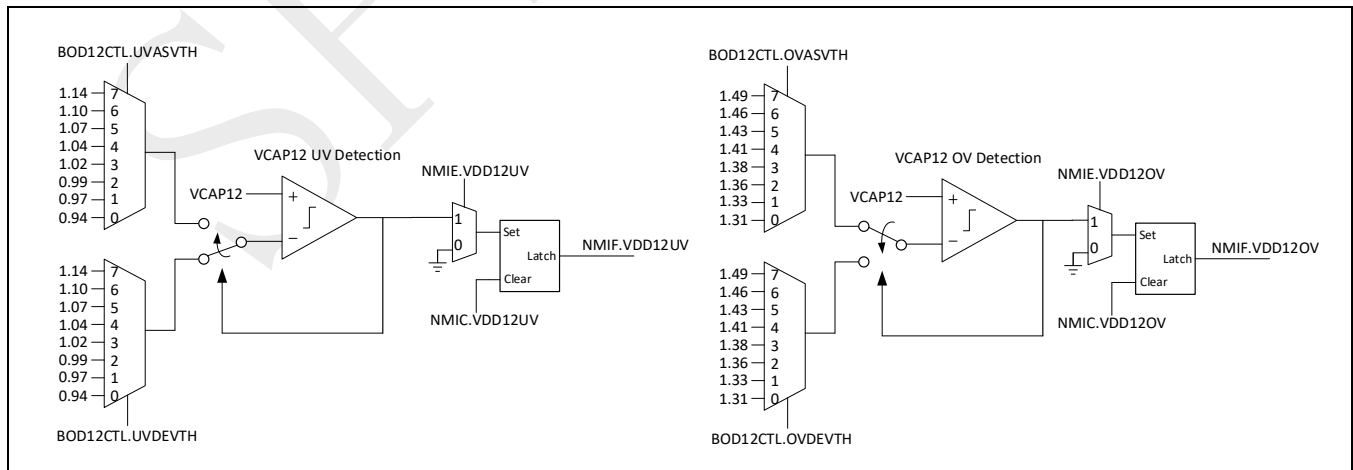


Figure 2-6: 1.2V domain brown-out detector diagram

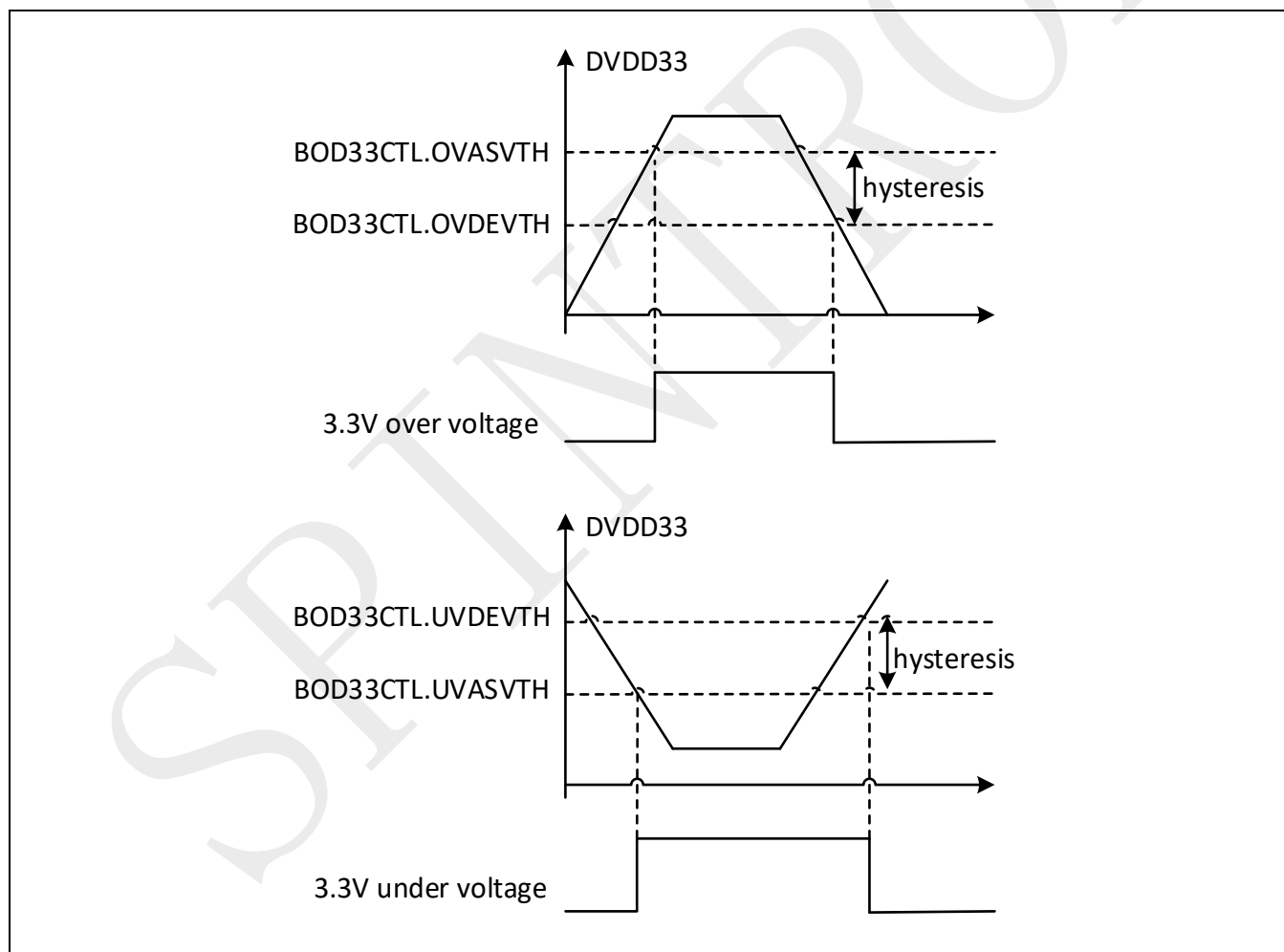


Note: For proper operation of each BOD:

1. In over voltage detection, the 0→1 transition threshold OVASVTH in BOD33CTL register and OVASVTH in BOD12CTL register must be larger than 1→0 transition threshold OVDEVTH in BOD33CTL register and OVDEVTH in BOD12CTL register.
2. In under voltage detection, the 0→1 transition threshold UVASVTH in BOD33CTL register and UVASVTH in BOD12CTL register must be smaller than 1→0 transition threshold UVDEVTH in BOD33CTL register and UVDEVTH in BOD12CTL register.

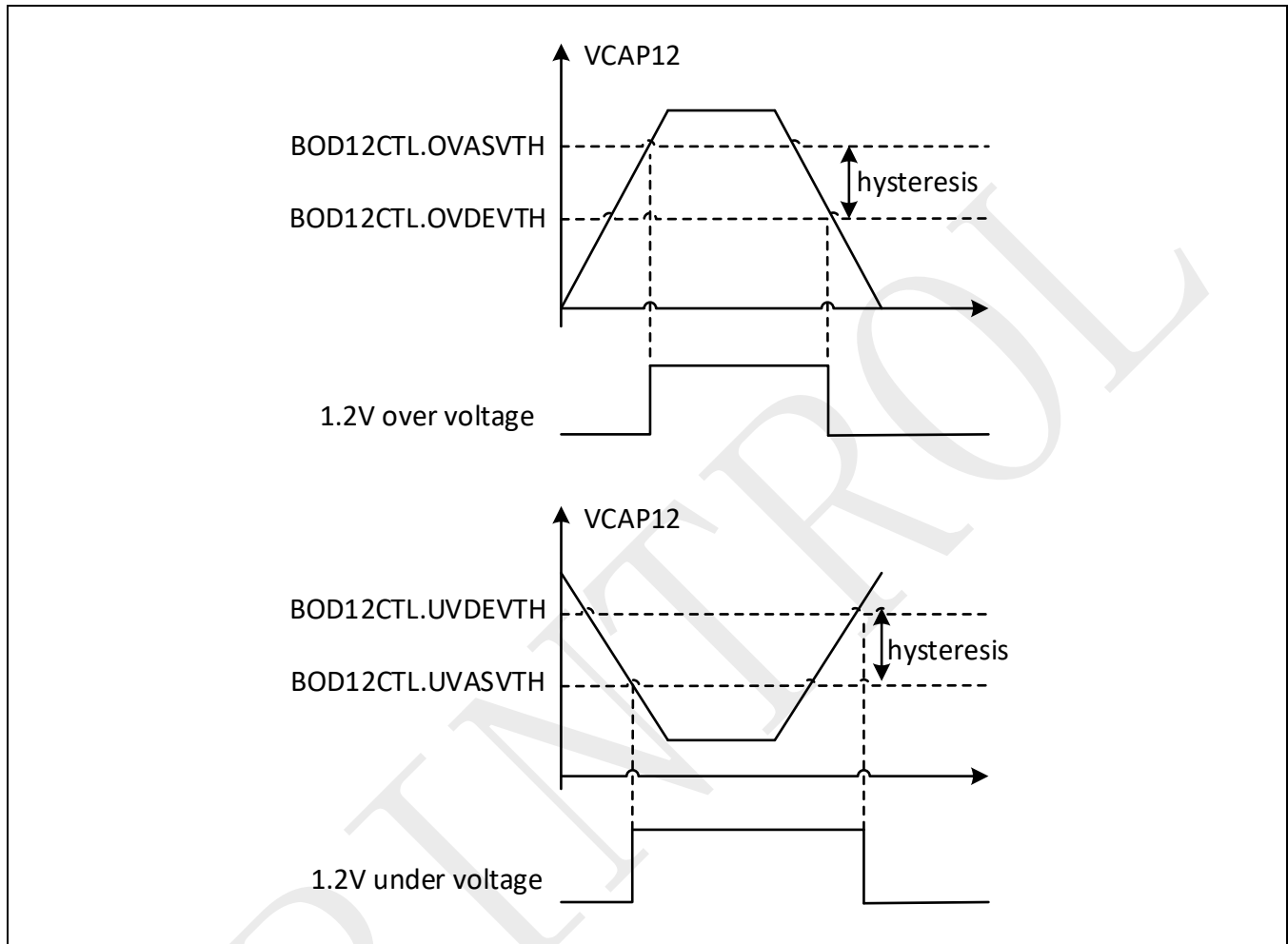
The 3.3V domain over-voltage or under-voltage interrupt can be generated when DVDD33 rises above the over-voltage threshold or when DVDD33 drops below the under-voltage threshold, as shown in Figure 2-7.

Figure 2-7: DVDD33 thresholds



The 1.2V over-voltage or under-voltage interrupt can be generated when VCAP12 rises above the over-voltage threshold or when VCAP12 drops below the under-voltage threshold, as shown in [Figure 2-8](#).

Figure 2-8: VCAP12 thresholds



2.5 System wake up

- Stop mode:
 - Any edge change in the GPIO pin can be used as wake-up source in stop mode. GPIO selection are shown in [Table 25-66](#).
- Sleep mode:
 - Only GPIO4 and GPIO28 pin can be used as wake-up source in sleep mode. The detail register description are shown in [Table 25-66](#).

Note: In stop mode, XRSTn is also a valid source. If XRSTn is low level, chip will wake-up as system reset.

2.6 Registers

2.6.1 MCU Power register map

Table 2-1: POWER Module Base Address

Peripheral Module	Base Address
POWER	0x4000 0100

Table 2-2: POWER Register Map

Register	Offset	Description	Reset Value
LDOCTL *	0x0	Power Control Register	0x00000007
BOD33CTL *	0x4	3.3V BOD Control Register	0x06070100
BOD12CTL *	0x8	1.2V BOD Control Register	0x06070100
PWRMODECMD *	0xC	Power Mode Command Register	0x00000000
PWRREGKEY	0x10	Power Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the PWRREGKEY=0x1ACCE551.

2.6.2 MCU Power registers

Table 2-3: Power Control Register (LDOCTL) Layout

LDOCTL (Power Control Register) Offset: 0x0 Default: 0x00000007 Access: POWER->LDOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VOUT			

Table 2-4: Power Control Register (LDOCTL) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3:0	VOUT	RW	0x7	1.2V LDO output voltage The output will be saturated at the minimum voltage when the code goes lower below certain code (usually <4). The output will be saturated at the maximum voltage when the code goes above certain code (usually >11). The exact threshold varies from chip to chip. 0000: 1.06V 0001: 1.08V 0010: 1.10V 0011: 1.12V 0100: 1.14V 0101: 1.16V 0110: 1.18V 0111: 1.20V 1000: 1.22V 1001: 1.24V 1010: 1.26V 1011: 1.28V 1100: 1.30V 1101: 1.32V 1110: 1.34V 1111: 1.36V

Table 2-5: 3.3V BOD Control Register (BOD33CTL) Layout

BOD33CTL (3.3V BOD Control Register) Offset: 0x4 Default: 0x06070100 Access: POWER->BOD33CTL								
31	30	29	28	27	26	25	24	
		RESERVED				OVDEVTH		
23	22	21	20	19	18	17	16	
		RESERVED				OVASVTH		
15	14	13	12	11	10	9	8	
		RESERVED				UVDEVTH		
7	6	5	4	3	2	1	0	
		RESERVED				UVASVTH		

Table 2-6: 3.3V BOD Control Register (BOD33CTL) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	OVDEVTH	RW	0x6	Select the threshold voltage to deassert VDD33OV when 3.3V supply is going low 000: 3.31V 001: 3.42V 010: 3.53V 011: 3.66V 100: 3.79V 101: 3.93V 110: 4.08V 111: 4.24V
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	OVASVTH	RW	0x7	Select the threshold voltage to assert VDD33OV when 3.3V supply is going high 000: 3.31V 001: 3.42V 010: 3.53V 011: 3.66V 100: 3.79V 101: 3.93V 110: 4.08V 111: 4.24V
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	UVDEVTH	RW	0x1	Select the threshold voltage to deassert VDD33UV when 3.3V supply is going high 000: 2.58V 001: 2.65V 010: 2.72V 011: 2.79V 100: 2.86V

Bits	Field Name	Type	Reset	Description
				101: 2.94V 110: 3.03V 111: 3.12V
7:3	RESERVED_7_3	RO	0x0	Reserved.
2:0	UVASVTH	RW	0x0	Select the threshold voltage to assert VDD33UV when 3.3V supply is going low 000: 2.58V 001: 2.65V 010: 2.72V 011: 2.79V 100: 2.86V 101: 2.94V 110: 3.03V 111: 3.12V

Table 2-7: 1.2V BOD Control Register (BOD12CTL) Layout

BOD12CTL (1.2V BOD Control Register) Offset: 0x8 Default: 0x06070100 Access: POWER->BOD12CTL								
31	30	29	28	27	26	25	24	
RESERVED					OVDEVTH			
23	22	21	20	19	18	17	16	
RESERVED					OVASVTH			
15	14	13	12	11	10	9	8	
RESERVED					UVDEVTH			
7	6	5	4	3	2	1	0	
RESERVED					UVASVTH			

Table 2-8: 1.2V BOD Control Register (BOD12CTL) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	OVDEVTH	RW	0x6	Select the threshold voltage to deassert VDD12OV when 1.2V supply is going low 000: 1.31V 001: 1.33V 010: 1.36V 011: 1.38V 100: 1.41V 101: 1.43V 110: 1.46V 111: 1.49V
23:19	RESERVED_23_19	RO	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
18:16	OVASVTH	RW	0x7	Select the threshold voltage to assert VDD12OV when 1.2V supply is going high 000: 1.31V 001: 1.33V 010: 1.36V 011: 1.38V 100: 1.41V 101: 1.43V 110: 1.46V 111: 1.49V
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	UVDEVTH	RW	0x1	Select the threshold voltage to deassert VDD12UV when 1.2V supply is going high 000: 0.94V 001: 0.97V 010: 0.99V 011: 1.02V 100: 1.04V 101: 1.07V 110: 1.10V 111: 1.14V
7:3	RESERVED_7_3	RO	0x0	Reserved.
2:0	UVASVTH	RW	0x0	Select the threshold voltage to assert VDD12UV when 1.2V supply is going low 000: 0.94V 001: 0.97V 010: 0.99V 011: 1.02V 100: 1.04V 101: 1.07V 110: 1.10V 111: 1.14V

Table 2-9: Power Mode Command Register (PWRMODECMD) Layout

PWRMODECMD (Power Mode Command Register) Offset: 0xC Default: 0x00000000							
Access: POWER->PWRMODECMD							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 2-10: Power Mode Command Register (PWRMODECMD) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	W1S	0x0	Write 0xA5A51EE9 to make the MCU go to deep-sleep Write 0x5A5A5709 to make the MCU go to stop

Table 2-11: Power Register Write-Allow Key Register (PWRREGKEY) Layout

PWRREGKEY (Power Register Write-Allow Key Register) Offset: 0x10 Default: 0x1ACCE551 Access: POWER->PWRREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 2-12: Power Register Write-Allow Key Register (PWRREGKEY) Description

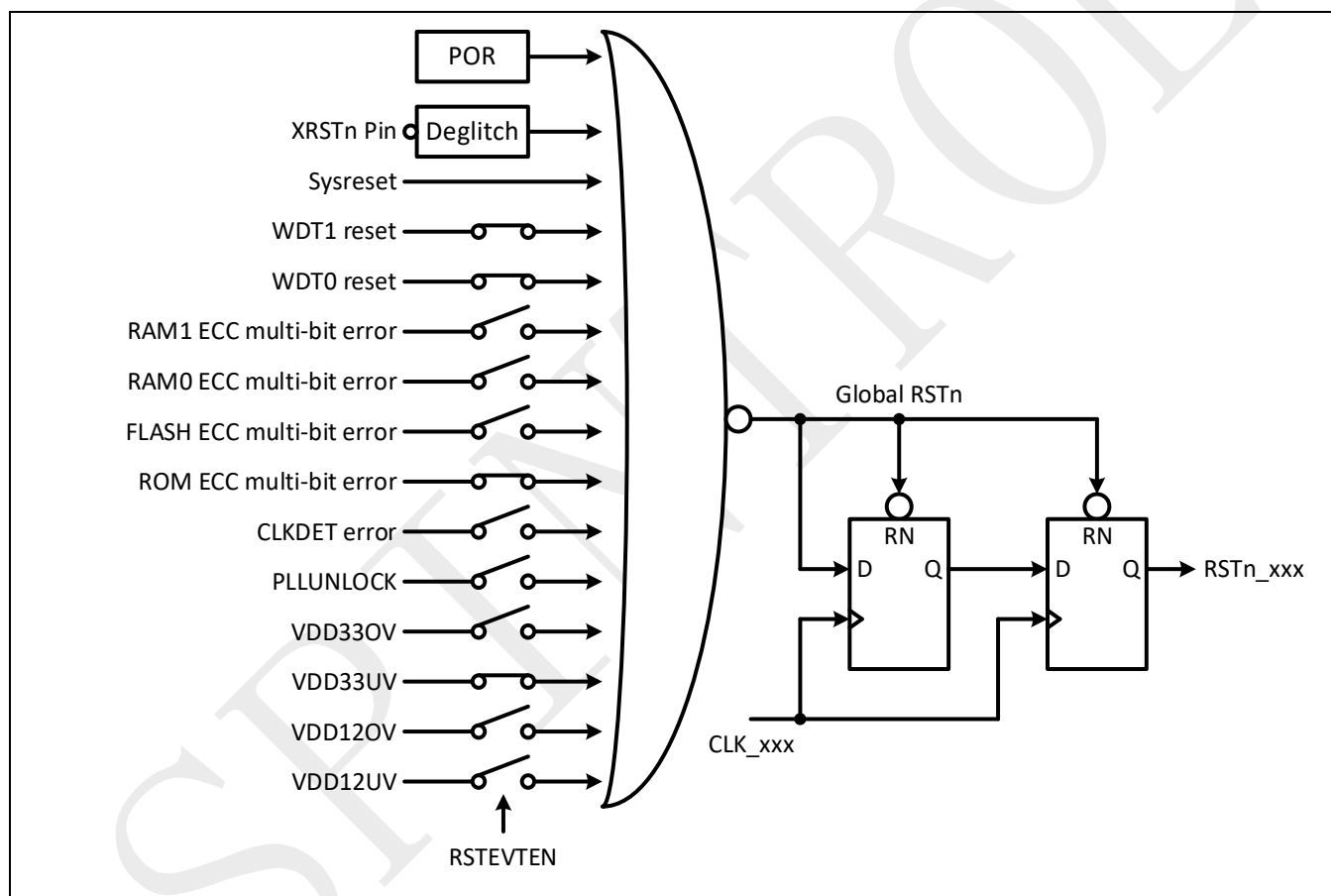
Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected power registers Write 0xACCE55CD to unlock PWRMODECMD register

3 Clocks and reset

3.1 Reset architecture

The global active low reset signal of the SPC1169 is a logic NOR of the reset signals from the power-on-reset (POR) block, the XRSTn pin and the functional logic. It is further shaped as asynchronously asserted and synchronously released in each peripheral's clock domain for timing integrity as shown in [Figure 3-1](#). The functional reset source can be the memory errors, the brown-out detector events, the clock error events, the watchdogs, and the Cortex-M4 SYSRESET request.

Figure 3-1: Active-low POR reset signal diagram



As a diagnosis feature, RSTEV TFLG register records the functional reset events, which can be cleared by corresponding bits in RSTEV TCLR register. Besides, the functional reset sources are programmable via RSTEV TEN register. Details are shown in [Table 25-89](#) and [Table 25-90](#).

3.2 Pin reset

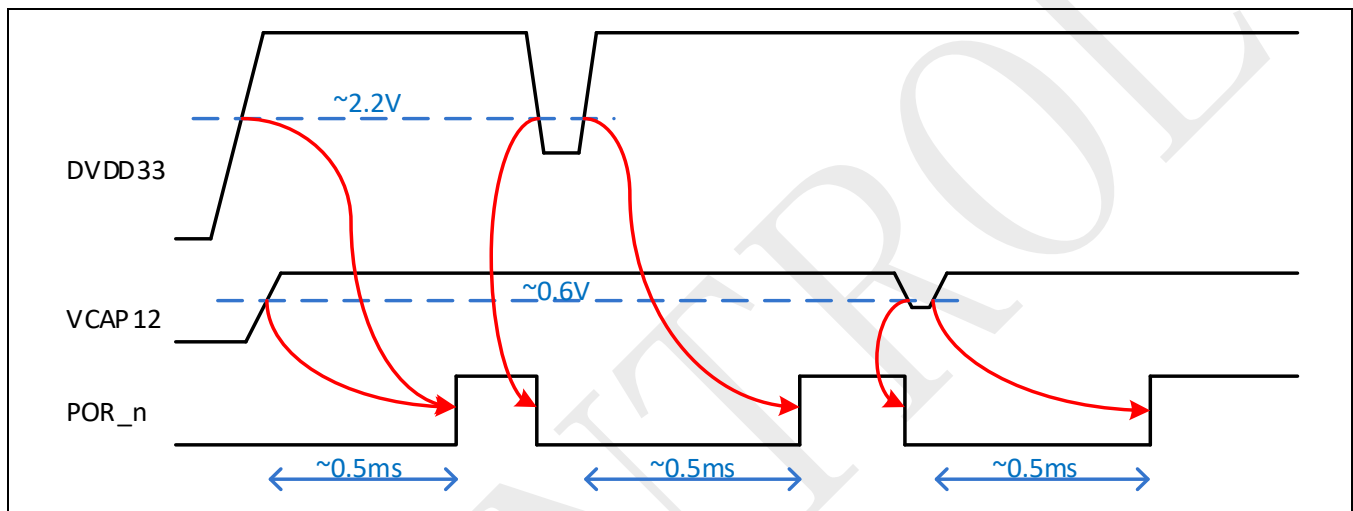
The XRSTn pin of the SPC1169 provides super-set global reset for all functional blocks. It is pull-up internally to make the chip in normal operation mode. The original XRSTn signal goes through a glitch filter. To assert a valid reset, the XRSTn pin should be kept low for time longer than typically 500us.

Besides, if the XRSTn pin is kept low for more than typically 5 seconds, it will further trigger the reset of all analog building blocks, including the 3.3V logic. In this case, it is similar to the power on reset.

3.3 POR reset

The built-in POR circuits in the SPC1169 guarantee all power up reset sequence requirements and make chip easy to use. It will be asserted for about 0.5ms after power up. The global active-low reset POR_n will go low whenever power levels of DVDD33 and VCAP12 are below approximately 2.2V and 0.6V respectively as shown in Figure 3-2.

Figure 3-2: Active-low POR reset signal timing diagram

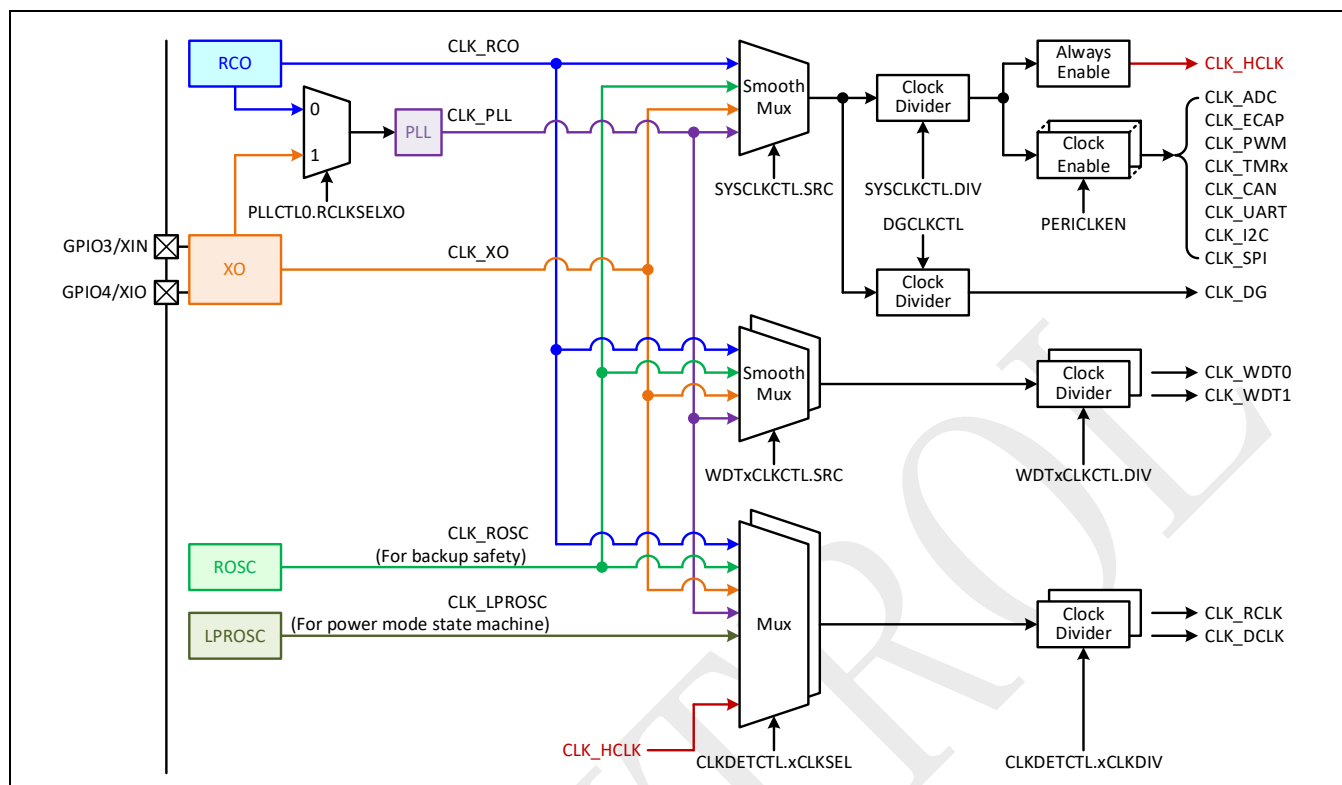


3.4 Clock architecture

The SPC1169 can be clocked by the internal RC oscillators, ring oscillators, or the PLL. It can also be clocked by an external oscillator, or by a crystal attached to the on-chip oscillator circuit. The PLL supports high frequency clock from 25MHz to 100MHz while the input frequency for the PLL can be from 4MHz to 56MHz. Combined with programmable dividers and gating blocks, a flexible clock tree is implemented in SPC1169 as shown in Figure 3-3.

-
- Note:
1. Before using the peripheral module, user should enable the corresponding clock first.
 2. The clocks can be brought out to GPIO2 for observation. Refer to Table 4-1 for details.
-

Figure 3-3: Clock overview



3.5 High frequency RC oscillator

There are two zero-pin oscillators implemented on SPC1169. One has been trimmed for minimum frequency variation over temperature and centered at 32MHz. The other is a free-running ring oscillator with a typical frequency of 32MHz, which is mainly used for clock safety.

3.6 On-chip crystal oscillator and external clock

Both on-chip and external modes can be used for the crystal oscillator. For the on-chip mode, a crystal need to be attached with the frequency range from 4MHz to 32MHz as shown in Figure 3-4.

Amplitude control is implemented in the crystal oscillator to constrain the power dissipated in the crystal, thus ensure long term frequency stability. Figure 3-4 shows the structure of the crystal oscillator, it has an oscillator working with a crystal and a amplitude control module which is able to sense the amplitude between XIN and XOUT pins during oscillation and adjust it to a preset value.

There are 2 methods provided to control the amplitude of oscillation between XIN and XOUT pins. The first one is called open loop amplitude control, with XOCTL.FRCCODEEN=1, it directly control the supply current goes into the oscillator, by setting the register XOCTL.FRCCODE, and max current it can source is 2mA. In this mode, customer need to choose the right amount of supply current for safe oscillation.

The second one is called close loop amplitude control. After desired oscillation amplitude is set through the register XOCTL.AMP, with 0.32V to 1.33V amplitude voltage selectable, amplitude control module will sense the oscillation amplitude between XIN and XOUT and compare it with the preset value. If the

oscillation amplitude is smaller or bigger than set amplitude, logic engine will increase the bias current until it reaches to the pre-set amplitude within one amplitude voltage level defined in XOCTL.AMP. For each calibration step to tune the amplitude, certain clock period need to be wait to make sure control loop stability, this clock period can be adjusted through register XOCTL.ACTLPRD, and default code=2 with 8192 clock cycle is recommended, this value should be in the same order of magnitude as the crystal's Q value. Right after XO is enabled, a pseudo-random noise is injected in the XIN pin with a 69.6us duration. After that DAC current can be set by selecting the 4 MSB through the register XOCTL.INITCODE, whose value is recommended to be set to the maximum for a safe start up. If user already know roughly the final control code after amplitude calibration, XOCTL.OPCODE which defines the 4 MSB code one update clock period after starting up can be set to close to the final value and reduce the calibration time. After calibration, if for whatever reason the amplitude is out of pre-set range, the calibration procedure will start all over again. To avoid mis triggering the calibration procedure, XOCTL.ACTLRELAX register can be set to 1 which will increase the triggering window of the amplitude calibration. Finally, external modes is provided. If XOCTL.BYPASS=1, crystal oscillator clock output will follow XIN input as shown in Figure 3-5.

Figure 3-4: On-chip crystal oscillator

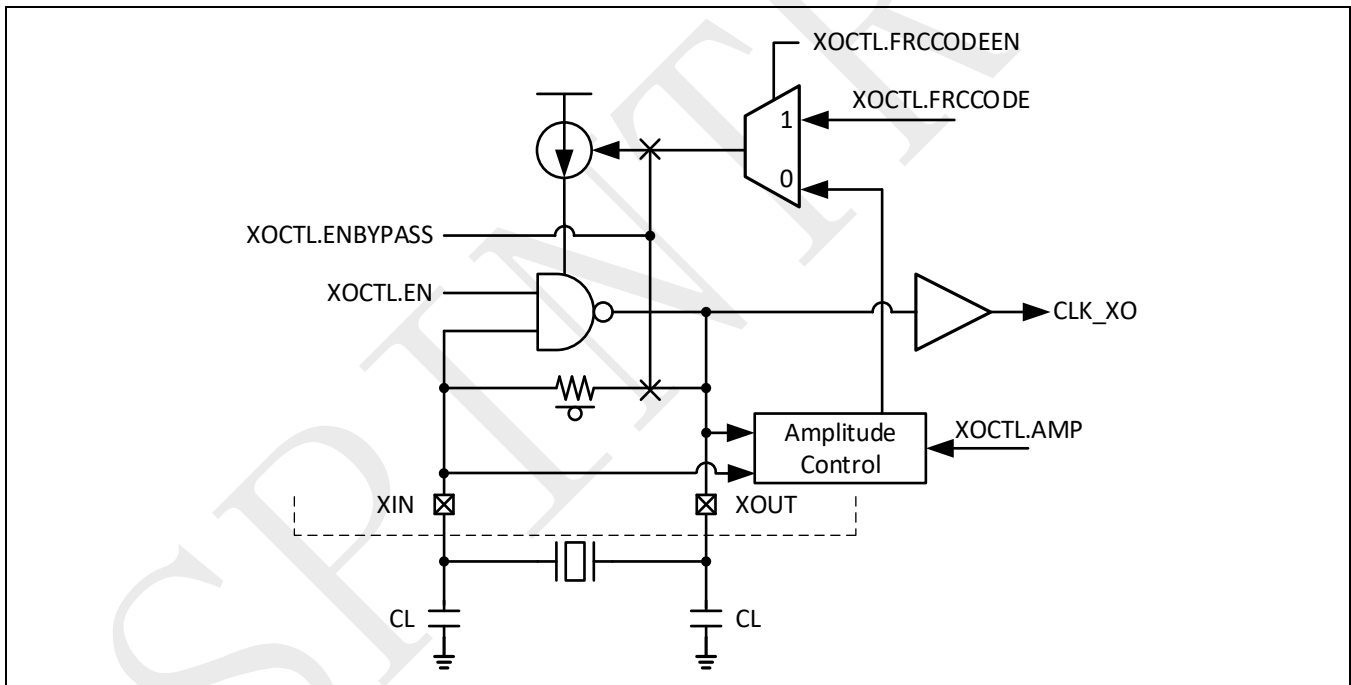
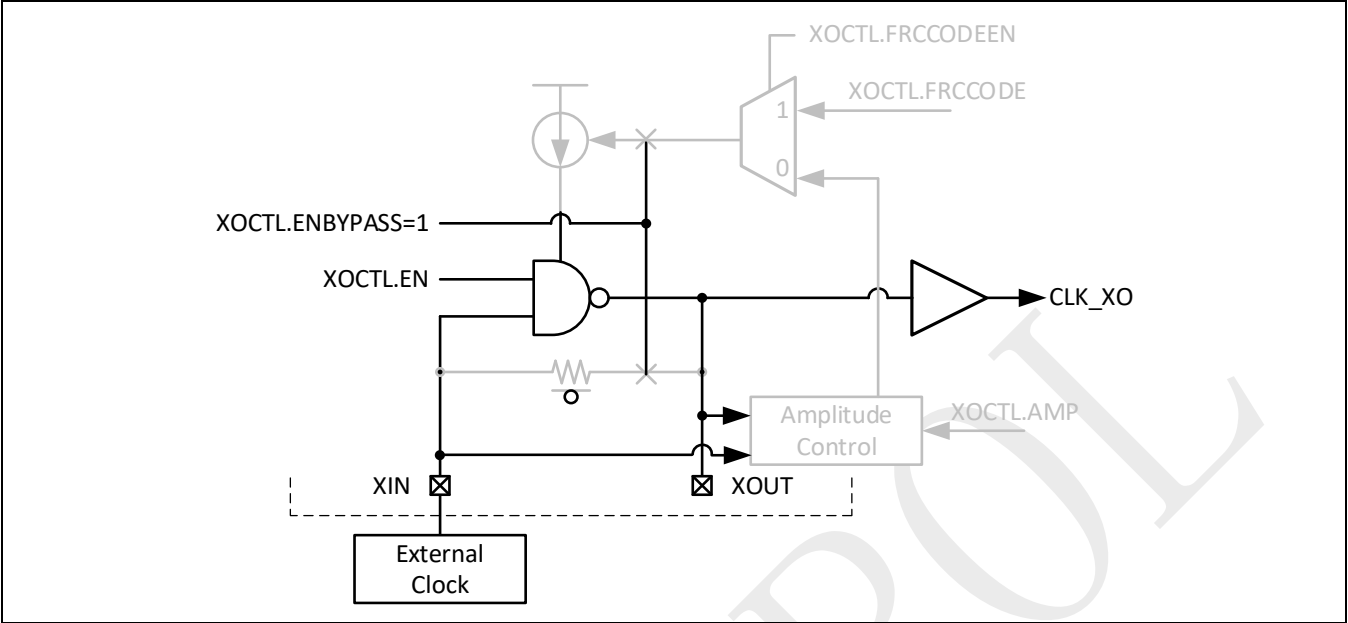


Figure 3-5: External clock



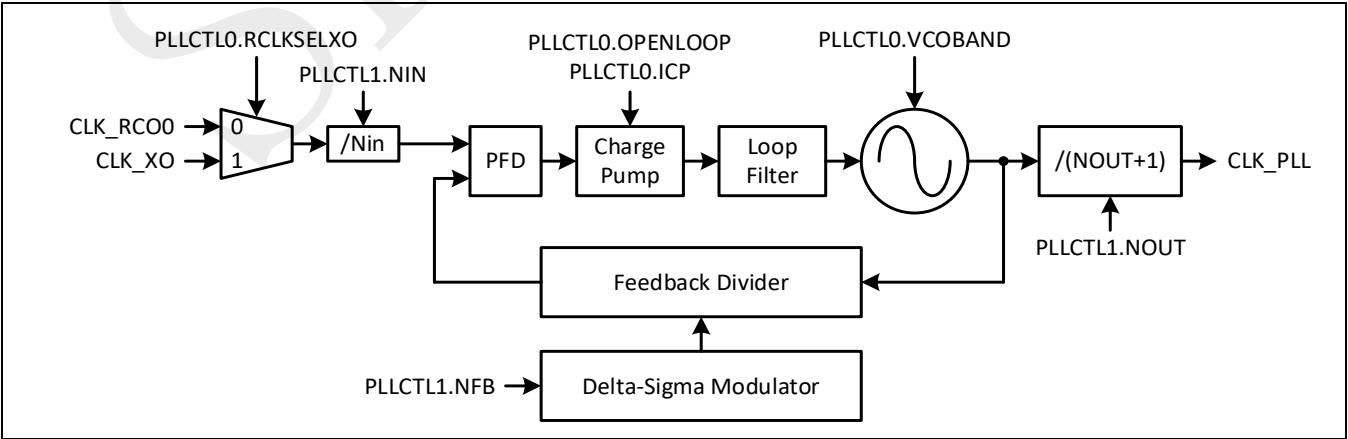
Note: If the CLK_XO is used as reference for PLL, the external clock frequency should be within from 4MHz to 56MHz.

3.7 PLL clock

The architecture of the phase-locked loop in SPC1169 is shown in Figure 3-6. It can take the clock from RCO or crystal oscillator as input reference. While the loop is locked, the VCO frequency will be from 400MHz to 600MHz. It is then divided down by programmable divider to provide typically 25MHz to 100MHz clock. The output frequency is given by:

$$f_{PLL} = \frac{f_{REF}}{NIN} \times \frac{NFB}{65536} \times \frac{1}{NOUT + 1}$$

Figure 3-6: PLL clock overview



3.7.1 Reference input and divider

The PLL is designed to take input reference clock frequency (f_{REF}) from 4MHz to 56MHz, while the reference clock at PFD input (f_{PFD}) is from 4MHz to 8MHz. Therefore, the input dividing ratio (f_{REF}/f_{PFD}) should be configured according to [Table 3-1](#). Here, NIN is the 1st register control the user needs to derive.

Table 3-1: Input dividing ratio configuration

Input Frequency (MHz)	Input Dividing Ratio ($NIN = f_{REF}/f_{PFD}$)
4~8	1
8~16	2
16~24	3
24~32	4
32~40	5
40~48	6
48~56	7

3.7.2 Output divider

The tuning range of the VCO is designed to be 400MHz to 600MHz. Combined with the output divider, the output clock frequency at CLK_PLL is shown in [Table 3-2](#). Here, NOUT is the 2nd register control the user needs to derive.

Table 3-2: Output dividing ratio and clock frequency

Desired Output Frequency	Dividing Ratio ($NOUT+1$)
75MHz ~ 100MHz	6
50MHz ~ 75MHz	8
37.5MHz ~ 50MHz	12
25MHz ~ 37.5MHz	16

3.7.3 Feedback divider

Based on the criterion on NIN & NOUT derived from [Table 3-1](#) and [Table 3-2](#), the user can calculate the 3rd register control as:

$$NFB = \frac{f_{PLL} \times (NOUT + 1)}{f_{REF} / NIN} \times 65536$$

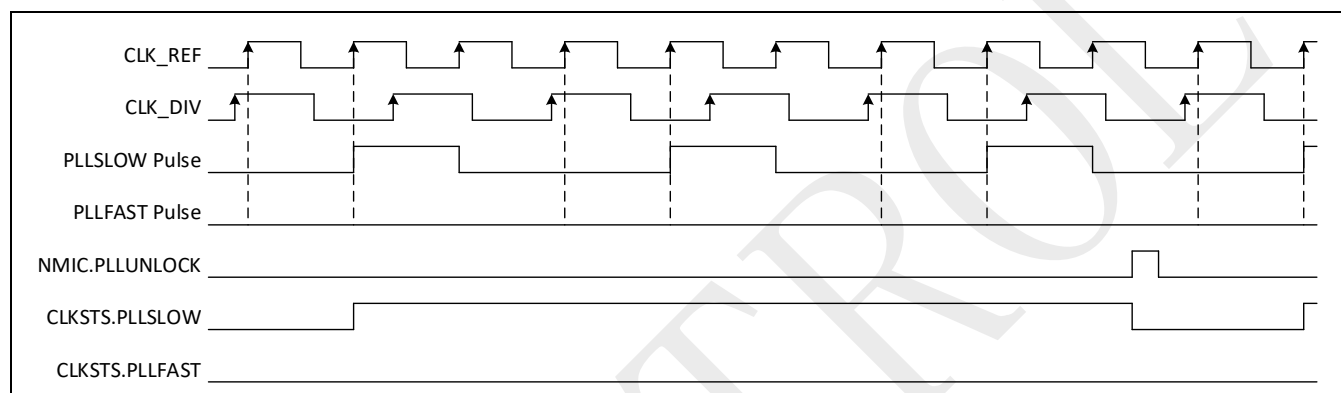
3.7.4 PLL unlock detection

A loop unlock detector is included in the phase-frequency detector (PFD) of SPC1169 to provide the diagnosis information for PLL locking status. For the two clocks at the PFD input, if there are two or more rising edges of one clock appears between the two adjacent rising edges of the other clock, internal PLLFAST/PLLSLOW pulse will be generated. These pulses will update the corresponding bit in CLKSTS register. The status bit can be cleared via write a 1 to NMIC.PLLUNLOCK. [Figure 3-7](#) illustrates a timing

diagram for the scenario that reference clock is faster than the divider clock and CLKSTS.PLLSLOW bit is updated.

During the initial locking or a significant change on the feedback dividing ratio of a locked PLL, the frequency of the reference clock and divider clock are different and there will be inevitable PLLFAST or PLLSLOW events latched into CLKSTS register. To avoid false alarm, it should wait some time (about 20us) for the loop to be settled and then assert an NMIC.PLLUNLOCK to clear the latched status and enable interrupt upon PLL unlock event via NMIE.PLLUNLOCK if wanted. Neither CLKSTS.PLLFAST nor CLKSTS.PLLSLOW will be set afterwards if the PLL is locked.

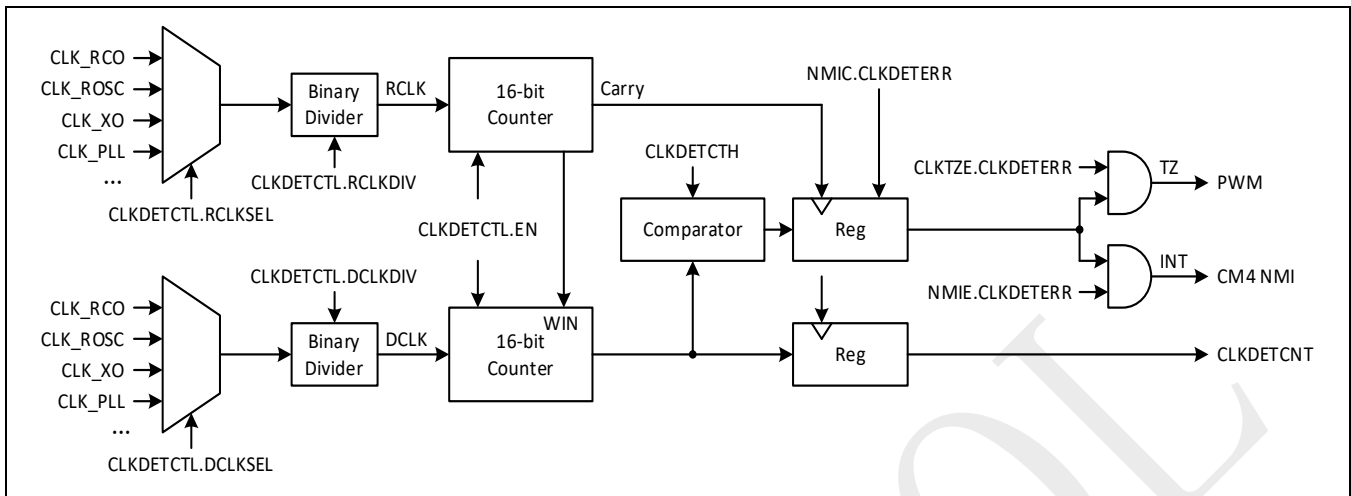
Figure 3-7: PLL unlock detection timing diagram example (FREF > FDIV)



3.8 Clock safety

The SPC1169 provides clock cross-detection circuits for clock safety concern. As shown in [Figure 3-8](#), reference clock and detected clock are selected independently. Each of them will go through a binary divider to get the RCLK and DCLK. The RCLK is used to generate the time window, during which period the DCLK is counting. When the counting window is finished, the final value of DCLK counter is latched into CLKDETCNT register. The counted value is also compared with the high and low threshold given in CLKDETCTH register and the error status is latched into NMIF.CLKDETERR if the value is out of the boundary. It further triggers a CPU interrupt or a PWM trip-zone event if as enabled by corresponding bit in NMIE and CLKTZE registers. The DCLK can be brought out to GPIO2 by setting GPIO2.MUXSEL=1. Refer to [Table 4-1](#) for the details.

Figure 3-8: Clock cross-detection



3.9 Registers

3.9.1 Clock register map

Table 3-3: CLOCK Module Base Address

Peripheral Module	Base Address
CLOCK	0x4000 0200

Table 3-4: CLOCK Register Map

Register	Offset	Description	Reset Value
CLKSTS	0x00	Clock Status Register	0x00000003
RCOCTL *	0x04	RCO Control Register	0x00000001
XOCTL *	0x08	Crystal Oscillator Control Register	0x00007F58
PLLCTL0 *	0x0C	PLL Control Register 0	0x00000270
PLLCTL1 *	0x10	PLL Control Register 1	0x2C4B0000
CLKDECTL *	0x14	Clock Detection Control Register	0x00000000
CLKDECTH *	0x18	Clock Detection Counter Threshold Register	0x00000000
CLKDETCNT	0x1C	Clock Detection Counter Register	0x00000000
CLKTZE *	0x20	Clock Trip-zone Event Enable Register	0x00000000
SYSCLKCTL *	0x24	System Clock Control Register	0x00000000
DGCLKCTL *	0x28	Deglitch Clock Control Register	0x00000200
WDT0CLKCTL *	0x2C	WDT0 Clock Control Register	0x00000000
WDT1CLKCTL *	0x30	WDT1 Clock Control Register	0x01000000
PERICKEN *	0x34	Peripheral Clock Enable Register	0x00000000
PERICKSTS	0x38	Peripheral Clock Status Register	0x0000F000
CLKREGKEY	0x3C	Clock Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the CLKREGKEY=0x1ACCE551.

3.9.2 Clock registers

Table 3-5: Clock Status Register (CLKSTS) Layout

CLKSTS (Clock Status Register) Offset: 0x0 Default: 0x00000003 Access: CLOCK->CLKSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							CLKDETERR
7	6	5	4	3	2	1	0
PLLSLOW	PLLFAST	VCOFREQVLD	VCOFREQ	PLLRDY	XORDY	ROSCRDY	RCORDY

Table 3-6: Clock Status Register (CLKSTS) Description

Bits	Field Name	Type	Reset	Description
31:9	RESERVED_31_9	RO	0x0	Reserved.
8	CLKDETERR	RO	0x0	Indicate error detected by CLKDET 0: No clock error is detected 1: Clock error is detected
7	PLLSLOW	RO	0x0	Indicate PLL is unlocked and slower than expected. Possible reason can be: 1. $f_{REF}/NIN*NFB/65536$ is higher than maximum VCO free-running frequency 2. VCO fails to oscillate 3. Malfunction of feedback divider so that there is no output or the equivalent dividing ratio is higher than expected
6	PLLFAST	RO	0x0	Indicate PLL is unlocked and faster than expected. Possible reason can be: 1. $f_{REF}/NIN*NFB/65536$ is lower than minimum VCO free-running frequency 2. The reference clock input is somehow missing 3. Malfunction of feedback divider so that the equivalent dividing ratio is lower than expected.
5	VCOFREQVLD	RO	0x0	VCOFREQ flag valid indicator 0: VCOFREQ is invalid 1: VCOFREQ is valid
4	VCOFREQ	RO	0x0	VCO frequency flag for trimming 0: VCO in PLL is too slow 1: VCO in PLL is too fast

Bits	Field Name	Type	Reset	Description
3	PLLRDY	RO	0x0	PLL clock ready indicator 0: Not ready 1: Ready
2	XORDY	RO	0x0	XO clock ready indicator 0: Not ready 1: Ready
1	ROSCRDY	RO	0x1	ROSC clock ready indicator 0: Not ready 1: Ready
0	RCORDY	RO	0x1	RCO clock ready indicator 0: Not ready 1: Ready

Table 3-7: RCO Control Register (RCOCTL) Layout

RCOCTL (RCO Control Register) Offset: 0x4 Default: 0x00000001 Access: CLOCK->RCOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						RESERVED	EN

Table 3-8: RCO Control Register (RCOCTL) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	RESERVED_1	RW	0x0	Reserved.
0	EN	RW	0x1	RCO enable 0: Disable 1: Enable

Table 3-9: Crystal Oscillator Control Register (XOCTL) Layout

XOCTL (Crystal Oscillator Control Register) Offset: 0x8 Default: 0x00007F58 Access: CLOCK->XOCTL						
31 CURRENT	30 FEEDPIN	29 BYPASS	28 NINJLEN	27	26 FRCCODEEN	25 24 FRCCODE
23	22	21	20	19	18	17 16 FRCCODE
15	14	13	12	11	10	9 8 INITCODE
7	6 ACTLPRD	5	4 ACTLRELAX	3	2 AMP	1 0 EN

Table 3-10: Crystal Oscillator Control Register (XOCTL) Description

Bits	Field Name	Type	Reset	Description
31	CURRENT	RW	0x0	Current bias option 0: 1x current 1: 2x current
30	FEEDPIN	RW	0x0	Pin select in bypass mode 0: Apply clock to XIN (GPIO3) pin 1: Apply clock to XOUT (GPIO4) pin
29	BYPASS	RW	0x0	Directly feed in the external clock and bypass internal circuit 0: Internal circuit is in normal operation 1: Internal circuit is bypassed
28:27	RESERVED_28_27	RW	0x0	Reserved.
26	FRCCODEEN	RW	0x0	Forced amplitude code enable 0: Amplitude is controlled by the engine 1: Amplitude is controlled by FRCCODE bits
25:16	FRCCODE	RW	0x0	Forced amplitude code
15:12	OPCODE	RW	0x7	Estimated amplitude code (4MSBs) in operation
11:8	INITCODE	RW	0xF	Amplitude code (4MSBs) during initial startup
7:5	ACTLPRD	RW	0x2	Amplitdue control code update period 000: Update DAC code every 2048 clock cycles 001: Update DAC code every 4096 clock cycles 010: Update DAC code every 8192 clock cycles 011: Update DAC code every 16384 clock cycles 100: Update DAC code every 32768 clock cycles 101: Update DAC code every 65536 clock cycles 110: Update DAC code every 131072 clock cycles 111: Update DAC code every 262144 clock cycles

Bits	Field Name	Type	Reset	Description
4	ACTLRELAX	RW	0x1	Relax the threshold of amplitude control 0: Normal threshold 1: Relaxed threshold
3:1	AMP	RW	0x4	Oscillation amplitude when locked 000: 0.32 V 001: 0.48 V 010: 0.64 V 011: 0.77 V 100: 0.92 V 101: 1.06 V 110: 1.19 V 111: 1.33 V
0	EN	RW	0x0	Crystal oscillator clock enable 0: Disable 1: Enable

Table 3-11: PLL Control Register 0 (PLLCTL0) Layout

PLLCTL0 (PLL Control Register 0) Offset: 0xC Default: 0x00000270 Access: CLOCK->PLLCTL0							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					VCOBAND		ICP
7	6	5	4	3	2	1	0
ICP				RCLKSELXO	FCALWIN	FCALEN	EN

Table 3-12: PLL Control Register 0 (PLLCTL0) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10:9	VCOBAND	RW	0x1	PLL VCO band control 00: roughly 350 ~ 450MHz 01: roughly 450 ~ 550MHz 10: roughly 550 ~ 650MHz 11: roughly 600 ~ 700MHz
8:4	ICP	RW	0x7	PLL charge pump current setting For better stability, it could be set to $0.96 \cdot \text{NFB} / (11 - \text{VCOBAND})$, where NFB is the PLL feedback factor in real number

Bits	Field Name	Type	Reset	Description
3	RCLKSELXO	RW	0x0	PLL reference clock select 0: CLK_RCO (RC oscillator) 1: CLK_XO (Crystal oscillator)
2	FCALWIN	RW	0x0	PLL frequency calibration time window For each calibration cycle, keep it as 1 until VCOFREQVLD=1 and release it as 0 0: Window is disabled 1: Window is enabled
1	FCALEN	RW	0x0	PLL frequency calibration enable 0: Disable 1: Enable
0	EN	RW	0x0	PLL enable 0: Disable 1: Enable

Table 3-13: PLL Control Register 1 (PLLCTL1) Layout

PLLCTL1 (PLL Control Register 1) Offset: 0x10 Default: 0x2C4B0000 Access: CLOCK->PLLCTL1							
31 RESERVED	30	29	28	27	26	25	24
			NOUT			NIN	
23	22	21	20	19	18	17	16
				NFB			
15	14	13	12	11	10	9	8
				NFB			
7	6	5	4	3	2	1	0
				NFB			

Table 3-14: PLL Control Register 1 (PLLCTL1) Description

Bits	Field Name	Type	Reset	Description
31	RESERVED_31	RO	0x0	Reserved.
30:27	NOUT	RW	0x5	Set VCO to PLL output clock dividing ratio (f_{VCO}/f_{PLL}) to be (NOUT + 1)
26:24	NIN	RW	0x4	Input reference clock dividing ratio (f_{REF}/f_{PFD}) 001: Set to 1 if f_{REF} is within [4MHz, 8MHz] 010: Set to 2 if f_{REF} is within (8MHz, 16MHz] 011: Set to 3 if f_{REF} is within (16MHz, 24MHz] 100: Set to 4 if f_{REF} is within (24MHz, 32MHz] 101: Set to 5 if f_{REF} is within (32MHz, 40MHz]

Bits	Field Name	Type	Reset	Description
				110: Set to 6 if f_{REF} is within (40MHz, 48MHz] 111: Set to 7 if f_{REF} is within (48MHz, 56MHz]
23:0	NFB	RW	0x4B0000	Set feedback dividing ratio (f_{VCO}/f_{PFD}) to be NFB/65536

Table 3-15: Clock Detection Control Register (CLKDETCTL) Layout

CLKDETCTL (Clock Detection Control Register) Offset: 0x14 Default: 0x00000000				Access: CLOCK->CLKDETCTL			
31	30	29	28	27	26	25	24
EN	RESERVED				DCLKDIV		
23	22	21	20	19	18	17	16
RESERVED					DCLKSEL		
15	14	13	12	11	10	9	8
RESERVED					RCLKDIV		
7	6	5	4	3	2	1	0
RESERVED					RCLKSEL		

Table 3-16: Clock Detection Control Register (CLKDETCTL) Description

Bits	Field Name	Type	Reset	Description
31	EN	RW	0x0	Clock detection enable When disabled, the error events will be cleared 0: Disable clock detection 1: Enable clock detection
30:27	RESERVED_30_27	RO	0x0	Reserved.
26:24	DCLKDIV	RW	0x0	Detected clock dividing ratio 000: Divide by 1 001: Divide by 2 010: Divide by 4 011: Divide by 8 100: Divide by 16 101: Divide by 32 110: Divide by 64 111: Divide by 128
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	DCLKSEL	RW	0x0	Detected clock select 000: RCO clock 001: Ring oscillator clock 010: Crystal oscillator clock 011: PLL clock 100: Low power ring oscillator clock

Bits	Field Name	Type	Reset	Description
				101: HV clock via ePower TZ0 pin in diagnosis mode 110: HV clock via ePower TZ1 pin in diagnosis mode 111: AHB clock, which is also the CPU clock
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	RCLKDIV	RW	0x0	Reference clock dividing ratio 000: Divide by 1 001: Divide by 2 010: Divide by 4 011: Divide by 8 100: Divide by 16 101: Divide by 32 110: Divide by 64 111: Divide by 128
7:3	RESERVED_7_3	RO	0x0	Reserved.
2:0	RCLKSEL	RW	0x0	Reference clock select 000: RCO clock 001: Ring oscillator clock 010: Crytal oscillator clock 011: PLL clock 100: Low power ring oscillator clock 101: HV clock via ePower TZ0 pin in diagnosis mode 110: HV clock via ePower TZ1 pin in diagnosis mode 111: AHB clock, which is also the CPU clock

Table 3-17: Clock Detection Counter Threshold Register (CLKDETCTH) Layout

CLKDETCTH (Clock Detection Counter Threshold Register) Offset: 0x18 Default: 0x00000000							
Access: CLOCK->CLKDETCTH							
31	30	29	28	27	26	25	24
HI							
23	22	21	20	19	18	17	16
HI							
15	14	13	12	11	10	9	8
LO							
7	6	5	4	3	2	1	0
LO							

Table 3-18: Clock Detection Counter Threshold Register (CLKDETCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	HI	RW	0x0	Upper counter threshold for clock detect If the final counter value > HI, a trip-zone event to PWM and an interrupt request to CPU will be generated
15:0	LO	RW	0x0	Lower counter threshold for clock detect If the final counter value < LO, a trip-zone event to PWM and an interrupt request to CPU will be generated

Table 3-19: Clock Detection Counter Register (CLKDETCNT) Layout

CLKDETCNT (Clock Detection Counter Register) Offset: 0x1C Default: 0x00000000 Access: CLOCK->CLKDETCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 3-20: Clock Detection Counter Register (CLKDETCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Final counter value of the clock detect

Table 3-21: Clock Trip-zone Event Enable Register (CLKTZE) Layout

CLKTZE (Clock Trip-zone Event Enable Register) Offset: 0x20 Default: 0x00000000 Access: CLOCK->CLKTZE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						CLKDETERR	PLLUNLOCK

Table 3-22: Clock Trip-zone Event Enable Register (CLKTZE) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	CLKDETERR	RW	0x0	CLKDET error trip-zone event enable 0: CLKDET error will not trigger PWM trip-zone 1: CLKDET error will trigger PWM trip-zone
0	PLLUNLOCK	RW	0x0	PLL unlock trip-zone event enable 0: PLL unlock will not trigger PWM trip-zone 1: PLL unlock will trigger PWM trip-zone

Table 3-23: System Clock Control Register (SYSCLKCTL) Layout

SYSCLKCTL (System Clock Control Register) Offset: 0x24 Default: 0x00000000 Access: CLOCK->SYSCLKCTL							
31	30	29	28	27	26	25	24
RESERVED						SRC	
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DIV							

Table 3-24: System Clock Control Register (SYSCLKCTL) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	SRC	RW	0x0	Clock source select 00: RCO 01: RO SC 10: XO 11: PLL
23:8	RESERVED_23_8	RO	0x0	Reserved.
7:0	DIV	RW	0x0	Set dividing ratio from the source clock to be (DIV+1)

Table 3-25: Deglitch Clock Control Register (DGCLKCTL) Layout

DGCLKCTL (Deglitch Clock Control Register) Offset: 0x28 Default: 0x00000200 Access: CLOCK->DGCLKCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					PDIV		
7	6	5	4	3	2	1	0
DIV							

Table 3-26: Deglitch Clock Control Register (DGCLKCTL) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10:8	PDIV	RW	0x2	Clock binary pre-dividing ratio from source clock 000: Divide by 1 001: Divide by 2 010: Divide by 4 011: Divide by 8 100: Divide by 16 101: Divide by 32 110: Divide by 64 111: Divide by 128
7:0	DIV	RW	0x0	Set overall dividing ratio to be $(DIV+1)*2^{PDIV}$

Table 3-27: WDT0 Clock Control Register (WDT0CLKCTL) Layout

WDT0CLKCTL (WDT0 Clock Control Register) Offset: 0x2C Default: 0x00000000 Access: CLOCK->WDT0CLKCTL							
31	30	29	28	27	26	25	24
RESERVED						SRC	
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DIV							

Table 3-28: WDT0 Clock Control Register (WDT0CLKCTL) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	SRC	RW	0x0	Clock source select 00: RCO clock 01: Ring oscillator clock 10: Crytal oscillator clock 11: PLL clock
23:8	RESERVED_23_8	RO	0x0	Reserved.
7:0	DIV	RW	0x0	Set dividing ratio from the source clock to be (DIV+1)

Table 3-29: WDT1 Clock Control Register (WDT1CLKCTL) Layout

WDT1CLKCTL (WDT1 Clock Control Register) Offset: 0x30 Default: 0x01000000 Access: CLOCK->WDT1CLKCTL							
31	30	29	28	27	26	25	24
RESERVED						SRC	
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DIV							

Table 3-30: WDT1 Clock Control Register (WDT1CLKCTL) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	SRC	RW	0x1	Clock source select 00: RCO clock 01: Ring oscillator clock 10: Crytal oscillator clock 11: PLL clock
23:8	RESERVED_23_8	RO	0x0	Reserved.
7:0	DIV	RW	0x0	Set dividing ratio from the source clock to be (DIV+1)

Table 3-31: Peripheral Clock Enable Register (PERICKEN) Layout

PERICKEN (Peripheral Clock Enable Register) Offset: 0x34 Default: 0x00000000 Access: CLOCK->PERICKEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				CANCLK	I2CCLK	SPI1CLK	SPIOCLK
7	6	5	4	3	2	1	0
UART1CLK	UART0CLK	TIMER2CLK	TIMER1CLK	TIMER0CLK	ECAPCLK	PWMCLK	ADCCLK

Table 3-32: Peripheral Clock Enable Register (PERICKEN) Description

Bits	Field Name	Type	Reset	Description
31:12	RESERVED_31_12	RO	0x0	Reserved.
11	CANCLK	RW	0x0	CAN module clock enable 0: Disable clock output 1: Enable clock output
10	I2CCLK	RW	0x0	I2C module clock enable 0: Disable clock output 1: Enable clock output
9	SPI1CLK	RW	0x0	SPI1 module clock enable 0: Disable clock output 1: Enable clock output
8	SPIOCLK	RW	0x0	SPIO module clock enable 0: Disable clock output 1: Enable clock output
7	UART1CLK	RW	0x0	UART1 module clock enable 0: Disable clock output 1: Enable clock output
6	UART0CLK	RW	0x0	UART0 module clock enable 0: Disable clock output 1: Enable clock output
5	TIMER2CLK	RW	0x0	TIMER2 module clock enable 0: Disable clock output 1: Enable clock output
4	TIMER1CLK	RW	0x0	TIMER1 module clock enable 0: Disable clock output 1: Enable clock output

Bits	Field Name	Type	Reset	Description
3	TIMER0CLK	RW	0x0	TIMER0 module clock enable 0: Disable clock output 1: Enable clock output
2	ECAPCLK	RW	0x0	ECAP module clock enable 0: Disable clock output 1: Enable clock output
1	PWMCLK	RW	0x0	PWM module clock enable 0: Disable clock output 1: Enable clock output
0	ADCCLK	RW	0x0	ADC module clock enable 0: Disable clock output 1: Enable clock output

Table 3-33: Peripheral Clock Status Register (PERICKSTS) Layout

PERICKSTS (Peripheral Clock Status Register) Offset: 0x38 Default: 0x0000F000 Access: CLOCK->PERICKSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 WDT1	14 WDT0	13 DG	12 AHB	11 CAN	10 I2C	9 SPI1	8 SPI0
7 UART1	6 UART0	5 TIMER2	4 TIMER1	3 TIMER0	2 ECAP	1 PWM	0 ADC

Table 3-34: Peripheral Clock Status Register (PERICKSTS) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	WDT1	RO	0x1	Watchdog 1 clock status 0: Clock is off 1: Clock is on
14	WDT0	RO	0x1	Watchdog 0 clock status 0: Clock is off 1: Clock is on
13	DG	RO	0x1	Deglitch module clock status 0: Clock is off 1: Clock is on
12	AHB	RO	0x1	Bus and CPU clock status 0: Clock is off 1: Clock is on

Bits	Field Name	Type	Reset	Description
11	CAN	RO	0x0	CAN module clock status 0: Clock is off 1: Clock is on
10	I2C	RO	0x0	I2C module clock status 0: Clock is off 1: Clock is on
9	SPI1	RO	0x0	SPI1 module clock status 0: Clock is off 1: Clock is on
8	SPI0	RO	0x0	SPI0 module clock status 0: Clock is off 1: Clock is on
7	UART1	RO	0x0	UART1 module clock status 0: Clock is off 1: Clock is on
6	UART0	RO	0x0	UART0 module clock status 0: Clock is off 1: Clock is on
5	TIMER2	RO	0x0	TIMER2 module clock status 0: Clock is off 1: Clock is on
4	TIMER1	RO	0x0	TIMER1 module clock status 0: Clock is off 1: Clock is on
3	TIMER0	RO	0x0	TIMER0 module clock status 0: Clock is off 1: Clock is on
2	ECAP	RO	0x0	ECAP module clock status 0: Clock is off 1: Clock is on
1	PWM	RO	0x0	PWM module clock status 0: Clock is off 1: Clock is on
0	ADC	RO	0x0	ADC module clock status 0: Clock is off 1: Clock is on

Table 3-35: Clock Register Write-Allow Key Register (CLKREGKEY) Layout

CLKREGKEY (Clock Register Write-Allow Key Register) Offset: 0x3C Default: 0x1ACCE551 Access: CLOCK->CLKREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 3-36: Clock Register Write-Allow Key Register (CLKREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected clock registers

4 General-purpose I/O (GPIO)

4.1 Overview

In SPC1169, the I/Os can be divided into the following categories.

- 3.3V I/O and multiplexed with analog input functions
 - GPIO0 ~ GPIO4, GPIO28 ~ GPIO31: Can be used as analog input channel 0 ~ 8
 - GPIO36 ~ GPIO39: Internal pins which should be used as the default function of BEMF voltage sensing for U-phase, V-phase, W-phase and the common node.
 - GPIO40: Internal pin which should be used as the default function of divided VBAT voltage sensing.
- 3.3V I/O with 5V input tolerance
 - GPIO25, GPIO26: internal pins, connected to the LIN transceiver (GPIO25 shall be used for TXD of UART1, GPIO26 shall be used for RXD).
 - GPIO27: Internal pins connected to the MON module.

Among them, GPIO26 and GPIO27 do not go through PINMUX.

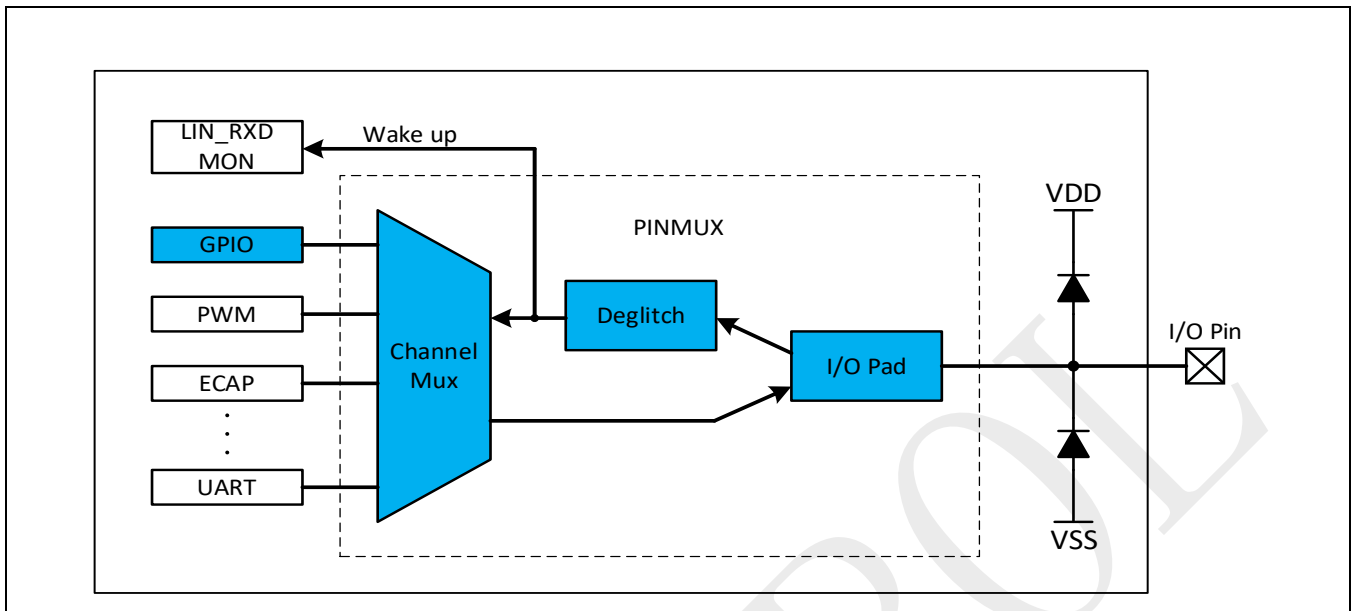
- True 5V I/O
 - GPIO5 ~ GPIO18, GPIO32 ~ GPIO35: Multiplexed with various functions
 - GPIO19 ~ GPIO24: Internal pins connected to the pre-driver module and should be configured as PWM output

This chapter describes how these I/O pins are multiplexed so that they can be configured as either general-purpose I/Os or any one of the alternative functions. As shown in [Figure 4-1](#), the I/O control consists of two parts:

- PINMUX module
 - I/O Pad: Pull-up/pull-down control, input/output control and output driving strength control
 - Deglitch: Input deglitch control
 - MUX: Pin multiplexing
- GPIO module
 - GPIO function IP control

Each I/O pin can be programmed and used independently. And there is one diode from the I/O to VDD and one diode from VSS to the I/O for ESD protection as shown in [Figure 4-2](#).

Figure 4-1: Block diagram of I/O control

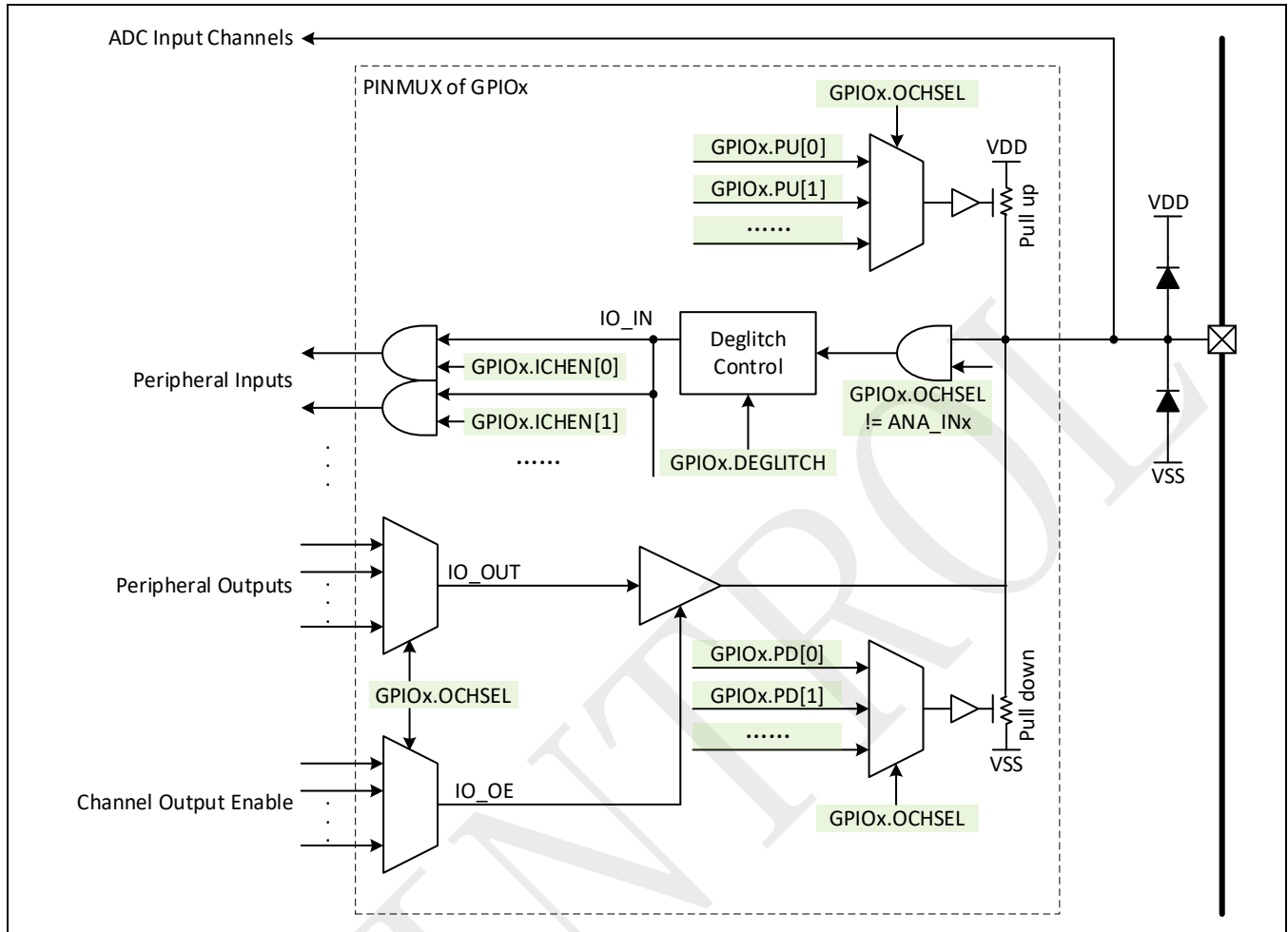


4.2 Pin multiplexing (PINMUX)

The block diagram of the PINMUX is shown in Figure 4-2. Multiplexers are used for each I/O pin to support up to 4 function channels. The function channel can be either input or output. If the I/O pin is used as input, the output buffer will be disabled and set into high-impedance state. The input buffer is disabled only when the I/O is used as analog input to avoid the potential current leakage. For all other cases, the input buffer is always enabled. The input signal can also pass through a filter to suppress the possible glitches.

The multiplexing on output function and pull-up/pull-down control is exclusive, and selected by the corresponding GPIOx.OCHSEL register field. For example, if GPIO4.OCHSEL=3, pull-up and pull-down control for GPIO4 pin is determined by GPIO4.PU[3] and GPIO4.PD[3] respectively, and the output function is set to be channel 3.

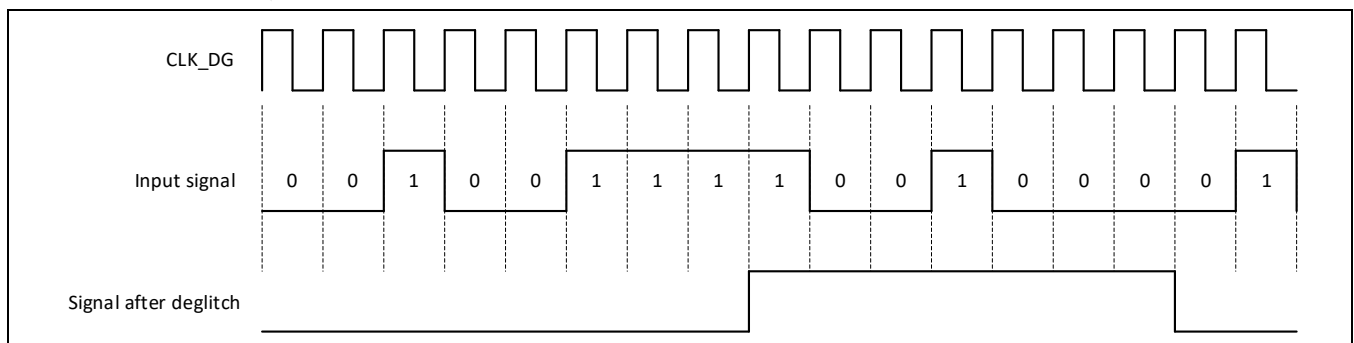
The multiplexing on input function for each channel is non-exclusive. Each channel has its own enable control via GPIOx.ICHEN[y]. Therefore, it can be determined independently whether the GPIOx input will be used as the functional input for channel-y.

Figure 4-2: PINMUX block diagram


Note: When GPIO3 and GPIO4 work in XIN and XOUT, the external crystal and the internal clock circuit are directly connected, and there is no input-output relationship in the figure.

4.2.1 Deglitch control

The input deglitching can be enabled by setting register GPIOx.DEGLITCH. Figure 4-3 shows the principle of the input deglitch. When the input signal toggles to a new value and holds it for 3 consecutive CLK_DG cycles, the output would be updated to the new value, otherwise the output keeps the original value.

Figure 4-3: Input deglitching


4.2.2 I/O alternate functions

The channel functions for all the I/O pins are summarized in [Table 4-1](#).

Table 4-1: I/O Pin function channel definition

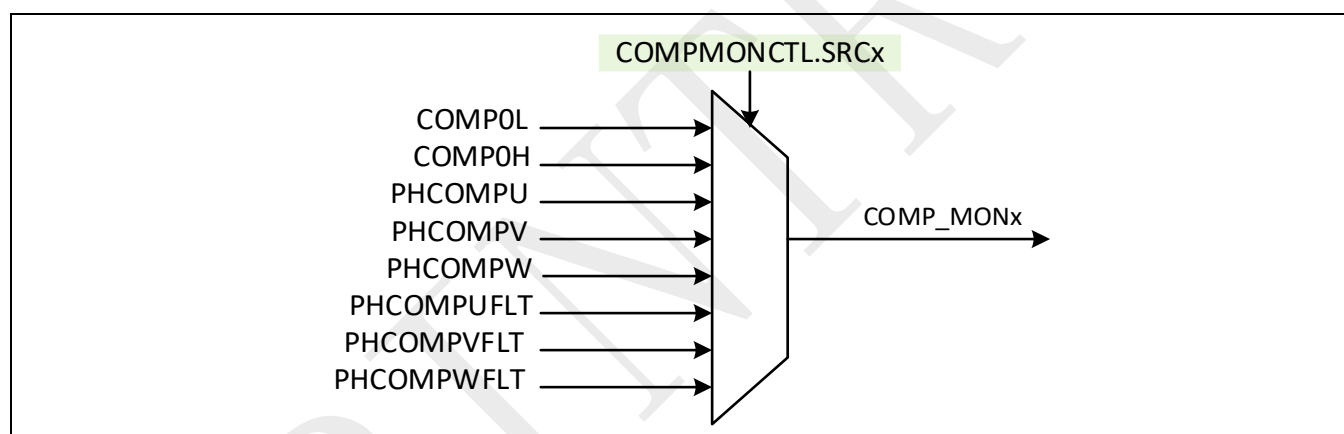
Name	Type	Channel (Default channel is highlight in red)			
		0	1	2	3
PIN_GPIO0	3.3V IO	GPIO0	ANA_IN0	PWMSOCAO	PWMSOCCO
PIN_GPIO1	3.3V IO	GPIO1	ANA_IN1	PWMSOCBO	ECAP_APWMO
PIN_GPIO2	3.3V IO	GPIO2	ANA_IN2	CLK_DCLK	-
PIN_GPIO3	3.3V IO	GPIO3	ANA_IN3	XIN	COMP_MON0
PIN_GPIO4	3.3V IO	GPIO4	ANA_IN4	XIO	COMP_MON1
PIN_GPIO5	5V IO	GPIO5	-	PWMSOCO	ECAP_APWMO
PIN_GPIO6	5V IO	GPIO6	SPI0_SCLK	UART1_TXD	COMP_MON2
PIN_GPIO7	5V IO	GPIO7	SPI0_SFRM	UART1_RXD	COMP_MON3
PIN_GPIO8	5V IO	GPIO8	SPI0_MOSI	CAN_TXD	COMP_MON4
PIN_GPIO9	5V IO	GPIO9	SPI0_MISO	CAN_RXD	COMP_MON5
PIN_GPIO10	5V IO	GPIO10	UART0_TXD	PWM0A	I2C_SCL
PIN_GPIO11	5V IO	GPIO11	UART0_RXD	PWM0B	I2C_SDA
PIN_GPIO12	5V IO	GPIO12	SPI1_SCLK	PWM1A	PWM3A
PIN_GPIO13	5V IO	GPIO13	SPI1_SFRM	PWM1B	PWM3B
PIN_GPIO14	5V IO	GPIO14	SPI1_MOSI	PWM2A	I2C_SCL
PIN_GPIO15	5V IO	GPIO15	SPI1_MISO	PWM2B	I2C_SDA
PIN_GPIO16	5V IO	GPIO16	UART1_TXD	CAN_TXD	SPI1_SFRM
PIN_GPIO17	5V IO	GPIO17	UART1_RXD	CAN_RXD	SPI1_SCLK
PIN_GPIO18	5V IO	GPIO18	PWMSOCO	PWMSYNCO	ECAP_APWMO
PIN_GPIO19	5V IO	GPIO19	PWM2B	PWM0A	ECAP_APWMO
PIN_GPIO20	5V IO	GPIO20	PWM2A	PWM1A	ECAP_APWMO
PIN_GPIO21	5V IO	GPIO21	PWM1B	PWM2A	ECAP_APWMO
PIN_GPIO22	5V IO	GPIO22	PWM1A	PWM0B	ECAP_APWMO
PIN_GPIO23	5V IO	GPIO23	PWM0B	PWM1B	ECAP_APWMO
PIN_GPIO24	5V IO	GPIO24	PWM0A	PWM2B	ECAP_APWMO
PIN_GPIO25	3.3V/5V IO	GPIO25	UART1_TXD	CAN_TXD	I2C_SCL
PIN_GPIO26	3.3V/5V IO	GPIO26	UART1_RXD	CAN_RXD	I2C_SDA
PIN_GPIO27	3.3V/5V IO	GPIO27	-	-	-
PIN_GPIO28	3.3V IO	GPIO28	ANA_IN5	SPI1_SCLK	PWMSOCAO
PIN_GPIO29	3.3V IO	GPIO29	ANA_IN6	SPI1_SFRM	PWMSOCBO
PIN_GPIO30	3.3V IO	GPIO30	ANA_IN7	SPI1_MOSI	PWMSOCCO
PIN_GPIO31	3.3V IO	GPIO31	ANA_IN8	SPI1_MISO	PWMSYNCO
PIN_GPIO32	5V IO	GPIO32	UART1_TXD	I2C_SCL	COMP_MON6
PIN_GPIO33	5V IO	GPIO33	UART1_RXD	I2C_SDA	COMP_MON7
PIN_GPIO34	5V IO	GPIO34	CAN_TXD	I2C_SCL	-

Name	Type	Channel (Default channel is highlight in red)			
		0	1	2	3
PIN_GPIO35	5V IO	GPIO35	CAN_RXD	I2C_SDA	-
PIN_GPIO36	3.3V IO	GPIO36	ANA_IN9(PHU)	SPI1_SCLK	UART1_TXD
PIN_GPIO37	3.3V IO	GPIO37	ANA_IN10(PHV)	SPI1_SFRM	UART1_RXD
PIN_GPIO38	3.3V IO	GPIO38	ANA_IN11(PHW)	SPI1_MOSI	CAN_TXD
PIN_GPIO39	3.3V IO	GPIO39	ANA_IN12(PHC)	SPI1_MISO	CAN_RXD
PIN_GPIO40	3.3V IO	GPIO40	ANA_IN13(VBAT)	-	-

- [1] All GPIO pins can be configured as ECAP input via channel 0 (GPIO function channel)
- [2] In SPC1169, GPIO19 ~ GPIO27, and GPIO36 ~ GPIO40 are not bonded out to the external pin, reserved for other purposes, as shown in [Figure 1-1](#).
- [3] Note: GPIO5 is BOOT pin and cannot be configured for other functions in the chip startup phase, otherwise it will cause a constant level of GPIO5 and affect the entry of ISP mode or Flash startup mode.

The channel functions for COMP_MONx are summarized in [Figure 4-4](#).

Figure 4-4: Select source of COMP_MONx



4.2.3 Loopback path for communication

To provide loopback test testing on I/O ports, a loopback option is provided for ports with communication data functions, including SPIx_MISO, SPIx_MOSI, UARTx_RXD, UARTx_TXD, CAN_RXD, CAN_TXD, controlled by the corresponding GPIOx.LPBK register bit.

When the path is enabled, the output channel selected by GPIOx.OCHSEL will not only output to I/O, but also loop back to the input and can be used as input to channel y by GPIOx.ICHEN[y].

For example, if both ICHEN[1] and LPBK bit of PINMUX->GPIO10 is 1, the UART0_TXD output to the GPIO10 will be looped back to UART0_RXD.

4.2.4 Debug interface

When TRSTn pin is pulled-down, GPIO15 ~ GPIO18 pins are used as the functional channel controlled by the software. When the TRSTn pin is pulled-up, they would be used for the debug interface as defined in [Table 4-2](#).

Table 4-2: Debug interface definition

Pin Name	Debug interface signal	
	SYSTEM->DBGIFCTL=0 (SWD mode)	SYSTEM->DBGIFCTL=1 (JTAG mode)
GPIO15	Software controlled function channel	TDI
GPIO16	SWV ^[1]	TDO
GPIO17	SWD	TMS
GPIO18	SWCK	TCK

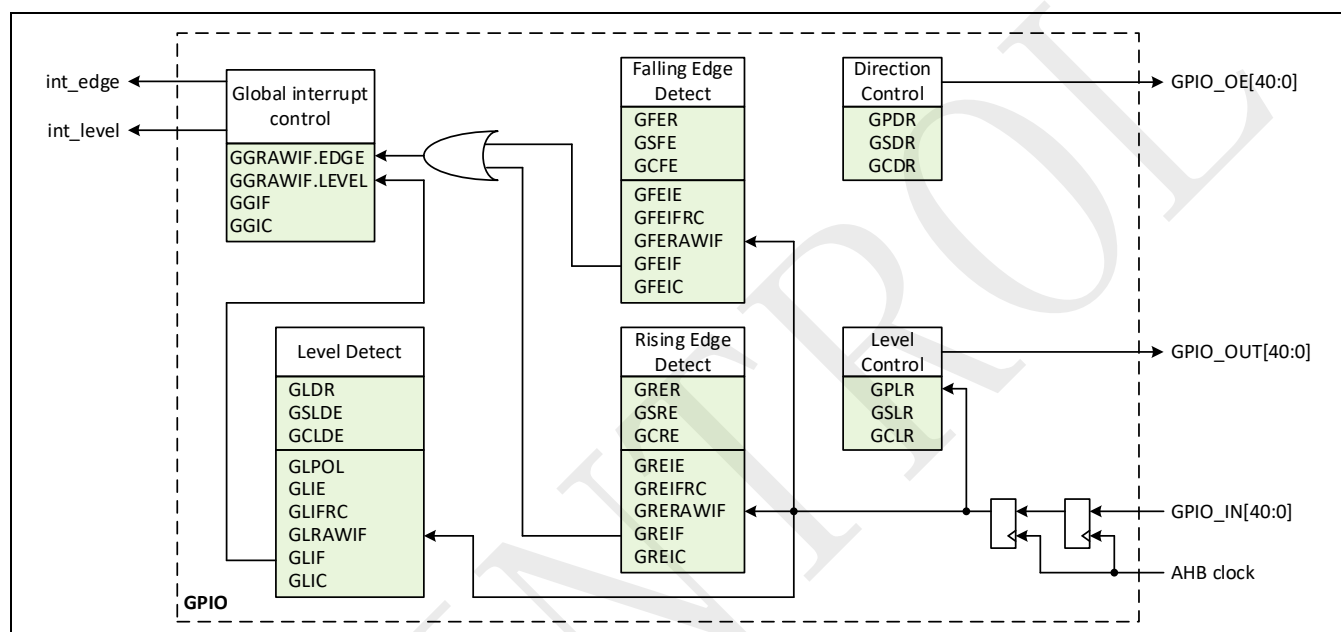
[1] User can disable SWV output function in SWD mode so as to use GPIO16 for normal function channel by code "CoreDebug->DEMCR &= ~(1U << 24);". It clears bit 24 for register at 0xE000EDFC.

4.3 GPIO control module (GPIO)

As shown in Figure 4-5, the GPIO control module consists of level control, direction control, edge detection and level detection.

Each bit in the register corresponds to a GPIO pin. For ease of use, bitwise control registers are provided so user can simply set control bits via writing 1 to “GS-” register bits or clear control bits via writing 1 to “GC-” register bits. Writing 0 to these bitwise control registers has no effect.

Figure 4-5: GPIO block diagram



4.3.1 Direction and level control

All the GPIO pins are used as input by default. The direction of the GPIO pins can be configured by programming the GPDR[x]. User can also write 1 to GSDR bits so as to set those pins as output, or write 1 to GCDR bits so as to set those pins as input. The GPDR register will be updated accordingly.

When configured as output pin, the output level can be programmed via GPLR[x] or GSLR[x]/GCLR[x] registers. No matter what has been written to GPLR[x] or GPDR[x] register, readback from GPLR[x] register will always reflect the GPIO ports status, i.e. the level of the I/O pins.

Please note that cross domain input resampling is applied to avoid logic metastability. Therefore, if the level is changed via writing GPLR[x] register for the output pin, the readback from GPLR[x] register will not be updated to the written value until 2 AHB clocks later. For this reason, to generate waveform on the output I/O pin by software, it is recommended to write GSLR[x] register bits to output high, and write GCLR[x] register bits to output low. Coding style via read-modify-write GPLR[x] register does not work as expected unless manual delay (i.e. NOP instruction) is inserted after the write.

4.3.2 Edge-triggered interrupt

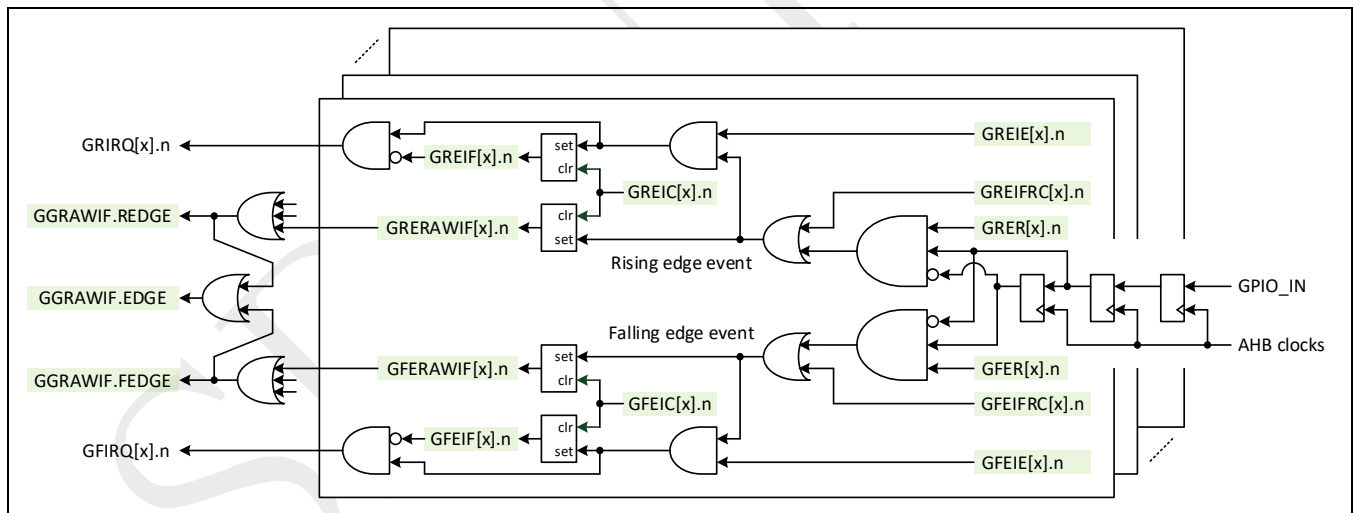
As shown in Figure 4-6, both the rising edge event and the falling edge event can be detected on the resampled pin input using AHB clock. This feature is enabled when $GRER[x]$ or $GFER[x]$ is set. $GSRE[x]/GSFE[x]$ and $GCRE[x]/GCFE[x]$ are bitwise control registers, which can also be used to set or clear $GREF[x]/GFER[x]$ register bits.

Whenever the rising edge event or the falling edge event is detected on the resampled pin input, it will be always latched into the corresponding bit in $GRERAWIF[x]$ or $GFERAWIF[x]$ register. On the contrary, the interrupt will not be triggered and latched as $GREIF[x]$ or $GFEIF[x]$ interrupt flag unless it is enabled by setting the corresponding bit in $GREIE[x]$ or $GFEIE[x]$. Write $GREIC[x]$ or $GFEIC[x]$ bit with 1 will clear both the latched event flag in $GRERAWIF[x]/GFERAWIF[x]$ and the interrupt flag in $GREIF[x]$ or $GFEIF[x]$.

For ease of debug, the rising edge and falling edge event can also be software forced via writing 1 to $GREIFRC[x]$ and $GFEIFRC[x]$ register bits.

The $REDGE$ bit in $GGRWIF$ register is the global rising edge event flag, which is the logic OR of all $GRERAWIF$ bits. Similarly, The $FEDGE$ bit in $GGRWIF$ register is the global falling edge event flag, which is the logic OR of all $GFERAWIF$ bits. The $REDGE$ and $FEDGE$ are Ored to generate the $EDGE$ bit in $GGRWIF$ register. Therefore, whenever the rising edge is detected on any pin input, both $REDGE$ bit and $EDGE$ bit of $GGRWIF$ would be latched as 1. Whenever the falling edge is detected on any pin input, both $FEDGE$ bit and $EDGE$ bit of $GGRWIF$ would be latched as 1.

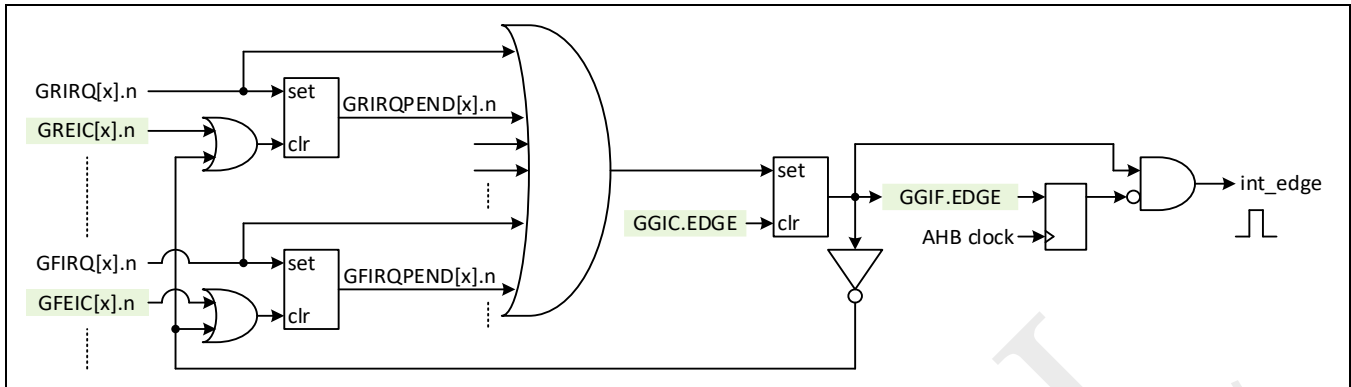
Figure 4-6: GPIO edge detect



The signals named as $GRIRQ[x].n$ and $GFIRQ[x].n$ in Figure 4-6 are interrupt requests triggered by rising and falling edges, and they can trigger both rising and falling edge interrupt events, but it is important to note that if the value in the $GREIF[x]$ or $GFEIF[x]$ register is 1, it will not be possible to trigger the $GRIRQ[x].n$ and $GFIRQ[x].n$ interrupt requests, so remember to clear the $GREIF[x]$ or $GFEIF[x]$ registers after interrupt processing is complete.

As shown in Figure 4-7, all the valid interrupt requests are Ored together to generate the final edge-triggered interrupt to the CPU, and latched to the $EDGE$ bit of the global interrupt flag register $GGIF$. The latched flag can be cleared via writing 1 to $EDGE$ bit of $GGIC$ register.

Figure 4-7: GPIO edge-triggered interrupt



If PIN x triggers an edge interrupt request first, the interrupt flag register GGIF.EDGE is set to 1 and the interrupt is triggered, then the edge interrupt requests of PIN y and PIN z will be pending at the same time.

If the PIN x interrupt service program is executed and writes 1 to the interrupt clear register GGIC.EDGE, then the GGIF.EDGE flag bit is cleared to 0. Subsequently, on the next beat of the clock, two actions will occur:

1. the PIN y, PIN z pending request again sets the interrupt flag register GGIF.EDGE flag bit 1 and triggers an interrupt;
2. the PIN x interrupt clear action (writing 1 to the interrupt clear register GGIC.EDGE) will clear the PIN y, PIN z pending interrupt request via the inverter.

To summarize: when one interrupt is responded to, there is only one subsequent interrupt that can be responded to, no matter how many other interrupts occur.

As mentioned above, when there are two interrupt requests at the same time, only one interrupt request will be responded to. In this case, if you want to handle all the requests, it is recommended to implement the ISR program according to the following steps:

- First read the GREIF or GFEIF register to determine the request to be processed;
- Process the request according to the request;
- Clear the GREIF or GFEIF registers according to the requests processed;
- Clear the GGIF.EDGE bit (otherwise, GPIO edge-triggered interrupt generation will be blocked).

4.3.3 Level-triggered interrupt

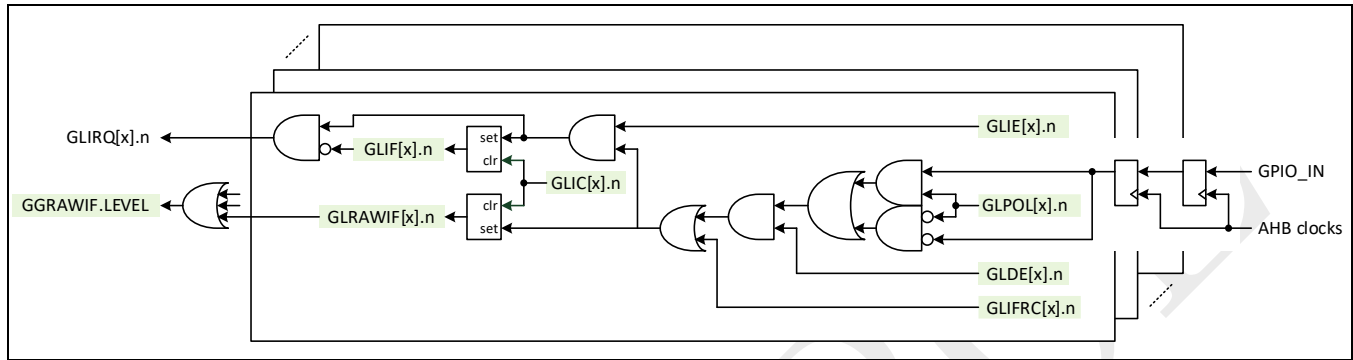
As shown in Figure 4-8, the level event can be detected on the resampled pin input using AHB clock. This feature is enabled when GLDE[x] is set. GSLDE[x] and GCLDE[x] are bitwise control registers, which can also be used to set or clear GLDE[x] register bits.

Whenever the level of the resampled pin input is the same as defined in GLPOL[x], it will be regarded as a valid level event and always latched into the corresponding bit in GLRAWIF[x] register. On the contrary,

the interrupt will not be triggered and latched as GLIF[x] interrupt flag unless it is enabled by setting the corresponding bit in GLIE[x]. Write GLIC[x] bit with 1 will clear both the latched event flag in GLRAWIF[x] and the interrupt flag in GLIF[x].

For ease of debug, the level event can also be software forced via writing 1 to GLIFRC[x] register bits.

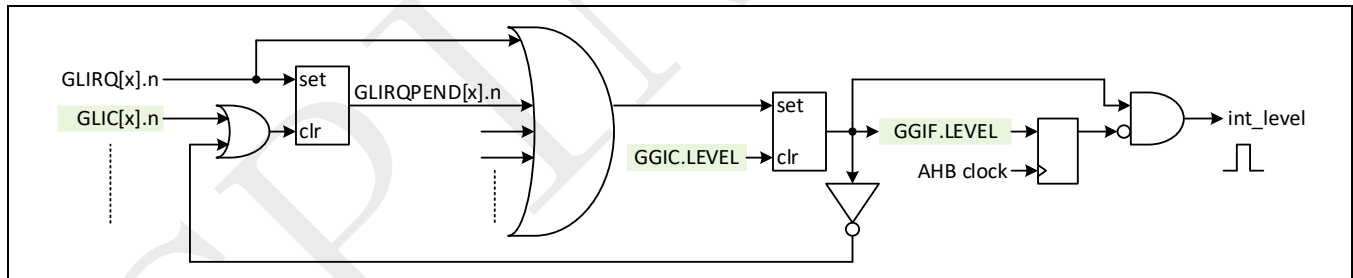
Figure 4-8: GPIO level detect



The LEVEL bit in GRAWIF register is the global level event flag, which is the logic OR of all GLRAWIF bits. Therefore, upon the valid level detection on any pin input, LEVEL bit of GRAWIF would be latched as 1.

The signals named as GLIRQ[x].n in Figure 4-8 are valid level triggered interrupt requests. They are the enabled level event while the corresponding interrupt flag bit in GLIF[x] register is still 0. Namely, only the new event can trigger the interrupt.

Figure 4-9: GPIO level-triggered interrupt



As shown in Figure 4-9, all the valid interrupt requests are ORed together to generate the final level-triggered interrupt to the CPU, and latched to the LEVEL bit of the global interrupt flag register GGIF. The latched flag can be cleared via writing 1 to LEVEL bit of GGIC register.

If PIN x triggers an level interrupt request first, the interrupt flag register GGIF.LEVEL is set to 1 and the interrupt is triggered, then the level interrupt requests of PIN y and PIN z will be pending at the same time.

If the PIN x interrupt service program is executed and writes 1 to the interrupt clear register GGIC.LEVEL, then the GGIF.LEVEL flag bit is cleared to 0. Subsequently, on the next beat of the clock, two actions will occur:

1. The PIN y, PIN z pending request again sets the interrupt flag register GGIF.LEVEL flag bit 1 and triggers an interrupt;

2. The PIN x interrupt clear action (writing 1 to the interrupt clear register GGIC.LEVEL) will clear the PIN y, PIN z pending interrupt request via the inverter.

To summarize: when one interrupt is responded to, there is only one subsequent interrupt that can be responded to, no matter how many other interrupts occur.

As mentioned above, when there are two interrupt requests at the same time, only one interrupt request will be responded to. In this case, if you want to handle all the requests, it is recommended to implement the ISR program according to the following steps:

- First read the GLIF register to determine the request to be processed;
- Process the request according to the request;
- Clear the GLIF register according to the requests processed;
- Clear the GGIF.LEVEL bit (otherwise, GPIO level-triggered interrupt generation will be blocked).

4.4 Registers

4.4.1 PINMUX register map

Table 4-3: PINMUX Module Base Address

Peripheral Module	Base Address
PINMUX	0x4000 0300

Table 4-4: PINMUX Register Map

Register	Offset	Description	Reset Value
GPIO0*	0x00	GPIO0 Pin Control Register	0x00010001
GPIO1*	0x04	GPIO1 Pin Control Register	0x00010001
GPIO2*	0x08	GPIO2 Pin Control Register	0x00010001
GPIO3*	0x0C	GPIO3 Pin Control Register	0x00010001
GPIO4*	0x10	GPIO4 Pin Control Register	0x00010001
GPIO5*	0x14	GPIO5 Pin Control Register	0x00010100
GPIO6*	0x18	GPIO6 Pin Control Register	0x05000100
GPIO7*	0x1C	GPIO7 Pin Control Register	0x05000100
GPIO8*	0x20	GPIO8 Pin Control Register	0x05000100
GPIO9*	0x24	GPIO9 Pin Control Register	0x05000100
GPIO10*	0x28	GPIO10 Pin Control Register	0x0B000100
GPIO11*	0x2C	GPIO11 Pin Control Register	0x0B000100
GPIO12*	0x30	GPIO12 Pin Control Register	0x00000100
GPIO13*	0x34	GPIO13 Pin Control Register	0x00000100
GPIO14*	0x38	GPIO14 Pin Control Register	0x09000100
GPIO15*	0x3C	GPIO15 Pin Control Register	0x09000100
GPIO16*	0x40	GPIO16 Pin Control Register	0x07000100
GPIO17*	0x44	GPIO17 Pin Control Register	0x07000100
GPIO18*	0x48	GPIO18 Pin Control Register	0x00010100
GPIO19*	0x4C	GPIO19 Pin Control Register	0x00000100
GPIO20*	0x50	GPIO20 Pin Control Register	0x00000100
GPIO21*	0x54	GPIO21 Pin Control Register	0x00000100
GPIO22*	0x58	GPIO22 Pin Control Register	0x00000100
GPIO23*	0x5C	GPIO23 Pin Control Register	0x00000100
GPIO24*	0x60	GPIO24 Pin Control Register	0x00000100
GPIO25*	0x64	GPIO25 Pin Control Register	0x0F000100
GPIO26*	0x68	GPIO26 Pin Control Register	0x0F000100
GPIO27*	0x6C	GPIO27 Pin Control Register	0x00010100
GPIO28*	0x70	GPIO28 Pin Control Register	0x00010001
GPIO29*	0x74	GPIO29 Pin Control Register	0x00010001
GPIO30*	0x78	GPIO30 Pin Control Register	0x00010001

Register	Offset	Description	Reset Value
GPIO31 *	0x7C	GPIO31 Pin Control Register	0x00010001
GPIO32 *	0x80	GPIO32 Pin Control Register	0x07000100
GPIO33 *	0x84	GPIO33 Pin Control Register	0x07000100
GPIO34 *	0x88	GPIO34 Pin Control Register	0x07000100
GPIO35 *	0x8C	GPIO35 Pin Control Register	0x07000100
GPIO36 *	0x90	GPIO36 Pin Control Register	0x09000001
GPIO37 *	0x94	GPIO37 Pin Control Register	0x09000001
GPIO38 *	0x98	GPIO38 Pin Control Register	0x09000001
GPIO39 *	0x9C	GPIO39 Pin Control Register	0x09000001
GPIO40 *	0xA0	GPIO40 Pin Control Register	0x00010001
COMPMONCTL *	0xA4	Comparator Output Monitor Control Register	0x76543210
PINMUXREGKEY	0xA8	PINMUX Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the PINMUXREGKEY = 0x1ACCE551.

4.4.2 PINMUX registers

Table 4-5: GPIO0 Pin Control Register (GPIO0) Layout

GPIO0 (GPIO0 Pin Control Register) Offset: 0x0 Default: 0x00010001 Access: PINMUX->GPIO0											
31	30	29	28	27	26	25		24			
RESERVED						PU					
23	22	21	20	19	18	17		16			
RESERVED						PD					
15	14	13	12	11	10	9		8			
RESERVED								ICHEN			
7	6	5	4	3	2	1		0			
STRENGTH		DEGLITCH	RESERVED			OCHSEL					

Table 4-6: GPIO0 Pin Control Register (GPIO0) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	PU	RW	0x0	Pull up enable
23:18	RESERVED_23_18	RO	0x0	Reserved.
17:16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO0 01: ANA_IN0 10: PWM SOCA output for monitor 11: PWM SOCC output for monitor

Table 4-7: GPIO1 Pin Control Register (GPIO1) Layout

GPIO1 (GPIO1 Pin Control Register) Offset: 0x4 Default: 0x00010001 Access: PINMUX->GPIO1											
31	30	29	28	27	26	25		24			
RESERVED						PU					
23	22	21	20	19	18	17		16			
RESERVED						PD					
15	14	13	12	11	10	9		8			
RESERVED								ICHEN			
7	6	5	4	3	2	1		0			
STRENGTH		DEGLITCH	RESERVED			OCHSEL					

Table 4-8: GPIO1 Pin Control Register (GPIO1) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	PU	RW	0x0	Pull up enable
23:18	RESERVED_23_18	RO	0x0	Reserved.
17:16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO1 01: ANA_IN1 10: PWM SOCB output for monitor 11: ECAP APWM mode output

Table 4-9: GPIO2 Pin Control Register (GPIO2) Layout

GPIO2 (GPIO2 Pin Control Register) Offset: 0x8 Default: 0x00010001 Access: PINMUX->GPIO2											
31	30	29	28	27	26	25		24			
RESERVED						PU					
23	22	21	20	19	18	17		16			
RESERVED						PD					
15	14	13	12	11	10	9			8		
RESERVED									ICHEN		
7	6	5	4	3	2	1		0			
STRENGTH		DEGLITCH	RESERVED			OCHSEL					

Table 4-10: GPIO2 Pin Control Register (GPIO2) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	PU	RW	0x0	Pull up enable
23:18	RESERVED_23_18	RO	0x0	Reserved.
17:16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO2 01: ANA_IN2 10: CLK_DCLK 11: Reserve

Table 4-11: GPIO3 Pin Control Register (GPIO3) Layout

GPIO3 (GPIO3 Pin Control Register) Offset: 0xC Default: 0x00010001 Access: PINMUX->GPIO3											
31	30	29	28	27	26	25		24			
RESERVED						PU					
23	22	21	20	19	18	17		16			
RESERVED						PD					
15	14	13	12	11	10	9		8			
RESERVED								ICHEN			
7	6	5	4	3	2	1		0			
STRENGTH		DEGLITCH	RESERVED			OCHSEL					

Table 4-12: GPIO3 Pin Control Register (GPIO3) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	PU	RW	0x0	Pull up enable
23:18	RESERVED_23_18	RO	0x0	Reserved.
17:16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO3 01: ANA_IN3 10: XIN 11: COMP_MON0

Table 4-13: GPIO4 Pin Control Register (GPIO4) Layout

GPIO4 (GPIO4 Pin Control Register) Offset: 0x10 Default: 0x00010001 Access: PINMUX->GPIO4											
31	30	29	28	27	26	25		24			
RESERVED						PU					
23	22	21	20	19	18	17		16			
RESERVED						PD					
15	14	13	12	11	10	9			8		
RESERVED									ICHEN		
7	6	5	4	3	2	1		0			
STRENGTH		DEGLITCH	RESERVED			OCHSEL					

Table 4-14: GPIO4 Pin Control Register (GPIO4) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	PU	RW	0x0	Pull up enable
23:18	RESERVED_23_18	RO	0x0	Reserved.
17:16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO4 01: ANA_IN4 10: XOUT 11: COMP_MON1

Table 4-15: GPIO5 Pin Control Register (GPIO5) Layout

GPIO5 (GPIO5 Pin Control Register) Offset: 0x14 Default: 0x00010100 Access: PINMUX->GPIO5							
31	30	29	28	27	26	25	24
RESERVED							PU
23	22	21	20	19	18	17	16
RESERVED							PD
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

Table 4-16: GPIO5 Pin Control Register (GPIO5) Description

Bits	Field Name	Type	Reset	Description
31:25	RESERVED_31_25	RO	0x0	Reserved.
24	PU	RW	0x0	Pull up enable
23:17	RESERVED_23_17	RO	0x0	Reserved.
16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO5 01: Reserve 10: PWM SOC output for monitor It is logic OR of PWM_SOCAO, PWM_SOCBO and PWM_SOCCO 11: ECAP APWM mode output

Table 4-17: GPIO6 Pin Control Register (GPIO6) Layout

GPIO6 (GPIO6 Pin Control Register) Offset: 0x18 Default: 0x05000100 Access: PINMUX->GPIO6								
31	30	29	28	27	26	25	24	
RESERVED						PU		
23	22	21	20	19	18	17	16	
RESERVED						PD		
15	14	13	12	11	10	9	8	
RESERVED						ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-18: GPIO6 Pin Control Register (GPIO6) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x5	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Channel select 00: GPIO6 01: SPI0_SCLK 10: UART1_TXD 11: COMP_MON2

Table 4-19: GPIO7 Pin Control Register (GPIO7) Layout

GPIO7 (GPIO7 Pin Control Register) Offset: 0x1C Default: 0x05000100 Access: PINMUX->GPIO7								
31	30	29	28	27	26	25	24	
		RESERVED				PU		
23	22	21	20	19	18	17	16	
		RESERVED				PD		
15	14	13	12	11	10	9	8	
		RESERVED				ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-20: GPIO7 Pin Control Register (GPIO7) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x5	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO7 01: SPI0_SFRM 10: UART1_RXD 11: COMP_MON3

Table 4-21: GPIO8 Pin Control Register (GPIO8) Layout

GPIO8 (GPIO8 Pin Control Register) Offset: 0x20 Default: 0x05000100 Access: PINMUX->GPIO8								
31	30	29	28	27	26	25	24	
		RESERVED				PU		
23	22	21	20	19	18	17	16	
		RESERVED				PD		
15	14	13	12	11	10	9	8	
		RESERVED				ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-22: GPIO8 Pin Control Register (GPIO8) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x5	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI0 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO8 01: SPI0_MOSI 10: CAN_TXD 11: COMP_MON4

Table 4-23: GPIO9 Pin Control Register (GPIO9) Layout

GPIO9 (GPIO9 Pin Control Register) Offset: 0x24 Default: 0x05000100 Access: PINMUX->GPIO9								
31	30	29	28	27	26	25	24	
RESERVED						PU		
23	22	21	20	19	18	17	16	
RESERVED						PD		
15	14	13	12	11	10	9	8	
RESERVED						ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-24: GPIO9 Pin Control Register (GPIO9) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x5	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI0 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO9 01: SPI0_MISO 10: CAN_RXD 11: COMP_MON5

Table 4-25: GPIO10 Pin Control Register (GPIO10) Layout

GPIO10 (GPIO10 Pin Control Register) Offset: 0x28 Default: 0x0B000100 Access: PINMUX->GPIO10							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-26: GPIO10 Pin Control Register (GPIO10) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0xB	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART0 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO10 01: UART0_TXD 10: PWM0A 11: I2C_SCL

Table 4-27: GPIO11 Pin Control Register (GPIO11) Layout

GPIO11 (GPIO11 Pin Control Register) Offset: 0x2C Default: 0x0B000100 Access: PINMUX->GPIO11							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-28: GPIO11 Pin Control Register (GPIO11) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0xB	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART0 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO11 01: UART0_RXD 10: PWM0B 11: I2C_SDA

Table 4-29: GPIO12 Pin Control Register (GPIO12) Layout

GPIO12 (GPIO12 Pin Control Register) Offset: 0x30 Default: 0x00000100 Access: PINMUX->GPIO12							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED						ICHEN	
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

Table 4-30: GPIO12 Pin Control Register (GPIO12) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x0	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:10	RESERVED_15_10	RO	0x0	Reserved.
9:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO12 01: SPI1_SCLK 10: PWM1A 11: PWM3A

Table 4-31: GPIO13 Pin Control Register (GPIO13) Layout

GPIO13 (GPIO13 Pin Control Register) Offset: 0x34 Default: 0x00000100 Access: PINMUX->GPIO13							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED						ICHEN	
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

Table 4-32: GPIO13 Pin Control Register (GPIO13) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x0	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:10	RESERVED_15_10	RO	0x0	Reserved.
9:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO13 01: SPI1_SFRM 10: PWM1B 11: PWM3B

Table 4-33: GPIO14 Pin Control Register (GPIO14) Layout

GPIO14 (GPIO14 Pin Control Register) Offset: 0x38 Default: 0x09000100 Access: PINMUX->GPIO14							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-34: GPIO14 Pin Control Register (GPIO14) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x9	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO14 01: SPI1_MOSI 10: PWM2A 11: I2C_SCL

Table 4-35: GPIO15 Pin Control Register (GPIO15) Layout

GPIO15 (GPIO15 Pin Control Register) Offset: 0x3C Default: 0x09000100 Access: PINMUX->GPIO15							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-36: GPIO15 Pin Control Register (GPIO15) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x9	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO15 01: SPI1_MISO 10: PWM2B 11: I2C_SDA

Table 4-37: GPIO16 Pin Control Register (GPIO16) Layout

GPIO16 (GPIO16 Pin Control Register) Offset: 0x40 Default: 0x07000100 Access: PINMUX->GPIO16							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-38: GPIO16 Pin Control Register (GPIO16) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x7	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO16 01: UART1_TXD 10: CAN_TXD 11: SPI1_SFRM

Table 4-39: GPIO17 Pin Control Register (GPIO17) Layout

GPIO17 (GPIO17 Pin Control Register) Offset: 0x44 Default: 0x07000100 Access: PINMUX->GPIO17							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-40: GPIO17 Pin Control Register (GPIO17) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x7	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO17 01: UART1_RXD 10: CAN_RXD 11: SPI1_SCLK

Table 4-41: GPIO18 Pin Control Register (GPIO18) Layout

GPIO18 (GPIO18 Pin Control Register) Offset: 0x48 Default: 0x00010100 Access: PINMUX->GPIO18							
31	30	29	28	27	26	25	24
RESERVED							PU
23	22	21	20	19	18	17	16
RESERVED							PD
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL	

Table 4-42: GPIO18 Pin Control Register (GPIO18) Description

Bits	Field Name	Type	Reset	Description
31:25	RESERVED_31_25	RO	0x0	Reserved.
24	PU	RW	0x0	Pull up enable
23:17	RESERVED_23_17	RO	0x0	Reserved.
16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO18 01: PWM_SOC output for monitor It is logic OR of PWM_SOCAO, PWM_SOCBO and PWM_SOCCO 10: PWM_SYNC output for monitor 11: ECAP APWM mode output

Table 4-43: GPIO19 Pin Control Register (GPIO19) Layout

GPIO19 (GPIO19 Pin Control Register) Offset: 0x4C Default: 0x00000100 Access: PINMUX->GPIO19									
31 30 29 28 27					26 25 24				
RESERVED					PU				
23 22 21 20 19					18 17 16				
RESERVED					PD				
15 14 13 12 11 10 9							8		
RESERVED							ICHEN		
7	6	5	4 3 2			1	0		
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL			

Table 4-44: GPIO19 Pin Control Register (GPIO19) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO19 01: PWM2B 10: PWM0A 11: ECAP APWM mode output

Table 4-45: GPIO20 Pin Control Register (GPIO20) Layout

GPIO20 (GPIO20 Pin Control Register)										Offset: 0x50		Default: 0x00000100			
Access: PINMUX->GPIO20															
31		30		29		28		27		26		25		24	
RESERVED										PU					
23		22		21		20		19		18		17		16	
RESERVED										PD					
15		14		13		12		11		10		9		8	
RESERVED													ICHEN		
7		6		5		4		3		2		1		0	
RESERVED		STRENGTH		DEGLITCH		RESERVED						OCHSEL			

Table 4-46: GPIO20 Pin Control Register (GPIO20) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO20 01: PWM2A 10: PWM1A 11: ECAP APWM mode output

Table 4-47: GPIO21 Pin Control Register (GPIO21) Layout

GPIO21 (GPIO21 Pin Control Register) Offset: 0x54 Default: 0x00000100 Access: PINMUX->GPIO21									
31 30 29 28 27					26 25 24				
RESERVED					PU				
23 22 21 20 19					18 17 16				
RESERVED					PD				
15 14 13 12 11					10 9			8	
RESERVED								ICHEN	
7	6	5	4	3	2	1	0		
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL			

Table 4-48: GPIO21 Pin Control Register (GPIO21) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO21 01: PWM1B 10: PWM2A 11: ECAP APWM mode output

Table 4-49: GPIO22 Pin Control Register (GPIO22) Layout

GPIO22 (GPIO22 Pin Control Register)										Offset: 0x58		Default: 0x00000100			
Access: PINMUX->GPIO22															
31		30		29		28		27		26		25		24	
RESERVED										PU					
23		22		21		20		19		18		17		16	
RESERVED										PD					
15		14		13		12		11		10		9		8	
RESERVED										ICHEN					
7		6		5		4		3		2		1		0	
RESERVED		STRENGTH		DEGLITCH		RESERVED						OCHSEL			

Table 4-50: GPIO22 Pin Control Register (GPIO22) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO22 01: PWM1A 10: PWM0B 11: ECAP APWM mode output

Table 4-51: GPIO23 Pin Control Register (GPIO23) Layout

GPIO23 (GPIO23 Pin Control Register) Offset: 0x5C Default: 0x00000100 Access: PINMUX->GPIO23									
31 30 29 28 27					26 25 24				
RESERVED					PU				
23 22 21 20 19					18 17 16				
RESERVED					PD				
15 14 13 12 11					10 9			8	
RESERVED								ICHEN	
7	6	5	4 3 2			1		0	
RESERVED	STRENGTH	DEGLITCH	RESERVED			OCHSEL			

Table 4-52: GPIO23 Pin Control Register (GPIO23) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO23 01: PWM0B 10: PWM1B 11: ECAP APWM mode output

Table 4-53: GPIO24 Pin Control Register (GPIO24) Layout

GPIO24 (GPIO24 Pin Control Register) Offset: 0x60 Default: 0x00000100 Access: PINMUX->GPIO24									
31 30 29 28 27					26		25		24
RESERVED							PU		
23 22 21 20 19					18		17		16
RESERVED							PD		
15 14 13 12 11					10		9		8
RESERVED									ICHEN
7	6	5	4		3	2	1		0
RESERVED	STRENGTH	DEGLITCH	RESERVED				OCHSEL		

Table 4-54: GPIO24 Pin Control Register (GPIO24) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO24 01: PWM0A 10: PWM2B 11: ECAP APWM mode output

Table 4-55: GPIO25 Pin Control Register (GPIO25) Layout

GPIO25 (GPIO25 Pin Control Register) Offset: 0x64 Default: 0x0F000100 Access: PINMUX->GPIO25							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-56: GPIO25 Pin Control Register (GPIO25) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0xF	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO25 01: UART1_TXD 10: CAN_TXD 11: I2C_SCL

Table 4-57: GPIO26 Pin Control Register (GPIO26) Layout

GPIO26 (GPIO26 Pin Control Register) Offset: 0x68 Default: 0x0F000100 Access: PINMUX->GPIO26							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-58: GPIO26 Pin Control Register (GPIO26) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0xF	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x1	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO26 01: UART1_RXD 10: CAN_RXD 11: I2C_SDA

Table 4-59: GPIO27 Pin Control Register (GPIO27) Layout

GPIO27 (GPIO27 Pin Control Register) Offset: 0x6C Default: 0x00010100 Access: PINMUX->GPIO27							
31	30	29	28	27	26	25	24
RESERVED							PU
23	22	21	20	19	18	17	16
RESERVED							PD
15	14	13	12	11	10	9	8
RESERVED							ICHEN
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED				

Table 4-60: GPIO27 Pin Control Register (GPIO27) Description

Bits	Field Name	Type	Reset	Description
31:25	RESERVED_31_25	RO	0x0	Reserved.
24	PU	RW	0x0	Pull up enable
23:17	RESERVED_23_17	RO	0x0	Reserved.
16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x1	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:0	RESERVED_4_0	RO	0x0	Reserved.

Table 4-61: GPIO28 Pin Control Register (GPIO28) Layout

GPIO28 (GPIO28 Pin Control Register) Offset: 0x70 Default: 0x00010001 Access: PINMUX->GPIO28								
31	30	29	28	27	26	25	24	
		RESERVED				PU		
23	22	21	20	19	18	17	16	
		RESERVED				PD		
15	14	13	12	11	10	9	8	
		RESERVED				ICHEN		
7	6	5	4	3	2	1	0	
STRENGTH		DEGLITCH	RESERVED			OCHSEL		

Table 4-62: GPIO28 Pin Control Register (GPIO28) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x1	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO28 01: ANA_IN5 10: SPI1_SCLK 11: PWM SOCA output for monitor

Table 4-63: GPIO29 Pin Control Register (GPIO29) Layout

GPIO29 (GPIO29 Pin Control Register) Offset: 0x74 Default: 0x00010001 Access: PINMUX->GPIO29								
31	30	29	28	27	26	25	24	
		RESERVED				PU		
23	22	21	20	19	18	17	16	
		RESERVED				PD		
15	14	13	12	11	10	9	8	
		RESERVED				ICHEN		
7	6	5	4	3	2	1	0	
STRENGTH		DEGLITCH	RESERVED			OCHSEL		

Table 4-64: GPIO29 Pin Control Register (GPIO29) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x1	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:2	RESERVED_4_2	RO	0x0	Reserved.
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO29 01: ANA_IN6 10: SPI1_SFRM 11: PWM SOCB output for monitor

Table 4-65: GPIO30 Pin Control Register (GPIO30) Layout

GPIO30 (GPIO30 Pin Control Register) Offset: 0x78 Default: 0x00010001 Access: PINMUX->GPIO30								
31	30	29	28	27	26	25	24	
		RESERVED				PU		
23	22	21	20	19	18	17	16	
		RESERVED				PD		
15	14	13	12	11	10	9	8	
		RESERVED				ICHEN		
7	6	5	4	3	2	1	0	
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-66: GPIO30 Pin Control Register (GPIO30) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x1	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO30 01: ANA_IN7 10: SPI1_MOSI 11: PWM SOCC output for monitor

Table 4-67: GPIO31 Pin Control Register (GPIO31) Layout

GPIO31 (GPIO31 Pin Control Register) Offset: 0x7C Default: 0x00010001 Access: PINMUX->GPIO31								
31	30	29	28	27	26	25	24	
		RESERVED				PU		
23	22	21	20	19	18	17	16	
		RESERVED				PD		
15	14	13	12	11	10	9	8	
		RESERVED				ICHEN		
7	6	5	4	3	2	1	0	
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-68: GPIO31 Pin Control Register (GPIO31) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x0	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x1	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO31 01: ANA_IN8 10: SPI1_MISO 11: PWM_SYNC output for monitor

Table 4-69: GPIO32 Pin Control Register (GPIO32) Layout

GPIO32 (GPIO32 Pin Control Register) Offset: 0x80 Default: 0x07000100 Access: PINMUX->GPIO32								
31	30	29	28	27	26	25	24	
RESERVED						PU		
23	22	21	20	19	18	17	16	
RESERVED						PD		
15	14	13	12	11	10	9	8	
RESERVED						ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-70: GPIO32 Pin Control Register (GPIO32) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x7	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO32 01: UART1_TXD 10: I2C_SCL 11: COMP_MON6

Table 4-71: GPIO33 Pin Control Register (GPIO33) Layout

GPIO33 (GPIO33 Pin Control Register) Offset: 0x84 Default: 0x07000100 Access: PINMUX->GPIO33								
31	30	29	28	27	26	25	24	
RESERVED						PU		
23	22	21	20	19	18	17	16	
RESERVED						PD		
15	14	13	12	11	10	9	8	
RESERVED						ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-72: GPIO33 Pin Control Register (GPIO33) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x7	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO33 01: UART1_RXD 10: I2C_SDA 11: COMP_MON7

Table 4-73: GPIO34 Pin Control Register (GPIO34) Layout

GPIO34 (GPIO34 Pin Control Register) Offset: 0x88 Default: 0x07000100 Access: PINMUX->GPIO34								
31	30	29	28	27	26	25	24	
RESERVED						PU		
23	22	21	20	19	18	17	16	
RESERVED						PD		
15	14	13	12	11	10	9	8	
RESERVED						ICHEN		
7	6	5	4	3	2	1	0	
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL		

Table 4-74: GPIO34 Pin Control Register (GPIO34) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x7	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO34 01: CAN_TXD 10: I2C_SCL 11: Reserve

Table 4-75: GPIO35 Pin Control Register (GPIO35) Layout

GPIO35 (GPIO35 Pin Control Register) Offset: 0x8C Default: 0x07000100 Access: PINMUX->GPIO35									
31	30	29	28	27	26	25	24		
RESERVED						PU			
23	22	21	20	19	18	17	16		
RESERVED						PD			
15	14	13	12	11	10	9	8		
RESERVED						ICHEN			
7	6	5	4	3	2	1	0		
RESERVED	STRENGTH	DEGLITCH	RESERVED		LOOPBACK	OCHSEL			

Table 4-76: GPIO35 Pin Control Register (GPIO35) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26:24	PU	RW	0x7	Pull up enable
23:19	RESERVED_23_19	RO	0x0	Reserved.
18:16	PD	RW	0x0	Pull down enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	ICHEN	RW	0x1	Input channel enable
7	RESERVED_7	RO	0x0	Reserved.
6	STRENGTH	RW	0x0	Output Drive Strength (for detailed data, refer to the datasheet) 0: Weak drive strength (approximately 1/10 of full drive strength) 1: Full drive strength
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x0	Output channel select 00: GPIO35 01: CAN_RXD 10: I2C_SDA 11: Reserve

Table 4-77: GPIO36 Pin Control Register (GPIO36) Layout

GPIO36 (GPIO36 Pin Control Register) Offset: 0x90 Default: 0x09000001 Access: PINMUX->GPIO36							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-78: GPIO36 Pin Control Register (GPIO36) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x9	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO36 01: ANA_IN9_PHU 10: SPI1_SCLK 11: UART1_TXD

Table 4-79: GPIO37 Pin Control Register (GPIO37) Layout

GPIO37 (GPIO37 Pin Control Register) Offset: 0x94 Default: 0x09000001 Access: PINMUX->GPIO37							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-80: GPIO37 Pin Control Register (GPIO37) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x9	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for UART1 0: Disable 1: Enable
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO37 01: ANA_IN10_PHV 10: SPI1_SFRM 11: UART1_RXD

Table 4-81: GPIO38 Pin Control Register (GPIO38) Layout

GPIO38 (GPIO38 Pin Control Register) Offset: 0x98 Default: 0x09000001 Access: PINMUX->GPIO38							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-82: GPIO38 Pin Control Register (GPIO38) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x9	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI1 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO38 01: ANA_IN11_PHW 10: SPI1_MOSI 11: CAN_TXD

Table 4-83: GPIO39 Pin Control Register (GPIO39) Layout

GPIO39 (GPIO39 Pin Control Register) Offset: 0x9C Default: 0x09000001 Access: PINMUX->GPIO39							
31	30	29	28	27	26	25	24
RESERVED				PU			
23	22	21	20	19	18	17	16
RESERVED				PD			
15	14	13	12	11	10	9	8
RESERVED				ICHEN			
7	6	5	4	3	2	1	0
STRENGTH		DEGLITCH	RESERVED		LOOPBACK	OCHSEL	

Table 4-84: GPIO39 Pin Control Register (GPIO39) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:24	PU	RW	0x9	Pull up enable
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	PD	RW	0x0	Pull down enable
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:3	RESERVED_4_3	RO	0x0	Reserved.
2	LOOPBACK	RW	0x0	Data loopback for SPI1 and CAN 0: Disable 1: Enable
1:0	OCHSEL	RW	0x1	Output channel select 00: GPIO39 01: ANA_IN12_PHC 10: SPI1_MISO 11: CAN_RXD

Table 4-85: GPIO40 Pin Control Register (GPIO40) Layout

GPIO40 (GPIO40 Pin Control Register) Offset: 0xA0 Default: 0x00010001 Access: PINMUX->GPIO40									
31	30	29	28	27	26	25		24	
RESERVED						PU			
23	22	21	20	19	18	17		16	
RESERVED						PD			
15	14	13	12	11	10	9		8	
RESERVED								ICHEN	
7	6	5	4	3	2	1		0	
STRENGTH		DEGLITCH	RESERVED					OCHSEL	

Table 4-86: GPIO40 Pin Control Register (GPIO40) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25:24	PU	RW	0x0	Pull up enable
23:18	RESERVED_23_18	RO	0x0	Reserved.
17:16	PD	RW	0x1	Pull down enable
15:9	RESERVED_15_9	RO	0x0	Reserved.
8	ICHEN	RW	0x0	Input channel enable
7:6	STRENGTH	RW	0x0	Output driving strength 00: 5mA 01: 10mA 10: 15mA 11: 20mA
5	DEGLITCH	RW	0x0	Input deglitch 0: Disable 1: Enable
4:1	RESERVED_4_1	RO	0x0	Reserved.
0	OCHSEL	RW	0x1	Output channel select 0: GPIO40 1: ANA_IN13

Table 4-87: Comparator Output Monitor Control Register (COMPMONCTL) Layout

COMPMONCTL (Comparator Output Monitor Control Register) Offset: 0xA4 Default: 0x76543210 Access: PINMUX->COMPMONCTL							
31 RESERVED	30	29 SRC7	28	27 RESERVED	26	25 SRC6	24
23 RESERVED	22	21 SRC5	20	19 RESERVED	18	17 SRC4	16
15 RESERVED	14	13 SRC3	12	11 RESERVED	10	9 SRC2	8
7 RESERVED	6	5 SRC1	4	3 RESERVED	2	1 SRC0	0

Table 4-88: Comparator Output Monitor Control Register (COMPMONCTL) Description

Bits	Field Name	Type	Reset	Description
31	RESERVED_31	RO	0x0	Reserved.
30:28	SRC7	RW	0x7	COMP output monitor 7 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output
27	RESERVED_27	RO	0x0	Reserved.
26:24	SRC6	RW	0x6	COMP output monitor 6 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output
23	RESERVED_23	RO	0x0	Reserved.
22:20	SRC5	RW	0x5	COMP output monitor 5 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output

Bits	Field Name	Type	Reset	Description
19	RESERVED_19	RO	0x0	Reserved.
18:16	SRC4	RW	0x4	COMP output monitor 4 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output
15	RESERVED_15	RO	0x0	Reserved.
14:12	SRC3	RW	0x3	COMP output monitor 3 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output
11	RESERVED_11	RO	0x0	Reserved.
10:8	SRC2	RW	0x2	COMP output monitor 2 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output
7	RESERVED_7	RO	0x0	Reserved.
6:4	SRC1	RW	0x1	COMP output monitor 1 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output 011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPU filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output
3	RESERVED_3	RO	0x0	Reserved.
2:0	SRC0	RW	0x0	COMP output monitor 0 source select 000: COMP0L output 001: COMP0H output 010: PHCOMPU original output

Bits	Field Name	Type	Reset	Description
				011: PHCOMPV original output 100: PHCOMPW original output 101: PHCOMPV filtered output 110: PHCOMPV filtered output 111: PHCOMPW filtered output

Table 4-89: PINMUX Register Write-Allow Key Register (PINMUXREGKEY) Layout

PINMUXREGKEY (PINMUX Register Write-Allow Key Register)					Offset: 0xA8	Default: 0x1ACCE551	
Access: PINMUX->PINMUXREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 4-90: PINMUX Register Write-Allow Key Register (PINMUXREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected pinmux registers

4.4.3 GPIO register map

Table 4-91: GPIO Module Base Address

Peripheral Module	Base Address
GPIO	0x4000 1200

Table 4-92: GPIO Register Map

Register	Offset	Description	Reset Value
GPLR *	0x000	GPIO Pin Level Register n ([0:1])	0x00000000
GSLR *	0x010	GPIO Pin Output Set Register n ([0:1])	0x00000000
GCLR *	0x020	GPIO Pin Output Clear Register n ([0:1])	0x00000000
GPDR *	0x030	GPIO Pin Direction Register n ([0:1])	0x00000000
GSDR *	0x040	GPIO Pin Bitwise Set Direction Register n ([0:1])	0x00000000
GCDR *	0x050	GPIO Pin Bitwise Clear Direction Register n ([0:1])	0x00000000
GRER *	0x060	GPIO Pin Rising Edge Detect Enable Register n ([0:1])	0x00000000
GSRE *	0x070	GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1])	0x00000000
GCRE *	0x080	GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1])	0x00000000

Register	Offset	Description	Reset Value
GFER*	0x090	GPIO Pin Falling Edge Detect Enable Register n ([0:1])	0x00000000
GSFE*	0x0A0	GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1])	0x00000000
GCFE*	0x0B0	GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1])	0x00000000
GRERAWIF	0x0C0	GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1])	0x00000000
GFERAWIF	0x0D0	GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1])	0x00000000
GREIF	0x0E0	GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1])	0x00000000
GFEIF	0x0F0	GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1])	0x00000000
GREIE*	0x100	GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1])	0x00000000
GFEIE*	0x110	GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1])	0x00000000
GREIC	0x120	GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1])	0x00000000
GFEIC	0x130	GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1])	0x00000000
GREIFRC*	0x140	GPIO Pin Rising Edge Event Force Register n ([0:1])	0x00000000
GFEIFRC*	0x150	GPIO Pin Falling Edge Event Force Register n ([0:1])	0x00000000
GLDE*	0x160	GPIO Pin Level Detect Enable Register n ([0:1])	0x00000000
GSLDE*	0x170	GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1])	0x00000000
GCLDE*	0x180	GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1])	0x00000000
GLPOL*	0x190	GPIO Pin Level Detect Polarity Register n ([0:1])	0x00000000
GLRAWIF	0x1A0	GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1])	0x00000000
GLIF	0x1B0	GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1])	0x00000000
GLIE*	0x1C0	GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1])	0x00000000
GLIC	0x1D0	GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1])	0x00000000
GLIFRC*	0x1E0	GPIO Pin Level-Triggered Event Force Register n ([0:1])	0x00000000
GGRAWIF	0x1F0	GPIO Global Interrupt Raw Flag Register	0x00000000
GGIF	0x1F4	GPIO Global Interrupt Flag Register	0x00000000
GGIC	0x1F8	GPIO Global Interrupt Clear Register	0x00000000
GPIOREGKEY	0x1FC	GPIO Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the GPIOREGKEY = 0x1ACCE551.

4.4.4 GPIO registers

Table 4-93: GPIO Pin Level Register n ([0:1]) (GPLR) Layout

GPLR (GPIO Pin Level Register n ([0:1]))							
Access: GPIO->GPLR		Offset: 0x000		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-94: GPIO Pin Level Register n ([0:1]) (GPLR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	GPIO pin level 0: Write a 0 sets the pin level to low Read a 0 indicates current pin level is low 1: Write a 1 sets the pin level to high Read a 1 indicates current pin level is high

Table 4-95: GPIO Pin Output Set Register n ([0:1]) (GSLR) Layout

GSLR (GPIO Pin Output Set Register n ([0:1])) Access: GPIO->GSLR Offset: 0x010 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-96: GPIO Pin Output Set Register n ([0:1]) (GSLR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Set GPIO output to high 0: Write a 0 has no effect and always reads back 0 1: Write a 1 sets output to high. This bit is self-cleared to 0

Table 4-97: GPIO Pin Output Clear Register n ([0:1]) (GCLR) Layout

GCLR (GPIO Pin Output Clear Register n ([0:1])) Access: GPIO->GCLR Offset: 0x020 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-98: GPIO Pin Output Clear Register n ([0:1]) (GCLR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear GPIO output to low 0: Write a 0 has no effect and always reads back 0 1: Write a 1 sets output to low. This bit is self-cleared to 0

Table 4-99: GPIO Pin Direction Register n ([0:1]) (GPDR) Layout

GPDR (GPIO Pin Direction Register n ([0:1])) Access: GPIO->GPDR Offset: 0x030 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-100: GPIO Pin Direction Register n ([0:1]) (GPDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Set GPIO direction 0: Set GPIO as input 1: Set GPIO as output

Table 4-101: GPIO Pin Bitwise Set Direction Register n ([0:1]) (GSDR) Layout

GSDR (GPIO Pin Bitwise Set Direction Register n ([0:1])) Access: GPIO->GSDR Offset: 0x040 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-102: GPIO Pin Bitwise Set Direction Register n ([0:1]) (GSDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Set GPDR register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 sets GPDR bit. This bit is self-cleared to 0

Table 4-103: GPIO Pin Bitwise Clear Direction Register n ([0:1]) (GCDR) Layout

GCDR (GPIO Pin Bitwise Clear Direction Register n ([0:1])) Access: GPIO->GCDR Offset: 0x050 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-104: GPIO Pin Bitwise Clear Direction Register n ([0:1]) (GCDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear GPDR register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears GPDR bit. This bit is self-cleared to 0

Table 4-105: GPIO Pin Rising Edge Detect Enable Register n ([0:1]) (GRER) Layout

GRER (GPIO Pin Rising Edge Detect Enable Register n ([0:1])) Access: GPIO->GRER Offset: 0x060 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-106: GPIO Pin Rising Edge Detect Enable Register n ([0:1]) (GRER) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Enable GPIO rising edge detection 0: Disable 1: Enable

Table 4-107: GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1]) (GSRE) Layout

GSRE (GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1])) Access: GPIO->GSRE Offset: 0x070 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-108: GPIO Pin Bitwise Set Rising Edge Detect Enable Register n ([0:1]) (GSRE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Set GRER register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 sets GRER bit. This bit is self-cleared to 0

Table 4-109: GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1]) (GCRE) Layout

GCRE (GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1])) Access: GPIO->GCRE Offset: 0x080 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-110: GPIO Pin Bitwise Clear Rising Edge Detect Enable Register n ([0:1]) (GCRE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear GRER register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears GRER bit. This bit is self-cleared to 0

Table 4-111: GPIO Pin Falling Edge Detect Enable Register n ([0:1]) (GFER) Layout

GFER (GPIO Pin Falling Edge Detect Enable Register n ([0:1])) Access: GPIO->GFER Offset: 0x090 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-112: GPIO Pin Falling Edge Detect Enable Register n ([0:1]) (GFER) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Enable GPIO falling edge detection 0: Disable 1: Enable

Table 4-113: GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1]) (GSFE) Layout

GSFE (GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1])) Access: GPIO->GSFE Offset: 0x0A0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-114: GPIO Pin Bitwise Set Falling Edge Detect Enable Register n ([0:1]) (GSFE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Set GFER register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 sets GFER bit. This bit is self-cleared to 0

Table 4-115: GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1]) (GCFE) Layout

GCFE (GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1])) Access: GPIO->GCFE Offset: 0x0B0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-116: GPIO Pin Bitwise Clear Falling Edge Detect Enable Register n ([0:1]) (GCFE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear GFER register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clear GFER bit. This bit is self-cleared to 0

Table 4-117: GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1]) (GRERAWIF) Layout

GRERAWIF (GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1])) Access: GPIO->GRERAWIF Offset: 0x0C0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-118: GPIO Pin Rising Edge Detect Interrupt Raw Flag Register n ([0:1]) (GRERAWIF)
Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	GPIO rising edge event flag 0: No detected rising edge on the GPIO 1: Rising edge on the GPIO was detected

Table 4-119: GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1]) (GFERAWIF) Layout

GFERAWIF (GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1])) Access: GPIO->GFERAWIF Offset: 0x0D0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-120: GPIO Pin Falling Edge Detect Interrupt Raw Flag Register n ([0:1]) (GFERAWIF)
Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	GPIO falling edge event flag 0: No detected falling edge on the GPIO 1: Falling edge on the GPIO was detected

Table 4-121: GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1]) (GREIF) Layout

GREIF (GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1])) Access: GPIO->GREIF Offset: 0x0E0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-122: GPIO Pin Rising Edge Triggered Interrupt Flag Register n ([0:1]) (GREIF) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Latched GPIO rising edge triggered interrupt flag 0: The interrupt does not occur for the GPIO 1: The interrupt occurred for the GPIO

Table 4-123: GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1]) (GFEIF) Layout

GFEIF (GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1])) Access: GPIO->GFEIF Offset: 0x0F0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-124: GPIO Pin Falling Edge Triggered Interrupt Flag Register n ([0:1]) (GFEIF) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Latched GPIO falling edge triggered interrupt flag 0: The interrupt does not occur for the GPIO 1: The interrupt occurred for the GPIO

Table 4-125: GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1]) (GREIE) Layout

GREIE (GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1])) Access: GPIO->GREIE Offset: 0x100 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-126: GPIO Pin Rising Edge Triggered Interrupt Enable Register n ([0:1]) (GREIE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	GPIO rising edge triggered interrupt enable 0: Disable 1: Enable

Table 4-127: GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1]) (GFEIE) Layout

GFEIE (GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1])) Access: GPIO->GFEIE Offset: 0x110 Default: 0x00000000							
31	30	29	28	27	26	25	24
				VAL			
23	22	21	20	19	18	17	16
				VAL			
15	14	13	12	11	10	9	8
				VAL			
7	6	5	4	3	2	1	0
				VAL			

Table 4-128: GPIO Pin Falling Edge Triggered Interrupt Enable Register n ([0:1]) (GFEIE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	GPIO falling edge triggered interrupt enable 0: Disable 1: Enable

Table 4-129: GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1]) (GREIC) Layout

GREIC (GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1])) Access: GPIO->GREIC Offset: 0x120 Default: 0x00000000							
31	30	29	28	27	26	25	24
				VAL			
23	22	21	20	19	18	17	16
				VAL			
15	14	13	12	11	10	9	8
				VAL			
7	6	5	4	3	2	1	0
				VAL			

Table 4-130: GPIO Pin Rising Edge Triggered Interrupt Clear Register n ([0:1]) (GREIC) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear latched rising edge triggered event and interrupt flag 0: Write a 0 has no effect and always reads back 0. 1: Write a 1 clears the GRERAWIF bit and GREIF bit. This bit is self-cleared to 0.

Table 4-131: GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1]) (GFEIC) Layout

GFEIC (GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1]))							
Access: GPIO->GFEIC		Offset: 0x130		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-132: GPIO Pin Falling Edge Triggered Interrupt Clear Register n ([0:1]) (GFEIC) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear latched falling edge triggered event and interrupt flag 0: Write a 0 has no effect and always reads back 0. 1: Write a 1 clears the GFERAWIF bit and GFEIF bit. This bit is self-cleared to 0.

Table 4-133: GPIO Pin Rising Edge Event Force Register n ([0:1]) (GREIFRC) Layout

GREIFRC (GPIO Pin Rising Edge Event Force Register n ([0:1]))							
Access: GPIO->GREIFRC		Offset: 0x140		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-134: GPIO Pin Rising Edge Event Force Register n ([0:1]) (GREIFRC) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Force GPIO rising edge event 0: Write a 0 has no effect and always reads back 0. 1: Write a 1 forces a rising edge event on GPIO. This bit is self-cleared to 0.

Table 4-135: GPIO Pin Falling Edge Event Force Register n ([0:1]) (GFEIFRC) Layout

GFEIFRC (GPIO Pin Falling Edge Event Force Register n ([0:1]))							
Access: GPIO->GFEIFRC		Offset: 0x150		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-136: GPIO Pin Falling Edge Event Force Register n ([0:1]) (GFEIFRC) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Force GPIO falling edge event 0: Write a 0 has no effect and always reads back 0. 1: Write a 1 forces a falling edge event on GPIO. This bit is self-cleared to 0.

Table 4-137: GPIO Pin Level Detect Enable Register n ([0:1]) (GLDE) Layout

GLDE (GPIO Pin Level Detect Enable Register n ([0:1]))							
Access: GPIO->GLDE		Offset: 0x160		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-138: GPIO Pin Level Detect Enable Register n ([0:1]) (GLDE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	GPIO level detect enable 0: Disable 1: Enable

Table 4-139: GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1]) (GSLDE) Layout

GSLDE (GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1])) Access: GPIO->GSLDE Offset: 0x170 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-140: GPIO Pin Bitwise Set Level Detect Enable Register n ([0:1]) (GSLDE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Set GLDE register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 sets GLDE bit. This bit is self-cleared to 0

Table 4-141: GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1]) (GCLDE) Layout

GCLDE (GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1])) Access: GPIO->GCLDE Offset: 0x180 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-142: GPIO Pin Bitwise Clear Level Detect Enable Register n ([0:1]) (GCLDE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear GLDE register bit 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears GLDE bit. This bit is self-cleared to 0

Table 4-143: GPIO Pin Level Detect Polarity Register n ([0:1]) (GLPOL) Layout

GLPOL (GPIO Pin Level Detect Polarity Register n ([0:1])) Access: GPIO->GLPOL Offset: 0x190 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-144: GPIO Pin Level Detect Polarity Register n ([0:1]) (GLPOL) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	GPIO level detect polarity 0: Detected level is valid when low 1: Detected level is valid when high

Table 4-145: GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1]) (GLRAWIF) Layout

GLRAWIF (GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1])) Access: GPIO->GLRAWIF Offset: 0x1A0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-146: GPIO Pin Level Detect Interrupt Raw Flag Register n ([0:1]) (GLRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Latched GPIO level event flag 0: Specified level was not detected 1: Specified level was detected

Table 4-147: GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1]) (GLIF) Layout

GLIF (GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1])) Access: GPIO->GLIF Offset: 0x1B0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-148: GPIO Pin Level-Triggered Interrupt Flag Register n ([0:1]) (GLIF) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Latched GPIO level-triggered interrupt flag 0: The interrupt does not occur for the GPIO 1: The interrupt occurred for the GPIO

Table 4-149: GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1]) (GLIE) Layout

GLIE (GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1])) Access: GPIO->GLIE Offset: 0x1C0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-150: GPIO Pin Level-Triggered Interrupt Enable Register n ([0:1]) (GLIE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	GPIO level-triggered interrupt enable 0: Disable 1: Enable

Table 4-151: GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1]) (GLIC) Layout

GLIC (GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1])) Access: GPIO->GLIC Offset: 0x1D0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-152: GPIO Pin Level-Triggered Interrupt Clear Register n ([0:1]) (GLIC) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1C	0x0	Clear GPIO level-triggered event and interrupt flag 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the GLRAWIF bit and GLIF bit. This bit is self-cleared to 0.

Table 4-153: GPIO Pin Level-Triggered Event Force Register n ([0:1]) (GLIFRC) Layout

GLIFRC (GPIO Pin Level-Triggered Event Force Register n ([0:1])) Access: GPIO->GLIFRC Offset: 0x1E0 Default: 0x00000000							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 4-154: GPIO Pin Level-Triggered Event Force Register n ([0:1]) (GLIFRC) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	W1S	0x0	Force GPIO level-triggered event 0: Write a 0 has no effect and always reads back 0 1: Write a 1 forces a GPIO level-triggered event. This bit is self-cleared to 0.

Table 4-155: GPIO Global Interrupt Raw Flag Register (GGRAWIF) Layout

GGRAWIF (GPIO Global Interrupt Raw Flag Register)							
Access: GPIO->GGRAWIF		Offset: 0x1F0		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				FEDGE	REDGE	EDGE	LEVEL

Table 4-156: GPIO Global Interrupt Raw Flag Register (GGRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	FEDGE	RO	0x0	Global GPIO falling edge event flag This bit is the logic OR of GFERAWIF bits 0: No falling edge was detected on GPIO 1: Falling edge was detected on GPIO
2	REDGE	RO	0x0	Global GPIO rising edge event flag This bit is the logic OR of GRERAWIF bits 0: No rising edge was detected on GPIO 1: Rising edge was detected on GPIO
1	EDGE	RO	0x0	Global GPIO edge event flag This bit is the logic OR of GRERAWIF and GFERAWIF bits 0: No edge was detected on GPIO 1: Edge was detected on GPIO
0	LEVEL	RO	0x0	Global GPIO level event flag This bit is the logic OR of GLRAWIF bits 0: Specified level was not detected on GPIO 1: Specified level was detected on GPIO

Table 4-157: GPIO Global Interrupt Flag Register (GGIF) Layout

GGIF (GPIO Global Interrupt Flag Register)							
Access: GPIO->GGIF		Offset: 0x1F4		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						EDGE	LEVEL

Table 4-158: GPIO Global Interrupt Flag Register (GGIF) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	EDGE	RO	0x0	Global GPIO edge-triggered interrupt flag 0: Edge-triggered interrupt does not occur 1: Edge-triggered interrupt occurred and issued to the CPU. Further interrupt will not be generated until this bit is cleared.
0	LEVEL	RO	0x0	Global GPIO level-triggered interrupt flag 0: Level-triggered interrupt does not occur 1: Level-triggered interrupt occurred and issued to the CPU. Further interrupt will not be generated until this bit is cleared.

Table 4-159: GPIO Global Interrupt Clear Register (GGIC) Layout

GGIC (GPIO Global Interrupt Clear Register)							
Access: GPIO->GGIC		Offset: 0x1F8		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						EDGE	LEVEL

Table 4-160: GPIO Global Interrupt Clear Register (GGIC) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	EDGE	W1C	0x0	Global GPIO edge-triggered interrupt clear 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the flag. This bit is self-cleared.
0	LEVEL	W1C	0x0	Global GPIO level-triggered interrupt clear 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the flag. This bit is self-cleared.

Table 4-161: GPIO Register Write-Allow Key Register (GPIOREGKEY) Layout

GPIOREGKEY (GPIO Register Write-Allow Key Register)							
Access: GPIO->GPIOREGKEY				Offset: 0x1FC		Default: 0x1ACCE551	
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 4-162: GPIO Register Write-Allow Key Register (GPIOREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected GPIO registers

5 Interrupts and events

5.1 Nested vectored interrupt controller (NVIC)

The Nested Vectored Interrupt Controller (NVIC) is an integral part of ARM Cortex-M4 processor unit. Due to a tight coupling with the CPU, it provides the lowest interrupt latency and efficient processing of interrupts arrived later.

5.1.1 Features

The NVIC supports the following features:

- Up to 54 vectored interrupts (not including the 16 Cortex-M4 interrupts)
- 16 programmable interrupt priority levels for each interrupt (4 bits of interrupt priority are used). A higher level corresponds to a lower priority, so level 0 is the highest interrupt priority
- Controls system exceptions and peripheral interrupts
- Support for interrupt tail-chaining and late-arrival
- Support for interrupt lazy-stacking
- Support for NMI

5.1.2 Interrupt and exception vectors

When an exception event takes place and is being handled by the Cortex-M4, the processor needs to locate the starting address of the exception handler. Therefore, a vector table mechanism is used. The vector table is an array of word data inside the system memory, each representing the starting address of one exception. The vector table location is controlled by a programmable register in the NVIC called the VTOR (address 0xE000ED08). After reset, the VTOR is reset to 0; therefore, the vector table is located at address 0x0 after reset. The vector address is arranged according to 4 x the exception number. [Table 5-1](#) is the vector table for SPC1169.

Table 5-1: Vector table for SPC1169

Vector#	IRQ#	Acronym	Priority	Description	Address Offset
-	-	MSP	-	Initial main stack pointer	0x0000_0000
1	-	Reset	-3, the highest	Reset vector	0x0000_0004
2	-14	NMI	-2	Non mask-able interrupt. The watchdog timer 0 is linked to the NMI vector.	0x0000_0008
3	-13	Hard Fault	-1	Hard fault	0x0000_000C
4	-12	MemManage	Configurable	Memory management	0x0000_00010
5	-11	Bus Fault	Configurable	Bus fault	0x0000_0014
6	-10	Usage Fault	Configurable	Usage fault	0x0000_0018
7-10	-	-	-	Reserved	0x0000_001C – 0x0000_002B

Vector#	IRQ#	Acronym	Priority	Description	Address Offset
11	-5	SVCall	Configurable	Supervisor call	0x0000_002C
12	-4	Debug Monitor	Configurable	Debug monitor	0x0000_0030
13	-	-	-	Reserved	0x0000_0034
14	-2	PendSV	Configurable	Pendable service call	0x0000_0038
15	-1	SysTick	Configurable	SysTick timer interrupt	0x0000_003C
16	0	WDT1	Configurable	Watchdog timer 1 interrupt	0x0000_0040
17	1	MEM	Configurable	Memory ECC single-bit error interrupt	0x0000_0044
18	2	-	-	Reserved	0x0000_0048
19	3	-	-	Reserved	0x0000_004C
20	4	GPIO_LEVEL	Configurable	GPIO level-triggered interrupt	0x0000_0050
21	5	GPIO_EDGE	Configurable	GPIO edge-triggered interrupt	0x0000_0054
22	6	UART0	Configurable	UART0 interrupt	0x0000_0058
23	7	SPI0	Configurable	SPI0 interrupt	0x0000_005C
24	8	UART1	Configurable	UART1 interrupt	0x0000_0060
25	9	SPI1	Configurable	SPI1 interrupt	0x0000_0064
26	10	CANTRG0	Configurable	CAN Transfer Group 0 interrupt	0x0000_0068
27	11	CANTRG1	Configurable	CAN Transfer Group 1 interrupt	0x0000_006C
28	12	CANTRG2	Configurable	CAN Transfer Group 2 interrupt	0x0000_0070
29	13	CANTRG3	Configurable	CAN Transfer Group 3 interrupt	0x0000_0074
30	14	CANTRG4	Configurable	CAN Transfer Group 4 interrupt	0x0000_0078
31	15	CANTRG5	Configurable	CAN Transfer Group 5 interrupt	0x0000_007C
32	16	CANTRG6	Configurable	CAN Transfer Group 6 interrupt	0x0000_0080
33	17	CANTRG7	Configurable	CAN Transfer Group 7 interrupt	0x0000_0084
34	18	CANMISC	Configurable	CAN Miscellaneous interrupt	0x0000_0088
35	19	I2C	Configurable	I2C interrupt	0x0000_008C
36	20	ADCCH0	Configurable	ADC channel 0 EOC interrupt	0x0000_0090
37	21	ADCCH1	Configurable	ADC channel 1 EOC interrupt	0x0000_0094
38	22	ADCCH2	Configurable	ADC channel 2 EOC interrupt	0x0000_0098
39	23	ADCCH3	Configurable	ADC channel 3 EOC interrupt	0x0000_009C
40	24	ADCCH4	Configurable	ADC channel 4 EOC interrupt	0x0000_00A0
41	25	ADCCH5	Configurable	ADC channel 5 EOC interrupt	0x0000_00A4
42	26	ADCCH6	Configurable	ADC channel 6 EOC interrupt	0x0000_00A8
43	27	ADCCH7	Configurable	ADC channel 7 EOC interrupt	0x0000_00AC
44	28	ADCCH8	Configurable	ADC channel 8 EOC interrupt	0x0000_00B0
45	29	ADCCH9	Configurable	ADC channel 9 EOC interrupt	0x0000_00B4
46	30	ADCCH10	Configurable	ADC channel 10 EOC interrupt	0x0000_00B8
47	31	ADCCH11	Configurable	ADC channel 11 EOC interrupt	0x0000_00BC
48	32	ADCCH12	Configurable	ADC channel 12 EOC interrupt	0x0000_00C0
49	33	ADCCH13	Configurable	ADC channel 13 EOC interrupt	0x0000_00C4
50	34	ADCCH14	Configurable	ADC channel 14 EOC interrupt	0x0000_00C8

Vector#	IRQ#	Acronym	Priority	Description	Address Offset
51	35	ADCCH15	Configurable	ADC channel 15 EOC interrupt	0x0000_00CC
52	36	ADCPPU0	Configurable	ADCPPU0 interrupt	0x0000_00D0
53	37	ADCPPU1	Configurable	ADCPPU1 interrupt	0x0000_00D4
54	38	ADCPPU2	Configurable	ADCPPU2 interrupt	0x0000_00D8
55	39	PHCOMP	Configurable	Phase comparator interrupt	0x0000_00DC
56	40	PWM0	Configurable	PWM0 interrupt	0x0000_00E0
57	41	PWM1	Configurable	PWM1 interrupt	0x0000_00E4
58	42	PWM2	Configurable	PWM2 interrupt	0x0000_00E8
59	43	PWM3	Configurable	PWM3 interrupt	0x0000_00EC
60	44	PWM0TZ	Configurable	PWM0 trip-zone interrupt	0x0000_00F0
61	45	PWM1TZ	Configurable	PWM1 trip-zone interrupt	0x0000_00F4
62	46	PWM2TZ	Configurable	PWM2 trip-zone interrupt	0x0000_00F8
63	47	PWM3TZ	Configurable	PWM3 trip-zone interrupt	0x0000_00FC
64	48	ECAP	Configurable	ECAP interrupt	0x0000_0100
65	49	DMAC	Configurable	DMAC interrupt	0x0000_0104
66	50	TIMER0	Configurable	General purpose timer 0 interrupt	0x0000_0108
67	51	TIMER1	Configurable	General purpose timer 1 interrupt	0x0000_010C
68	52	TIMER2	Configurable	General purpose timer 2 interrupt	0x0000_0110
69	53	CRC	Configurable	CRC controller interrupt	0x0000_0114

5.1.3 Non-maskable interrupt

Following events can be enabled via SYSTEM->NMIE register as the source to trigger non-maskable interrupt

- BOD events
 - VCAP12 over-voltage
 - VCAP12 under-voltage
 - DVDD33 over-voltage
 - DVDD33 under-voltage
- Clock events
 - PLL unlock
 - Clock detector reported error
- Memory ECC multi-bit error
- Watchdog timer 0 timeout
- GPIO pin triggered event
- 13-bit monitoring ADC detected error
- ECC multi-bit error for AHB address and data
- JTAG/SWD debug mode entry
- Production test mode entry

5.1.4 NVIC interrupt configuration

NVIC interrupts can be enabled by writing to the SETENA register or disabled by writing to the CLRENA register. In this way, enabling or disabling an interrupt will not affect other interrupt enable status. The SETENA/CLRENA registers are 32 bits wide; each bit represents one interrupt input.

The NVIC interrupt-pending status can be accessed through the Interrupt Set Pending (SETPEND) and Interrupt Clear Pending (CLRPEND) registers. You can cancel a current pended exception through the CLRPEND register, or generate software interrupts through the SETPEND register.

Each external interrupt has an associated priority-level register, which has a maximum width of 8 bits and a minimum width of 3 bits. Each register can be further divided into preempt priority level and sub-priority level based on priority group settings. The SPC1169 implements 4 bits of priority level, so the chip supports 16 levels of programmable priority.

Please see ARM released documents for more information of NVIC configurations.

5.2 SysTick Timer

The Cortex-M4 processor includes an integrated system timer, called SysTick. The SYSTICK timer is a 24-bit down counter. Once it reaches zero, the counter loads the reload value from the RELOAD register. It does not stop until the enable bit in the SYSTICK Control and Status register is cleared.

When the SYSTICK timer changes from 1 to 0, it will set the COUNTFLAG bit in the SYSTICK Control and Status register. The COUNTFLAG can be cleared by one of the following:

- Read of the SYSTICK Control and Status register by the processor
- Clear of the SYSTICK counter value by writing any value to the SYSTICK Current Value register

The SYSTICK counter can be used to generate SYSTICK exceptions at regular intervals. This is often necessary for the OS, for task and resources management. To enable SYSTICK exception generation, the TICKINT bit should be set.

Please see ARM released documents for more information of SysTick Timer configurations.

6 General-purpose timers

6.1 Timer overview

The SPC1169 includes three 32-bit General Purpose Timers (GPT) with following features:

- 32-bit down-counter with dedicated clock
- Generate interrupt upon counter reaching zero
- Generate ADCSOC event upon counter reaching zero
- Generate PWMSYNCl event upon counter reaching zero
- Support 4 modes
 - General timer: Periodic down-counter
 - Gated timer: Down-counter enabled by external input level
 - Event counter: Down-counter upon external input edge
 - Event capture: Down-counter with timestamp latched upon external input edge

6.2 Functional description

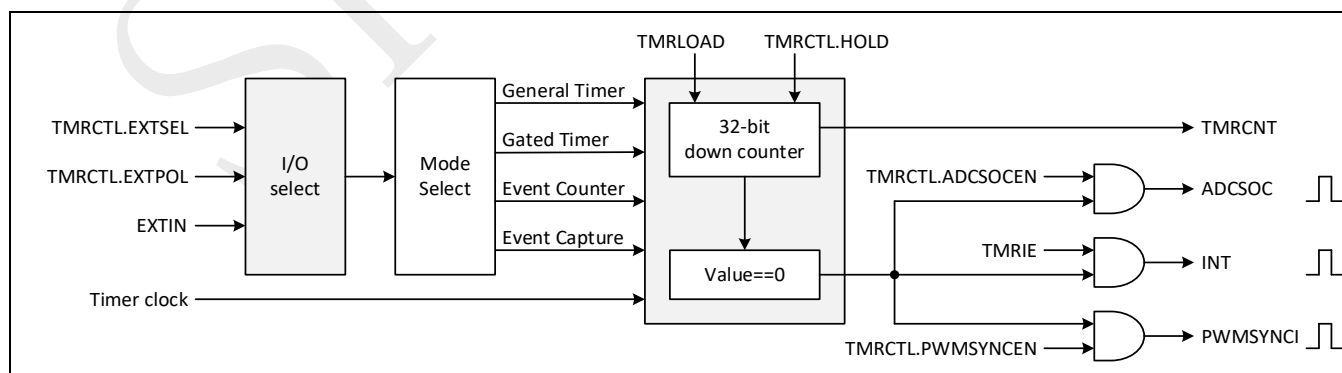
As shown in [Figure 6-1](#), each GPT is based on a 32-bit down-counter. When the timer is enabled, the counter loads the value from TMRLOAD register and starts to count down until it reaches 0. Then the counter restarts from the load value and decrease agains. Upon the counter hitting 0:

- An interrupt will be generated if it is enabled. The interrupt flag held until it is manually cleared
- An ADCSOC event will be generated if TMRCTL.ADCSOCEN is enabled
- A PWMSYNCl event will be generated if TMRCTL.PWMSYNCEN is enabled

The I/O used for external input and the valid level or edge polarity can be selected by TMRCTL.EXTSEL and TMRCTL.EXTPOL. Please refer to [Table 6-7](#) and [Table 6-8](#) for the details.

All GPTs are disabled by default after reset.

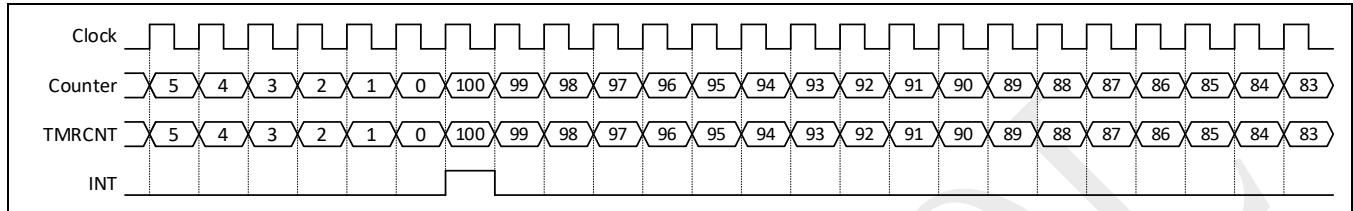
Figure 6-1: General-purpose timer topology



6.2.1 General timer mode

As shown in Figure 6-2, the 32-bit down counter simply decreases by 1 from the load value upon each clock edge of the module in this mode. It wraps back to the load value after it reaches 0. The TMRCNT register readback is always the realtime count value.

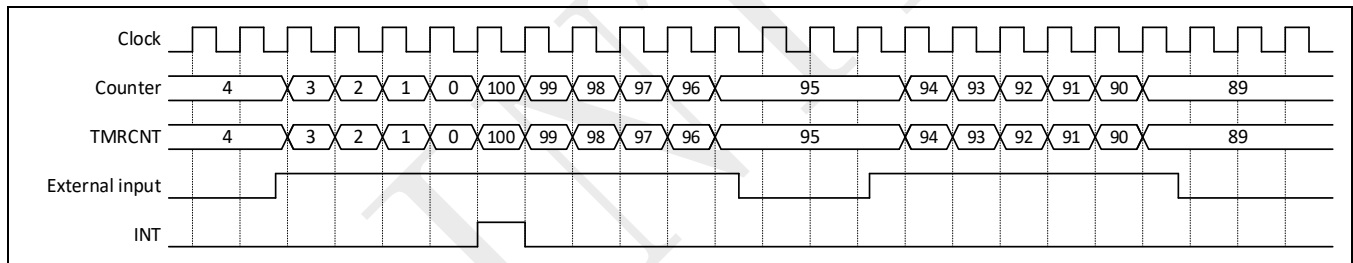
Figure 6-2: Timer waveform in general timer mode (TMRLOAD=100)



6.2.2 Gated timer mode

As shown in Figure 6-3, this mode is similar to the general timer mode, except that the counter decreases only when the external input level is valid. The I/O used for the external input and the valid level are programmable. The TMRCNT register readback is always the realtime count value.

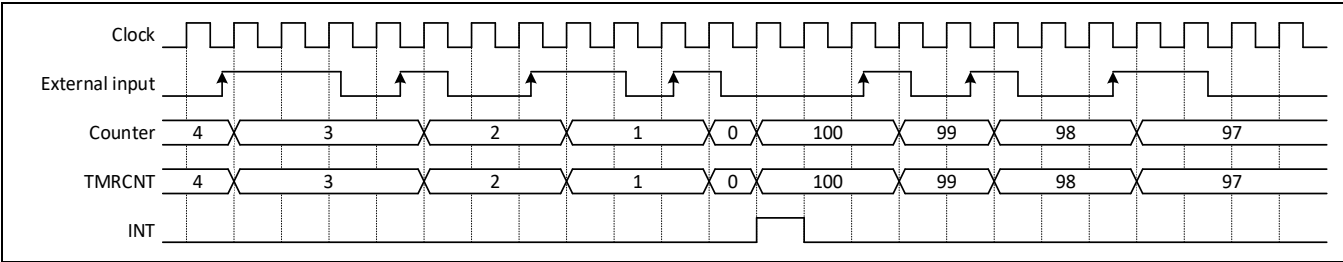
Figure 6-3: Timer waveform in gated timer mode (TMRLOAD=100)



6.2.3 Event counter mode

As shown in Figure 6-4, counter decreases only when the valid edge is detected at external input in this mode. When it reaches zero, the counter is immediately reloaded so as to be ready to count for the next round. Therefore, the interrupt is generated each time the accumulated number of valid edges at the I/O pin is TMRLOAD. This is useful if the user wants to detect a predefined number of edges at the I/O pin and be notified by the interrupt. The I/O used for the external input and the valid edge are programmable. The TMRCNT register readback is always the realtime count value. For correct detection, the duration for the external input to be high or low should be above the timer module clock period.

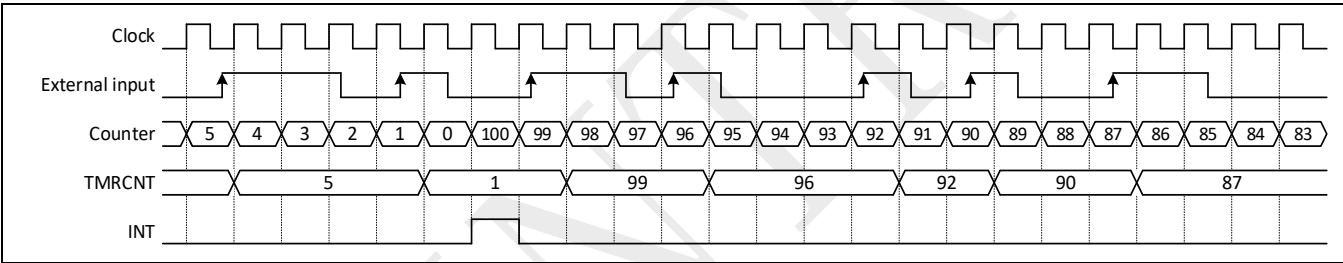
Figure 6-4: Timer waveform in event counter mode (TMRLOAD=100)



6.2.4 Event capture mode

As shown in [Figure 6-5](#), the counter behaves exactly the same as in general timer mode. However, the TMRCNT register readback is not the realtime count value, but the latched count value when the valid edge at the external input is detected. The I/O used for the external input and the valid edge are programmable. For correct detection, the duration for the external input to be high or low should be above the timer module clock period.

Figure 6-5: Timer waveform in event capture mode (TMRLOAD=100)



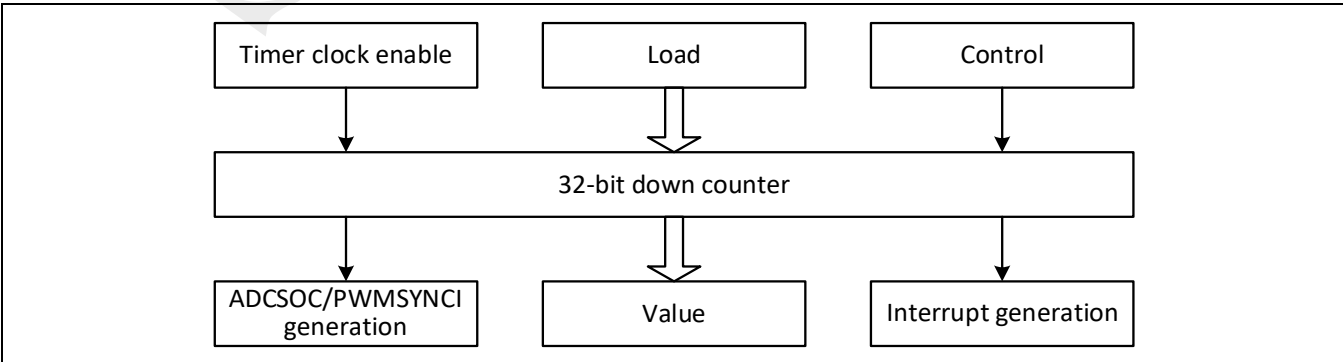
6.3 Initialization sequence

The sequence below should be followed to start the timer.

- Disable the timer and clear the timer interrupt
- Configure the timer load value register (TMRLOAD)
- Configure the timer control register (TMRCTL)
- Enable the timer to start the timer counter

A block diagram of the timer run mode is shown in [Figure 6-6](#).

Figure 6-6: GPT run flow diagram



6.4 Registers

6.4.1 TIMER register map

Table 6-1: Timer Module Base Address

Peripheral Module	Base Address
TIMER0	0x4000 1800
TIMER1	0x4000 1900
TIMER2	0x4000 1A00

Table 6-2: TIMER Register Map

Register	Offset	Description	Reset Value
<u>TMRLOAD</u> *	0x00	Timer Load Register	0x00000000
<u>TMRCNT</u> *	0x04	Timer Counter Value Register	0x00000000
<u>TMRCTL</u> *	0x08	Timer Control Register	0x00000000
<u>TMRRAWIF</u>	0x0C	Timer Interrupt Raw Flag Register	0x00000000
<u>TMRIF</u>	0x10	Timer Interrupt Flag Register	0x00000000
<u>TMRIC</u> *	0x14	Timer Interrupt Clear Register	0x00000000
<u>TMRIE</u> *	0x18	Timer Interrupt Enable Register	0x00000000
<u>TMRREGKEY</u>	0x1C	Timer Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the TMRREGKEY = 0x1ACCE551.

6.4.2 TIMER registers

Table 6-3: Timer Load Register (TMRLOAD) Layout

TMRLOAD (Timer Load Register)							
Access: TIMERx->TMRLOAD		Offset: 0x00		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 6-4: Timer Load Register (TMRLOAD) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Load value from which the counter is to decrease. A write to this register also sets the current counter value.

Table 6-5: Timer Counter Value Register (TMRCNT) Layout

TMRCNT (Timer Counter Value Register)							
Access: TIMERx->TMRCNT		Offset: 0x04		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 6-6: Timer Counter Value Register (TMRCNT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	A write to this register sets the current count value. For event capture mode, the readback is the latched count value when the valid edge at the external input is detected. For other modes, the readback is current count value.

Table 6-7: Timer Control Register (TMRCTL) Layout

TMRCTL (Timer Control Register)							
Access: TIMERx->TMRCTL		Offset: 0x08		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		EXTSEL					
7	6	5	4	3	2	1	0
EXTSEL	EXTPOL	PWMSYNCEN	ADCSOCEN	HOLD	MODE		EN

Table 6-8: Timer Control Register (TMRCTL) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13:7	EXTSEL	RW	0x0	External input source select (GPIO number)
6	EXTPOL	RW	0x0	External input polarity 0: Active low Enable counter when input is low for gated timer mode Trigger event upon falling edge of input for event counter and event capture modes 1: Active high Enable counter when input is high for gated timer mode Trigger event upon rising edge of input for event counter and event capture modes
5	PWMSYNCEN	RW	0x0	PWMSYNCl generation enable 0: Do not generate PWMSYNCl 1: Generate PWMSYNCl whenever TMRCNT counts down to 0
4	ADCSOCEN	RW	0x0	ADCSOC generation enable 0: Do not generate ADCSOC 1: Generate ADCSOC whenever TMRCNT counts down to 0
3	HOLD	RW	0x0	Hold counter value upon falling edge of TMRCTL.EN 0: Reset counter to 0 upon falling edge of TMRCTL.EN 1: Hold counter value upon falling edge of TMRCTL.EN

Bits	Field Name	Type	Reset	Description
2:1	MODE	RW	0x0	Operation mode 00: General timer (down count) 01: Down counter timer enabled by external input 10: Down counter for external input event 11: Capture counter value upon external input event
0	EN	RW	0x0	Timer enable 0: Disable 1: Enable

Table 6-9: Timer Interrupt Raw Flag Register (TMRRAWIF) Layout

TMRRAWIF (Timer Interrupt Raw Flag Register)							
Access: TIMERx->TMRRAWIF Offset: 0x0C Default: 0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 6-10: Timer Interrupt Raw Flag Register (TMRRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	RO	0x0	Latched timer event flag. This bit is set whenever TMRCNT counts down to 0 and cleared only by writing a 1 to TMRIC 0: Event does not occur 1: Event occurred

Table 6-11: Timer Interrupt Flag Register (TMRIF) Layout

TMRIF (Timer Interrupt Flag Register)							
Access: TIMERx->TMRIF Offset: 0x10 Default: 0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 6-12: Timer Interrupt Flag Register (TMRIF) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	RO	0x0	Latched timer interrupt flag 0: Interrupt does not occur 1: Interrupt occurred and issued to the CPU. Further interrupt will not be generated until this bit is cleared.

Table 6-13: Timer Interrupt Clear Register (TMRIC) Layout

TMRIC (Timer Interrupt Clear Register)							
Access: TIMERx->TMRIC Offset: 0x14 Default: 0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 6-14: Timer Interrupt Clear Register (TMRIC) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	W1C	0x0	Timer interrupt clear 0: Write a 0 has no effect and always reads back 0. 1: Writing a 1 clears the TMRRAWIF and TMRIF flag This bit is self-cleared to 0.

Table 6-15: Timer Interrupt Enable Register (TMRIE) Layout

TMRIE (Timer Interrupt Enable Register)							
Access: TIMERx->TMRIE		Offset: 0x18		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 6-16: Timer Interrupt Enable Register (TMRIE) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	RW	0x0	Timer interrupt enable. This bit does not affect TMRRWIF and PWMSYNCI/ADCSOC generation 0: Disable 1: Enable

Table 6-17: Timer Register Write-Allow Key Register (TMRREGKEY) Layout

TMRREGKEY (Timer Register Write-Allow Key Register)							
Access: TIMERx->TMRREGKEY		Offset: 0x1C		Default: 0x1ACCE551			
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 6-18: Timer Register Write-Allow Key Register (TMRREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected timer registers

7 Watchdog timers

7.1 Watchdog overview

The watchdog timer (WDT) regains control in case of system failure (due to a software error) to increase application reliability. The WDT can generate a reset or an interrupt when the counter reaches a given timeout value. The SPC1169 includes two 32-bit WDTs.

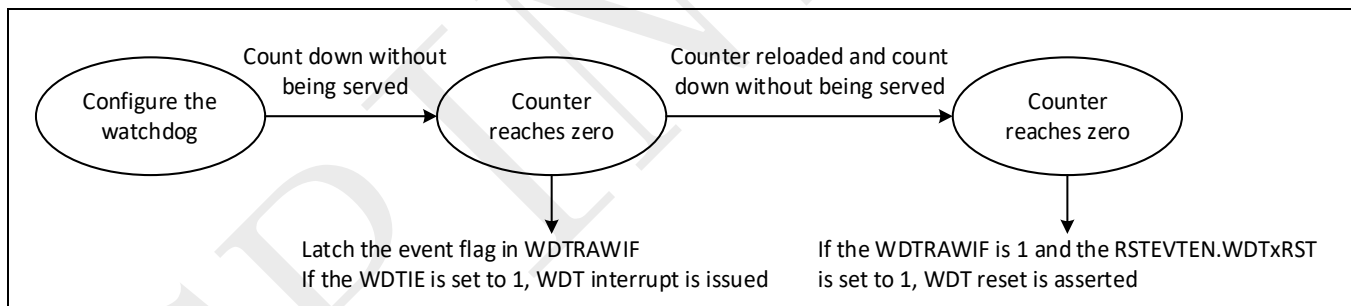
Each WDT has following features:

- AHB register interface
- Dedicated WDT clock controlled by the corresponding WDTxCLKCTL register.
- 32-bit down-counter with programmable serve window
- Configurable reset or interrupt generation with the given timeout value

7.2 Functional description

As shown in [Figure 7-1](#), when it is enabled after WDTCTL.EN is set, the watchdog counter decreases from the preset timeout value defined by WDTLOAD register. Each time when the counter reaches zero, it reloads from WDTLOAD and counts down again. Meanwhile, the timeout event would be latched to WDTRAWIF register. If it is enabled by setting WDTIE register as 1, the interrupt would be generated and the flag is latched to WDTIF register. If the latched event flag in WDTRAWIF is not cleared when timeout occurs for the second time and the reset is enabled, a system reset request would be generated.

Figure 7-1: Watchdog operation flow



Upon the write to WDTLOAD register, the counter restarts from the new value immediately. Writing 1 to WDTIC register clears the latched time event flag in WDTRAWIF and the interrupt flag in WDTIF.

The watchdog would be fed or served (i.e. counter reset to WDTLOAD) by writing 1 to WDTIC when the counter value is within the window within [WDTTHL, WDTTHH].

The WDT can be closed by configuring WDTCTL register when CPU is in HALTED mode or LOCKUP mode.

As a safety feature to prevent rogue software from disabling the watchdog functionality, all the critical control registers of the watchdog are write protected. Before configuring the watchdog, user has to write 0x1ACCE551 to WDTREGKEY register so as to enable write access to other registers. After the configuration, user should resume the WDTREGKEY to any other value so as enable the protection again.

Note: The WDT0 timeout event is used for non-maskable interrupt.

7.3 Initialization sequence

Sequence below should be followed to start the watchdog timer:

- Enable the watchdog configuration by writing WDTREGKEY register as 0x1ACCE551
- Configure the timeout threshold via WDTLOAD register
- Configure the serve window via WDTTHL and WDTTHH registers
- Configure the reset request via RSTEV TEN.WDTORST/WDT1RST bit
- Configure the interrupt via WDTIE
- Enable the watchdog counter via writing 1 to WDTCTL.EN
- Disable and protect the watchdog configuration by writing WDTREGKEY register as 0x0 (any value other than 0x1ACCE551)

7.4 Registers

7.4.1 WDT register map

Table 7-1: WDT Module Base Address

Peripheral Module	Base Address
WDT0	0x4000 1000
WDT1	0x4000 1100

Table 7-2: WDT Register Map

Register	Offset	Description	Reset Value
<u>WDTLOAD*</u>	0x00	Watchdog Timer Load Register	0x03D09000
<u>WDTCNT</u>	0x04	Watchdog Timer Current Value Register	0x03D09000
<u>WDTCTL*</u>	0x08	Watchdog Timer Control Register	0x00000007
<u>WDTRAWIF</u>	0x0C	Watchdog Timer Interrupt Raw Flag Register	0x00000000
<u>WDTIF</u>	0x10	Watchdog Timer Interrupt Flag Register	0x00000000
<u>WDTIC</u>	0x14	Watchdog Timer Interrupt Clear Register	0x00000000
<u>WDTIE*</u>	0x18	Watchdog Timer Interrupt Enable Register	0x00000001
<u>WDTTHL*</u>	0x1C	Watchdog Windowed-Feed Low Threshold Register	0x00000000
<u>WDTTHH*</u>	0x20	Watchdog Windowed-Feed High Threshold Register	0xFFFFFFFF
<u>WDTREGKEY</u>	0x24	Watchdog Register Write-Allow Key Register	0x00000000

Note: Registers marked with * are write-allowed only when the WDTREGKEY=0x1ACCE551.

7.4.2 WDT registers

Table 7-3: Watchdog Timer Load Register (WDTLOAD) Layout

WDTLOAD (Watchdog Timer Load Register)							
Access: WDTx->WDTLOAD				Offset: 0x00		Default: 0x03D09000	
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 7-4: Watchdog Timer Load Register (WDTLOAD) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x3D09000	Load value (Minimum valid value is 1) from which the counter is to decrease. When this register is written to, the count is immediately restarted from the new value.

Table 7-5: Watchdog Timer Current Value Register (WDTCNT) Layout

WDTCNT (Watchdog Timer Current Value Register)							
Access: WDTx->WDTCNT				Offset: 0x04		Default: 0x03D09000	
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 7-6: Watchdog Timer Current Value Register (WDTCNT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x3D09000	Current value of the decrementing counter.

Table 7-7: Watchdog Timer Control Register (WDTCTL) Layout

WDTCTL (Watchdog Timer Control Register)							
Access: WDTx->WDTCTL				Offset: 0x08		Default: 0x00000007	
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				LOCKUPRUN		HALTEDRUN	EN

Table 7-8: Watchdog Timer Control Register (WDTCTL) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	LOCKUPRUN	RW	0x1	Allow watchdog to run during CPU lockup mode 0: Disable the counter and reset it to WDTLOAD when CPU is in lockup mode 1: Enable the counter when CPU is in lockup mode
1	HALTEDRUN	RW	0x1	Allow watchdog to run during CPU halted mode 0: Disable the counter and reset it to WDTLOAD when CPU is in halted mode 1: Enable the counter when CPU is in halted mode
0	EN	RW	0x1	Enable the watchdog counter 0: Disable the counter and reset it to WDTLOAD 1: Enable the counter

Table 7-9: Watchdog Timer Interrupt Raw Flag Register (WDTRAWIF) Layout

WDTRAWIF (Watchdog Timer Interrupt Raw Flag Register)							
Access: WDTx->WDTRAWIF				Offset: 0x0C		Default: 0x00000000	
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 7-10: Watchdog Timer Interrupt Raw Flag Register (WDTRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	RO	0x0	Timeout (counter reaches zero) event flag 0: Timeout event does not occur 1: Timeout event occurred

Table 7-11: Watchdog Timer Interrupt Flag Register (WDTIF) Layout

WDTIF (Watchdog Timer Interrupt Flag Register)							
Access: WDTx->WDTIF		Offset: 0x10		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 7-12: Watchdog Timer Interrupt Flag Register (WDTIF) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	RO	0x0	Interrupt flag from the counter 0: Interrupt does not occur 1: Interrupt occurred and issued to the CPU.

Table 7-13: Watchdog Timer Interrupt Clear Register (WDTIC) Layout

WDTIC (Watchdog Timer Interrupt Clear Register)							
Access: WDTx->WDTIC		Offset: 0x14		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 7-14: Watchdog Timer Interrupt Clear Register (WDTIC) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	W1C	0x0	Interrupt clear. The WDTCNT will be reset to WDTLOAD if it is within [WDTTHL, WDTTHH] when this bit is written a 1. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears both the WDTRAWIF and WDTIF. This bit is self-cleared to 0.

Table 7-15: Watchdog Timer Interrupt Enable Register (WDTIE) Layout

WDTIE (Watchdog Timer Interrupt Enable Register)							
Access: WDTx->WDTIE		Offset: 0x18		Default: 0x00000001			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							INT

Table 7-16: Watchdog Timer Interrupt Enable Register (WDTIE) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	INT	RW	0x1	Interrupt enable 0: Disable 1: Enable

Table 7-17: Watchdog Windowed-Feed Low Threshold Register (WDTTHL) Layout

WDTTHL (Watchdog Windowed-Feed Low Threshold Register)							
Access: WDTx->WDTTHL		Offset: 0x1C		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 7-18: Watchdog Windowed-Feed Low Threshold Register (WDTTHL) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Watchdog can be fed via writing a 1 to WDTIC only when WDCNT is within [WDTTHL, WDTTHH]

Table 7-19: Watchdog Windowed-Feed High Threshold Register (WDTTHH) Layout

WDTTHH (Watchdog Windowed-Feed High Threshold Register)							
Access: WDTx->WDTTHH			Offset: 0x20		Default: 0xFFFFFFFF		
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 7-20: Watchdog Windowed-Feed High Threshold Register (WDTTHH) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0xFFFFFFFF	Watchdog can be fed via writing a 1 to WDTIC only when WDCNT is within [WDTTHL, WDTTHH]

Table 7-21: Watchdog Register Write-Allow Key Register (WDTREGKEY) Layout

WDTREGKEY (Watchdog Register Write-Allow Key Register)							
Access: WDTx->WDTREGKEY			Offset: 0x24		Default: 0x00000000		
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 7-22: Watchdog Register Write-Allow Key Register (WDTREGKEY) Description

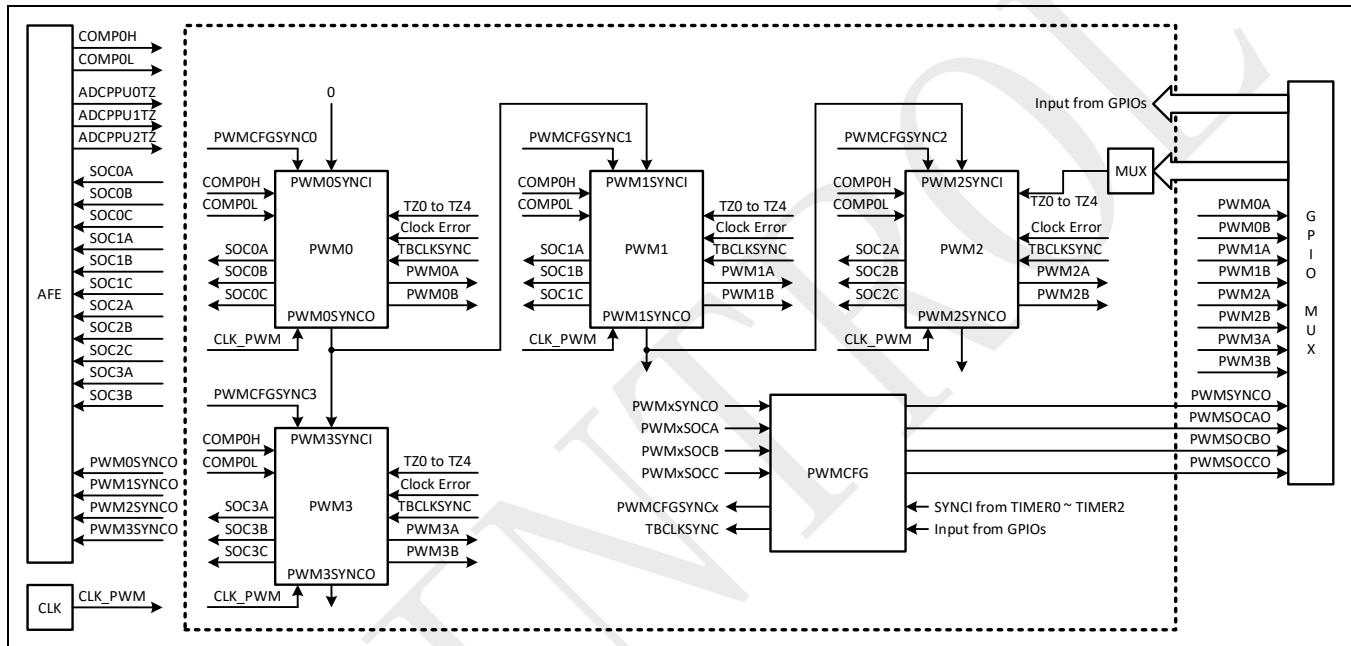
Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x0	Write 0x1ACCE551 to unlock protected watchdog registers

8 PWM

8.1 PWM overview

The PWM plays a key role in power electronic systems such as digital motor control, switch mode power supply control, and uninterruptible power supplies (UPS). The SPC1169 provides 4 PWM modules while each PWM has two outputs: PWMxA and PWMxB. They are chained together via a clock synchronization scheme as shown in [Figure 8-1](#).

Figure 8-1: PWM overview



Each PWM module supports the following features:

- 16-bit time-based counter with period and frequency control.
- Software override control of PWM signals.
- Independent and programmable phase-control support.
- Synchronized phase relationship on a cycle-by-cycle basis.
- Dead-band generation with independent rising and falling edge delay control.
- Programmable trip zone allocation of both cycle-by-cycle trip and one-shot trip on fault conditions.
- Either high, low, or high-impedance state logic levels can be forced at the PWM outputs on a trip condition.
- Comparator module outputs and trip zone inputs can generate events, filtered events, or trip conditions.
- All events can trigger both CPU interrupt and ADC start of conversion (SOC).

Figure 8-2: PWM architecture

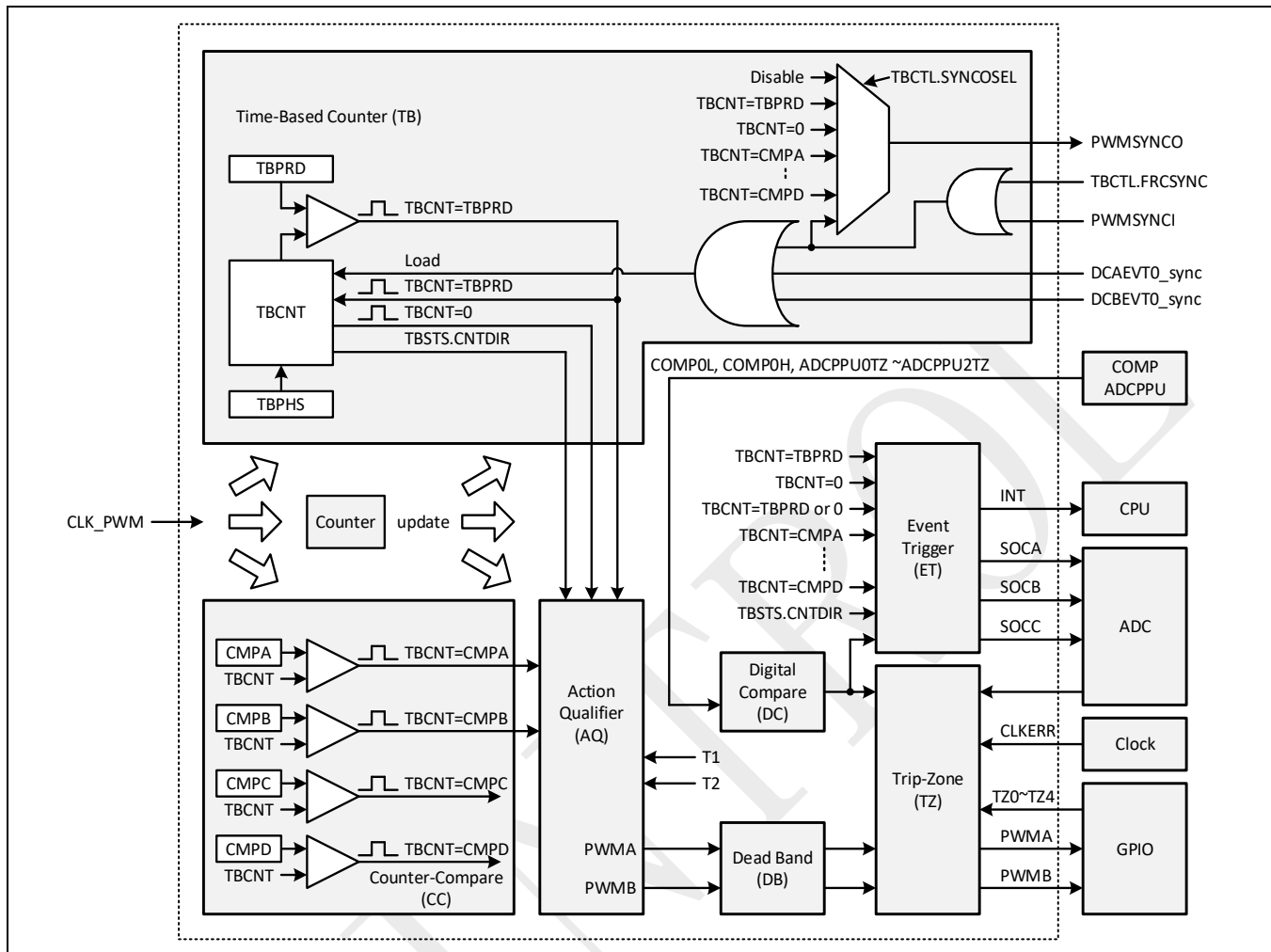


Figure 8-2 shows more internal details of a single PWM module. The key signals of a PWM module includes:

- Synchronization IO (PWMSYNCI and PWMSYNCO)

It is used to synchronize the phase relationship of multiple PWMs. In SPC1169, the PWMSYNCI can be from the GPIO, the timer or the PWMSYNCO of the preceding PWM.

Note: It takes one PWM clock delay from the synchronization input to the actual synchronization action, results in one time-base cycle delay if both TB DIVBIN and TB DIVLIN bits of TBCTL register are 0. Please take this into account when setting the TBPHS register. The register FRCSYNC in PWMCFG module provides a more flexible and convenient synchronization among the PWMs.

- Trip-zone signals

Each PWM has 5 trip-zone signals (TZ0~TZ4) configured from the GPIOs to alert the external fault conditions. Clock error event from the CLKDET module or the PLL and the halted/lockup event from the CPU can also be configured as trip-zone events for each PWM.

Note: The signal polarity from the GPIO to the trip-zone logic is programmable. By default, the trip-zone event is triggered when the GPIO is high.

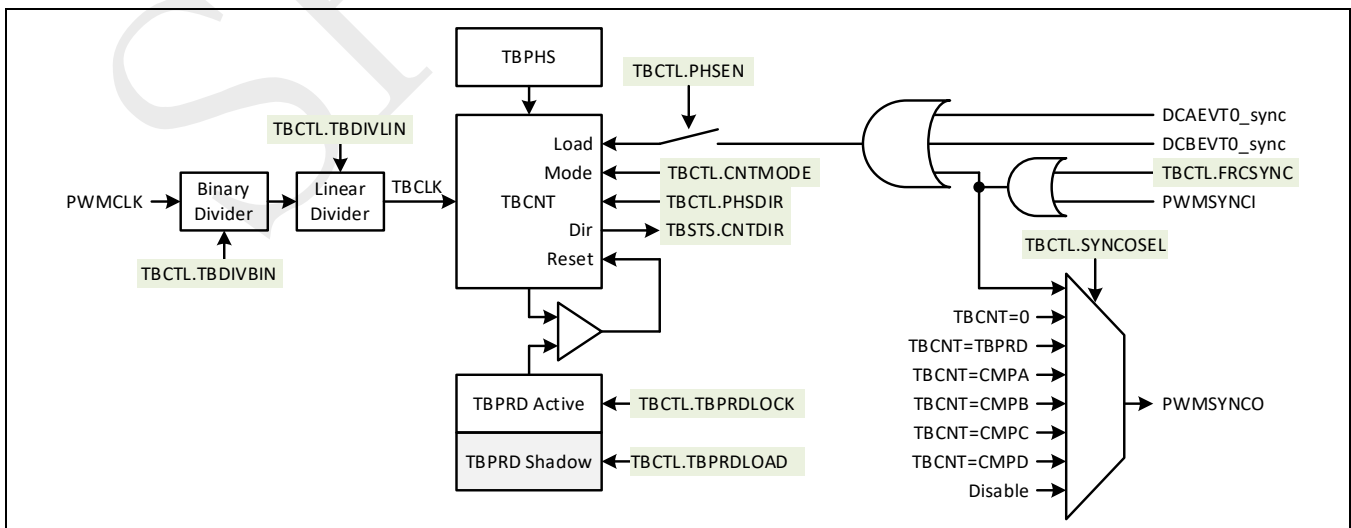
- Comparator results
The results from the 2 external comparators can generate digital compare events.
- ADC start-of-conversion (PWMxSOCA , PWMxSOCB and PWMxSOCC)
Each PWM has three ADC start of conversion signals. The event to trigger the start of conversion signal is configured in the Event Trigger sub-module.
- PWM output (PWMxA and PWMxB)
The PWM outputs can be configured to be multiplexed with the GPIO as in [Table 4-1](#).

8.2 Time-Base (TB) sub-module

The time-based sub-module handles the timing for the PWM module. The key signals is given in [Figure 8-3](#). It is programmable as below:

- Specify the time-base clock (TBCLK) dividing ratio from the input clock.
- Set the time-base counter (TBCNT) mode:
 - Count-up
 - Count-down
 - Count-up-and-down
- Specify the time-base counter (TBCNT) period.
- Determine the phase relationship with other PWM module.
- Generate the events TBCNT=TBPRD and TBCNT=0.
- Specify the synchronization trigger events input and the action taken on the counter.

Figure 8-3: Time-Base sub-module signals



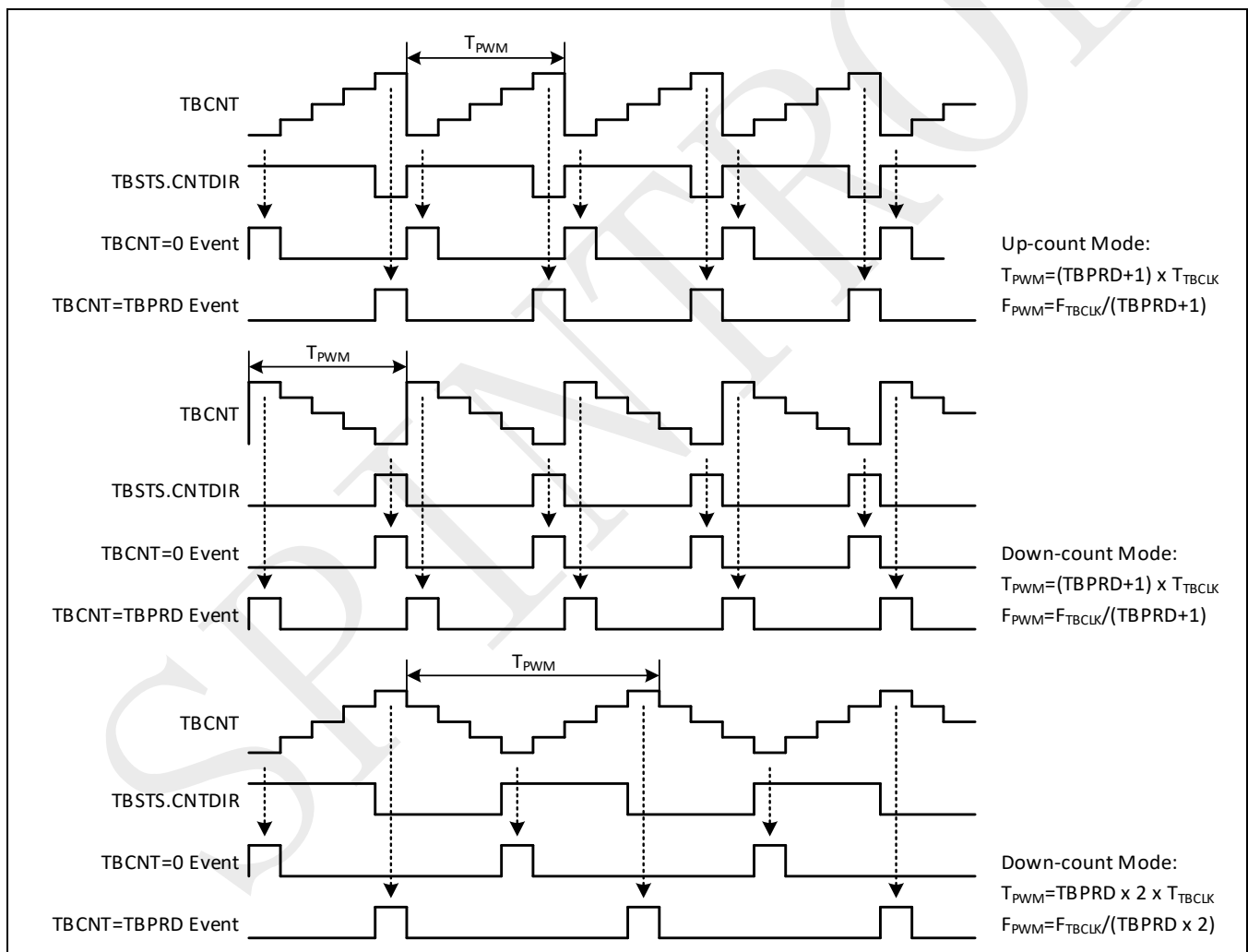
The frequency of the PWM events is controlled by the counter mode (TBCTL.CNTMODE) and the time-base period (TBPRD) register. Figure 8-4 shows the period and frequency relationships for the up-count, down-count, and up-down-count time-base counter modes.

In up-count mode, the time-base counter starts from zero and increases until it reaches TBPRD. Then the TBCNT is reset to zero and starts to increase again.

In down-count mode, the time-base counter starts from TBPRD and decreases until it reaches zero. Then the TBCNT is reset to TBPRD and starts to decrease again.

In up-down-count mode, the time-base counter increases from zero until it reaches TBPRD and then decrease until it reaches zero. Read back from the CNTDIR bit in the TBSTS register is 1 when the counter increases from zero to (TBPRD-1) and 0 when the counter decreases from TBPRD to 1.

Figure 8-4: Time-base counter waveform



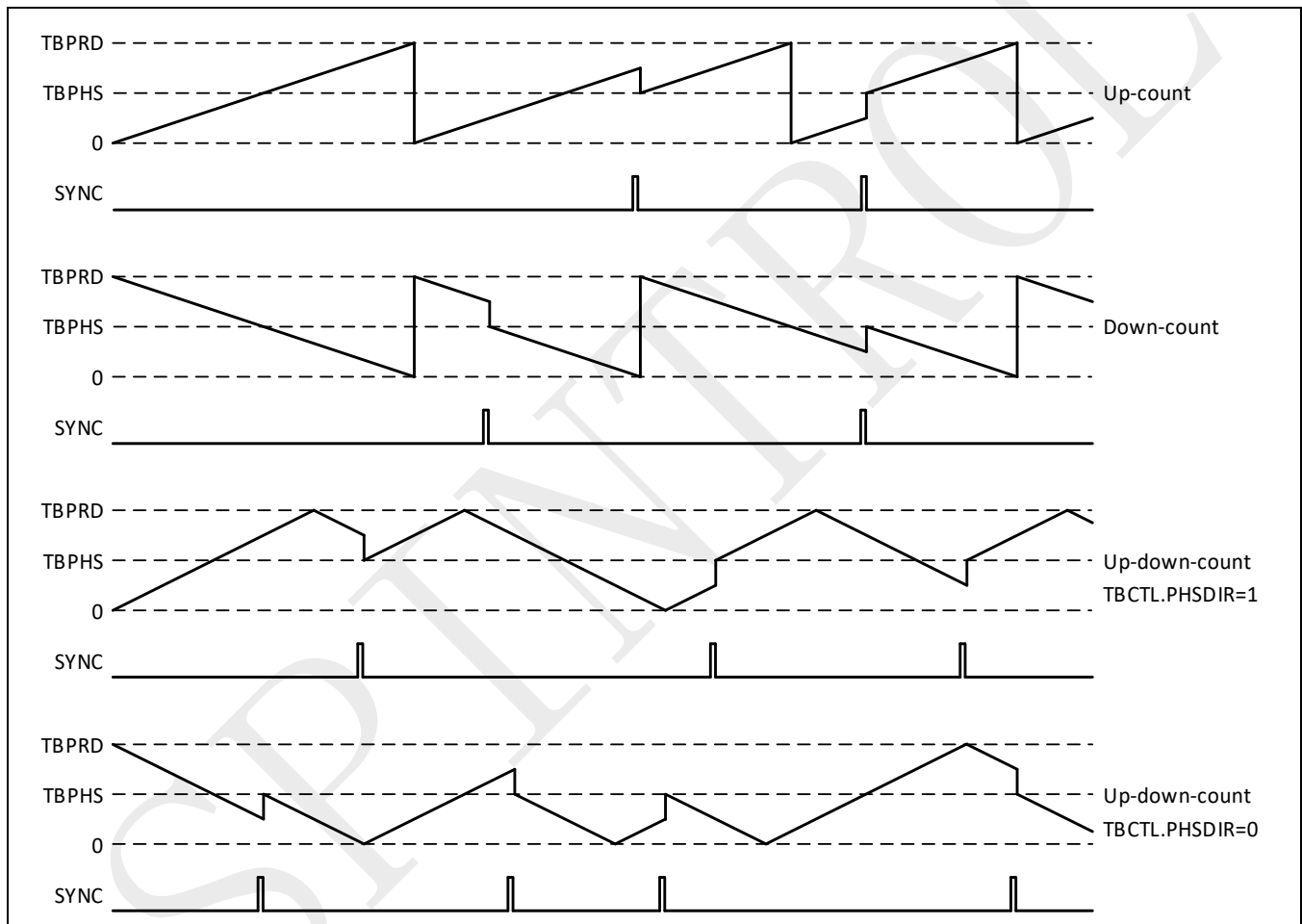
Shadow scheme is provided for TBPRD register in the SPC1169. In this mode, write to the TBPRD register will not immediately change the active value. Instead, the active value of TBPRD is updated from the shadow register when TBCNT=0 and/or SYNC event as defined by TBCTL.TBPRDLOAD.

While TBCTL.PHSEN=1, phase synchronization is enabled for the TBCNT. In this case, whenever there is a synchronization event, the TBCNT will load the value in time-base phase (TBPHS) register. A timing diagram is given in Figure 8-5. For up-count and down-count mode, the counter keeps the direction after

synchronization. For up-down-count mode, the direction is defined by PHSDIR bit in time-base control (TBCTL) register. As shown in Figure 8-3, the synchronization event can be the external input PWMSYNCI signal, the digital compare synchronization A and B, local software forced synchronization, and the global software forced synchronization.

The phase synchronization enables the PWM module to be automatically synchronized to the time base of another PWM module. While TBCTL.PHSEN=0, the PWM will ignore the synchronization input pulse. However, the pulse will go through to PWMSYNCO and used to synchronize other PWM modules.

Figure 8-5: Time-base phase synchronization waveform

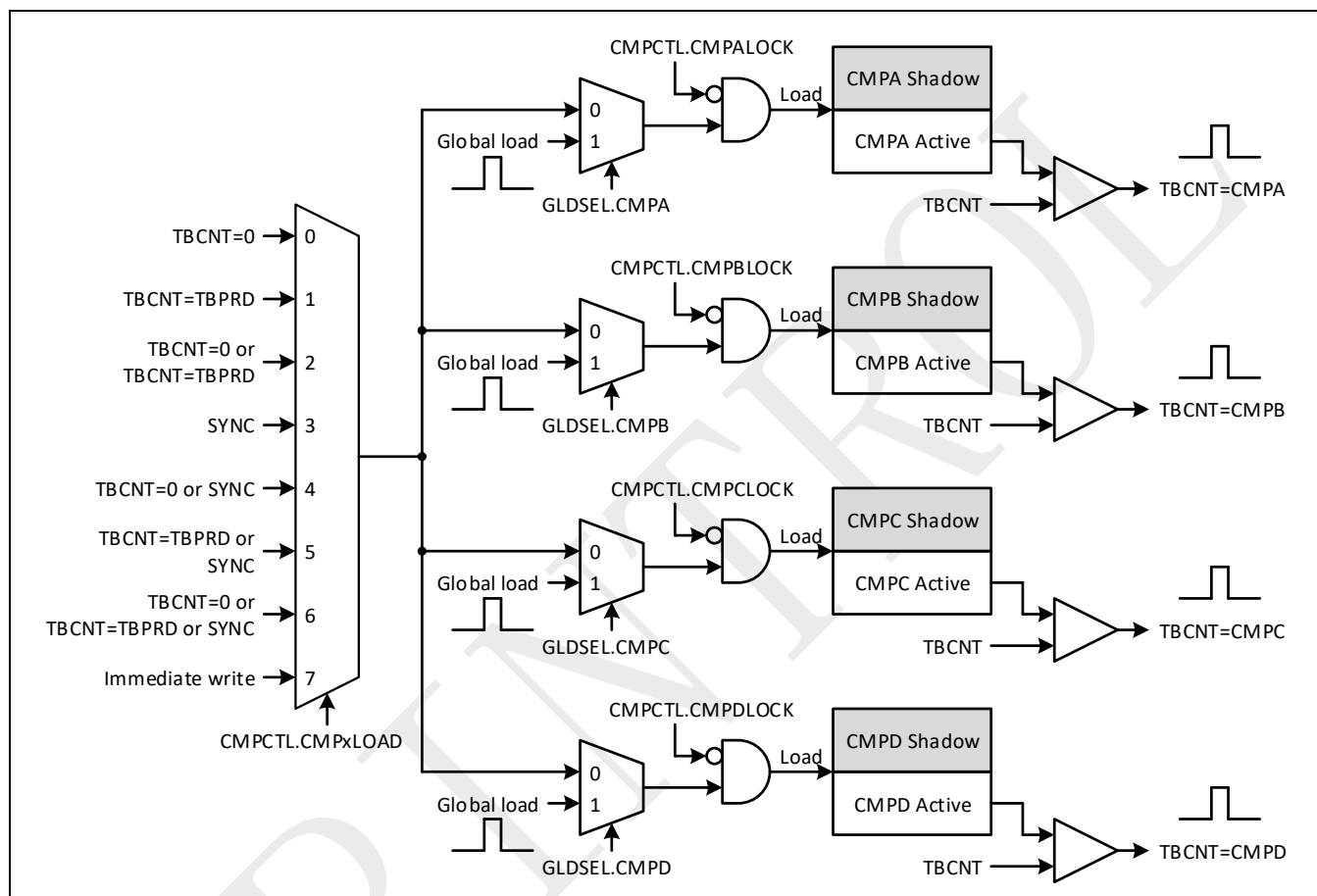


Note: It takes one PWM clock delay from the synchronization input to the actual synchronization action, results in one time-base cycle delay if TBCTL.TBDIVBIN=0 and TBCTL.TBDIVLIN=0. Please take this into account when setting the TBPHS register.
The register FRCSYNC in PWMCFG module provides a more flexible and convenient synchronization among the PWMs.

8.3 Counter Compare (CC) sub-module

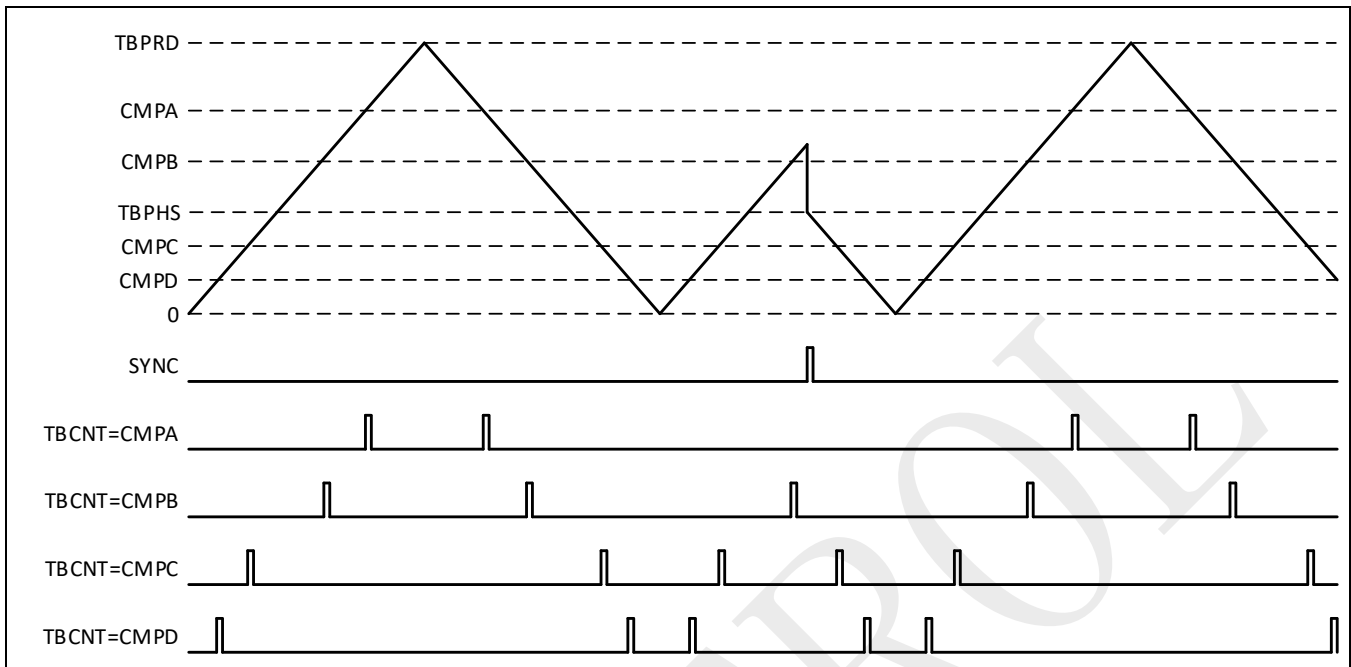
The counter-compare sub-module continuously compare the TBCNT to the counter-compare A (CMPA), counter-compare B (CMPB), counter-compare C (CMPC), and counter-compare D (CMPD) registers. When TBCNT is equal to the counter-compare register, a corresponding event will be generated.

Figure 8-6: Counter-Compare sub-module structure



The basic structure of counter-compare sub-module is shown in [Figure 8-6](#). The counter-compare registers can be configured in shadow mode that the write to the register affects only the shadow register. The active values of the counter-compare register are updated from the shadow register upon the event defined by CMPxLOAD bit in counter compare control (CMPCTL) register. Global shadow load option can be enabled via CMPx bit in GLDSEL register as described in detail in [Table 8-14](#) and [Table 8-15](#). A sample waveform is shown in [Figure 8-7](#).

Figure 8-7: Counter-Compare sub-module waveform



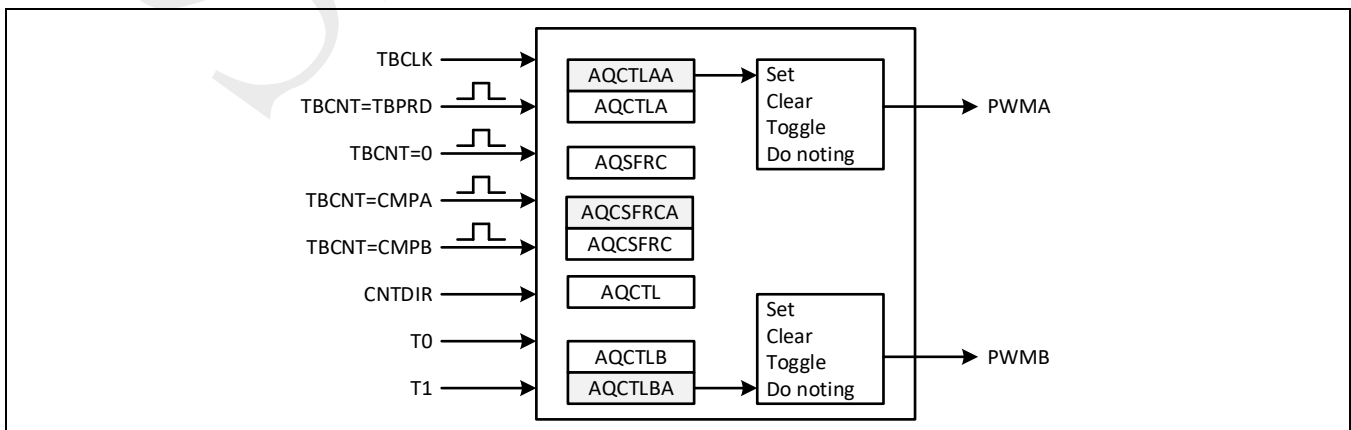
8.4 Action-Qualifier (AQ) sub-module

The action-qualifier sub-module determines which events are converted into various action types and produces the required switched waveforms at PWMA and PWMB outputs.

As shown in Figure 8-8, the events includes:

- TBCNT=TBPRD
- TBCNT=0
- TBCNT=CMPA
- TBCNT=CMPB
- T0/T1 events, which are based on comparator, trip-zone and sync input events
- Software forced event

Figure 8-8: Action-Qualifier sub-module inputs and outputs



The action-qualifier handles these events as well as their priority if they occur concurrently. Based on the events, it takes one of following actions as defined by AQCTLA and AQCTLB registers.

- Set output to a high level
- Clear output to a low level
- Toggle the output
- Do nothing

Shadow mode or the immediate write mode can be selected via AQCTL.AQCTLALOAD, AQCTL.AQCTLBL OAD and AQSFRC.CSFLOAD. Global load is also supported by setting GLDSEL.AQCTLA, GLDSEL.AQCTLB and GLDSEL.AQCSFRC. It is similar to the shadow for CMPx registers as shown in [Figure 8-6](#).

[Table 8-1](#) through [Table 8-3](#) list the priority of action-qualifier in different count modes.

Table 8-1: Action-Qualifier event priority for Up-Count mode

Priority Level	Event
1(Highest)	Software forced event
2	TBCNT=TBPRD (PRD)
3	T0 on up-count (T0U)
4	T1 on up-count (T1U)
5	TBCNT=CMPB on up-count (CBU)
6	TBCNT=CMPA on up-count (CAU)
7 (Lowest)	TBCNT=0 (ZRO)

Table 8-2: Action-Qualifier Event Priority for Down-Count Mode

Priority Level	Event
1(Highest)	Software forced event
2	TBCNT=0 (ZRO)
3	T0 on down-count (T0D)
4	T1 on down-count (T1D)
5	TBCNT=CMPB on down-count (CBD)
6	TBCNT=CMPA on down-count (CAD)
7 (Lowest)	TBCNT=TBPRD (PRD)

Table 8-3: Action-Qualifier Event Priority for Up-Down-Count Mode

Priority Level	Event	
	When increase from 0 to TBPRD-1	When decrease from TBPRD to 1
1(Highest)	Software forced event	Software forced event
2	T0 on up-count (T0U)	T0 on down-count (T0D)
3	T1 on up-count (T1U)	T1 on down-count (T1D)
4	TBCNT=CMPB on up-count (CBU)	TBCNT=CMPB on down-count (CBD)
5	TBCNT=CMPA on up-count (CAU)	TBCNT=CMPA on down-count (CAD)
6 (Lowest)	TBCNT=0 (ZRO)	TBCNT=TBPRD (PRD)

Figure 8-9 and Figure 8-10 demonstrate timing diagrams for the symmetric and asymmetric scenarios of the action qualifier.

Figure 8-9: Up-Down-Count, dual edge symmetric waveform of Action-Qualifier

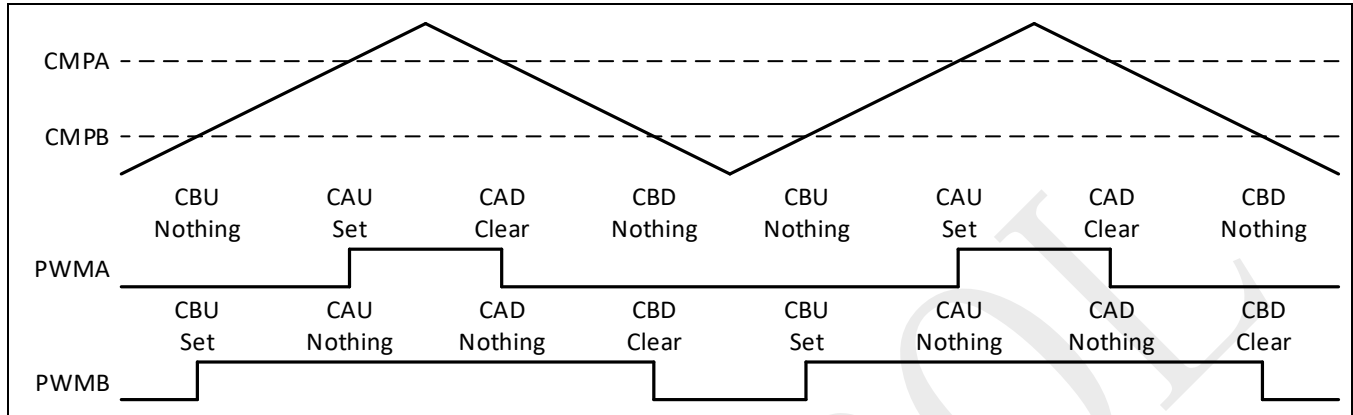
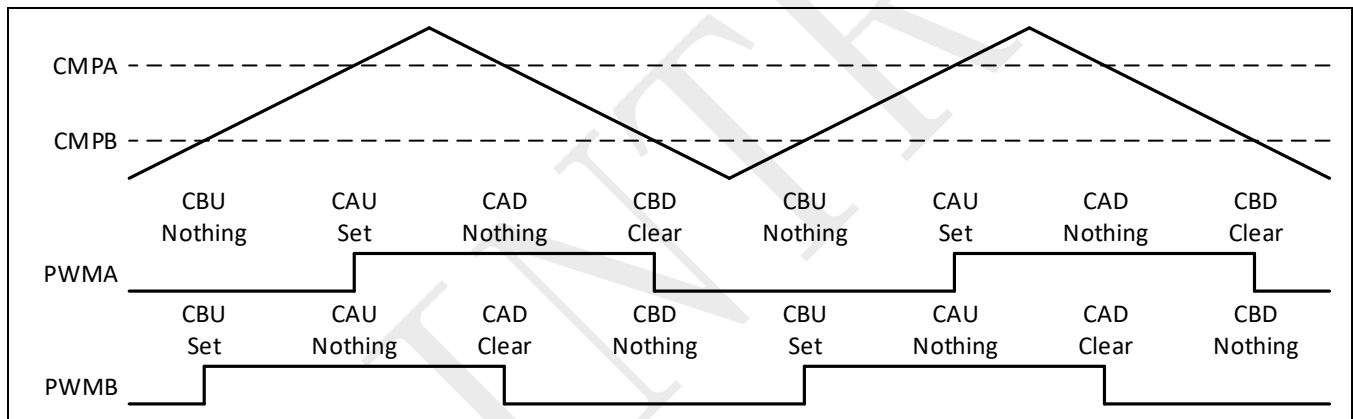


Figure 8-10: Up-Down-Count, dual edge asymmetric waveform of Action-Qualifier

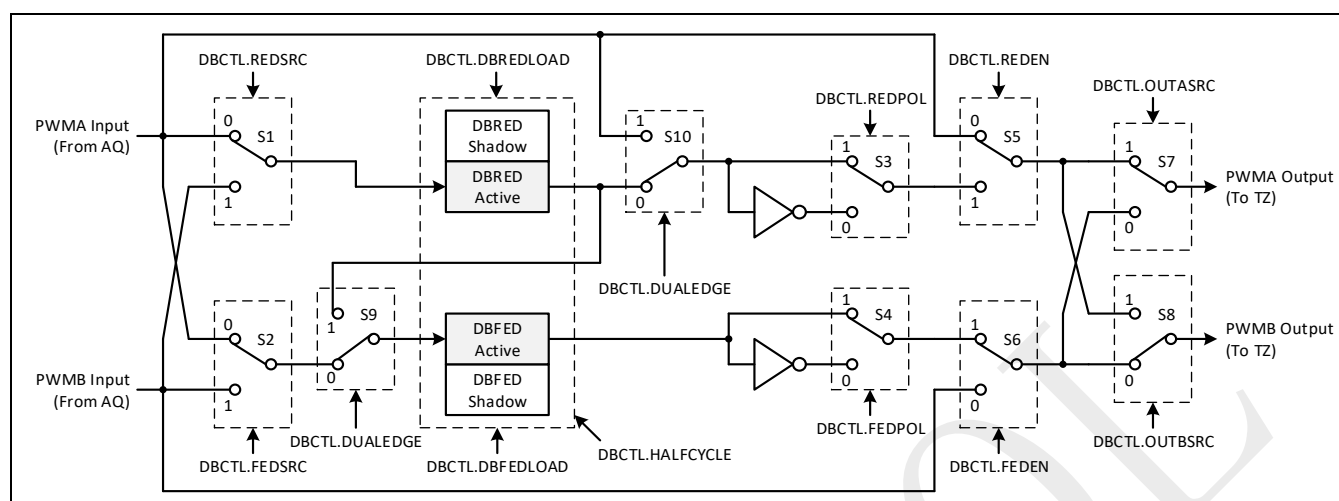


8.5 Dead-Band (DB) sub-module

The dead-band sub-module deals with the signals from the action qualifier to further shaping the waveform with independent and programmable rising-edge and falling-edge delay as well as polarity control.

The key functions of the module includes:

- Generate signal pairs output (PWMA and PWMB) with dead-band relationship from a single PWMA input
- Add programmable delay to rising edge
- Add programmable delay to falling edge
- Support half clock cycle delay
- Adjust the polarity of the output signal
- Bypass the dead-band signal shaping and directly feed through the signals from the action qualifier

Figure 8-11: Dead-Band sub-module structure

Basic structure of the dead-band sub-module is given in [Figure 8-11](#). The REDSRC and FEDSRC bits in dead-band control (DBCTL) register determines the input signal source of the rising edge delay and falling edge delay. All combinations to control the switches from S3 to S10 are supported. Taking DBCTL.REDSRC=0 and DBCTL.FEDSRC=0 for example, when PWMA is configured as the input for both rising-edge and falling-edge delay, the typical operating modes are listed in [Table 8-4](#).

As a special case, dual-edge delay can be achieved by setting the DUALEDGE bit in DBCTL register, when both rising-edge and falling-edge are applied to the same path. This is useful to generate a waveform pair with the same period and duty-cycle but specified phase shift between each other.

Table 8-4: Typical Dead-Band operating modes

Mode	Mode Description	REDPOL S3	FEDPOL S4	REDEN S5	FEDEN S6
1	Both PWMA and PWMB pass through	X	X	0	0
2	Active high complementary	1	0	1	1
3	Active low complementary	0	1	1	1
4	Active high	0	0	1	1
5	Active low	1	1	1	1
6	PWMA output = PWMA input with no delay PWMB output = PWMA input with falling-edge delay	0 or 1	0 or 1	0	1
7	PWMA output = PWMA input with rising-edge delay PWMB output = PWMA input with no delay	0 or 1	0 or 1	1	0

Figure 8-12 illustrates the waveform of typical dead-band sub-module.

Figure 8-12: Typical Dead-Band waveform

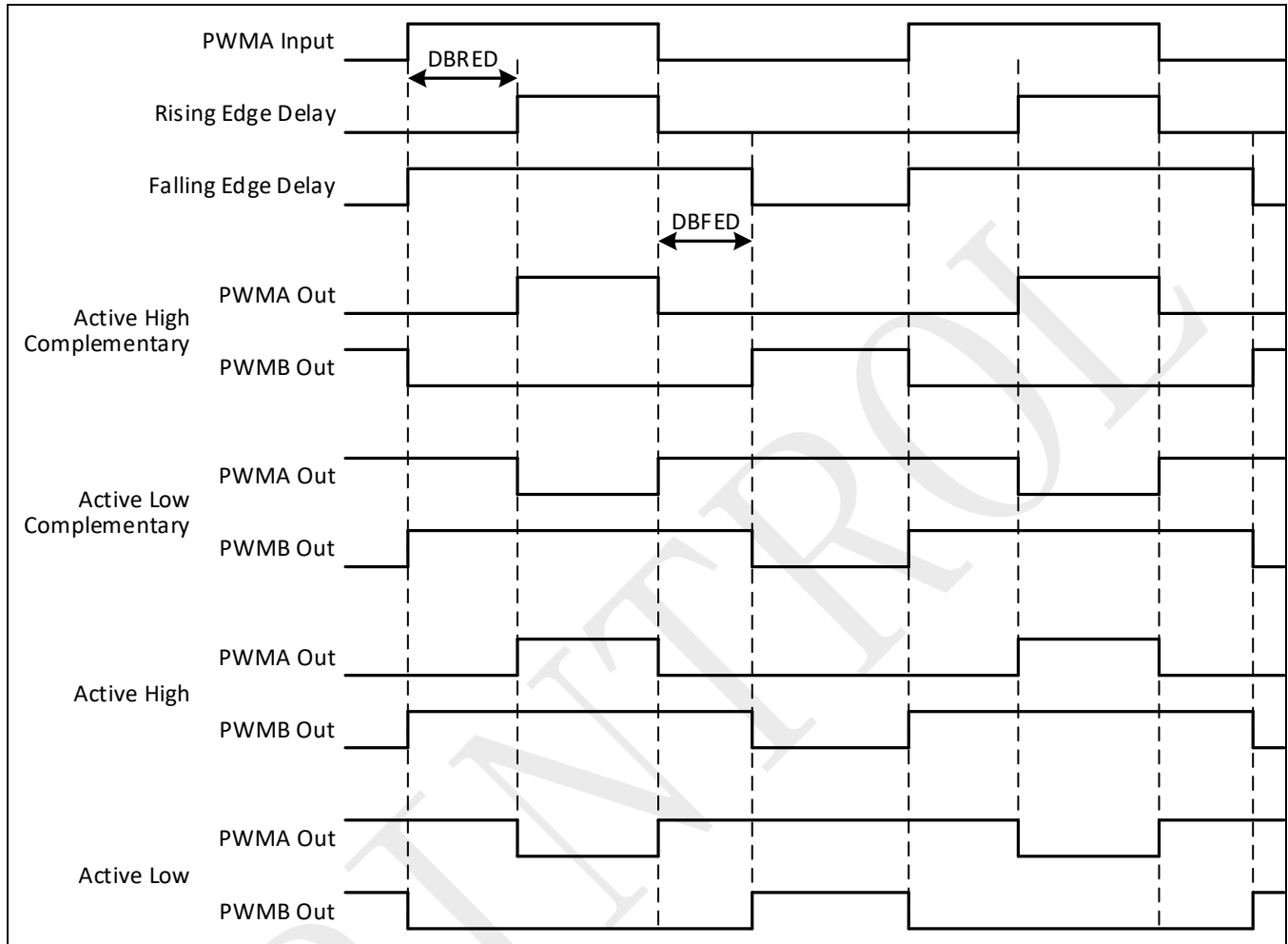
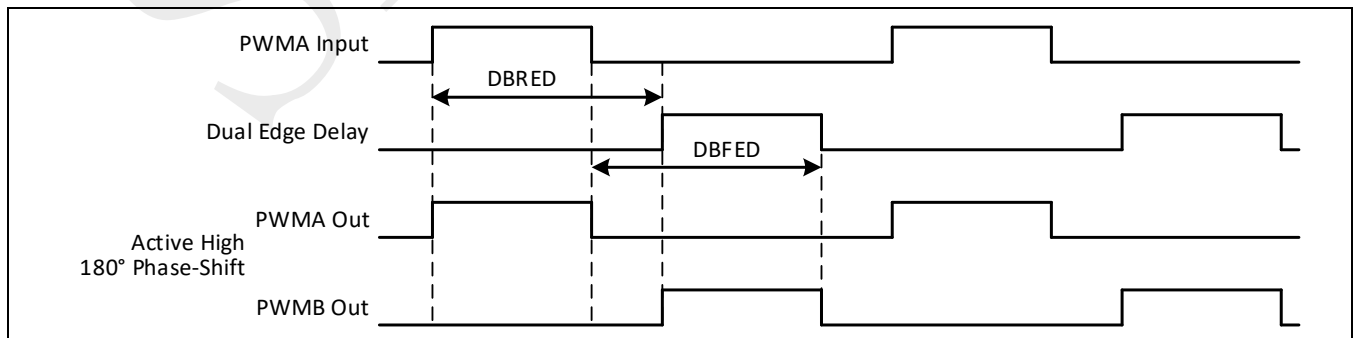


Figure 8-13 illustrates the case to generate waveform with 180° phase shift by enable dual-edge delay and setting both DBRED and DBFED to half the waveform period.

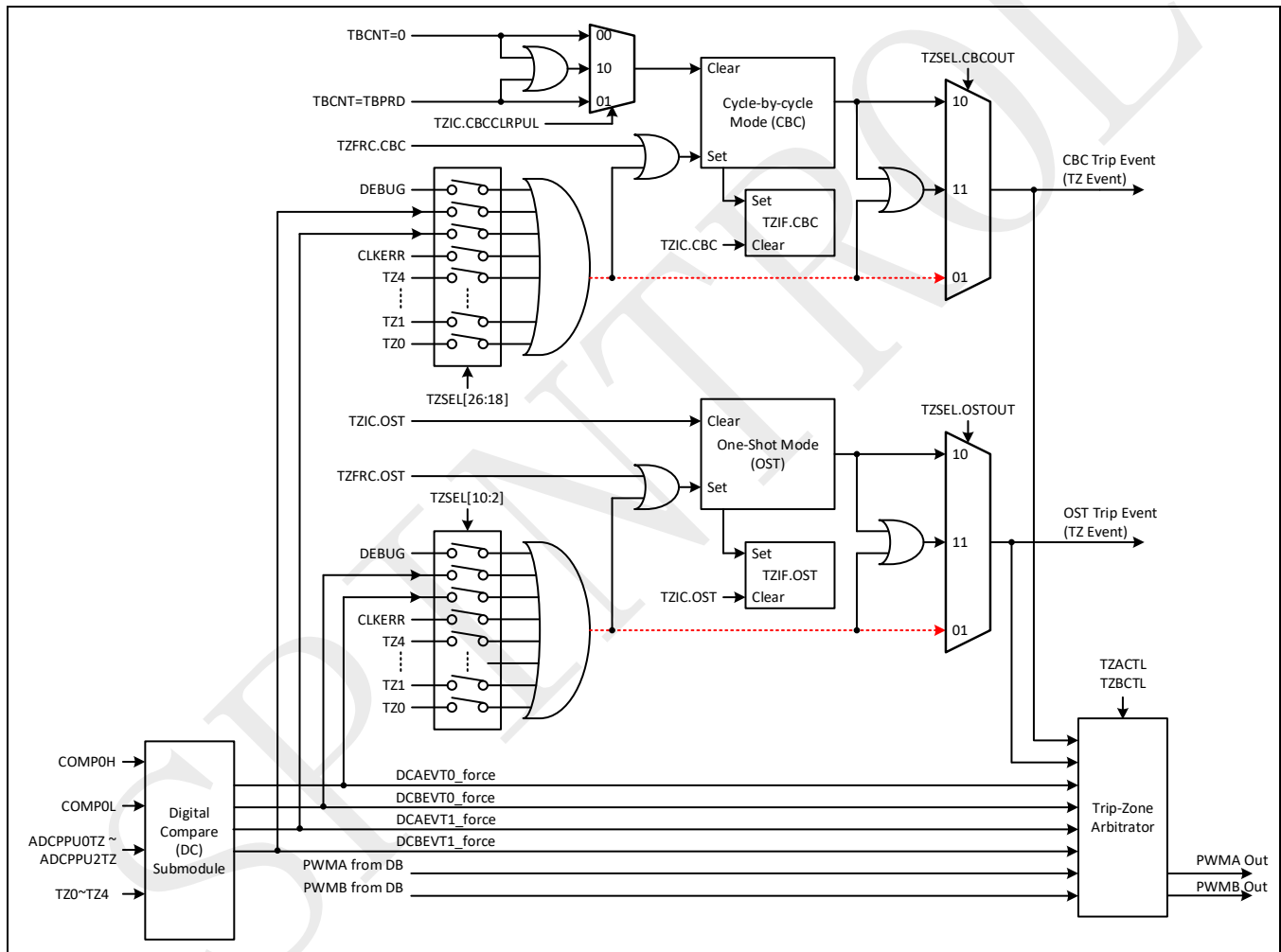
Figure 8-13: Dual-edge delay Dead-Band waveform



8.6 Trip-Zone (TZ) sub-module

The trip-zone sub-module determines the PWM outputs when external fault or trip condition occurs. As shown in Figure 8-14, the input can be from the TZ signal, the clock error event from the PLL unlock detection or the CLKDET module, the lockup or halted condition from the CPU, or the digital compare (DC) output. The five TZ signals (TZ0~TZ4) are sourced from the GPIO multiplexer with programmable signal polarity according to the TZ#SRCCTL registers in PWMCFG module. The DCxEVTy.force signals are generated by the DC submodule as described in the next Section. Software-forced tripping is also supported and interrupt can be generated on any trip-zone input.

Figure 8-14: Trip-Zone sub-module logic



Upon a trip-zone condition, the PWMA and PMWB outputs can be forced the to one of the following:

- High
- Low
- High-impedance
- No action

The event actions on PWMA output is controlled by register TZACTL, and the priority of these event actions is TZU > TZD > DCAEVT0U > DCAEVT0D > DCAEVT1U > DCAEVT1D. It means that when both high priority event and low priority event happen, the output is determined by the configuration of the high

priority event. For example, when both TZU and DCAEVT1U events happen, TZU register field determines the PWM output while DCAEVT1U field is ignored.

Both one-shot trip (OST) and cycle-by-cycle (CBC) trip are supported. The one-shot trip is typically used for major over-current or short circuits condition while the cycle-by-cycle trip is usually used for current limiting operation.

When a one-shot trip event occurs, the action specified in the TZACTL and TZBCTL registers is immediately taken on the PWMA and PWMB outputs. Meanwhile, the one-shot trip event flag (TZIF.OST) is set and a PWM_TZINT interrupt is generated if it is enabled in trip-zone enable interrupt (TZIE) register. The TZIF.OST flag must be manually cleared by writing 1 to the TZIC.OST bit.

When a cycle-by-cycle trip event occurs, the action specified in the TZACTL and TZBCTL registers is immediately taken on the PWMA and PWMB outputs. Meanwhile, the cycle-by-cycle trip event flag (TZIF.CBC) is set and a PWM_TZINT interrupt is generated if it is enabled in the TZIE register. The specified condition is automatically cleared if the trip event is no longer present when the TBCNT reaches zero or TBPRD as specified by TZIC.CBCCLRPUL. However, the TZIF.CBC flag bit remains set until it is manually cleared by writing 1 to the TZIC.CBC bit. If the cycle-by-cycle trip event is still present when the TZIF.CBC flag is cleared, it will be set again immediately.

Note: The asynchronous logic path shown as dashed line in [Figure 8-14](#) is used to observe the original trip-zone event signal when TZSEL.OSTOUT or TZSEL.CBCOUT is set to 1. In real application, the latched path should be enabled to implement the one-shot or cycle-by-cycle trip function as described above. i.e. TZSEL.OSTOUT and TZSEL.CBCOUT should be 2 or 3.

The digital compare DCAEVT0/1 or DCBEVT0/1 event is generated based on the DCAH/DCAL and DCBH/DCBL bits of trip-zone digital compare selection (TZDCSEL) register. When a digital compare event occurs, the action specified in the TZACTL and TZBCTL registers is immediately taken on the PWMA and PWMB outputs. Meanwhile, the DC trip event flag (TZIF.DCAEVT0/1 and TZIF.DCBEVT0/1) is set and a PWM_TZINT interrupt is generated if it is enabled in the TZIE register. The specified condition is automatically cleared when the DC trip event is no longer present. However, the flag bits in the TZIF register remains set until it is manually cleared by writing 1 to the corresponding bits in the TZIC register. If the DC trip event is still present when the TZIF bit is cleared, it will be set again immediately.

[Figure 8-15](#) illustrates the trip-zone interrupt logic.

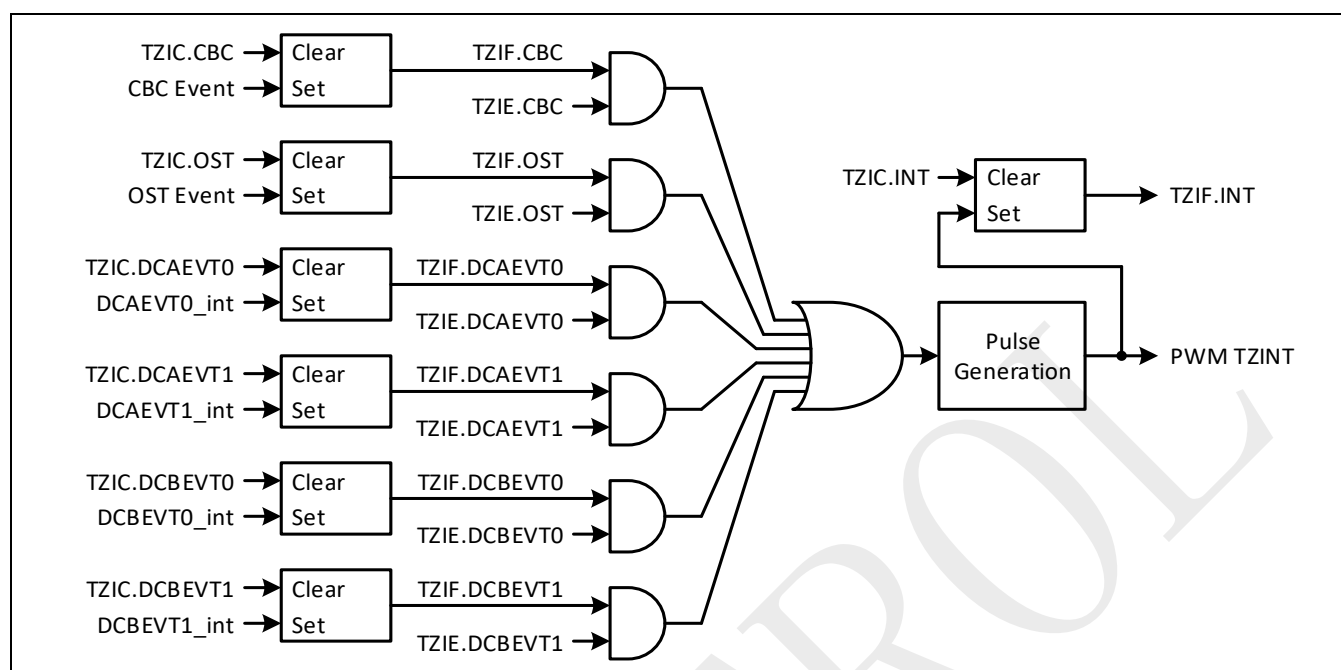
Figure 8-15: Trip-Zone interrupt generation

Table 8-5 lists the priority of events that effect the final PWM outputs.

Table 8-5: Priority of events that effect the final PWM outputs

Priority Level	Event
1(Highest)	Trip event
2	Continuous software force via AQCSFRC register
3	Dead-Band control logic
4	One-time software force via AQSFRC register
5 (Lowest)	Action-Qualifier control logic

8.7 Digital Compare (DC) sub-module

The digital compare sub-module takes the analog comparator (COMP) module outputs, the ADC post-processing unit (ADCPPU) module trip-zone outputs, and TZ0, TZ1, TZ2, TZ3 and TZ4 inputs to generate digital compare A high/low (DCAH, DCAL) and digital compare B high/low (DCBH, DCBL) signals. As shown in Figure 8-16, these signals generate raw DCAEVT0/1 and DCBEVT0/1 events, which can be either filtered or directly fed to the event processing logic, so as to generate signals for time-base, trip-zone and event-trigger sub-modules.

The raw DCAEVT0/1 and DCBEVT0/1 events can be filtered to remove noise by inserting a blank window to gate the event for a certain time as shown in Figure 8-17. Upon the TBCNT=TBPRD or TBCNT=0 pulse, the offset counter loads the value from the DCOFFSET register and starts to count down. When the counter expires, the blank window starts and the duration is certain TBCLK clocks determined by DCFWINDOW register. During the blank window, all events are ignored. Before the window starts and after the window ends, events can generate SOC, SYNC, interrupt and force signals as before. To take into account the cross-talk from/to other blocks in real application, the generated blank windows from PWM modules can be enabled to filter local raw DC events. The polarity of the blank window can be inverted by setting the BLANKINV bit of the DCFCTL register. As shown in the timing diagram in Figure 8-18, the new blank window will immediately start even if the old one does not expire.

Figure 8-16: Digital Compare sub-module structure

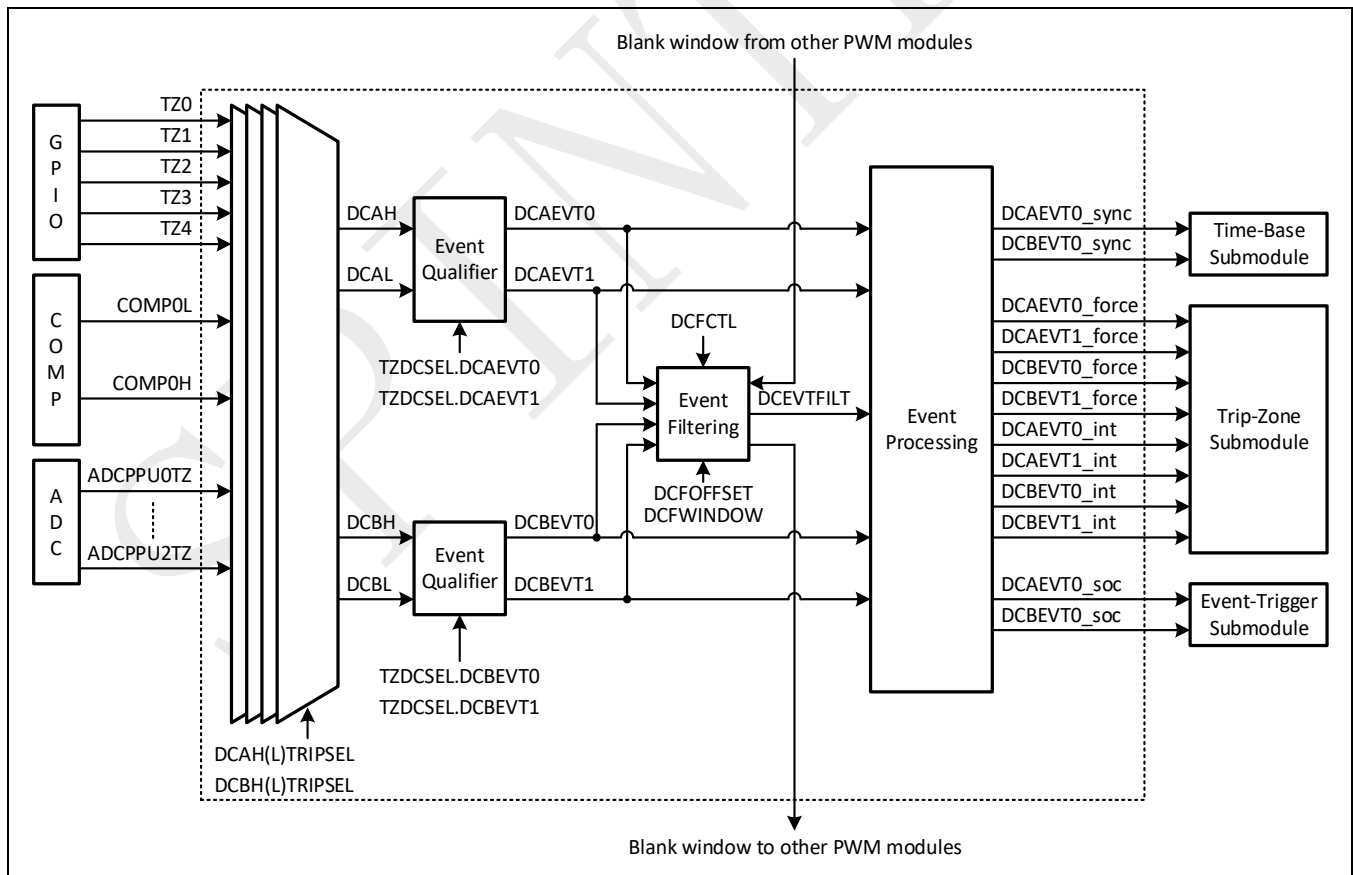


Figure 8-17: Event filtering logic

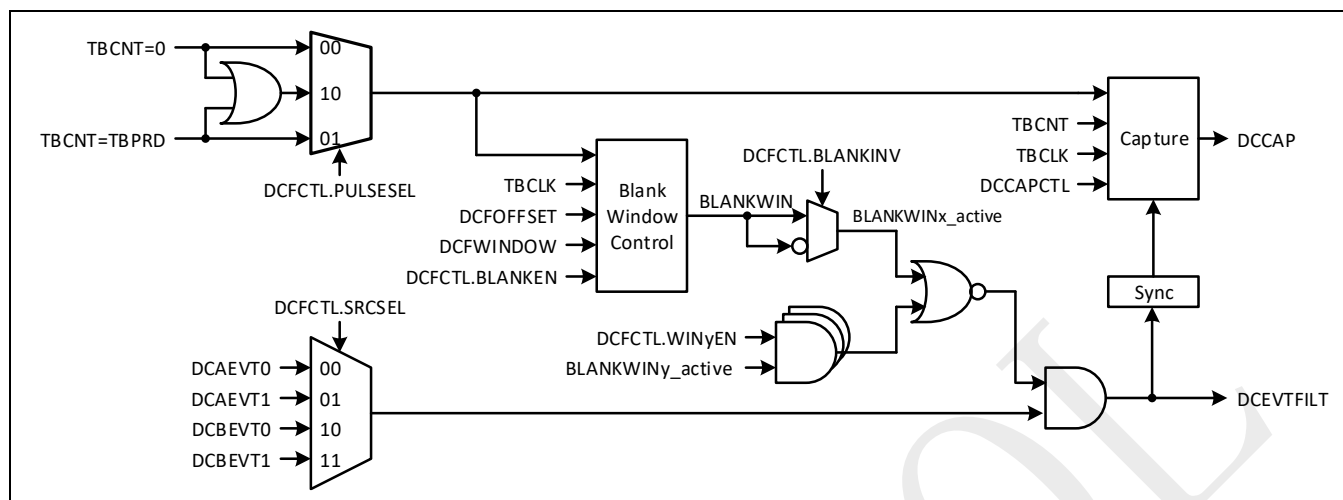
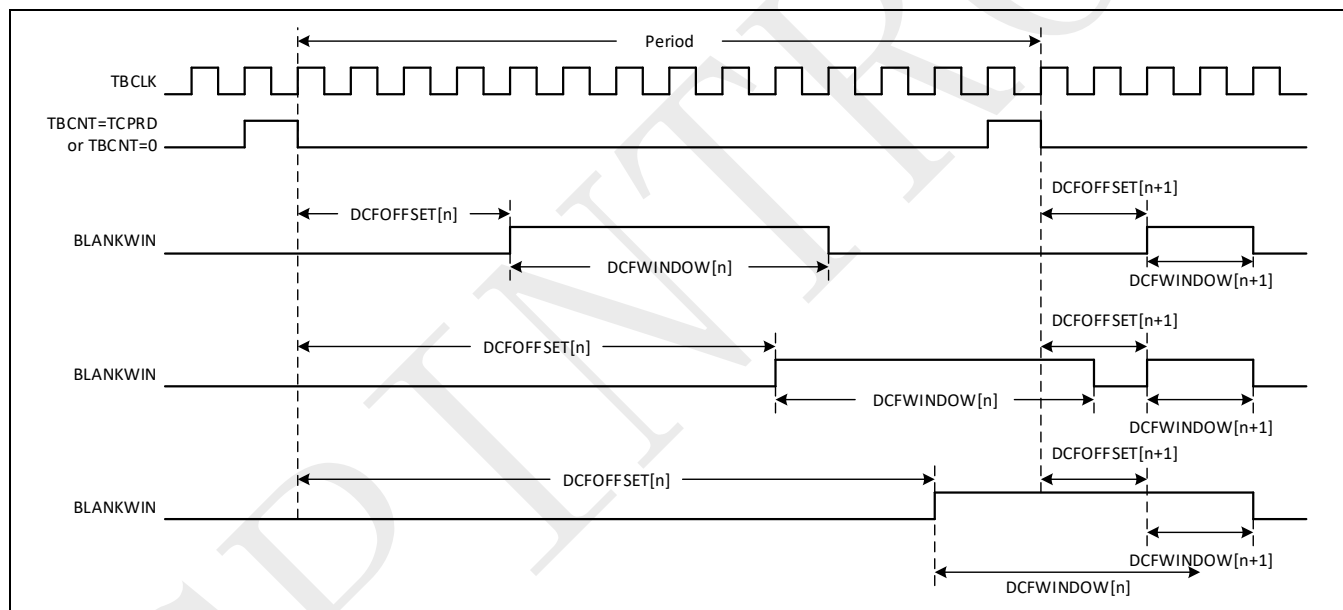


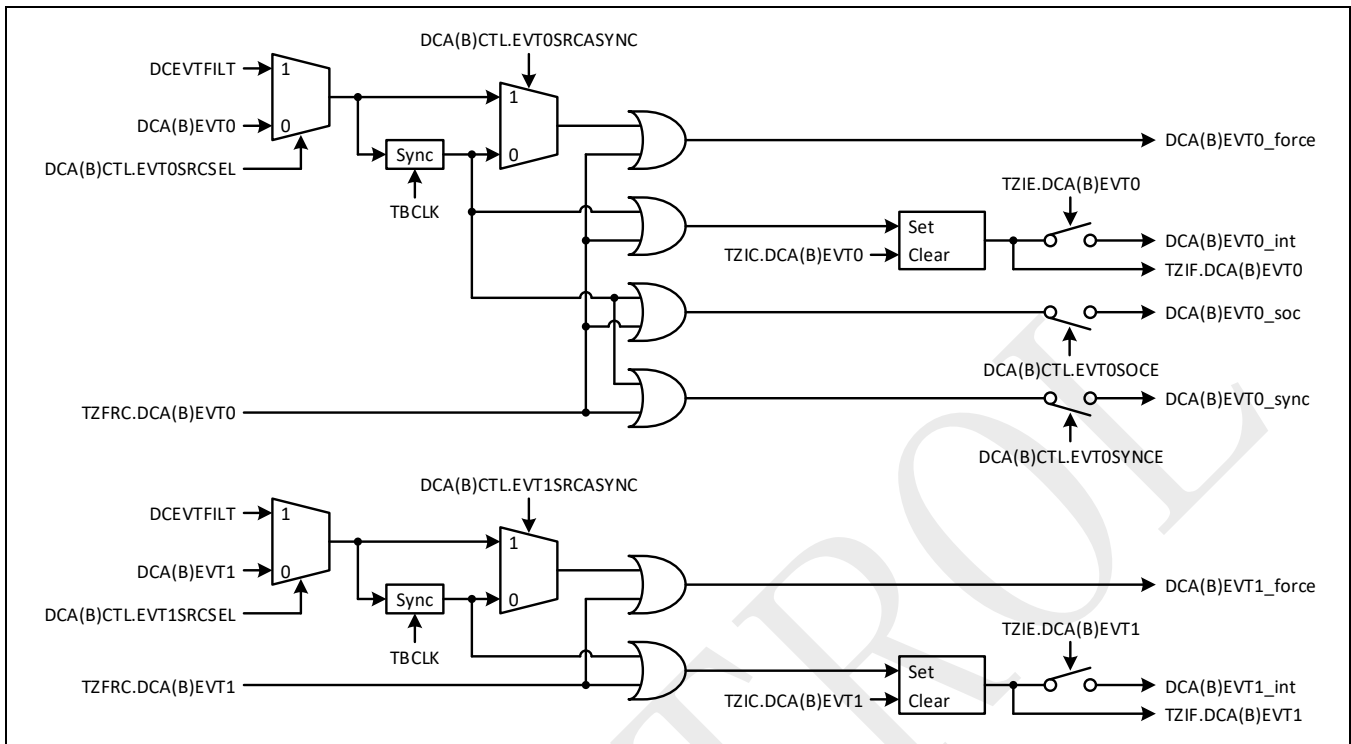
Figure 8-18: Blank window timing diagram



As shown in [Figure 8-19](#), the raw DCAEVT0, DCAEVT1, DCBEVT0, DCBEVT1 and the filtered DCEVTFILT pass through the event processing block and generate the final DCAEVT0, DCAEVT1, DCBEVT0 and DCBEVT1 events according to EVT0SRCSEL and EVT1SRCSEL fields in DCACTL and DCBCTL registers. Based on these events, further signals can be generated as below:

- Force signals for trip-zone sub-module
DCAEVT0_force, DCAEVT1_force, DCBEVT0_force, DCBEVT1_force
- Synchronous signal for time-base sub-module
DCAEVT0_sync, DCBEVT0_sync
- ADC start of conversion
DCAEVT0_soc, DCBEVT0_soc
- Interrupt
DCAEVT0_int, DCAEVT1_int, DCBEVT0_int, DCBEVT1_int

Figure 8-19: Event processing



Below is an example in the scenario that COMP0 are used for shut-current monitoring and generate filtered DC trip-zone events to control both PWMxA and PWMxB outputs.

- DCAH is set as the logic OR of COMP0H, and COMP0L via DCAHTRIPSEL register.
- Use DCAH=high to generate the raw DCAEVT0 signal via setting TZDCCSEL.RAWDCAEVT0 = 4.
- Use the raw DCAEVT0 as the DC filter input via setting DCFCTL.SRCSEL=0.
- Use the filtered signal DCEVTFILT as the final DCAEVT0 event via setting DCACTL.EVT0SRCSEL=1.
- Use the filtered signal DCEVTFILT as the final DCBEVT0 event via setting DCBCTL.EVT0SRCSEL=1.
- Configure the PWMxA output action upon final DCAEVT0 trip-zone event via DCAEVT0U and DCAEVT0D fields in TZACTL register.
- Configure the PWMxB output action upon final DCBEVT0 trip-zone event via DCBEVT0U and DCBEVT0D fields in TCBCTL register.

8.8 Event-Trigger (ET) sub-module

The event-trigger sub-module manages the events generated by the time-base sub-module, the counter-compare sub-module, and the digital-compare sub-module to generate an interrupt to the CPU and/or a start-of-conversion (SOC) pulse to the ADC when a selected event occurs.

Figure 8-20: Event-Trigger sub-module structure

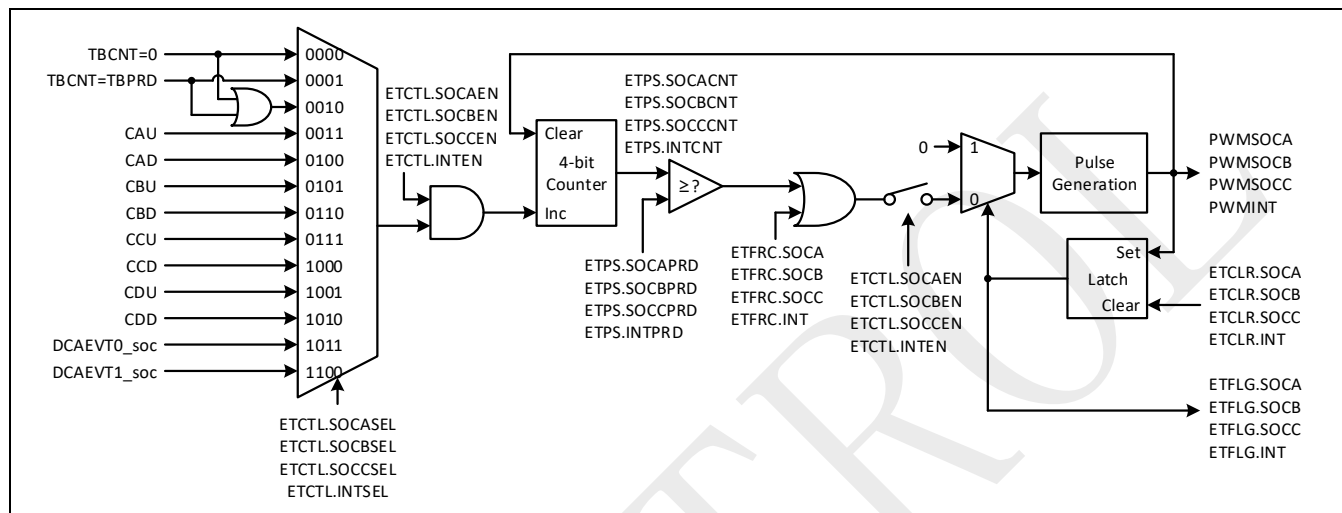


Figure 8-20 illustrates the basic structure of the event-trigger sub-module. Software forced SOCA, SOCB, SOCC and INT pulses are supported in the SPC1169. By using a 4-bit counter, the pulse can be generated in following scenario:

- Never generate
- Generate a pulse on every event
- Generate a pulse on every second event
-
- Generate a pulse up to every fifteenth event

8.9 Global reload

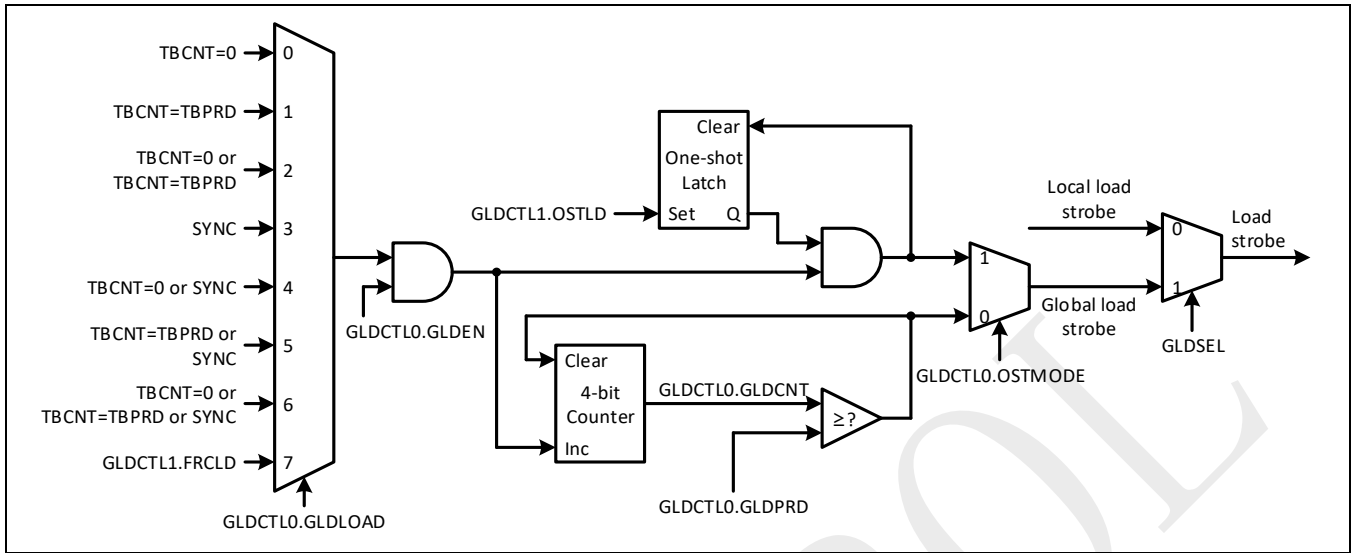
The content transfer from the shadow register to the active register can be done upon the global load strobe as shown in [Figure 8-21](#). This feature is enabled by setting the GLDEN bit of GLDCTL0 register to 1. And the bits in GLDSEL register decide whether the corresponding active register is updated upon the global load strobe or the local load strobe.

When GLDCTL0.OSTMODE=0, the pre-scaler is applied so that the global load strobe is generated every GLDCTL0.GLDPRD occurrences of the global reload events.

When GLDCTL0.OSTMODE=1, one-shot reload mode is enabled so that only the first global reload event will generate the global load strobe after GLDCTL1.OSTLD is setting to 1. Once the global load strobe is generated, the GLDCTL1.OSTLD is cleared to 0 until it is manually set to 1 again. Global reload events during GLDCTL1.OSTLD=0 do not generate any global load strobe.

Software forced global reload can be enabled via writing 1 to GLDCTL1.FRCLD.

Figure 8-21: Global reload structure



8.10 Simultaneous write

In some applications, there is a need of simultaneous writes to the registers of same name in different PWMs. SPC1169 provides a PWM link feature by PWMLINK register, so that write to the register of another PWM module can also update the value of the same register in current PWM module. The registers supporting this feature are TBPRD, CMPA, CMPB, CMPC, CMPD, DBRED, DBFED and GLDCTL1.

8.11 Registers

8.11.1 PWM register map

Table 8-6: PWM Module Base Address

Peripheral Module	Base Address
PWM0	0x4000 2000
PWM1	0x4000 2100
PWM2	0x4000 2200
PWM3	0x4000 2300

Table 8-7: PWM Register Map

Register	Offset	Description	Reset Value
SHADOWSTS	0x00	Shadow Status Register	0x00000000
GLDCTL0 *	0x04	Global Shadow to Active Load Control Register 0	0x00000000
GLDCTL1	0x08	Global Shadow to Active Load Control Register 1	0x00000000
GLDSEL *	0x0C	Global Shadow to Active Load Select Register	0x00000000
PWMLINK *	0x10	PWM Link Control Register	0xFFFFFFFF
TBCTL *	0x14	Time-Base Control Register	0x00000006
TBPRD	0x18	Time-Base Period Register	0x00000000
TBPRDA	0x1C	Time-Base Period Active Register	0x00000000
TBPHS	0x20	Time-Base Phase Register	0x00000000
TBCNT	0x24	Time-Base Counter Register	0x00000000
TBFLG	0x28	Time-Base Status Register	0x00000000
TBFLGCLR	0x2C	Time-Base Status Clear Register	0x00000000
CMPCTL *	0x30	Counter-Compare Control Register	0x00000000
CMPA	0x34	Counter-Compare A Threshold Register	0x00000000
CMPAA	0x38	Counter-Compare A Threshold Active Register	0x00000000
CMPB	0x3C	Counter-Compare B Threshold Register	0x00000000
CMPBA	0x40	Counter-Compare B Threshold Active Register	0x00000000
CMPC	0x44	Counter-Compare C Threshold Register	0x00000000
CMPCA	0x48	Counter-Compare C Threshold Active Register	0x00000000
CMPD	0x4C	Counter-Compare D Threshold Register	0x00000000
CMPDA	0x50	Counter-Compare D Threshold Active Register	0x00000000
AQCTL *	0x54	Action-Qualifier Control Register	0x00000000
AQCTLA	0x58	Action-Qualifier Output A Control Register	0x00000000
AQCTLAA	0x5C	Action-Qualifier Output A Control Active Register	0x00000000
AQCTLB	0x60	Action-Qualifier Output B Control Register	0x00000000
AQCTLBA	0x64	Action-Qualifier Output B Control Active Register	0x00000000
AQSFRC	0x68	Action-Qualifier Software Force Register	0x00000000
AQCSFRC	0x6C	Action-Qualifier Continuous Software Force Register	0x00000000

Register	Offset	Description	Reset Value
AQCSFRCA	0x70	Action-Qualifier Continuous Software Force Active Register	0x00000000
DBCTL *	0x74	Dead-Band Generator Control Register	0x00000070
DBCTLA	0x78	Dead-Band Generator Control Active Register	0x00000070
DBRED	0x7C	Dead-Band Generator Rising Edge Delay Register	0x00000000
DBREDA	0x80	Dead-Band Generator Rising Edge Delay Active Register	0x00000000
DBFED	0x84	Dead-Band Generator Falling Edge Delay Register	0x00000000
DBFEDA	0x88	Dead-Band Generator Falling Edge Delay Active Register	0x00000000
TZSEL *	0x8C	Trip-Zone Event Select Register	0x00030003
TZFLG	0x90	Trip-Zone Flag Register	0x00000000
TZFLGCLR	0x94	Trip-Zone Flag Clear Register	0x00000000
TZDCSEL *	0x98	Trip-Zone Digital Compare Event Select Register	0x00000000
TZACTL	0x9C	Trip-Zone Output A Control Register	0x00000000
TZBCTL	0xA0	Trip-Zone Output B Control Register	0x00000000
TZIF	0xA4	Trip-Zone Interrupt Flag Register	0x00000000
TZIC	0xA8	Trip-Zone Clear Register	0x00000000
TZIE *	0xAC	Trip-Zone Interrupt Enable Register	0x00000000
TZFRC *	0xB0	Trip-Zone Force Register	0x00000000
DCALTRIPSEL *	0xB4	Digital Compare AL Trip Select Register	0x00000000
DCAHTRIPSEL *	0xB8	Digital Compare AH Trip Select Register	0x00000000
DCBLTRIPSEL *	0xBC	Digital Compare BL Trip Select Register	0x00000000
DCBHTRIPSEL *	0xC0	Digital Compare BH Trip Select Register	0x00000000
DCACTL *	0xC4	Digital Compare A Control Register	0x00000000
DCBCTL *	0xC8	Digital Compare B Control Register	0x00000000
DCFCTL *	0xCC	Digital Compare Filter Register	0x00000010
DCFOFFSET *	0xD0	Digital Compare Filter Offset Register	0x00000000
DCFOFFSETCNT	0xD4	Digital Compare Filter Offset Counter Register	0x00000000
DCFWINDOW *	0xD8	Digital Compare Filter Window Register	0x00000000
DCFWINDOWCNT	0xDC	Digital Compare Filter Window Counter Register	0x00000000
DCCAPCTL	0xE0	Digital Compare Capture Control Register	0x00000000
DCCAP	0xE4	Digital Compare Counter Capture Register	0x00000000
ETCTL *	0xE8	Event-Trigger Control Register	0x1E1E1E1E
ETPS *	0xEC	Event-Trigger Prescale Register	0x00000000
ETFLG	0xF0	Event-Trigger Flag Register	0x00000000
ETCLR	0xF4	Event-Trigger Clear Register	0x00000000
ETFRC *	0xF8	Event-Trigger Force Register	0x00000000
PWMREGKEY	0xFC	PWM Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the PWMREGKEY=0x1ACCE551.

8.11.2 PWM registers

Table 8-8: Shadow Status Register (SHADOWSTS) Layout

SHADOWSTS (Shadow Status Register) Offset: 0x0 Default: 0x00000000 Access: PWMx->SHADOWSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					DBFED	DBRED	DBCTL
7	6	5	4	3	2	1	0
AQCSFRC	AQCTLB	AQCTLA	CMPD	CMPC	CMPB	CMPA	TBPRD

Table 8-9: Shadow Status Register (SHADOWSTS) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10	DBFED	RO	0x0	DBFED shadow status 0: A load from DBFED to DBFEDA clears this bit 1: Write to DBFED set this bit and overwrite current value in DBFED
9	DBRED	RO	0x0	DBRED shadow status 0: A load from DBRED to DBREDA clears this bit 1: Write to DBRED set this bit and overwrite current value in DBRED
8	DBCTL	RO	0x0	DBCTL shadow status 0: A load from DBCTL to DBCTLA clears this bit 1: Write to DBCTL set this bit and overwrite current value in DBCTL
7	AQCSFRC	RO	0x0	AQCSFRC shadow status 0: A load from AQCSFRC to AQCSFRCA clears this bit 1: Write to AQCSFRC set this bit and overwrite current value in AQCSFRC
6	AQCTLB	RO	0x0	AQCTLB shadow status 0: A load from AQCTLB to AQCTLBA clears this bit 1: Write to AQCTLB set this bit and overwrite current value in AQCTLB
5	AQCTLA	RO	0x0	AQCTLA shadow status 0: A load from AQCTLA to AQCTLAA clears this bit

Bits	Field Name	Type	Reset	Description
				1: Write to AQCTLA set this bit and overwrite current value in AQCTLA
4	CMPD	RO	0x0	CMPD shadow status 0: A load from CMPD to CMPDA clears this bit 1: Write to CMPD set this bit and overwrite current value in CMPD
3	CMPC	RO	0x0	CMPC shadow status 0: A load from CMPC to CMPCA clears this bit 1: Write to CMPC set this bit and overwrite current value in CMPC
2	CMPB	RO	0x0	CMPB shadow status 0: A load from CMPB to CMPBA clears this bit 1: Write to CMPB set this bit and overwrite current value in CMPB
1	CMPA	RO	0x0	CMPA shadow status 0: A load from CMPA to CMPAA clears this bit 1: Write to CMPA set this bit and overwrite current value in CMPA
0	TBPRD	RO	0x0	TBPRD shadow status 0: A load from TBPRD to TBPRDA clears this bit 1: Write to TBPRD set this bit and overwrite current value in TBPRD

Table 8-10: Global Shadow to Active Load Control Register 0 (GLDCTL0) Layout

GLDCTL0 (Global Shadow to Active Load Control Register 0) Offset: 0x4 Default: 0x00000000 Access: PWMx->GLDCTL0							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			OSTMODE	GLDCNT			
7	6	5	4	3	2	1	0
GLDPRD				GLDLOAD			GLDEN

Table 8-11: Global Shadow to Active Load Control Register 0 (GLDCTL0) Description

Bits	Field Name	Type	Reset	Description
31:13	RESERVED_31_13	RO	0x0	Reserved.
12	OSTMODE	RW	0x0	One-shot global load mode 0: Global strobe is continuously controlled by GLDPRD and GLDCNT 1: Global strobe is blocked until GLDCTL1[OSTLOAD] is written with 1
11:8	GLDCNT	RO	0x0	Global shadow to active load strobe counter It indicates how many selected global load events have occurred
7:4	GLDPRD	RW	0x0	Global shadow to active load strobe period No global load pulse will be generated if GLDPRD=0. Otherwise, the global load pulse is generated when new event comes upon GLDCNT=GLDPRD-1. In other words, the load pulse is generated every GLDPRD events. It has no effect if OSTMODE=1
3:1	GLDLOAD	RW	0x0	Global shadow to active load mode 000: Load from shadow to active on TBCNT=0 001: Load from shadow to active on TBCNT=TBPRD 010: Load from shadow to active on TBCNT=0 or TBCNT=TBPRD 011: Load from shadow to active on SYNC event 100: Load from shadow to active on SYNC event or TBCNT=0 101: Load from shadow to active on SYNC event or TBCNT=TBPRD 110: Load from shadow to active on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Load from shadow to active on software by writing 1 to GLDCTL1[FRCLOAD]
0	GLDEN	RW	0x0	Global shadow to active load enable 0: Shadow to active load is defined per register regardless of GLDSEL. i.e. CMPA load is always controlled by CMPCTL[CMPALOAD] 1: Shadow to active load is globally controlled

Table 8-12: Global Shadow to Active Load Control Register 1 (GLDCTL1) Layout

GLDCTL1 (Global Shadow to Active Load Control Register 1) Offset: 0x8 Default: 0x00000000 Access: PWMx->GLDCTL1							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						FRCLOAD	OSTLOAD

Table 8-13: Global Shadow to Active Load Control Register 1 (GLDCTL1) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	FRCLOAD	W1S	0x0	Global force shadow to active load 0: Write 0 has no effect and always reads back 0 1: Write 1 generates global load event to the event counter
0	OSTLOAD	W1S	0x0	Start the one-shot load pending 0: Write 0 has no effect. Read 0 indicates there is no pending one-shot load 1: Write 1 initialize a one-shot load pending. Upon event defined by GLDCTL[GLDLOAD], the global load happens and this bit is self-cleared to 0.

Table 8-14: Global Shadow to Active Load Select Register (GLDSEL) Layout

GLDSEL (Global Shadow to Active Load Select Register) Offset: 0xC Default: 0x00000000 Access: PWMx->GLDSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					DBFED	DBRED	DBCTL
7	6	5	4	3	2	1	0
AQCSFRC	AQCTLB	AQCTLA	CMPD	CMPC	CMPB	CMPA	TBPRD

Table 8-15: Global Shadow to Active Load Select Register (GLDSEL) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10	DBFED	RW	0x0	DBFED global shadow to active load select 0: Shadow to active load is controlled by DBCTL[DBFEDLOAD] 1: Shadow to active load is controlled by GLDCTL
9	DBRED	RW	0x0	DBRED global shadow to active load select 0: Shadow to active load is controlled by DBCTL[DBREDLOAD] 1: Shadow to active load is controlled by GLDCTL
8	DBCTL	RW	0x0	DBCTL global shadow to active load select 0: Shadow to active load is controlled by DBCTL[DBCTLLOAD] 1: Shadow to active load is controlled by GLDCTL
7	AQCSFRC	RW	0x0	AQCSFRC global shadow to active load select 0: Shadow to active load is controlled by AQCTL[AQCSFRCLOAD] 1: Shadow to active load is controlled by GLDCTL
6	AQCTLB	RW	0x0	AQCTLB global shadow to active load select 0: Shadow to active load is controlled by AQCTL[AQCTLBLOAD] 1: Shadow to active load is controlled by GLDCTL
5	AQCTLA	RW	0x0	AQCTLA global shadow to active load select 0: Shadow to active load is controlled by AQCTL[AQCTLALOAD] 1: Shadow to active load is controlled by GLDCTL
4	CMPD	RW	0x0	CMPD global shadow to active load select 0: Shadow to active load is controlled by CMPCTL[CMPDLOAD] 1: Shadow to active load is controlled by GLDCTL
3	CMPC	RW	0x0	CMPC global shadow to active load select 0: Shadow to active load is controlled by CMPCTL[CMPCLOAD] 1: Shadow to active load is controlled by GLDCTL
2	CMPB	RW	0x0	CMPB global shadow to active load select 0: Shadow to active load is controlled by CMPCTL[CMPBLOAD] 1: Shadow to active load is controlled by GLDCTL
1	CMPA	RW	0x0	CMPA global shadow to active load select 0: Shadow to active load is controlled by CMPCTL[CMPALOAD] 1: Shadow to active load is controlled by GLDCTL

Bits	Field Name	Type	Reset	Description
0	TBPRD	RW	0x0	TBPRD global shadow to active load select 0: Shadow to active load is controlled by TBCTL[TBPRDLOAD] 1: Shadow to active load is controlled by GLDCTL

Table 8-16: PWM Link Control Register (PWMLINK) Layout

PWMLINK (PWM Link Control Register) Offset: 0x10 Default: 0xFFFFFFFF Access: PWMx->PWMLINK							
31	30	29	28	27	26	25	24
GLDCTL1				DBFED			
23	22	21	20	19	18	17	16
DBRED				CMPD			
15	14	13	12	11	10	9	8
CMPC				CMPB			
7	6	5	4	3	2	1	0
CMPA				TBPRD			

Table 8-17: PWM Link Control Register (PWMLINK) Description

Bits	Field Name	Type	Reset	Description
31:28	GLDCTL1	RW	0xF	Write to the GLDCTL1 of PWM selected by PWMLINK[GLDCTL1] results in simultaneous write to GLDCTL1 of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected
27:24	DBFED	RW	0xF	Write to the DBFED of PWM selected by PWMLINK[DBFED] results in simultaneous write to DBFED of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected
23:20	DBRED	RW	0xF	Write to the DBRED of PWM selected by PWMLINK[DBRED] results in simultaneous write to DBRED of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected

Bits	Field Name	Type	Reset	Description
19:16	CMPD	RW	0xF	Write to the CMPD of PWM selected by PWMLINK[CMPD] results in simultaneous write to CMPD of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected
15:12	CMPC	RW	0xF	Write to the CMPC of PWM selected by PWMLINK[CMPC] results in simultaneous write to CMPC of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected
11:8	CMPB	RW	0xF	Write to the CMPB of PWM selected by PWMLINK[CMPB] results in simultaneous write to CMPB of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected
7:4	CMPA	RW	0xF	Write to the CMPA of PWM selected by PWMLINK[CMPA] results in simultaneous write to CMPA of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected
3:0	TBPRD	RW	0xF	Write to the TBPRD of PWM selected by PWMLINK[TBPRD] results in simultaneous write to TBPRD of current PWM module 0000: Select PWM0 0001: Select PWM1 0010: Select PWM2 0011: Select PWM3 1111: No PWM selected

Table 8-18: Time-Base Control Register (TBCTL) Layout

TBCTL (Time-Base Control Register) Offset: 0x14 Default: 0x00000006 Access: PWMx->TBCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				DBGRUN		TBPDIV	
15	14	13	12	11	10	9	8
TBPDIV	TBDIV			FRCSYNC	SYNCOSEL		
7	6	5	4	3	2	1	0
TBPRDLOCK	TBPRDLOAD		PHSDIR	PHSEN	CNTMODE		CNTRUN

Table 8-19: Time-Base Control Register (TBCTL) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:18	DBGRUN	RW	0x0	PWM counter behavior upon CPU lockup or halted condition Note: Such condition can be caused by debug operation and CPU fault exceptions 00: Stop after the next time-base counter increment or decrement 01: Stop when TBCNT completes a whole cycle (TBCNT=TBPRD for up-count mode and TBCNT=0 for other modes) 10: Counter keep running when CPU is halted
17:15	TBPDIV	RW	0x0	TBCLK pre-dividing ratio from PWM clock input. Overall dividing ratio is $2^{TBPDIV} * (TBDIV+1)$ 000: /1 (default on reset) 001: /2 010: /4 011: /8 100: /16 101: /32 110: /64 111: /128
14:12	TBDIV	RW	0x0	TBCLK dividing ratio from pre-divider output. Overall dividing ratio is $2^{TBPDIV} * (TBDIV+1)$ 000: /1 (default on reset) 001: /2 010: /3 011: /4 100: /5 101: /6 110: /7 111: /8

Bits	Field Name	Type	Reset	Description
11	FRCSYNC	W1S	0x0	Software forced synchronization This event is ORed with SYNCI input and is valid only when SYNCSEL=000 0: Write 0 has no effect and always reads back 0 1: Write 1 forces a one-time synchronization pulse. This bit is self-cleared to 0 and not affected by PWMREGKEY protection
10:8	SYNCSEL	RW	0x0	Synchronization output select 000: SYNCI and the software forced sync 001: TBCNT=0 event 010: TBCNT=TBPRD event 011: TBCNT=CMPC event 100: TBCNT=CMPC event 101: TBCNT=CMPC event 110: TBCNT=CMPC event 111: Disable SYNCO
7	TBPRDLOCK	RW	0x0	TBPRD active value lock 0: Active value in TBPRDA is updated according to TBCTL[TBPRDLOAD] 1: Active value in TBPRDA will not change
6:5	TBPRDLOAD	RW	0x0	TBPRD active value load mode It has no effect when TBCTL[TBPRDLOCK]=1 00: Load TBPRD into TBPRDA on TBCNT=0 01: Load TBPRD into TBPRDA on SYNC event 10: Load TBPRD into TBPRDA on SYNC event or TBCNT=0 11: Write TBPRD will immediately change the active value in TBPRDA
4	PHSDIR	RW	0x0	Phase direction It is valid only when TBCNT is in up-down-count mode (CNTMODE=10) 0: Count down after synchronization event 1: Count up after synchronization event
3	PHSEN	RW	0x0	Phase register enable 0: Do not load time-base counter (TBCNT) from time-base phase register (TBPHS) 1: Load TBCNT from TBPHS when synchronization event occurs (SYNCI input, software synchronization or digital compare sync event)
2:1	CNTMODE	RW	0x3	Counter mode 00: Down-count 01: Up-count 10: Up-down-count 11: Stop and freeze (default on reset)

Bits	Field Name	Type	Reset	Description
0	CNTRUN	RW	0x0	Counter run/stop control 0: Counter stop 1: Counter free-running

Table 8-20: Time-Base Period Register (TBPRD) Layout

TBPRD (Time-Base Period Register) Offset: 0x18 Default: 0x00000000 Access: PWMx->TBPRD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-21: Time-Base Period Register (TBPRD) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Time-base counter period

Table 8-22: Time-Base Period Active Register (TBPRDA) Layout

TBPRDA (Time-Base Period Active Register) Offset: 0x1C Default: 0x00000000 Access: PWMx->TBPRDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-23: Time-Base Period Active Register (TBPRDA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Time-base counter active period When TBCTL[PRDLOAD]=0x3, write to TBPRD directly affects this register. Otherwise, the value is loaded from TBPRD upon the events defined in TBCTL[PRDLOAD].

Table 8-24: Time-Base Phase Register (TBPHS) Layout

TBPHS (Time-Base Phase Register) Offset: 0x20 Default: 0x00000000 Access: PWMx->TBPHS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-25: Time-Base Phase Register (TBPHS) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Time-base counter phase when synchronization event occurs TBCTL[PHSEN]=0: The synchronization event is ignored and TBCNT keeps its value. TBCTL[PHSEN]=1: Time-base counter (TBCNT) will be loaded with TBPHS when synchronization event occurs.

Table 8-26: Time-Base Counter Register (TBCNT) Layout

TBCNT (Time-Base Counter Register) Offset: 0x24 Default: 0x00000000 Access: PWMx->TBCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-27: Time-Base Counter Register (TBCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Current value of the time-base counter

Table 8-28: Time-Base Status Register (TBFLG) Layout

TBFLG (Time-Base Status Register) Offset: 0x28 Default: 0x00000000 Access: PWMx->TBFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							CNTDIR
7	6	5	4	3	2	1	0
CNTCMPD	CNTCMPC	CNTCMPB	CNTCMPA	CNTPRD	CNTZRO	CNTMAX	SYNCI

Table 8-29: Time-Base Status Register (TBFLG) Description

Bits	Field Name	Type	Reset	Description
31:9	RESERVED_31_9	RO	0x0	Reserved.
8	CNTDIR	RO	0x0	Time-base counter direction status 0: Time-base counter is currently counting down 1: Time-base counter is currently counting up
7	CNTCMPD	RO	0x0	Time-base counter reaching CMPD latched status 0: TBCNT never reaches CMPD 1: TBCNT reached CMPD
6	CNTCMPC	RO	0x0	Time-base counter reaching CMPC latched status 0: TBCNT never reaches CMPC 1: TBCNT reached CMPC
5	CNTCMPB	RO	0x0	Time-base counter reaching CMPB latched status 0: TBCNT never reaches CMPB 1: TBCNT reached CMPB
4	CNTCMPA	RO	0x0	Time-base counter reaching CMPA latched status 0: TBCNT never reaches CMPA 1: TBCNT reached CMPA
3	CNTPRD	RO	0x0	Time-base counter reaching period latched status 0: TBCNT never reaches TBPRD 1: TBCNT reached TBPRD
2	CNTZRO	RO	0x0	Time-base counter reaching zero latched status 0: TBCNT never reaches 0x0 1: TBCNT reached 0x0
1	CNTMAX	RO	0x0	Time-base counter reaching max latched status 0: TBCNT never reaches 0xFFFF 1: TBCNT reached 0xFFFF
0	SYNCI	RO	0x0	Input synchronization latched status 0: No input synchronization event has occurred 1: An input synchronization event has occurred

Table 8-30: Time-Base Status Clear Register (TBFLGCLR) Layout

TBFLGCLR (Time-Base Status Clear Register) Offset: 0x2C Default: 0x00000000 Access: PWMx->TBFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CNTCMPD	CNTCMPC	CNTCMPB	CNTCMPA	CNTPRD	CNTZRO	CNTMAX	SYNCI

Table 8-31: Time-Base Status Clear Register (TBFLGCLR) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CNTCMPD	W1C	0x0	Time-base counter reaching CMPD latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTCMPD. This bit is self-cleared to 0
6	CNTCMPC	W1C	0x0	Time-base counter reaching CMPC latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTCMPC. This bit is self-cleared to 0
5	CNTCMPB	W1C	0x0	Time-base counter reaching CMPB latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTCMPB. This bit is self-cleared to 0
4	CNTCMPA	W1C	0x0	Time-base counter reaching CMPA latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTCMPA. This bit is self-cleared to 0
3	CNTPRD	W1C	0x0	Time-base counter reaching period latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTPRD. This bit is self-cleared to 0

Bits	Field Name	Type	Reset	Description
2	CNTZRO	W1C	0x0	Time-base counter reaching zero latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTZRO. This bit is self-cleared to 0
1	CNTMAX	W1C	0x0	Time-base counter reaching max latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.CNTMAX. This bit is self-cleared to 0
0	SYNCI	W1C	0x0	Input synchronization latched status clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the TZFLG.SYNCI. This bit is self-cleared to 0

Table 8-32: Counter-Compare Control Register (CMPCTL) Layout

CMPCTL (Counter-Compare Control Register) Offset: 0x30 Default: 0x00000000 Access: PWMx->CMPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CMPDLOCK	14	13 CMPDLOAD	12	11 CMPDCLOCK	10	9 CMPCLOAD	8
7 CMPBLOCK	6	5 CMPBLOAD	4	3 CMPALOCK	2	1 CMPALOAD	0

Table 8-33: Counter-Compare Control Register (CMPCTL) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CMPDLOCK	RW	0x0	CMPD active value lock 0: Active value in CMPDA is updated according to CMPCTL.CMPDLOAD 1: Active value in CMPDA will not change
14:12	CMPDLOAD	RW	0x0	CMPD active value load mode It has no effect when CMPCTL[CMPDLOCK]=1 000: Load CMPD into CMPDA on TBCNT=0 001: Load CMPD into CMPDA on TBCNT=TBPRD 010: Load CMPD into CMPDA on TBCNT=0 or TBCNT=TBPRD 011: Load CMPD into CMPDA on SYNC event 100: Load CMPD into CMPDA on SYNC event or

Bits	Field Name	Type	Reset	Description
				TBCNT=0 101: Load CMPD into CMPDA on SYNC event or TBCNT=TBPRD 110: Load CMPD into CMPDA on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Write CMPD will immediately change the active value in CMPDA
11	CMPCLOCK	RW	0x0	CMPC active value lock 0: Active value in CMPCA is updated according to CMPCTL.CMPCLOAD 1: Active value in CMPCA will not change
10:8	CMPCLOAD	RW	0x0	CMPC active value load mode It has no effect when CMPCTL[CMPCLOCK]=1 000: Load CMPC into CMPCA on TBCNT=0 001: Load CMPC into CMPCA on TBCNT=TBPRD 010: Load CMPC into CMPCA on TBCNT=0 or TBCNT=TBPRD 011: Load CMPC into CMPCA on SYNC event 100: Load CMPC into CMPCA on SYNC event or TBCNT=0 101: Load CMPC into CMPCA on SYNC event or TBCNT=TBPRD 110: Load CMPC into CMPCA on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Write CMPC will immediately change the active value in CMPCA
7	CMPBLOCK	RW	0x0	CMPB active value lock 0: Active value in CMPBA is updated according to CMPCTL.CMPBLOAD 1: Active value in CMPBA will not change
6:4	CMPBLOAD	RW	0x0	CMPB active value load mode It has no effect when CMPCTL[CMPBLOCK]=1 000: Load CMPB into CMPBA on TBCNT=0 001: Load CMPB into CMPBA on TBCNT=TBPRD 010: Load CMPB into CMPBA on TBCNT=0 or TBCNT=TBPRD 011: Load CMPB into CMPBA on SYNC event 100: Load CMPB into CMPBA on SYNC event or TBCNT=0 101: Load CMPB into CMPBA on SYNC event or TBCNT=TBPRD 110: Load CMPB into CMPBA on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Write CMPB will immediately change the active value in CMPBA

Bits	Field Name	Type	Reset	Description
3	CMPALOCK	RW	0x0	CMPA active value lock 0: Active value in CMPAA is updated according to CMPATL.CMPALOAD 1: Active value in CMPAA will not change
2:0	CMPALOAD	RW	0x0	CMPA active value load mode It has no effect when CMPATL[CMPALOCK]=1 000: Load CMPA into CMPAA on TBCNT=0 001: Load CMPA into CMPAA on TBCNT=TBPRD 010: Load CMPA into CMPAA on TBCNT=0 or TBCNT=TBPRD 011: Load CMPA into CMPAA on SYNC event 100: Load CMPA into CMPAA on SYNC event or TBCNT=0 101: Load CMPA into CMPAA on SYNC event or TBCNT=TBPRD 110: Load CMPA into CMPAA on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Write CMPA will immediately change the active value in CMPAA

Table 8-34: Counter-Compare A Threshold Register (CMPA) Layout

CMPA (Counter-Compare A Threshold Register) Offset: 0x34 Default: 0x00000000 Access: PWMx->CMPA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-35: Counter-Compare A Threshold Register (CMPA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Comparator A threshold

Table 8-36: Counter-Compare A Threshold Active Register (CMPAA) Layout

CMPAA (Counter-Compare A Threshold Active Register) Offset: 0x38 Default: 0x00000000 Access: PWMx->CMPAA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-37: Counter-Compare A Threshold Active Register (CMPAA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Comparator A threshold active value A TBCNT=CMPA event will be generated while TBCNT is equal to the threshold value

Table 8-38: Counter-Compare B Threshold Register (CMPB) Layout

CMPB (Counter-Compare B Threshold Register) Offset: 0x3C Default: 0x00000000 Access: PWMx->CMPB							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-39: Counter-Compare B Threshold Register (CMPB) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Comparator B threshold

Table 8-40: Counter-Compare B Threshold Active Register (CMPBA) Layout

CMPBA (Counter-Compare B Threshold Active Register) Offset: 0x40 Default: 0x00000000 Access: PWMx->CMPBA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-41: Counter-Compare B Threshold Active Register (CMPBA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Comparator B threshold active value A TBCNT=CMPB event will be generated while TBCNT is equal to the threshold value

Table 8-42: Counter-Compare C Threshold Register (CMPC) Layout

CMPC (Counter-Compare C Threshold Register) Offset: 0x44 Default: 0x00000000 Access: PWMx->CMPC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-43: Counter-Compare C Threshold Register (CMPC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Comparator C threshold

Table 8-44: Counter-Compare C Threshold Active Register (CMPCA) Layout

CMPCA (Counter-Compare C Threshold Active Register) Offset: 0x48 Default: 0x00000000 Access: PWMx->CMPCA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-45: Counter-Compare C Threshold Active Register (CMPCA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Comparator C threshold active value A TBCNT=CMPC event will be generated while TBCNT is equal to the threshold value

Table 8-46: Counter-Compare D Threshold Register (CMPD) Layout

CMPD (Counter-Compare D Threshold Register) Offset: 0x4C Default: 0x00000000 Access: PWMx->CMPD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-47: Counter-Compare D Threshold Register (CMPD) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Comparator D threshold

Table 8-48: Counter-Compare D Threshold Active Register (CMPDA) Layout

CMPDA (Counter-Compare D Threshold Active Register) Offset: 0x50 Default: 0x00000000 Access: PWMx->CMPDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-49: Counter-Compare D Threshold Active Register (CMPDA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Comparator D threshold active value A TBCNT=CMPPD event will be generated while TBCNT is equal to the threshold value

Table 8-50: Action-Qualifier Control Register (AQCTL) Layout

AQCTL (Action-Qualifier Control Register) Offset: 0x54 Default: 0x00000000 Access: PWMx->AQCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
T1SEL				TOSEL			
7	6	5	4	3	2	1	0
AQCTLBLOCK	AQCTLBLOAD			AQCTLALOCK	AQCTLALOAD		

Table 8-51: Action-Qualifier Control Register (AQCTL) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:12	T1SEL	RW	0x0	T1 event source select 0000: DCAEVT0 0001: DCAEVT1 0010: DCBEVT0 0011: DCBEVT1 0100: TZ0 0101: TZ1 0110: TZ2 0111: TZ3 1000: TZ4 1001: SYNCI
11:8	T0SEL	RW	0x0	T0 event source select 0000: DCAEVT0 0001: DCAEVT1 0010: DCBEVT0 0011: DCBEVT1 0100: TZ0 0101: TZ1 0110: TZ2 0111: TZ3 1000: TZ4 1001: SYNCI
7	AQCTLBLOCK	RW	0x0	AQCTLB active value lock 0: Active value in AQCTLBA is updated according to AQCTL.AQCTLBLOAD 1: Active value in AQCTLBA will not change
6:4	AQCTLBLOAD	RW	0x0	AQCTLB active value load mode It has no effect when AQCTL[AQCTLBLOCK]=1 000: Load AQCTLB into AQCTLBA on TBCNT=0 001: Load AQCTLB into AQCTLBA on TBCNT=TBPRD 010: Load AQCTLB into AQCTLBA on TBCNT=0 or TBCNT=TBPRD 011: Load AQCTLB into AQCTLBA on SYNC event 100: Load AQCTLB into AQCTLBA on SYNC event or TBCNT=0 101: Load AQCTLB into AQCTLBA on SYNC event or TBCNT=TBPRD 110: Load AQCTLB into AQCTLBA on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Write AQCTLB immediately change the active value in AQCTLBA

Bits	Field Name	Type	Reset	Description
3	AQCTLALOCK	RW	0x0	AQCTLA active value lock 0: Active value in AQCTLAA is updated according to AQCTL[AQCTLALOAD] 1: Active value in AQCTLAA will not change
2:0	AQCTLALOAD	RW	0x0	AQCTLA active value load mode It has no effect when AQCTL[AQCTLALOCK]=1 000: Load AQCTLA into AQCTLAA on TBCNT=0 001: Load AQCTLA into AQCTLAA on TBCNT=TBPRD 010: Load AQCTLA into AQCTLAA on TBCNT=0 or TBCNT=TBPRD 011: Load AQCTLA into AQCTLAA on SYNC event 100: Load AQCTLA into AQCTLAA on SYNC event or TBCNT=0 101: Load AQCTLA into AQCTLAA on SYNC event or TBCNT=TBPRD 110: Load AQCTLA into AQCTLAA on SYNC event, TBCNT=0 or TBCNT=TBPRD 111: Write AQCTLA immediately change the active value in AQCTLAA

Table 8-52: Action-Qualifier Output A Control Register (AQCTLA) Layout

AQCTLA (Action-Qualifier Output A Control Register) Offset: 0x58 Default: 0x00000000 Access: PWMx->AQCTLA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
TOD		TOU		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

Table 8-53: Action-Qualifier Output A Control Register (AQCTLA) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:18	T1D	RW	0x0	Action on output A when T1 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)

Bits	Field Name	Type	Reset	Description
17:16	T1U	RW	0x0	Action on output A when T1 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
15:14	T0D	RW	0x0	Action on output A when T0 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
13:12	T0U	RW	0x0	Action on output A when T0 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
11:10	CBD	RW	0x0	Action on output A when TBCNT=CM PB and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
9:8	CBU	RW	0x0	Action on output A when TBCNT=CM PB and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
7:6	CAD	RW	0x0	Action on output A when TBCNT=CM PA and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
5:4	CAU	RW	0x0	Action on output A when TBCNT=CM PA and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)

Bits	Field Name	Type	Reset	Description
3:2	PRD	RW	0x0	Action on output A when TBCNT=TBPRD 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
1:0	ZRO	RW	0x0	Action on output A when TBCNT=Zero 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)

Table 8-54: Action-Qualifier Output A Control Active Register (AQCTLAA) Layout

AQCTLAA (Action-Qualifier Output A Control Active Register) Offset: 0x5C Default: 0x00000000 Access: PWMx->AQCTLAA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
TOD		TOU		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

Table 8-55: Action-Qualifier Output A Control Active Register (AQCTLAA) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:18	T1D	RO	0x0	Action on output A when T1 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
17:16	T1U	RO	0x0	Action on output A when T1 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
15:14	TOD	RO	0x0	Action on output A when T0 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low)

Bits	Field Name	Type	Reset	Description
				10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
13:12	T0U	RO	0x0	Action on output A when T0 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
11:10	CBD	RO	0x0	Action on output A when TBCNT=CMPB and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
9:8	CBU	RO	0x0	Action on output A when TBCNT=CMPB and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
7:6	CAD	RO	0x0	Action on output A when TBCNT=CMPA and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
5:4	CAU	RO	0x0	Action on output A when TBCNT=CMPA and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
3:2	PRD	RO	0x0	Action on output A when TBCNT=TBPRD 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)
1:0	ZRO	RO	0x0	Action on output A when TBCNT=Zero 00: Do nothing (action disabled) 01: Clear (force PWM output A to low) 10: Set (force PWM output A to high) 11: Toggle (invert current PWM output A status)

Table 8-56: Action-Qualifier Output B Control Register (AQCTLB) Layout

AQCTLB (Action-Qualifier Output B Control Register) Offset: 0x60 Default: 0x00000000 Access: PWMx->AQCTLB							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
TOD		TOU		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

Table 8-57: Action-Qualifier Output B Control Register (AQCTLB) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:18	T1D	RW	0x0	Action on output B when T1 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
17:16	T1U	RW	0x0	Action on output B when T1 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
15:14	TOD	RW	0x0	Action on output B when T0 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
13:12	TOU	RW	0x0	Action on output B when T0 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)

Bits	Field Name	Type	Reset	Description
11:10	CBD	RW	0x0	Action on output B when TBCNT=CMPA and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
9:8	CBU	RW	0x0	Action on output B when TBCNT=CMPB and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
7:6	CAD	RW	0x0	Action on output B when TBCNT=CMPA and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
5:4	CAU	RW	0x0	Action on output B when TBCNT=CMPA and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
3:2	PRD	RW	0x0	Action on output B when TBCNT=TBPRD 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
1:0	ZRO	RW	0x0	Action on output B when TBCNT=Zero 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)

Table 8-58: Action-Qualifier Output B Control Active Register (AQCTLBA) Layout

AQCTLBA (Action-Qualifier Output B Control Active Register) Offset: 0x64 Default: 0x00000000 Access: PWMx->AQCTLBA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				T1D		T1U	
15	14	13	12	11	10	9	8
TOD		TOU		CBD		CBU	
7	6	5	4	3	2	1	0
CAD		CAU		PRD		ZRO	

Table 8-59: Action-Qualifier Output B Control Active Register (AQCTLBA) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:18	T1D	RO	0x0	Action on output B when T1 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
17:16	T1U	RO	0x0	Action on output B when T1 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
15:14	TOD	RO	0x0	Action on output B when T0 event and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
13:12	TOU	RO	0x0	Action on output B when T0 event and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
11:10	CBD	RO	0x0	Action on output B when TBCNT=CMPT and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low)

Bits	Field Name	Type	Reset	Description
				10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
9:8	CBU	RO	0x0	Action on output B when TBCNT=CMPB and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
7:6	CAD	RO	0x0	Action on output B when TBCNT=CMPA and TBCNT is counting down 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
5:4	CAU	RO	0x0	Action on output B when TBCNT=CMPA and TBCNT is counting up 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
3:2	PRD	RO	0x0	Action on output B when TBCNT=TBPRD 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)
1:0	ZRO	RO	0x0	Action on output B when TBCNT=Zero 00: Do nothing (action disabled) 01: Clear (force PWM output B to low) 10: Set (force PWM output B to high) 11: Toggle (invert current PWM output B status)

Table 8-60: Action-Qualifier Software Force Register (AQSFRC) Layout

AQSFRC (Action-Qualifier Software Force Register) Offset: 0x68 Default: 0x00000000 Access: PWMx->AQSFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						OTSFRC	
7	6	5	4	3	2	1	0
RESERVED		CSFLOAD		ACTSFB		ACTSFA	

Table 8-61: Action-Qualifier Software Force Register (AQSFRC) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:8	OTSFRC	W1S	0x0	One-time software forced event on output 00: Write a 0 has no effect and always reads back 0 01: 10: 11: Write a 1 initialize a one-cycle force pulse. This bit is self-cleared to 0
7:6	RESERVED_7_6	RO	0x0	Reserved.
5:4	CSFLOAD	RW	0x0	Options to load continuous software force register (AQCSFRC) active value from shadow register 00: Load from AQCSFRC into AQCSFRCA on TBCNT=0 01: Load from AQCSFRC into AQCSFRCA on TBCNT=TBPRD 10: Load from AQCSFRC into AQCSFRCA on TBCNT=Zero or TBCNT=TBPRD 11: Write AQCSFRC immediately change the active value in AQCSFRCA
3:2	ACTSFB	RW	0x0	Action when one-time software force B is invoked 00: Do nothing (action disabled) 01: Clear (force PWM output B to low 10: Set (force PWM output B to high 11: Toggle (invert current PWM AQ output B status)
1:0	ACTSFA	RW	0x0	Action when one-time software force A is invoked 00: Do nothing (action disabled) 01: Clear (force PWM output A to low 10: Set (force PWM output A to high 11: Toggle (invert current PWM AQ output A status)

Table 8-62: Action-Qualifier Continuous Software Force Register (AQCSFRC) Layout

AQCSFRC (Action-Qualifier Continuous Software Force Register) Offset: 0x6C Default: 0x00000000 Access: PWMx->AQCSFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						CSFB	
7	6	5	4	3	2	1	0
RESERVED						CSFA	

Table 8-63: Action-Qualifier Continuous Software Force Register (AQCSFRC) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:8	CSFB	RW	0x0	Continuous software force on output B 00: Forcing disabled 01: Force a continuous low on output B 10: Force a continuous high on output B 11: Invalid option
7:2	RESERVED_7_2	RO	0x0	Reserved.
1:0	CSFA	RW	0x0	Continuous software force on output A 00: Forcing disabled 01: Force a continuous low on output A 10: Force a continuous high on output A 11: Invalid option

Table 8-64: Action-Qualifier Continuous Software Force Active Register (AQCSFRCA) Layout

AQCSFRCA (Action-Qualifier Continuous Software Force Active Register) Offset: 0x70 Default: 0x00000000 Access: PWMx->AQCSFRCA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						CSFB	
7	6	5	4	3	2	1	0
RESERVED						CSFA	

Table 8-65: Action-Qualifier Continuous Software Force Active Register (AQCSFRCA) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:8	CSFB	RO	0x0	Continuous software force on output B 00: Forcing disabled 01: Force a continuous low on output B 10: Force a continuous high on output B 11: Invalid option
7:2	RESERVED_7_2	RO	0x0	Reserved.
1:0	CSFA	RO	0x0	Continuous software force on output A 00: Forcing disabled 01: Force a continuous low on output A 10: Force a continuous high on output A 11: Invalid option

Table 8-66: Dead-Band Generator Control Register (DBCTL) Layout

DBCTL (Dead-Band Generator Control Register) Offset: 0x74 Default: 0x00000070 Access: PWMx->DBCTL							
31	30	29	28	27	26	25	24
RESERVED					DBFEDLOCK	DBFEDLOAD	
23	22	21	20	19	18	17	16
RESERVED	DBREDLOCK	DBREDLOAD		RESERVED	DBCTLLOCK	DBCTLLOAD	
15	14	13	12	11	10	9	8
RESERVED					CTLWE	HALFCYCLE	DUALEDGE
7	6	5	4	3	2	1	0
OUTBSRC	OUTASRC	FEDPOL	REDPOL	FEDSRC	REDSRC	FEDEN	REDEN

Table 8-67: Dead-Band Generator Control Register (DBCTL) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	DBFEDLOCK	RW	0x0	DBFED active value lock 0: Active value in DBFEDA is updated according to DBCTL[DBFEDLOAD] 1: Active value in DBFEDA will not change
25:24	DBFEDLOAD	RW	0x0	DBFED active value load mode It has no effect when DBCTL[DBFEDLOCK]=1 00: Load DBFED into DBFEDA on TBCNT=0 01: Load DBFED into DBFEDA on TBCNT=TBPRD 10: Load DBFED into DBFEDA on TBCNT=0 or TBCNT=TBPRD 11: Write DBFED immediately change the active value in DBFEDA
23	RESERVED_23	RO	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
22	DBREDLOCK	RW	0x0	DBRED active value lock 0: Active value in DBREDA is updated according to DBCTL[DBREDLOAD] 1: Active value in DBREDA will not change
21:20	DBREDLOAD	RW	0x0	DBRED active value load mode It has no effect when DBCTL[DBFEDLOCK]=1 00: Load DBRED into DBREDA on TBCNT=0 01: Load DBRED into DBREDA on TBCNT=TBPRD 10: Load DBRED into DBREDA on TBCNT=0 or TBCNT=TBPRD 11: Write DBRED will immediately change the active value in DBREDA
19	RESERVED_19	RO	0x0	Reserved.
18	DBCTLLOCK	RW	0x0	DBCTL[9:0] active value lock 0: Active value in DBCTLA is updated according to DBCTL[DBCTLLOAD] 1: Active value in DBCTLA will not change
17:16	DBCTLLOAD	RW	0x0	DBCTL[9:0] active value load mode It has no effect when DBCTL[DBCTLLOCK]=1 00: Load DBCTL[9:0] into DBCTLA on TBCNT=0 01: Load DBCTL[9:0] into DBCTLA on TBCNT=TBPRD 10: Load DBCTL[9:0] into DBCTLA on TBCNT=0 or TBCNT=TBPRD 11: Write DBCTL[9:0] will immediately change the active value in DBCTLA
15:11	RESERVED_15_11	RO	0x0	Reserved.
10	CTLWE	W1C	0x0	10 LSBs of the dead-band control write enable In order to write the 10 LSBs, this bit need to be write 1 simultaneously 0: 10 LSBs will not be written 1: 10 LSBs will be written
9	HALFCYCLE	RW	0x0	Half cycle clocking enable 0: Full cycle clocking enabled. The dead-band counter is clocked at TBCLK. 1: Half cycle clocking enabled. The dead-band counter is clocked at 2x TBCLK.
8	DUALEDGE	RW	0x0	Dual-edge delay mode 0: Rising-edge delay and falling-edge delay are two separate paths 1: Rising-edge delay path output serves as input for Falling-edge delay path
7	OUTBSRC	RW	0x0	Output B source 0: Falling-edge delay path is selected as output B 1: Rising-edge delay path is selected as output B

Bits	Field Name	Type	Reset	Description
6	OUTASRC	RW	0x1	Output A source 0: Falling-edge delay path is selected as output A 1: Rising-edge delay path is selected as output A
5	FEDPOL	RW	0x1	Falling-edge delay output polarity 0: Falling-edge delay output is inverted to have active-low polarity 1: Falling-edge delay output keeps the original active-high polarity
4	REDPOL	RW	0x1	Rising-edge delay output polarity 0: Rising-edge delay output is inverted to have active-low polarity 1: Rising-edge delay output keeps the original active-high polarity
3	FEDSRC	RW	0x0	Falling-edge delay source 0: Input A is the source for falling-edge delay 1: Input B is the source for falling-edge delay
2	REDSRC	RW	0x0	Rising-edge delay source 0: Input A is the source for rising-edge delay 1: Input B is the source for rising-edge delay
1	FEDEN	RW	0x0	Falling-edge delay enable 0: Falling-edge delay is bypassed and the input from action-qualifier goes to the submodule directly 1: Falling-edge delay is enabled and applied at output B
0	REDEN	RW	0x0	Rising-edge delay enable 0: Rising-edge delay is bypassed and the input from action-qualifier goes to the submodule directly 1: Rising-edge delay is enabled and applied at output A

Table 8-68: Dead-Band Generator Control Active Register (DBCTLA) Layout

DBCTLA (Dead-Band Generator Control Active Register) Offset: 0x78 Default: 0x00000070 Access: PWMx->DBCTLA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						HALFCYCLE	DUALEDGE
7	6	5	4	3	2	1	0
OUTBSRC	OUTASRC	FEDPOL	REDPOL	FEDSRC	REDSRC	FEDEN	REDEN

Table 8-69: Dead-Band Generator Control Active Register (DBCTLA) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	HALFCYCLE	RO	0x0	Half cycle clocking enable 0: Full cycle clocking enabled. The dead-band counter is clocked at TBCLK. 1: Half cycle clocking enabled. The dead-band counter is clocked at 2x TBCLK.
8	DUALEDGE	RO	0x0	Dual-edge delay mode 0: Rising-edge delay and falling-edge delay are two separate paths 1: Rising-edge delay path output serves as input for Falling-edge delay path
7	OUTBSRC	RO	0x0	Output B source 0: Falling-edge delay path is selected as output B 1: Rising-edge delay path is selected as output B
6	OUTASRC	RO	0x1	Output A source 0: Falling-edge delay path is selected as output A 1: Rising-edge delay path is selected as output A
5	FEDPOL	RO	0x1	Falling-edge delay output polarity 0: Falling-edge delay output is inverted to have active-low polarity 1: Falling-edge delay output keeps the original active-high polarity
4	REDPOL	RO	0x1	Rising-edge delay output polarity 0: Rising-edge delay output is inverted to have active-low polarity 1: Rising-edge delay output keeps the original active-high polarity
3	FEDSRC	RO	0x0	Falling-edge delay source 0: Input A is the source for falling-edge delay 1: Input B is the source for falling-edge delay
2	REDSRC	RO	0x0	Rising-edge delay source 0: Input A is the source for rising-edge delay 1: Input B is the source for rising-edge delay
1	FEDEN	RO	0x0	Falling-edge delay enable 0: Falling-edge delay is bypassed and the input from action-qualifier goes to the submodule directly 1: Falling-edge delay is enabled and applied at output B
0	REDEN	RO	0x0	Rising-edge delay enable 0: Rising-edge delay is bypassed and the input from action-qualifier goes to the submodule directly 1: Rising-edge delay is enabled and applied at output B

Bits	Field Name	Type	Reset	Description
				1: Rising-edge delay is enabled and applied at output A

Table 8-70: Dead-Band Generator Rising Edge Delay Register (DBRED) Layout

DBRED (Dead-Band Generator Rising Edge Delay Register)					Offset: 0x7C	Default: 0x00000000	
Access: PWMx->DBRED							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-71: Dead-Band Generator Rising Edge Delay Register (DBRED) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Rising-edge delay

Table 8-72: Dead-Band Generator Rising Edge Delay Active Register (DBREDA) Layout

DBREDA (Dead-Band Generator Rising Edge Delay Active Register)					Offset: 0x80	Default: 0x00000000	
Access: PWMx->DBREDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-73: Dead-Band Generator Rising Edge Delay Active Register (DBREDA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Rising-edge delay

Table 8-74: Dead-Band Generator Falling Edge Delay Register (DBFED) Layout

DBFED (Dead-Band Generator Falling Edge Delay Register) Offset: 0x84 Default: 0x00000000 Access: PWMx->DBFED							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-75: Dead-Band Generator Falling Edge Delay Register (DBFED) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Falling-edge delay

Table 8-76: Dead-Band Generator Falling Edge Delay Active Register (DBFEDA) Layout

DBFEDA (Dead-Band Generator Falling Edge Delay Active Register) Offset: 0x88 Default: 0x00000000 Access: PWMx->DBFEDA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-77: Dead-Band Generator Falling Edge Delay Active Register (DBFEDA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Falling-edge delay

Table 8-78: Trip-Zone Event Select Register (TZSEL) Layout

TZSEL (Trip-Zone Event Select Register) Offset: 0x8C Default: 0x00030003 Access: PWMx->TZSEL									
31 CBCWE	30	29	28	27	26 DBGCBC	25 DCBEVT1	24 DCAEVT1		
RESERVED									
23 CLKERRCBC	22 TZ4CBC	21 TZ3CBC	20 TZ2CBC	19 TZ1CBC	18 TZ0CBC	17	16 CBCOUT		
15 OSTWE	14	13	12	11	10 DBGOST	9 DCBEVT0	8 DCAEVT0		
RESERVED									
7 CLKERROST	6 TZ4OST	5 TZ3OST	4 TZ2OST	3 TZ1OST	2 TZ0OST	1	0 OSTOUT		

Table 8-79: Trip-Zone Event Select Register (TZSEL) Description

Bits	Field Name	Type	Reset	Description
31	CBCWE	W1S	0x0	Cycle-by-cycle trip-zone event select control bits write enable 0: Write 0 has no effect and always reads back 0 1: Write 1 enables concurrent write to 16 MSBs. This bit is self-cleared to 0
30:27	RESERVED_30_27	RO	0x0	Reserved.
26	DBGCBC	RW	0x0	CPU lockup or halted condition as cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
25	DCBEVT1	RW	0x0	Digital compare B event 1 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
24	DCAEVT1	RW	0x0	Digital compare A event 1 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
23	CLKERRCBC	RW	0x0	Clock error as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
22	TZ4CBC	RW	0x0	TZ4 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
21	TZ3CBC	RW	0x0	TZ3 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
20	TZ2CBC	RW	0x0	TZ2 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
19	TZ1CBC	RW	0x0	TZ1 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
18	TZ0CBC	RW	0x0	TZ0 as a cycle-by-cycle trip source (ORed) 0: Disable 1: Enable
17:16	CBCOUT	RW	0x3	Cycle-by-cycle trip-zone output select 00: Disable cycle-by-cycle trip-zone 01: Select only the asynchronous path. i.e. The original event OR gate output 10: Select only the latched path 11: Select both the asynchronous path and the latched path (ORed)
15	OSTWE	W1S	0x0	One-shot trip-zone event select control bits write enable 0: Write 0 has no effect and always reads back 0 1: Write 1 enable concurrent write to 16 LSBs. This bit is self-cleared to 0
14:11	RESERVED_14_11	RO	0x0	Reserved.
10	DBGOST	RW	0x0	CPU locked or halted condition as a one-shot trip source (ORed) 0: Disable 1: Enable
9	DCBEVT0	RW	0x0	Digital compare B event 0 as a one-shot trip source (ORed) 0: Disable 1: Enable
8	DCAEVT0	RW	0x0	Digital compare A event 0 as a one-shot trip source (ORed) 0: Disable 1: Enable
7	CLKERROST	RW	0x0	Clock error as a one-shot trip source (ORed) 0: Disable 1: Enable
6	TZ4OST	RW	0x0	TZ4 as a one-shot trip source (ORed) 0: Disable 1: Enable
5	TZ3OST	RW	0x0	TZ3 as a one-shot trip source (ORed) 0: Disable 1: Enable
4	TZ2OST	RW	0x0	TZ2 as a one-shot trip source (ORed) 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
3	TZ1OST	RW	0x0	TZ1 as a one-shot trip source (ORed) 0: Disable 1: Enable
2	TZ0OST	RW	0x0	TZ0 as a one-shot trip source (ORed) 0: Disable 1: Enable
1:0	OSTOUT	RW	0x3	One-shot trip-zone output select 00: Disable one-shot trip-zone 01: Select only the asynchronous path. i.e. The original event OR gate output 10: Select only the latched path 11: Select both the asynchronous path and the latched path (ORed)

Table 8-80: Trip-Zone Flag Register (TZFLG) Layout

TZFLG (Trip-Zone Flag Register) Offset: 0x90 Default: 0x00000000 Access: PWMx->TZFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DBGCBC	DCBEVT1
15	14	13	12	11	10	9	8
DCAEVT1	CLKERRCBC	TZ4CBC	TZ3CBC	TZ2CBC	TZ1CBC	TZ0CBC	DBGOST
7	6	5	4	3	2	1	0
DCBEVT0	DCAEVT0	CLKERROST	TZ4OST	TZ3OST	TZ2OST	TZ1OST	TZ0OST

Table 8-81: Trip-Zone Flag Register (TZFLG) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	DBGCBC	RO	0x0	Latched CPU lockup or halted cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
16	DCBEVT1	RO	0x0	Latched digital compare B event 1 cycle-by-cycle trip flag 0: No event has occurred 1: Event occurred
15	DCAEVT1	RO	0x0	Latched digital compare A event 1 cycle-by-cycle trip flag 0: No event has occurred 1: Event occurred

Bits	Field Name	Type	Reset	Description
14	CLKERRCBC	RO	0x0	Latched clock error cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
13	TZ4CBC	RO	0x0	Latched TZ4 cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
12	TZ3CBC	RO	0x0	Latched TZ3 cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
11	TZ2CBC	RO	0x0	Latched TZ2 cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
10	TZ1CBC	RO	0x0	Latched TZ1 cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
9	TZ0CBC	RO	0x0	Latched TZ0 cycle-by-cycle trip event flag 0: No event has occurred 1: Event occurred
8	DBGOST	RO	0x0	Latched CPU lockup or halted one-shot trip event flag 0: No event has occurred 1: Event occurred
7	DCBEVT0	RO	0x0	Latched digital compare B event 0 one-shot trip flag 0: No event has occurred 1: Event occurred
6	DCAEVT0	RO	0x0	Latched digital compare A event 0 one-shot trip flag 0: No event has occurred 1: Event occurred
5	CLKERROST	RO	0x0	Latched clock error one-shot trip event flag 0: No event has occurred 1: Event occurred
4	TZ4OST	RO	0x0	Latched TZ4 one-shot trip event flag 0: No event has occurred 1: Event occurred
3	TZ3OST	RO	0x0	Latched TZ3 one-shot trip event flag 0: No event has occurred 1: Event occurred
2	TZ2OST	RO	0x0	Latched TZ2 one-shot trip event flag 0: No event has occurred 1: Event occurred

Bits	Field Name	Type	Reset	Description
1	TZ1OST	RO	0x0	Latched TZ1 one-shot trip event flag 0: No event has occurred 1: Event occurred
0	TZ0OST	RO	0x0	Latched TZ0 one-shot trip event flag 0: No event has occurred 1: Event occurred

Table 8-82: Trip-Zone Flag Clear Register (TZFLGCLR) Layout

TZFLGCLR (Trip-Zone Flag Clear Register) Offset: 0x94 Default: 0x00000000 Access: PWMx->TZFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DBGCBC	DCBEVT1
15	14	13	12	11	10	9	8
DCAEVT1	CLKERRCBC	TZ4CBC	TZ3CBC	TZ2CBC	TZ1CBC	TZ0CBC	DBGOST
7	6	5	4	3	2	1	0
DCBEVT0	DCAEVT0	CLKERROST	TZ4OST	TZ3OST	TZ2OST	TZ1OST	TZ0OST

Table 8-83: Trip-Zone Flag Clear Register (TZFLGCLR) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	DBGCBC	W1C	0x0	Latched CPU lockup or halted cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
16	DCBEVT1	W1C	0x0	Latched digital compare B event 1 cycle-by-cycle trip flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
15	DCAEVT1	W1C	0x0	Latched digital compare A event 1 cycle-by-cycle trip flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
14	CLKERRCBC	W1C	0x0	Latched clock error cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0

Bits	Field Name	Type	Reset	Description
13	TZ4CBC	W1C	0x0	Latched TZ4 cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
12	TZ3CBC	W1C	0x0	Latched TZ3 cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
11	TZ2CBC	W1C	0x0	Latched TZ2 cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
10	TZ1CBC	W1C	0x0	Latched TZ1 cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
9	TZ0CBC	W1C	0x0	Latched TZ0 cycle-by-cycle trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
8	DBGOST	W1C	0x0	Latched CPU lockup or halted one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
7	DCBEVT0	W1C	0x0	Latched digital output B event 0 one-shot trip flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
6	DCAEVT0	W1C	0x0	Latched digital output A event 0 one-shot trip flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
5	CLKERROST	W1C	0x0	Latched clock error one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
4	TZ4OST	W1C	0x0	Latched TZ4 one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
3	TZ3OST	W1C	0x0	Latched TZ3 one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0

Bits	Field Name	Type	Reset	Description
				1: Write a 1 clears the latched status. This bit is self-cleared to 0
2	TZ2OST	W1C	0x0	Latched TZ2 one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
1	TZ1OST	W1C	0x0	Latched TZ1 one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0
0	TZ0OST	W1C	0x0	Latched TZ0 one-shot trip event flag clear 0: Write a 0 has no effect. Always reads back a 0 1: Write a 1 clears the latched status. This bit is self-cleared to 0

Table 8-84: Trip-Zone Digital Compare Event Select Register (TZDCSEL) Layout

TZDCSEL (Trip-Zone Digital Compare Event Select Register) Offset: 0x98 Default: 0x00000000 Access: PWMx->TZDCSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				RAWDCBEVT1			RAWDCBEVT0
7	6	5	4	3	2	1	0
RAWDCBEVT0		RAWDCAEVT1			RAWDCAEVT0		

Table 8-85: Trip-Zone Digital Compare Event Select Register (TZDCSEL) Description

Bits	Field Name	Type	Reset	Description
31:12	RESERVED_31_12	RO	0x0	Reserved.
11:9	RAWDCBEVT1	RW	0x0	Raw digital compare B event 1 option 000: Event disabled 001: DCBL=low, DCBH=don't care 010: DCBL=high, DCBH=don't care 011: DCBL=don't care, DCBH=low 100: DCBL=don't care, DCBH=high 101: DCBL=low, DCBH=high 110: DCBL=high, DCBH=low 111: DCBL=high, DCBH=high
8:6	RAWDCBEVT0	RW	0x0	Raw digital compare B event 0 option 000: Event disabled 001: DCBL=low, DCBH=don't care 010: DCBL=high, DCBH=don't care

Bits	Field Name	Type	Reset	Description
				011: DCBL=don't care, DCBH=low 100: DCBL=don't care, DCBH=high 101: DCBL=low, DCBH=high 110: DCBL=high, DCBH=low 111: DCBL=high, DCBH=high
5:3	RAWDCAEVT1	RW	0x0	Raw digital compare A event 1 option 000: Event disabled 001: DCAL=low, DCAH=don't care 010: DCAL=high, DCAH=don't care 011: DCAL=don't care, DCAH=low 100: DCAL=don't care, DCAH=high 101: DCAL=low, DCAH=high 110: DCAL=high, DCAH=low 111: DCAL=high, DCAH=high
2:0	RAWDCAEVT0	RW	0x0	Raw digital compare A event 0 option 000: Event disabled 001: DCAL=low, DCAH=don't care 010: DCAL=high, DCAH=don't care 011: DCAL=don't care, DCAH=low 100: DCAL=don't care, DCAH=high 101: DCAL=low, DCAH=high 110: DCAL=high, DCAH=low 111: DCAL=high, DCAH=high

Table 8-86: Trip-Zone Output A Control Register (TZACTL) Layout

TZACTL (Trip-Zone Output A Control Register) Offset: 0x9C Default: 0x00000000 Access: PWMx->TZACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DCAEVT1D	
15	14	13	12	11	10	9	8
DCAEVT1D		DCAEVT1U			DCAEVT0D		DCAEVT0U
7	6	5	4	3	2	1	0
DCAEVT0U		TZD			TZU		

Table 8-87: Trip-Zone Output A Control Register (TZACTL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17:15	DCAEVT1D	RW	0x0	Action on output A when a digital compare A event 1 occurs and TBCNT is counting down 000: Set output A to high-impedance 001: Force output A to low state

Bits	Field Name	Type	Reset	Description
				010: Force output A to high state 011: Force output A to invert current state (high->low, low->high) 111: Do nothing for output A
14:12	DCAEVT1U	RW	0x0	Action on output A when a digital compare A event 1 occurs and TBCNT is counting up 000: Set output A to high-impedance 001: Force output A to low state 010: Force output A to high state 011: Force output A to invert current state (high->low, low->high) 111: Do nothing for output A
11:9	DCAEVT0D	RW	0x0	Action on output A when a digital compare A event 0 occurs and TBCNT is counting down 000: Set output A to high-impedance 001: Force output A to low state 010: Force output A to high state 011: Force output A to invert current state (high->low, low->high) 111: Do nothing for output A
8:6	DCAEVT0U	RW	0x0	Action on output A when a digital compare A event 0 occurs and TBCNT is counting up 000: Set output A to high-impedance 001: Force output A to low state 010: Force output A to high state 011: Force output A to invert current state (high->low, low->high) 111: Do nothing for output A
5:3	TZD	RW	0x0	Action on output A when a trip event occurs and TBCNT is counting down 000: Set output A to high-impedance 001: Force output A to low state 010: Force output A to high state 011: Force output A to invert current state (high->low, low->high) 111: Do nothing for output A
2:0	TZU	RW	0x0	Action on output A when a trip event occurs and TBCNT is counting up 000: Set output A to high-impedance 001: Force output A to low state 010: Force output A to high state 011: Force output A to invert current state (high->low, low->high) 111: Do nothing for output A

Table 8-88: Trip-Zone Ouput B Control Register (TZBCTL) Layout

TZBCTL (Trip-Zone Ouput B Control Register) Offset: 0xA0 Default: 0x00000000 Access: PWMx->TZBCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						DCBEVT1D	
15	14	13	12	11	10	9	8
DCBEVT1D	DCBEVT1U			DCBEVT0D			DCBEVT0U
7	6	5	4	3	2	1	0
DCBEVT0U		TZD			TZU		

Table 8-89: Trip-Zone Ouput B Control Register (TZBCTL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17:15	DCBEVT1D	RW	0x0	Action on output B when a digital compare B event 1 occurs and TBCNT is counting down 000: Set output B to high-impedance 001: Force output B to high state 010: Force output B to low state 011: Force output B to invert current state (high->low, low->high) 111: Do nothing for output B
14:12	DCBEVT1U	RW	0x0	Action on output B when a digital compare B event 1 occurs and TBCNT is counting up 000: Set output B to high-impedance 001: Force output B to low state 010: Force output B to high state 011: Force output B to invert current state (high->low, low->high) 111: Do nothing for output B
11:9	DCBEVT0D	RW	0x0	Action on output B when a digital compare B event 0 occurs and TBCNT is counting down 000: Set output B to high-impedance 001: Force output B to low state 010: Force output B to high state 011: Force output B to invert current state (high->low, low->high) 111: Do nothing for output B
8:6	DCBEVT0U	RW	0x0	Action on output B when a digital compare B event 0 occurs and TBCNT is counting up 000: Set output B to high-impedance 001: Force output B to low state 010: Force output B to high state

Bits	Field Name	Type	Reset	Description
				011: Force output B to invert current state (high->low, low->high) 111: Do nothing for output B
5:3	TZD	RW	0x0	Action on output B when a trip event occurs and TBCNT is counting down 000: Set output B to high-impedance 001: Force output B to low state 010: Force output B to high state 011: Force output B to invert current state (high->low, low->high) 111: Do nothing for output B
2:0	TZU	RW	0x0	Action on output B when a trip event occurs and TBCNT is counting up 000: Set output B to high-impedance 001: Force output B to low state 010: Force output B to high state 011: Force output B to invert current state (high->low, low->high) 111: Do nothing for output B

Table 8-90: Trip-Zone Interrupt Flag Register (TZIF) Layout

TZIF (Trip-Zone Interrupt Flag Register) Offset: 0xA4 Default: 0x00000000 Access: PWMx->TZIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED		DCBEVT1RAW	DCBEVT0RAW	DCAEVT1RAW	DCAEVT0RAW	CBCRAW	OSTRAW
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	INT	DCBEVT1	DCBEVT0	DCAEVT1	DCAEVT0	CBC	OST

Table 8-91: Trip-Zone Interrupt Flag Register (TZIF) Description

Bits	Field Name	Type	Reset	Description
31:22	RESERVED_31_22	RO	0x0	Reserved.
21	DCBEVT1RAW	RO	0x0	Digital compare B event 1 Interrupt Raw flag 0: DCBEVT1 event does not occur 1: DCBEVT1 event occurred. It is cleared by writing 1 to TZIC.DCBEVT1.
20	DCBEVT0RAW	RO	0x0	Digital compare B event 0 Interrupt Raw flag 0: DCBEVT0 event does not occur 1: DCBEVT0 event occurred. It is cleared by writing 1 to TZIC.DCBEVT0.

Bits	Field Name	Type	Reset	Description
19	DCAEVT1RAW	RO	0x0	Digital compare A event 1 Interrupt Raw flag 0: DCAEVT1 event does not occur 1: DCAEVT1 event occurred. It is cleared by writing 1 to TZIC.DCAEVT1.
18	DCAEVT0RAW	RO	0x0	Digital compare A event 0 Interrupt Raw flag 0: DCAEVT0 event does not occur 1: DCAEVT0 event occurred. It is cleared by writing 1 to TZIC.DCAEVT0.
17	CBCRAW	RO	0x0	Cycle-by-cycle trip event Interrupt Raw flag 0: Cycle-by-cycle trip event does not occur 1: Cycle-by-cycle trip event occurred. It will remain set until it is manually cleared by writing 1 to TZCLR.CBC. If the cycle-by-cycle trip event is still present, it will be immediately set again.
16	OSTRAW	RO	0x0	One-shot trip event Interrupt Raw flag 0: One-shot trip event does not occur 1: One-shot trip event occurred. It is cleared by writing 1 to TZCLR.OST.
15:7	RESERVED_15_7	RO	0x0	Reserved.
6	INT	RO	0x0	Global trip interrupt flag 0: No trip-zone interrupt has been generated 1: Trip-zone interrupt was generated and issued to the CPU. No further trip-zone interrupt will be generated until this flag is cleared by writing 1 to TZIC.INT
5	DCBEVT1	RO	0x0	Digital compare B event 1 interrupt flag This bit is always 0 if TZIE.DCBEVT1=0 0: DCBEVT interrupt does not occur 1: DCBEVT1 interrupt occurred.
4	DCBEVT0	RO	0x0	Digital compare B event 0 interrupt flag This bit is always 0 if TZIE.DCAEVT1=0 0: DCBEVT0 interrupt does not occur 1: DCBEVT0 interrupt occurred.
3	DCAEVT1	RO	0x0	Digital compare A event 1 interrupt flag This bit is always 0 if TZIE.DCAEVT1=0 0: DCAEVT1 did not occur 1: DCAEVT1 has occurred.
2	DCAEVT0	RO	0x0	Digital compare A event 0 interrupt flag This bit is always 0 if TZIE.DCAEVT0=0 0: DCAEVT0 did not occur 1: DCAEVT0 has occurred.
1	CBC	RO	0x0	Cycle-by-cycle trip event interrupt flag This bit is always 0 if TZIE.CBC=0 0: Cycle-by-cycle trip interrupt does not occur 1: Cycle-by-cycle trip interrupt occurred. If the

Bits	Field Name	Type	Reset	Description
				cycle-by-cycle trip event is still present when the CBC bit is cleared, it will be immediately set again and a new interrupt is generated.
0	OST	RO	0x0	One-shot trip event interrupt flag This bit is always 0 if TZIE.OST=0 0: One-shot trip interrupt does not occur 1: One-shot trip interrupt occurred.

Table 8-92: Trip-Zone Clear Register (TZIC) Layout

TZIC (Trip-Zone Clear Register) Offset: 0xA8 Default: 0x00000000 Access: PWMx->TZIC							
31 CBCCLRMODEWE	30 CBCCLRMODE	29 CBCCLRMODE	28 RESERVED	27 RESERVED	26 RESERVED	25 RESERVED	24 RESERVED
23 RESERVED	22 RESERVED	21 RESERVED	20 RESERVED	19 RESERVED	18 RESERVED	17 RESERVED	16 RESERVED
15 RESERVED	14 RESERVED	13 RESERVED	12 RESERVED	11 RESERVED	10 RESERVED	9 RESERVED	8 RESERVED
7 RESERVED	6 INT	5 DCBEVT1	4 DCBEVT0	3 DCAEVT1	2 DCAEVT0	1 CBC	0 OST

Table 8-93: Trip-Zone Clear Register (TZIC) Description

Bits	Field Name	Type	Reset	Description
31	CBCCLRMODEWE	W1C	0x0	Latched CBC trip status clear mode write enable 0: Write 0 has no effect and always reads back 0 1: Write 1 enables concurrent write to CBCCLRMODE bit. This bit is self-cleared to 0
30:29	CBCCLRMODE	RW	0x0	Latched CBC trip status clear mode. This bit is writable only when CBCCLRMODEWE bit is 1 and PWMREGKEY=0x1acce551 00: Clear CBC trip on TBCNT=0 01: Clear CBC trip on TBCNT=TBPRD 10: Clear CBC trip on TBCNT=0 or TBCNT=TBPRD 11: Clear disabled. i.e. Never clear the CBC trip
28:7	RESERVED_28_7	RO	0x0	Reserved.
6	INT	W1C	0x0	Clear global interrupt status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.INT and TZ interrupt
5	DCBEVT1	W1C	0x0	Clear digital compare B event 1 status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.DCBEVT1

Bits	Field Name	Type	Reset	Description
4	DCBEVT0	W1C	0x0	Clear digital compare B event 0 status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.DCBEVT0
3	DCAEVT1	W1C	0x0	Clear digital compare A event 1 status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.DCAEVT1
2	DCAEVT0	W1C	0x0	Clear digital compare A event 0 status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.DCAEVT0
1	CBC	W1C	0x0	Clear cycle-by-cycle trip status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.CBC
0	OST	W1C	0x0	Clear one-shot trip status flag 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears TZIF.OST

Table 8-94: Trip-Zone Interrupt Enable Register (TZIE) Layout

TZIE (Trip-Zone Interrupt Enable Register) Offset: 0xAC Default: 0x00000000 Access: PWMx->TZIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		DCBEVT1	DCBEVT0	DCAEVT1	DCAEVT0	CBC	OST

Table 8-95: Trip-Zone Interrupt Enable Register (TZIE) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5	DCBEVT1	RW	0x0	Digital compare B event 1 interrupt enable 0: Disable 1: Enable
4	DCBEVT0	RW	0x0	Digital compare B event 0 interrupt enable 0: Disable 1: Enable
3	DCAEVT1	RW	0x0	Digital compare A event 1 interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
2	DCAEVT0	RW	0x0	Digital compare A event 0 interrupt enable 0: Disable 1: Enable
1	CBC	RW	0x0	Cycle-by-cycle trip-zone interrupt enable 0: Disable 1: Enable
0	OST	RW	0x0	One-shot trip-zone interrupt enable 0: Disable 1: Enable

Table 8-96: Trip-Zone Force Register (TZFRC) Layout

TZFRC (Trip-Zone Force Register) Offset: 0xB0 Default: 0x00000000 Access: PWMx->TZFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		DCBEVT1WE	DCBEVT0WE	DCAEVT1WE	DCAEVT0WE	CBCWE	OSTWE
7	6	5	4	3	2	1	0
RESERVED		DCBEVT1	DCBEVT0	DCAEVT1	DCAEVT0	CBC	OST

Table 8-97: Trip-Zone Force Register (TZFRC) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13	DCBEVT1WE	W1S	0x0	DCBEVT1 bit write enable 0: Disable 1: Enable
12	DCBEVT0WE	W1S	0x0	DCBEVT0 bit write enable 0: Disable 1: Enable
11	DCAEVT1WE	W1S	0x0	DCAEVT1 bit write enable 0: Disable 1: Enable
10	DCAEVT0WE	W1S	0x0	DAAEVT0 bit write enable 0: Disable 1: Enable
9	CBCWE	W1S	0x0	CBC bit write enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
8	OSTWE	W1S	0x0	OST bit write enable 0: Disable 1: Enable
7:6	RESERVED_7_6	RO	0x0	Reserved.
5	DCBEVT1	W1S	0x0	Software force a digital compare B event 1 DCBEVT1WE bit should be written 1 at the same time 0: Write 0 has no effect and always reads back 0 1: Write a 1 forces a digital compare B event 1 and sets TZIF bits
4	DCBEVT0	W1S	0x0	Software force a digital compare B event 0 DCBEVT0WE bit should be written 1 at the same time 0: Write 0 has no effect and always reads back 0 1: Write a 1 forces a digital compare B event 0 and sets TZIF bits
3	DCAEVT1	W1S	0x0	Software force a digital compare A event 1 DCAEVT1WE bit should be written 1 at the same time 0: Write 0 has no effect and always reads back 0 1: Write a 1 forces a digital compare A event 1 and sets TZIF bits
2	DCAEVT0	W1S	0x0	Software force a digital compare A event 0 DCAEVT0WE bit should be written 1 at the same time 0: Write 0 has no effect and always reads back 0 1: Write a 1 forces a digital compare A event 0 and sets TZIF bits
1	CBC	W1S	0x0	Software force a cycle-by-cycle trip event CBCWE bit should be written 1 at the same time 0: Write 0 has no effect and always reads back 0 1: Write a 1 forces a cycle-by-cycle trip event and sets TZIF bits
0	OST	W1S	0x0	Software force a one-shot trip event OSTWE bit should be written 1 at the same time 0: Write 0 has no effect and always reads back 0 1: Write a 1 forces a one-shot trip event and sets TZIF bits

Table 8-98: Digital Compare AL Trip Select Register (DCALTRIPSEL) Layout

DCALTRIPSEL (Digital Compare AL Trip Select Register) Offset: 0xB4 Default: 0x00000000 Access: PWMx->DCALTRIPSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

Table 8-99: Digital Compare AL Trip Select Register (DCALTRIPSEL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	COMP0H	RW	0x0	Enable COMP0H as DCAL trip event (ORed) 0: Disable 1: Enable
16	COMP0L	RW	0x0	Enable COMP0L as DCAL trip event (ORed) 0: Disable 1: Enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10	ADCPPU2TZ	RW	0x0	Enable ADCPPU2TZ as DCAL trip event (ORed) 0: Disable 1: Enable
9	ADCPPU1TZ	RW	0x0	Enable ADCPPU1TZ as DCAL trip event (ORed) 0: Disable 1: Enable
8	ADCPPU0TZ	RW	0x0	Enable ADCPPU0TZ as DCAL trip event (ORed) 0: Disable 1: Enable
7	RESERVED_7	RO	0x0	Reserved.
6	RESERVED_7	RW	0x0	Reserved.
5	RESERVED_7	RW	0x0	Reserved.
4	TZ4	RW	0x0	Enable TZ4 as DCAL trip event (ORed) 0: Disable 1: Enable
3	TZ3	RW	0x0	Enable TZ3 as DCAL trip event (ORed) 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
2	TZ2	RW	0x0	Enable TZ2 as DCAL trip event (ORed) 0: Disable 1: Enable
1	TZ1	RW	0x0	Enable TZ1 as DCAL trip event (ORed) 0: Disable 1: Enable
0	TZ0	RW	0x0	Enable TZ0 as DCAL trip event (ORed) 0: Disable 1: Enable

Table 8-100: Digital Compare AH Trip Select Register (DCAHTRIPSEL) Layout

DCAHTRIPSEL (Digital Compare AH Trip Select Register) Offset: 0xB8 Default: 0x00000000 Access: PWMx->DCAHTRIPSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

Table 8-101: Digital Compare AH Trip Select Register (DCAHTRIPSEL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	COMP0H	RW	0x0	Enable COMP0H as DCAH trip event (ORed) 0: Disable 1: Enable
16	COMP0L	RW	0x0	Enable COMP0L as DCAH trip event (ORed) 0: Disable 1: Enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10	ADCPPU2TZ	RW	0x0	Enable ADCPPU2TZ as DCAH trip event (ORed) 0: Disable 1: Enable
9	ADCPPU1TZ	RW	0x0	Enable ADCPPU1TZ as DCAH trip event (ORed) 0: Disable 1: Enable
8	ADCPPU0TZ	RW	0x0	Enable ADCPPU0TZ as DCAH trip event (ORed) 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
7	RESERVED_7	RO	0x0	Reserved.
6	RESERVED_6	RW	0x0	Reserved.
5	RESERVED_5	RW	0x0	Reserved.
4	TZ4	RW	0x0	Enable TZ4 as DCAH trip event (ORed) 0: Disable 1: Enable
3	TZ3	RW	0x0	Enable TZ3 as DCAH trip event (ORed) 0: Disable 1: Enable
2	TZ2	RW	0x0	Enable TZ2 as DCAH trip event (ORed) 0: Disable 1: Enable
1	TZ1	RW	0x0	Enable TZ1 as DCAH trip event (ORed) 0: Disable 1: Enable
0	TZ0	RW	0x0	Enable TZ0 as DCAH trip event (ORed) 0: Disable 1: Enable

Table 8-102: Digital Compare BL Trip Select Register (DCBLTRIPSEL) Layout

DCBLTRIPSEL (Digital Compare BL Trip Select Register) Offset: 0xBC Default: 0x00000000 Access: PWMx->DCBLTRIPSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

Table 8-103: Digital Compare BL Trip Select Register (DCBLTRIPSEL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	COMP0H	RW	0x0	Enable COMP0H as DCBL trip event (ORed) 0: Disable 1: Enable
16	COMP0L	RW	0x0	Enable COMP0L as DCBL trip event (ORed) 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
15:11	RESERVED_15_11	RO	0x0	Reserved.
10	ADCPPU2TZ	RW	0x0	Enable ADCPPU2TZ as DCBL trip event (ORed) 0: Disable 1: Enable
9	ADCPPU1TZ	RW	0x0	Enable ADCPPU1TZ as DCBL trip event (ORed) 0: Disable 1: Enable
8	ADCPPU0TZ	RW	0x0	Enable ADCPPU0TZ as DCBL trip event (ORed) 0: Disable 1: Enable
7	RESERVED_7	RO	0x0	Reserved.
6	RESERVED_6	RW	0x0	Reserved.
5	RESERVED_5	RW	0x0	Reserved.
4	TZ4	RW	0x0	Enable TZ4 as DCBL trip event (ORed) 0: Disable 1: Enable
3	TZ3	RW	0x0	Enable TZ3 as DCBL trip event (ORed) 0: Disable 1: Enable
2	TZ2	RW	0x0	Enable TZ2 as DCBL trip event (ORed) 0: Disable 1: Enable
1	TZ1	RW	0x0	Enable TZ1 as DCBL trip event (ORed) 0: Disable 1: Enable
0	TZ0	RW	0x0	Enable TZ0 as DCBL trip event (ORed) 0: Disable 1: Enable

Table 8-104: Digital Compare BH Trip Select Register (DCBHTRIPSEL) Layout

DCBHTRIPSEL (Digital Compare BH Trip Select Register) Offset: 0xC0 Default: 0x00000000 Access: PWMx->DCBHTRIPSEL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						COMP0H	COMP0L
15	14	13	12	11	10	9	8
RESERVED					ADCPPU2TZ	ADCPPU1TZ	ADCPPU0TZ
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	TZ4	TZ3	TZ2	TZ1	TZ0

Table 8-105: Digital Compare BH Trip Select Register (DCBHTRIPSEL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	COMP0H	RW	0x0	Enable COMP0H as DCBH trip event (ORed) 0: Disable 1: Enable
16	COMP0L	RW	0x0	Enable COMPL as DCBH trip event (ORed) 0: Disable 1: Enable
15:11	RESERVED_15_11	RO	0x0	Reserved.
10	ADCPPU2TZ	RW	0x0	Enable ADCPPU2TZ as DCBH trip event (ORed) 0: Disable 1: Enable
9	ADCPPU1TZ	RW	0x0	Enable ADCPPU1TZ as DCBH trip event (ORed) 0: Disable 1: Enable
8	ADCPPU0TZ	RW	0x0	Enable ADCPPU0TZ as DCBH trip event (ORed) 0: Disable 1: Enable
7	RESERVED_7	RO	0x0	Reserved.
6	RESERVED_6	RW	0x0	Reserved.
5	RESERVED_5	RW	0x0	Reserved.
4	TZ4	RW	0x0	Enable TZ4 as DCBH trip event (ORed) 0: Disable 1: Enable
3	TZ3	RW	0x0	Enable TZ3 as DCBH trip event (ORed) 0: Disable 1: Enable
2	TZ2	RW	0x0	Enable TZ2 as DCBH trip event (ORed) 0: Disable 1: Enable
1	TZ1	RW	0x0	Enable TZ1 as DCBH trip event (ORed) 0: Disable 1: Enable
0	TZ0	RW	0x0	Enable TZ0 as DCBH trip event (ORed) 0: Disable 1: Enable

Table 8-106: Digital Compare A Control Register (DCACTL) Layout

DCACTL (Digital Compare A Control Register) Offset: 0xC4 Default: 0x00000000 Access: PWMx->DCACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		EVT1SRCASYN	EVT1SRCSEL	EVT0SYNCE	EVT0SOCE	EVT0SRCASYN	EVT0SRCSEL

Table 8-107: Digital Compare A Control Register (DCACTL) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5	EVT1SRCASYN	RW	0x0	DCAEVT1 source signal is asynchronous/synchronous 0: Synchronous 1: Asynchronous
4	EVT1SRCSEL	RW	0x0	DCAEVT1 source signal select 0: Source signal is raw DCAEVT1 1: Source signal is DCEVTFILT
3	EVT0SYNCE	RW	0x0	DCAEVT0 SYNC generation 0: Disable 1: Enable
2	EVT0SOCE	RW	0x0	DCAEVT0 SOC generation 0: Disable 1: Enable
1	EVT0SRCASYN	RW	0x0	DCAEVT0 source signal is asynchronous/synchronous (1/0) 0: Synchronous 1: Asynchronous
0	EVT0SRCSEL	RW	0x0	DCAEVT0 source signal select 0: Source signal is raw DCAEVT0 1: Source signal is DCEVTFILT

Table 8-108: Digital Compare B Control Register (DCBCTL) Layout

DCBCTL (Digital Compare B Control Register) Offset: 0xC8 Default: 0x00000000 Access: PWMx->DCBCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		EVT1SRCASYNC	EVT1SRCSEL	EVT0SYNCE	EVT0SOCE	EVT0SRCASYNC	EVT0SRCSEL

Table 8-109: Digital Compare B Control Register (DCBCTL) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5	EVT1SRCASYNC	RW	0x0	DCBEVT1 source signal is asynchronous/synchronous (1/0) 0: Synchronous 1: Asynchronous
4	EVT1SRCSEL	RW	0x0	DCBEVT1 source signal select 0: Source signal is raw DCBEVT1 1: Source signal is DCEVTFILT
3	EVT0SYNCE	RW	0x0	DCBEVT0 SYNC generation 0: Disable 1: Enable
2	EVT0SOCE	RW	0x0	DCBEVT0 SOC generation 0: Disable 1: Enable
1	EVT0SRCASYNC	RW	0x0	DCBEVT0 source signal is asynchronous/synchronous 0: Synchronous 1: Asynchronous
0	EVT0SRCSEL	RW	0x0	DCBEVT0 source signal select 0: Source signal is raw DCBEVT0 1: Source signal is DCEVTFILT

Table 8-110: Digital Compare Filter Register (DCFCTL) Layout

DCFCTL (Digital Compare Filter Register) Offset: 0xCC Default: 0x00000010 Access: PWMx->DCFCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						WIN11EN	WIN10EN
15	14	13	12	11	10	9	8
WIN9EN	WIN8EN	WIN7EN	WIN6EN	WIN5EN	WIN4EN	WIN3EN	WIN2EN
7	6	5	4	3	2	1	0
WIN1EN	WIN0EN	PULSESEL		BLANKINV	BLANKEN	SRCSEL	

Table 8-111: Digital Compare Filter Register (DCFCTL) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	WIN11EN	RW	0x0	Enable blank window from PWM11 (ORed) 0: Disable 1: Enable
16	WIN10EN	RW	0x0	Enable blank window from PWM10 (ORed) 0: Disable 1: Enable
15	WIN9EN	RW	0x0	Enable blank window from PWM9 (ORed) 0: Disable 1: Enable
14	WIN8EN	RW	0x0	Enable blank window from PWM8 (ORed) 0: Disable 1: Enable
13	WIN7EN	RW	0x0	Enable blank window from PWM7 (ORed) 0: Disable 1: Enable
12	WIN6EN	RW	0x0	Enable blank window from PWM6 (ORed) 0: Disable 1: Enable
11	WIN5EN	RW	0x0	Enable blank window from PWM5 (ORed) 0: Disable 1: Enable
10	WIN4EN	RW	0x0	Enable blank window from PWM4 (ORed) 0: Disable 1: Enable
9	WIN3EN	RW	0x0	Enable blank window from PWM3 (ORed) 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
8	WIN2EN	RW	0x0	Enable blank window from PWM2 (ORed) 0: Disable 1: Enable
7	WIN1EN	RW	0x0	Enable blank window from PWM1 (ORed) 0: Disable 1: Enable
6	WIN0EN	RW	0x0	Enable blank window from PWM0 (ORed) 0: Disable 1: Enable
5:4	PULSESEL	RW	0x1	Pulse select for blanking and capture alignment for this module 00: Align on TBCNT=0 01: Align on TBCNT=TBPRD 10: Align on TBCNT=0 or TBCNT=TBPRD
3	BLANKINV	RW	0x0	blanking window inversion 0: Disable 1: Enable
2	BLANKEN	RW	0x0	blanking window 0: Disable 1: Enable
1:0	SRCSEL	RW	0x0	DCEVTFILT signal source select 00: Source is raw DCAEVT0 01: Source is raw DCAEVT1 10: Source is raw DCBEVT0 11: Source is raw DCBEVT1

Table 8-112: Digital Compare Filter Offset Register (DCFOFFSET) Layout

DCFOFFSET (Digital Compare Filter Offset Register) Offset: 0xD0 Default: 0x00000000 Access: PWMx->DCFOFFSET							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-113: Digital Compare Filter Offset Register (DCOFFSET) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Blanking window offset It defines the number of TBCLK cycles from the blanking window reference specified by DCFCTL.PULSESEL to the point when the blanking window is applied.

Table 8-114: Digital Compare Filter Offset Counter Register (DCOFFSETCNT) Layout

DCOFFSETCNT (Digital Compare Filter Offset Counter Register) Offset: 0xD4 Default: 0x00000000 Access: PWMx->DCOFFSETCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-115: Digital Compare Filter Offset Counter Register (DCOFFSETCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Blanking offset counter It is read only and indicates the current value of the offset counter. The counter counts down to zero and then stops until it is re-loaded on the next event defined by DCFCTL.PULSESEL.

Table 8-116: Digital Compare Filter Window Register (DCFWINDOW) Layout

DCFWINDOW (Digital Compare Filter Window Register) Offset: 0xD8 Default: 0x00000000 Access: PWMx->DCFWINDOW							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-117: Digital Compare Filter Window Register (DCFWINDOW) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Blanking window width in TBCLK cycles The blanking window begins when the offset counter expires. When this occurs, the window counter is loaded and begins to count down. If the blanking window is currently active and the offset counter expires, the blanking window counter is restarted. The blanking window can cross a PWM period boundary. No blanking window is generated when it is set to zero.

Table 8-118: Digital Compare Filter Window Counter Register (DCFWINDOWCNT) Layout

DCFWINDOWCNT (Digital Compare Filter Window Counter Register) Offset: 0xDC Default: 0x00000000 Access: PWMx->DCFWINDOWCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-119: Digital Compare Filter Window Counter Register (DCFWINDOWCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Blanking window counter It is read only and indicates the current value of the window counter. The counter counts down to zero and then stops until it is re-loaded when the offset counter reaches zero again.

Table 8-120: Digital Compare Capture Control Register (DCCAPCTL) Layout

DCCAPCTL (Digital Compare Capture Control Register) Offset: 0xE0 Default: 0x00000000 Access: PWMx->DCCAPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						CAPFLG	CAPCLR
7	6	5	4	3	2	1	0
RESERVED	MULTICAPWE	DIRECTWE	CAPENWE	RESERVED	MULTICAP	DIRECT	CAPEN

Table 8-121: Digital Compare Capture Control Register (DCCAPCTL) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	CAPFLG	RO	0x0	Latched capture flag 0: No DC capture event occurred 1: DC capture event has occurred
8	CAPCLR	W1C	0x0	Latched capture flag clear 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears the latched flag. This bit is self-cleared to 0
7	RESERVED_7	RO	0x0	Reserved.
6	MULTICAPWE	W1S	0x0	MULTICAP bit write enable 0: Write 0 has no effect and always reads back 0 1: Write 1 enables concurrent write to MULTICAP bit. This bit is self-cleared to 0
5	DIRECTWE	W1S	0x0	DIRECT bit write enable 0: Write 0 has no effect and always reads back 0 1: Write 1 enables concurrent write to DIRECT bit. This bit is self-cleared to 0
4	CAPENWE	W1S	0x0	CAPEN bit write enable 0: Write 0 has no effect and always reads back 0 1: Write 1 enables concurrent write to CAPEN bit. This bit is self-cleared to 0
3	RESERVED_3	RO	0x0	Reserved.
2	MULTICAP	RW	0x0	DCCAP multiple capture mode 0: Capture only once upon the first event in each cycle until next cycle is started by the pulse defined in DCFCTL.PULSESEL 1: Capture upon every events and always keep the latest captured value

Bits	Field Name	Type	Reset	Description
1	DIRECT	RW	0x0	DCCAP direct load mode 0: Shadow mode. The DCCAP active value is copied to shadow register on a TBCNT=TBPRD or TBCNT=zero event as defined by DCFCTL.PULSESEL. Reads of the DCCAP register always returns the shadow register contents. 1: Direct mode. Reads of the DCCAP register always returns the active value.
0	CAPEN	RW	0x0	TBCNT counter capture enable 0: Disable 1: Enable

Table 8-122: Digital Compare Counter Capture Register (DCCAP) Layout

DCCAP (Digital Compare Counter Capture Register) Offset: 0xE4 Default: 0x00000000 Access: PWMx->DCCAP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 8-123: Digital Compare Counter Capture Register (DCCAP) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Captured TBCNT value upon a rising-edge of DCEVTFLT In shadow mode, the captured value is copied to the shadow register on a TBCNT=TBPRD or TBCNT=zero event as defined by DCFCTL.PULSESEL. Reads of the DCCAP register always returns the shadow register contents. In direct mode, reads of the DCCAP register always returns the active value.

Table 8-124: Event-Trigger Control Register (ETCTL) Layout

ETCTL (Event-Trigger Control Register) Offset: 0xE8 Default: 0x1E1E1E1E Access: PWMx->ETCTL									
31	30	29	28	27	26	25	24		
RESERVED			SOCCSEL				SOCCEN		
23	22	21	20	19	18	17	16		
RESERVED			SOCBSEL				SOCBEN		
15	14	13	12	11	10	9	8		
RESERVED			SOCASEL				SOCAEN		
7	6	5	4	3	2	1	0		
RESERVED			INTSEL				INTEN		

Table 8-125: Event-Trigger Control Register (ETCTL) Description

Bits	Field Name	Type	Reset	Description
31:29	RESERVED_31_29	RO	0x0	Reserved.
28:25	SOCCSEL	RW	0xF	SOCC event select 0000: Enable event TBCNT=0 0001: Enable event TBCNT=TBPRD 0010: Enable event TBCNT=0 or TBCNT=TBPRD 0011: Enable event TBCNT=CMPC when TBCNT is counting up 0100: Enable event TBCNT=CMPC when TBCNT is counting down 0101: Enable event TBCNT=CMPC when TBCNT is counting up 0110: Enable event TBCNT=CMPC when TBCNT is counting down 0111: Enable event TBCNT=CMPC when TBCNT is counting up 1000: Enable event TBCNT=CMPC when TBCNT is counting down 1001: Enable event TBCNT=CMPC when TBCNT is counting up 1010: Enable event TBCNT=CMPC when TBCNT is counting down 1011: Enable DCAEVT0.SOC event 1100: Enable DCBEVT0.SOC event
24	SOCCEN	RW	0x0	ADC start of conversion C (SOCC) pulse enable 0: Disable 1: Enable
23:21	RESERVED_23_21	RO	0x0	Reserved.
20:17	SOCBSEL	RW	0xF	SOCB event select 0000: Select event TBCNT=0 0001: Select event TBCNT=TBPRD

Bits	Field Name	Type	Reset	Description
				0010: Select event TBCNT=0 or TBCNT=TBPRD 0011: Select event TBCNT=CMPA when TBCNT is counting up 0100: Select event TBCNT=CMPA when TBCNT is counting down 0101: Select event TBCNT=CMPB when TBCNT is counting up 0110: Select event TBCNT=CMPB when TBCNT is counting down 0111: Select event TBCNT=CMPC when TBCNT is counting up 1000: Select event TBCNT=CMPC when TBCNT is counting down 1001: Select event TBCNT=CMPD when TBCNT is counting up 1010: Select event TBCNT=CMPD when TBCNT is counting down 1011: Select DCAEVT0.SOC event 1100: Select DCBEVT0.SOC event
16	SOCBEN	RW	0x0	ADC start of conversion B (SOCB) pulse enable 0: Disable 1: Enable
15:13	RESERVED_15_13	RO	0x0	Reserved.
12:9	SOCASEL	RW	0xF	SOCA event select 0000: Select event TBCNT=0 0001: Select event TBCNT=TBPRD 0010: Select event TBCNT=0 or TBCNT=TBPRD 0011: Select event TBCNT=CMPA when TBCNT is counting up 0100: Select event TBCNT=CMPA when TBCNT is counting down 0101: Select event TBCNT=CMPB when TBCNT is counting up 0110: Select event TBCNT=CMPB when TBCNT is counting down 0111: Select event TBCNT=CMPC when TBCNT is counting up 1000: Select event TBCNT=CMPC when TBCNT is counting down 1001: Select event TBCNT=CMPD when TBCNT is counting up 1010: Select event TBCNT=CMPD when TBCNT is counting down 1011: Select DCAEVT0.SOC event 1100: Select DCBEVT0.SOC event

Bits	Field Name	Type	Reset	Description
8	SOCAEN	RW	0x0	ADC start of conversion A (SOCA) pulse enable 0: Disable 1: Enable
7:5	RESERVED_7_5	RO	0x0	Reserved.
4:1	INTSEL	RW	0xF	PWM interrupt select 0000: Select event TBCNT=0 0001: Select event TBCNT=TBPRD 0010: Select event TBCNT=0 or TBCNT=TBPRD 0011: Select event TBCNT=CMPC when TBCNT is counting up 0100: Select event TBCNT=CMPC when TBCNT is counting down 0101: Select event TBCNT=CMPC when TBCNT is counting up 0110: Select event TBCNT=CMPC when TBCNT is counting down 0111: Select event TBCNT=CMPC when TBCNT is counting up 1000: Select event TBCNT=CMPC when TBCNT is counting down 1001: Select event TBCNT=CMPC when TBCNT is counting up 1010: Select event TBCNT=CMPC when TBCNT is counting down 1011: Select DCAEVT0.SOC event 1100: Select DCBEVT0.SOC event
0	INTEN	RW	0x0	PWM interrupt enable 0: Disable 1: Enable

Table 8-126: Event-Trigger Prescale Register (ETPS) Layout

ETPS (Event-Trigger Prescale Register) Offset: 0xEC Default: 0x00000000							
Access: PWMx->ETPS							
31	30	29	28	27	26	25	24
SOCCCNT				SOCCPRD			
23	22	21	20	19	18	17	16
SOCBCNT				SOCBPRD			
15	14	13	12	11	10	9	8
SOCACNT				SOCAPRD			
7	6	5	4	3	2	1	0
INTCNT				INTPRD			

Table 8-127: Event-Trigger Prescale Register (ETPS) Description

Bits	Field Name	Type	Reset	Description
31:28	SOCCCNT	RW	0x0	PWM ADC start-of-conversion C event (SOCC) counter It indicates how many selected ETSEL.SOCCSEL events have occurred. It is automatically cleared when an SOCC pulse is generated. It can be initialized via write. 0000: No events have occurred 0001: 1 event has occurred 0010: 2 events have occurred 0011: 3 events have occurred 0100: 4 events have occurred 0101: 5 events have occurred 0110: 6 events have occurred 0111: 7 events have occurred 1000: 8 events have occurred 1001: 9 events have occurred 1010: 10 events have occurred 1011: 11 events have occurred 1100: 12 events have occurred 1101: 13 events have occurred 1110: 14 events have occurred 1111: 15 events have occurred
27:24	SOCCPRD	RW	0x0	PWM ADC start-of-conversion C event (SOCC) period select SOCC is generated upon $SOCCPRD \leq SOCCCNT$ and $ETCTL[SOCCEN]=1$. No SOCC will be generated if $SOCCPRD=0$ 0000: Disable SOCC event counter. No SOCC pulse will be generated. 0001: Generate SOCC pulse when $ETPS.SOCCCNT=0001$ (first event) 0010: Generate SOCC pulse when $ETPS.SOCCCNT=0010$ (second event) 0011: Generate SOCC pulse when $ETPS.SOCCCNT=0011$ (third event) 0100: Generate SOCC pulse when $ETPS.SOCCCNT=0100$ (fourth event) 0101: Generate SOCC pulse when $ETPS.SOCCCNT=0101$ (fifth event) 0110: Generate SOCC pulse when $ETPS.SOCCCNT=0110$ (sixth event) 0111: Generate SOCC pulse when $ETPS.SOCCCNT=0111$ (seventh event) 1000: Generate SOCC pulse when

Bits	Field Name	Type	Reset	Description
				ETPS.SOCCCNT=1000 (eighth event) 1001: Generate SOCC pulse when ETPS.SOCCCNT=1001 (ninth event) 1010: Generate SOCC pulse when ETPS.SOCCCNT=1010 (tenth event) 1011: Generate SOCC pulse when ETPS.SOCCCNT=1011 (eleventh event) 1100: Generate SOCC pulse when ETPS.SOCCCNT=1100(twelfth event) 1101: Generate SOCC pulse when ETPS.SOCCCNT=1101 (thirteenth event) 1110: Generate SOCC pulse when ETPS.SOCCCNT=1110(fourteenth event) 1111: Generate SOCC pulse when ETPS.SOCCCNT=1111 (fifteenth event)
23:20	SOCBCNT	RW	0x0	PWM ADC start-of-conversion B event (SOCB) counter It indicates how many selected ETSEL.SOCBSEL events have occurred. It is automatically cleared when an SOCB pulse is generated. It can be initialized via write. 0000: No events have occurred 0001: 1 event has occurred 0010: 2 events have occurred 0011: 3 events have occurred 0100: 4 events have occurred 0101: 5 events have occurred 0110: 6 events have occurred 0111: 7 events have occurred 1000: 8 events have occurred 1001: 9 events have occurred 1010: 10 events have occurred 1011: 11 events have occurred 1100: 12 events have occurred 1101: 13 events have occurred 1110: 14 events have occurred 1111: 15 events have occurred
19:16	SOCBPRD	RW	0x0	PWM ADC start-of-conversion B event (SOCB) period select SOCB is generated upon SOCBCNT<=SOCBPRD and ETCTL[SOCBEN]=1. No SOCB will be generated if SOCBCNT=0 0000: Disable SOCB event counter. No SOCB pulse will be generated. 0001: Generate SOCB pulse when ETPS.SOCBCNT=0001 (first event)

Bits	Field Name	Type	Reset	Description
				0010: Generate SOCB pulse when ETPS.SOCBCNT=0010 (second event) 0011: Generate SOCB pulse when ETPS.SOCBCNT=0011 (third event) 0100: Generate SOCB pulse when ETPS.SOCBCNT=0100 (fourth event) 0101: Generate SOCB pulse when ETPS.SOCBCNT=0101 (fifth event) 0110: Generate SOCB pulse when ETPS.SOCBCNT=0110(sixth event) 0111: Generate SOCB pulse when ETPS.SOCBCNT=0111 (seventh event) 1000: Generate SOCB pulse when ETPS.SOCBCNT=1000 (eighth event) 1001: Generate SOCB pulse when ETPS.SOCBCNT=1001 (ninth event) 1010: Generate SOCB pulse when ETPS.SOCBCNT=1010 (tenth event) 1011: Generate SOCB pulse when ETPS.SOCBCNT=1011 (eleventh event) 1100: Generate SOCB pulse when ETPS.SOCBCNT=1100(twelfth event) 1101: Generate SOCB pulse when ETPS.SOCBCNT=1101 (thirteenth event) 1110: Generate SOCB pulse when ETPS.SOCBCNT=1110(fourteenth event) 1111: Generate SOCB pulse when ETPS.SOCBCNT=1111 (fifteenth event)
15:12	SOCACNT	RW	0x0	PWM ADC start-of-conversion A event (SOCA) counter It indicates how many selected ETSEL.SOCASEL events have occurred. It is automatically cleared when an SOCA pulse is generated. It can be initialized via write. 0000: No events have occurred 0001: 1 event has occurred 0010: 2 events have occurred 0011: 3 events have occurred 0100: 4 events have occurred 0101: 5 events have occurred 0110: 6 events have occurred 0111: 7 events have occurred 1000: 8 events have occurred 1001: 9 events have occurred 1010: 10 events have occurred 1011: 11 events have occurred

Bits	Field Name	Type	Reset	Description
				1100: 12 events have occurred 1101: 13 events have occurred 1110: 14 events have occurred 1111: 15 events have occurred
11:8	SOCAPRD	RW	0x0	PWM ADC start-of-conversion A event (SOCA) period select SOCA is generated upon $SOCAPRD \leq SOCACNT$ and $ETCTL[SOCACNT] = 1$. No SOCA will be generated if $SOCAPRD = 0$ 0000: Disable SOCA event counter. No SOCA pulse will be generated. 0001: Generate SOCA pulse when $ETPS.SOCACNT = 01$ (first event) 0010: Generate SOCA pulse when $ETPS.SOCACNT = 10$ (second event) 0011: Generate SOCA pulse when $ETPS.SOCACNT = 11$ (third event) 0100: Generate SOCA pulse when $ETPS.SOCACNT = 0100$ (fourth event) 0101: Generate SOCA pulse when $ETPS.SOCACNT = 0101$ (fifth event) 0110: Generate SOCA pulse when $ETPS.SOCACNT = 0110$ (sixth event) 0111: Generate SOCA pulse when $ETPS.SOCACNT = 0111$ (seventh event) 1000: Generate SOCA pulse when $ETPS.SOCACNT = 1000$ (eighth event) 1001: Generate SOCA pulse when $ETPS.SOCACNT = 1001$ (ninth event) 1010: Generate SOCA pulse when $ETPS.SOCACNT = 1010$ (tenth event) 1011: Generate SOCA pulse when $ETPS.SOCACNT = 1011$ (eleventh event) 1100: Generate SOCA pulse when $ETPS.SOCACNT = 1100$ (twelfth event) 1101: Generate SOCA pulse when $ETPS.SOCACNT = 1101$ (thirteenth event) 1110: Generate SOCA pulse when $ETPS.SOCACNT = 1110$ (fourteenth event) 1111: Generate SOCA pulse when $ETPS.SOCACNT = 1111$ (fifteenth event)
7:4	INTCNT	RW	0x0	PWM interrupt counter It indicates how many selected $ETSEL.INTSEL$ events have occurred. It is automatically cleared when an interrupt pulse is generated. It can be initialized via write.

Bits	Field Name	Type	Reset	Description
				0000: No events have occurred 0001: 1 event has occurred 0010: 2 events have occurred 0011: 3 events have occurred 0100: 4 events have occurred 0101: 5 events have occurred 0110: 6 events have occurred 0111: 7 events have occurred 1000: 8 events have occurred 1001: 9 events have occurred 1010: 10 events have occurred 1011: 11 events have occurred 1100: 12 events have occurred 1101: 13 events have occurred 1110: 14 events have occurred 1111: 15 events have occurred
3:0	INTPRD	RW	0x0	PWM interrupt period select Interrupt is generated upon $INTPRD \leq INTCNT$ and $ETCTL[INTEN]=1$. No interrupt will be generated if $INTPRD=0$ 0000: Disable the interrupt event counter. No interrupt will be generated and $ETFRC.INT$ is ignored. 0001: Generate an interrupt when $ETPS.INTCNT=0001$ (first event) 0010: Generate an interrupt when $ETPS.INTCNT=0010$ (second event) 0011: Generate an interrupt when $ETPS.INTCNT=0011$ (third event) 0100: Generate an interrupt when $ETPS.INTCNT=0100$ (fourth event) 0101: Generate an interrupt when $ETPS.INTCNT=0101$ (fifth event) 0110: Generate an interrupt when $ETPS.INTCNT=0110$ (sixth event) 0111: Generate an interrupt when $ETPS.INTCNT=0111$ (seventh event) 1000: Generate an interrupt when $ETPS.INTCNT=1000$ (eighth event) 1001: Generate an interrupt when $ETPS.INTCNT=1001$ (ninth event) 1010: Generate an interrupt when $ETPS.INTCNT=1010$ (tenth event) 1011: Generate an interrupt when $ETPS.INTCNT=1011$ (eleventh event) 1100: Generate an interrupt when

Bits	Field Name	Type	Reset	Description
				ETPS.INTCNT=1100(twelfth event) 1101: Generate an interrupt when ETPS.INTCNT=1101 (thirteenth event) 1110: Generate an interrupt when ETPS.INTCNT=1110(fourteenth event) 1111: Generate an interrupt when ETPS.INTCNT=1111 (fifteenth event)

Table 8-128: Event-Trigger Flag Register (ETFLG) Layout

ETFLG (Event-Trigger Flag Register) Offset: 0xF0 Default: 0x00000000 Access: PWMx->ETFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				SOCC	SOCB	SOCA	INT

Table 8-129: Event-Trigger Flag Register (ETFLG) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	SOCC	RO	0x0	Latched PWM ADC start-of-conversion C flag 0: Indicates no event occurred 1: Indicates a SOCC event was generated. The pulse will continue to be generated even if the flag is set.
2	SOCB	RO	0x0	Latched PWM ADC start-of-conversion B flag 0: Indicates no event occurred 1: Indicates a SOCB event was generated. The pulse will continue to be generated even if the flag is set.
1	SOCA	RO	0x0	Latched PWM ADC start-of-conversion A flag 0: Indicates no event occurred 1: Indicates a SOCA event was generated. The pulse will continue to be generated even if the flag is set.
0	INT	RO	0x0	Latched PWM interrupt flag 0: Indicates no event occurred 1: Indicates a PWM interrupt was generated. No further interrupts will be generated until the flag bit is cleared.

Table 8-130: Event-Trigger Clear Register (ETCLR) Layout

ETCLR (Event-Trigger Clear Register) Offset: 0xF4 Default: 0x00000000 Access: PWMx->ETCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				SOCC	SOCB	SOCA	INT

Table 8-131: Event-Trigger Clear Register (ETCLR) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	SOCC	W1S	0x0	Latched PWM ADC start-of-conversion C flag clear 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears ETFLG.SOCC
2	SOCB	W1S	0x0	Latched PWM ADC start-of-conversion B flag clear 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears ETFLG.SOCB
1	SOCA	W1S	0x0	Latched PWM ADC start-of-conversion A flag clear 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears ETFLG.SOCA
0	INT	W1S	0x0	Latched PWM interrupt flag clear 0: Write 0 has no effect and always reads back 0 1: Write a 1 clears ETFLG.INT

Table 8-132: Event-Trigger Force Register (ETFRC) Layout

ETFRC (Event-Trigger Force Register) Offset: 0xF8 Default: 0x00000000 Access: PWMx->ETFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				SOCCEWE	SOCBWE	SOCWE	INTWE
7	6	5	4	3	2	1	0
RESERVED				SOCC	SOCB	SOCA	INT

Table 8-133: Event-Trigger Force Register (ETFRC) Description

Bits	Field Name	Type	Reset	Description
31:12	RESERVED_31_12	RO	0x0	Reserved.
11	SOCWE	W1S	0x0	SOC bit write enable 0: Disable 1: Enable
10	SOCBWE	W1S	0x0	SOCB bit write enable 0: Disable 1: Enable
9	SOCWE	W1S	0x0	SOC bit write enable 0: Disable 1: Enable
8	INTWE	W1S	0x0	INT bit write enable 0: Disable 1: Enable
7:4	RESERVED_7_4	RO	0x0	Reserved.
3	SOC	W1S	0x0	Software force a start-of-conversion B pulse SOCWE bit should be written 1 at the same time This bit is ignored if ETSSEL.SOCEN=0 0: Write 0 has no effect and always reads back 0 1: Write a 1 generates an interrupt and sets the ETSFLG.SOC flag
2	SOCB	W1S	0x0	Software force a start-of-conversion B pulse SOCBWE bit should be written 1 at the same time This bit is ignored if ETSSEL.SOCBEN=0 0: Write 0 has no effect and always reads back 0 1: Write a 1 generates an interrupt and sets the ETSFLG.SOCB flag
1	SOC	W1S	0x0	Software force a start-of-conversion A pulse SOCWE bit should be written 1 at the same time This bit is ignored if ETSSEL.SOCAEN=0 0: Write 0 has no effect and always reads back 0 1: Write a 1 generates an interrupt and sets the ETSFLG.SOC flag
0	INT	W1S	0x0	Software force a PWM interrupt INT bit should be written 1 at the same time This bit is ignored if ETSSEL.INTEN=0 or ETPS.INTPRD=00 0: Write 0 has no effect and always reads back 0 1: Write a 1 generates an interrupt and sets the ETSFLG.INT flag

Table 8-134: PWM Register Write-Allow Key Register (PWMREGKEY) Layout

PWMREGKEY (PWM Register Write-Allow Key Register) Offset: 0xFC Default: 0x1ACCE551 Access: PWMx->PWMREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 8-135: PWM Register Write-Allow Key Register (PWMREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected PWM registers

8.11.3 PWMCFG register map

Table 8-136: PWMCFG Module Base Address

Peripheral Module	Base Address
PWMCFG	0x4000 2F00

Table 8-137: PWMCFG Register Map

Register	Offset	Description	Reset Value
<u>TZ0SRCCTL</u> *	0x00	TZ0 Source Control Register	0x00000001
<u>TZ1SRCCTL</u> *	0x04	TZ1 Source Control Register	0x00000001
<u>TZ2SRCCTL</u> *	0x08	TZ2 Source Control Register	0x00000001
<u>TZ3SRCCTL</u> *	0x0C	TZ3 Source Control Register	0x00000001
<u>TZ4SRCCTL</u> *	0x10	TZ4 Source Control Register	0x00000001
<u>FRCSYNC</u>	0x14	Global PWM Force Synchronization Register	0x00000000
<u>GPIOSYNCTL</u> *	0x18	GPIO PWMSYNCTL Control Register	0x00000001
<u>GPIOSYNCTEN</u> *	0x1C	GPIO Force PWMSYNCTEN Enable Register	0x00000000
<u>TMROSYNCTEN</u> *	0x20	Timer 0 Force PWMSYNCTEN Enable Register	0x00000000
<u>TMR1SYNCTEN</u> *	0x24	Timer 1 Force PWMSYNCTEN Enable Register	0x00000000
<u>TMR2SYNCTEN</u> *	0x28	Timer 2 Force PWMSYNCTEN Enable Register	0x00000000
<u>SYNCOCTL</u> *	0x2C	PWMSYNCO Control Register	0x00000007
<u>SOCACCTL</u> *	0x30	PWMSOCA Output Control Register	0x00000007
<u>SOCBOCTL</u> *	0x34	PWMSOCB Output Control Register	0x00000007
<u>SOCOCCTL</u> *	0x38	PWMSOCC Output Control Register	0x00000007
<u>PWMCFGREGKEY</u>	0x3C	PWMCFG Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the PWMCFGKEY=0x1ACCE551.

8.11.4 PWMCFG registers

Table 8-138: TZ0 Source Control Register (TZ0SRCCTL) Layout

TZ0SRCCTL (TZ0 Source Control Register) Offset: 0x0 Default: 0x00000001 Access: PWMCFG->TZ0SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 8-139: TZ0 Source Control Register (TZ0SRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for TZ0 event
0	POL	RW	0x1	TZ0 polarity 0: Active when low 1: Active when high

Table 8-140: TZ1 Source Control Register (TZ1SRCCTL) Layout

TZ1SRCCTL (TZ1 Source Control Register) Offset: 0x4 Default: 0x00000001 Access: PWMCFG->TZ1SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 8-141: TZ1 Source Control Register (TZ1SRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for TZ1 event
0	POL	RW	0x1	TZ1 polarity 0: Active when low 1: Active when high

Table 8-142: TZ2 Source Control Register (TZ2SRCCTL) Layout

TZ2SRCCTL (TZ2 Source Control Register) Offset: 0x8 Default: 0x00000001 Access: PWMCFG->TZ2SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 8-143: TZ2 Source Control Register (TZ2SRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for TZ2 event
0	POL	RW	0x1	TZ2 polarity 0: Active when low 1: Active when high

Table 8-144: TZ3 Source Control Register (TZ3SRCCTL) Layout

TZ3SRCCTL (TZ3 Source Control Register) Offset: 0xC Default: 0x00000001 Access: PWMCFG->TZ3SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 8-145: TZ3 Source Control Register (TZ3SRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for TZ3 event
0	POL	RW	0x1	TZ3 polarity 0: Active when low 1: Active when high

Table 8-146: TZ4 Source Control Register (TZ4SRCCTL) Layout

TZ4SRCCTL (TZ4 Source Control Register) Offset: 0x10 Default: 0x00000001 Access: PWMCFG->TZ4SRCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 8-147: TZ4 Source Control Register (TZ4SRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for TZ4 event
0	POL	RW	0x1	TZ4 polarity 0: Active when low 1: Active when high

Table 8-148: Global PWM Force Synchronization Register (FRCSYNC) Layout

FRCSYNC (Global PWM Force Synchronization Register) Offset: 0x14 Default: 0x00000000 Access: PWMCFG->FRCSYNC							
31	30	29	28	27	26	25	24
PWMCLK	RESERVED						
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

Table 8-149: Global PWM Force Synchronization Register (FRCSYNC) Description

Bits	Field Name	Type	Reset	Description
31	PWMCLK	W1S	0x0	Software forced synchronization on clock for all PWMs 0: Write 0 has no effect and always read back 0 1: Write a 1 will force a SYNC on all PWM clocks This bit is self-cleared
30:4	RESERVED_30_4	RO	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
3	PWM3SYNC	W1S	0x0	Software forced synchronization on PWM3 0: Write 0 has no effect and always read back 0 1: Write a 1 will force a SYNC event on PWM3 This bit is self-cleared
2	PWM2SYNC	W1S	0x0	Software forced synchronization on PWM2 0: Write 0 has no effect and always read back 0 1: Write a 1 will force a SYNC event on PWM2 This bit is self-cleared
1	PWM1SYNC	W1S	0x0	Software forced synchronization on PWM1 0: Write 0 has no effect and always read back 0 1: Write a 1 will force a SYNC event on PWM1 This bit is self-cleared
0	PWM0SYNC	W1S	0x0	Software forced synchronization on PWM0 0: Write 0 has no effect and always read back 0 1: Write a 1 will force a SYNC event on PWM0 This bit is self-cleared

Table 8-150: GPIO PWMSYNCI Control Register (GPIOSYNICTL) Layout

GPIOSYNICTL (GPIO PWMSYNCI Control Register) Offset: 0x18 Default: 0x00000001 Access: PWMCFG->GPIOSYNICTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 8-151: GPIO PWMSYNCI Control Register (GPIOSYNICTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for PWMSYNCI input
0	POL	RW	0x1	PWMSYNCI input polarity 0: Active when low 1: Active when high

Table 8-152: GPIO Force PWMSYNCI Enable Register (GPIOSYNCIEN) Layout

GPIOSYNCIEN (GPIO Force PWMSYNCI Enable Register) Offset: 0x1C Default: 0x00000000 Access: PWMCFGx->GPIOSYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

Table 8-153: GPIO Force PWMSYNCI Enable Register (GPIOSYNCIEN) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	PWM3SYNC	RW	0x0	Enable PWM3 synchronization by GPIO 0: Disable 1: Enable
2	PWM2SYNC	RW	0x0	Enable PWM2 synchronization by GPIO 0: Disable 1: Enable
1	PWM1SYNC	RW	0x0	Enable PWM1 synchronization by GPIO 0: Disable 1: Enable
0	PWM0SYNC	RW	0x0	Enable PWM0 synchronization by GPIO 0: Disable 1: Enable

Table 8-154: Timer 0 Force PWMSYNCI Enable Register (TMR0SYNCIEN) Layout

TMR0SYNCIEN (Timer 0 Force PWMSYNCI Enable Register) Offset: 0x20 Default: 0x00000000 Access: PWMCFG->TMR0SYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

Table 8-155: Timer 0 Force PWMSYNCI Enable Register (TMR0SYNCIEN) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	PWM3SYNC	RW	0x0	Enable PWM3 synchronization by Timer 0 0: Disable 1: Enable
2	PWM2SYNC	RW	0x0	Enable PWM2 synchronization by Timer 0 0: Disable 1: Enable
1	PWM1SYNC	RW	0x0	Enable PWM1 synchronization by Timer 0 0: Disable 1: Enable
0	PWM0SYNC	RW	0x0	Enable PWM0 synchronization by Timer 0 0: Disable 1: Enable

Table 8-156: Timer 1 Force PWMSYNCI Enable Register (TMR1SYNCIEN) Layout

TMR1SYNCIEN (Timer 1 Force PWMSYNCI Enable Register) Offset: 0x24 Default: 0x00000000 Access: PWMCFG->TMR1SYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

Table 8-157: Timer 1 Force PWMSYNCI Enable Register (TMR1SYNCIEN) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	PWM3SYNC	RW	0x0	Enable PWM3 synchronization by Timer 1 0: Disable 1: Enable
2	PWM2SYNC	RW	0x0	Enable PWM2 synchronization by Timer 1 0: Disable 1: Enable
1	PWM1SYNC	RW	0x0	Enable PWM1 synchronization by Timer 1 0: Disable 1: Enable
0	PWM0SYNC	RW	0x0	Enable PWM0 synchronization by Timer 1 0: Disable 1: Enable

Table 8-158: Timer 2 Force PWMSYNCI Enable Register (TMR2SYNCIEN) Layout

TMR2SYNCIEN (Timer 2 Force PWMSYNCI Enable Register) Offset: 0x28 Default: 0x00000000 Access: PWMCFG->TMR2SYNCIEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				PWM3SYNC	PWM2SYNC	PWM1SYNC	PWM0SYNC

Table 8-159: Timer 2 Force PWMSYNCI Enable Register (TMR2SYNCIEN) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	PWM3SYNC	RW	0x0	Enable PWM3 synchronization by Timer 2 0: Disable 1: Enable
2	PWM2SYNC	RW	0x0	Enable PWM2 synchronization by Timer 2 0: Disable 1: Enable
1	PWM1SYNC	RW	0x0	Enable PWM1 synchronization by Timer 2 0: Disable 1: Enable
0	PWM0SYNC	RW	0x0	Enable PWM0 synchronization by Timer 2 0: Disable 1: Enable

Table 8-160: PWMSYNCO Control Register (SYNCOCTL) Layout

SYNCOCTL (PWMSYNCO Control Register) Offset: 0x2C Default: 0x00000007 Access: PWMCFG->SYNCOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SYNCO3EN	SYNCO2EN	SYNCO1EN	SYNCO0EN	POL	DURATION	

Table 8-161: PWMSYNCO Control Register (SYNCOCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	SYNCO3EN	RW	0x0	Enable PWM3SYNCO output to the pin 0: Disable 1: Enable
5	SYNCO2EN	RW	0x0	Enable PWM2SYNCO output to the pin 0: Disable 1: Enable
4	SYNCO1EN	RW	0x0	Enable PWM1SYNCO output to the pin 0: Disable 1: Enable
3	SYNCO0EN	RW	0x0	Enable PWM0SYNCO output to the pin 0: Disable 1: Enable
2	POL	RW	0x1	Polarity of PWMSYNCO to the pin 0: Active when low 1: Active when high
1:0	DURATION	RW	0x3	Pulse duration of PWMSYNCO to the pin 00: Duration of PWMSYNCO pulse to the pin is 4 PWM clocks 01: Duration of PWMSYNCO pulse to the pin is 8 PWM clocks 10: Duration of PWMSYNCO pulse to the pin is 16 PWM clocks 11: Duration of PWMSYNCO pulse to the pin is 32 PWM clocks

Table 8-162: PWMSOCA Output Control Register (SOCAOCTL) Layout

SOCAOCTL (PWMSOCA Output Control Register) Offset: 0x30 Default: 0x00000007 Access: PWMCFG->SOCAOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SOCA3EN	SOCA2EN	SOCA1EN	SOCA0EN	POL	DURATION	

Table 8-163: PWMSOCA Output Control Register (SOCAOCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	SOCA3EN	RW	0x0	Enable PWM3SOCA output to the pin 0: Disable 1: Enable
5	SOCA2EN	RW	0x0	Enable PWM2SOCA output to the pin 0: Disable 1: Enable
4	SOCA1EN	RW	0x0	Enable PWM1SOCA output to the pin 0: Disable 1: Enable
3	SOCA0EN	RW	0x0	Enable PWM0SOCA output to the pin 0: Disable 1: Enable
2	POL	RW	0x1	Polarity of PWMSOCA output to the pin 0: Active when low 1: Active when high
1:0	DURATION	RW	0x3	Pulse duration of PWMSOCA output to the pin 00: Duration of PWMSOCA pulse to the pin is 4 PWM clocks 01: Duration of PWMSOCA pulse to the pin is 8 PWM clocks 10: Duration of PWMSOCA pulse to the pin is 16 PWM clocks 11: Duration of PWMSOCA pulse to the pin is 32 PWM clocks

Table 8-164: PWMSOCB Output Control Register (SOCBOCTL) Layout

SOCBOCTL (PWMSOCB Output Control Register) Offset: 0x34 Default: 0x00000007							
Access: PWMCFG->SOCBOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SOCB3EN	SOCB2EN	SOCB1EN	SOCB0EN	POL	DURATION	

Table 8-165: PWMSOCB Output Control Register (SOCBOCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	SOCB3EN	RW	0x0	Enable PWM3SOCB output to the pin 0: Disable 1: Enable
5	SOCB2EN	RW	0x0	Enable PWM2SOCB output to the pin 0: Disable 1: Enable
4	SOCB1EN	RW	0x0	Enable PWM1SOCB output to the pin 0: Disable 1: Enable
3	SOCB0EN	RW	0x0	Enable PWM0SOCB output to the pin 0: Disable 1: Enable
2	POL	RW	0x1	Polarity of PWMSOCB output to the pin 0: Active when low 1: Active when high
1:0	DURATION	RW	0x3	Pulse duration of PWMSOCB output to the pin 00: Duration of PWMSOCB pulse to the pin is 4 PWM clocks 01: Duration of PWMSOCB pulse to the pin is 8 PWM clocks 10: Duration of PWMSOCB pulse to the pin is 16 PWM clocks 11: Duration of PWMSOCB pulse to the pin is 32 PWM clocks

Table 8-166: PWMSOCC Output Control Register (SOCCOCTL) Layout

SOCCOCTL (PWMSOCC Output Control Register) Offset: 0x38 Default: 0x00000007							
Access: PWMCFG->SOCCOCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	SOCC3EN	SOCC2EN	SOCC1EN	SOCC0EN	POL	DURATION	

Table 8-167: PWMSOCC Output Control Register (SOCCOCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	SOCC3EN	RW	0x0	Enable PWM3SOCC output to the pin 0: Disable 1: Enable
5	SOCC2EN	RW	0x0	Enable PWM2SOCC output to the pin 0: Disable 1: Enable
4	SOCC1EN	RW	0x0	Enable PWM1SOCC output to the pin 0: Disable 1: Enable
3	SOCC0EN	RW	0x0	Enable PWM0SOCC output to the pin 0: Disable 1: Enable
2	POL	RW	0x1	Polarity of PWMSOCC output to the pin 0: Active when low 1: Active when high
1:0	DURATION	RW	0x3	Pulse duration of PWMSOCC output to the pin 00: Duration of PWMSOCC pulse to the pin is 4 PWM clocks 01: Duration of PWMSOCC pulse to the pin is 8 PWM clocks 10: Duration of PWMSOCC pulse to the pin is 16 PWM clocks 11: Duration of PWMSOCC pulse to the pin is 32 PWM clocks

Table 8-168: PWMCFG Register Write-Allow Key Register (PWMCFGREGKEY) Layout

PWMCFGREGKEY (PWMCFG Register Write-Allow Key Register) Offset: 0x3C Default: 0x1ACCE551 Access: PWMCFG->PWMCFGREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 8-169: PWMCFG Register Write-Allow Key Register (PWMCFGREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected PWMCFG registers

9 ECAP

9.1 Introduction

The enhanced capture (ECAP) module is used in situations where accurate timing of external events is important. This chapter describes the module and how to use it.

Uses for ECAP include:

- Speed measurements of rotating machinery (e.g., e-bike motor with Hall sensors)
- Measure period and duty cycle of pulse signals
- Decoding information derived from duty cycle encoded systems

The ECAP module described in this guide includes the following features:

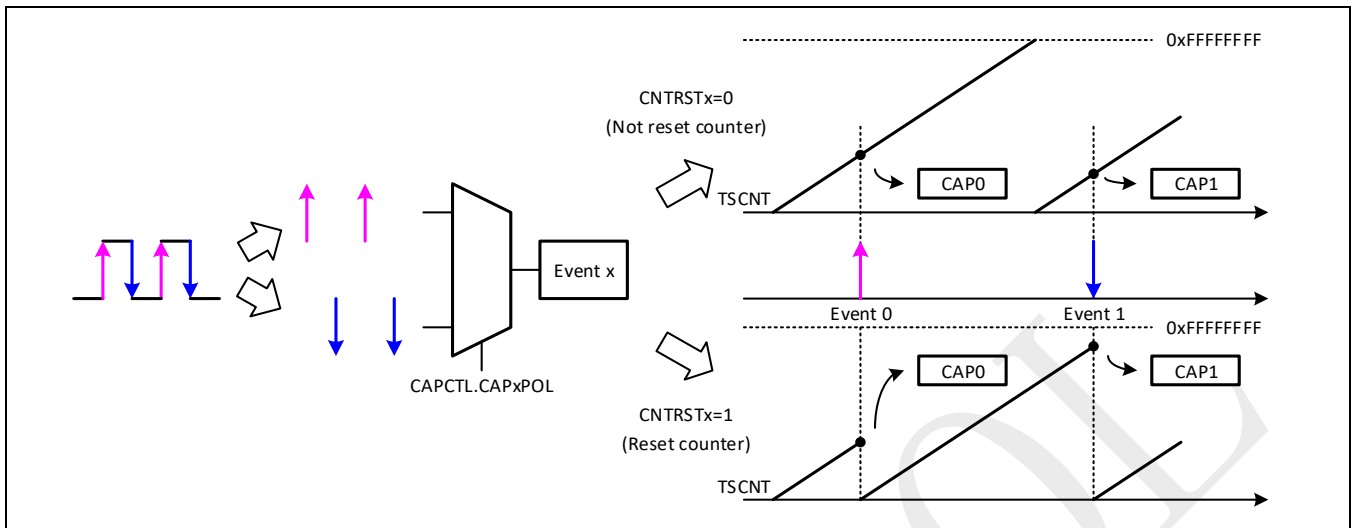
- 32-bit time base with 10 ns time resolution (clock frequency up to 100MHz)
- Four 32-bit event time-stamp registers
- Edge polarity selection for up to four sequenced time-stamp capture events
- Interrupt on either of the four events
- Single shot capture of up to four event time-stamps
- Continuous mode capture of time-stamps in circular manner
- Absolute time-stamp capture
- Difference (Delta) mode time-stamp capture
- All above resources dedicated to a single input pin or dual input pins
- Configurable to be PWM output if not used in capture mode

9.2 Description

Figure 9-1 shows the basic two operations of the enhanced capture (ECAP). The philosophy of the ECAP module is to treat and abstract the “rising” and “falling” edge of the captured signal as event sequence. They are called CEVT0, CEVT1, CEVT2 and CEVT3. Each of them can be programmed to be valid on rising or falling edge. This is set by CAPCTL.CAPxPOL field.

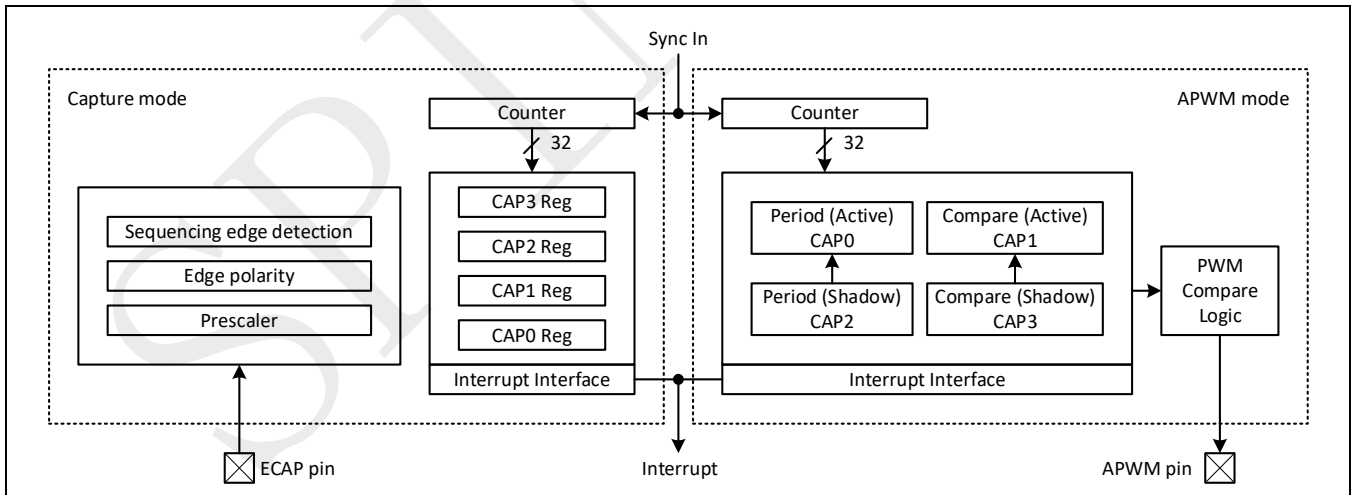
The second is to set counter behavior after event happened. The possible options can be resetting the counter or doing nothing, as specified by CAPCTL.CNTRSTx.

With the two basic configuration of events and counter, users can make complicated capture function such as period and duty measurement with other registers illustrated later.

Figure 9-1: Basic operation of enhanced capture

9.3 Capture and APWM operating mode

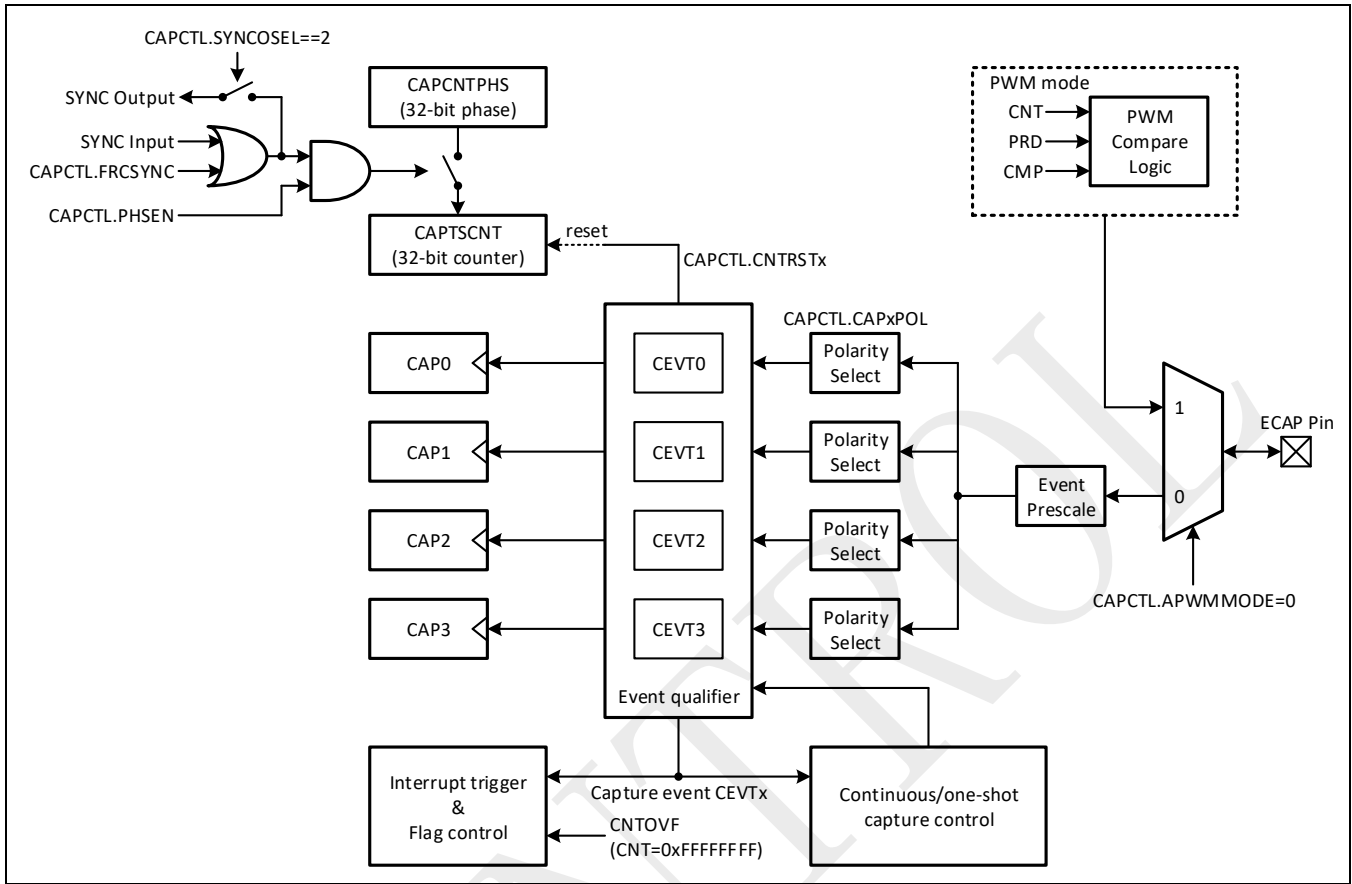
Figure 9-2 is a high-level view of both the capture and auxiliary pulse-width modulator (APWM) modes operation. User can take ECAP module resources to implement a single-channel PWM generator (with 32 bit capabilities) when it is not being used for input captures. The counter operates in count-up mode, providing a time-base for asymmetrical pulse width modulation (PWM) waveforms. The CAP0 and CAP1 registers become the period and compare active registers, respectively, while CAP2 and CAP3 registers become the period and capture shadow registers, respectively.

Figure 9-2: Capture and APWM modes of operation

9.4 Capture mode description

Figure 9-3 shows the various components that implement the capture function. The captured signal is sourced from GPIO pin according to CAPSRCCTL register.

Figure 9-3: Capture function diagram



9.4.1 Event prescaler

An input capture signal (continuous pulse series) can be prescaled by $N = \text{CAPCTL.EVTDIV} + 1$. This is useful when very high frequency signals are used as inputs. Figure 9-4 shows a functional diagram and Figure 9-5 shows the operation of the prescale function.

Figure 9-4: Event prescale control

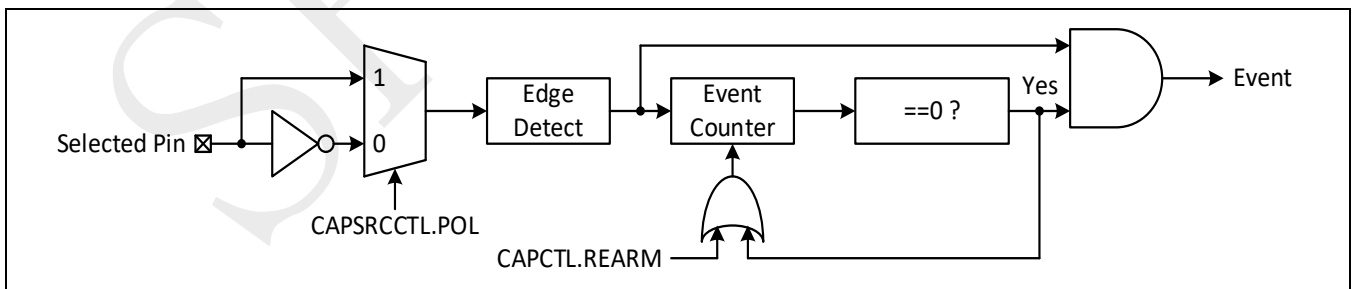
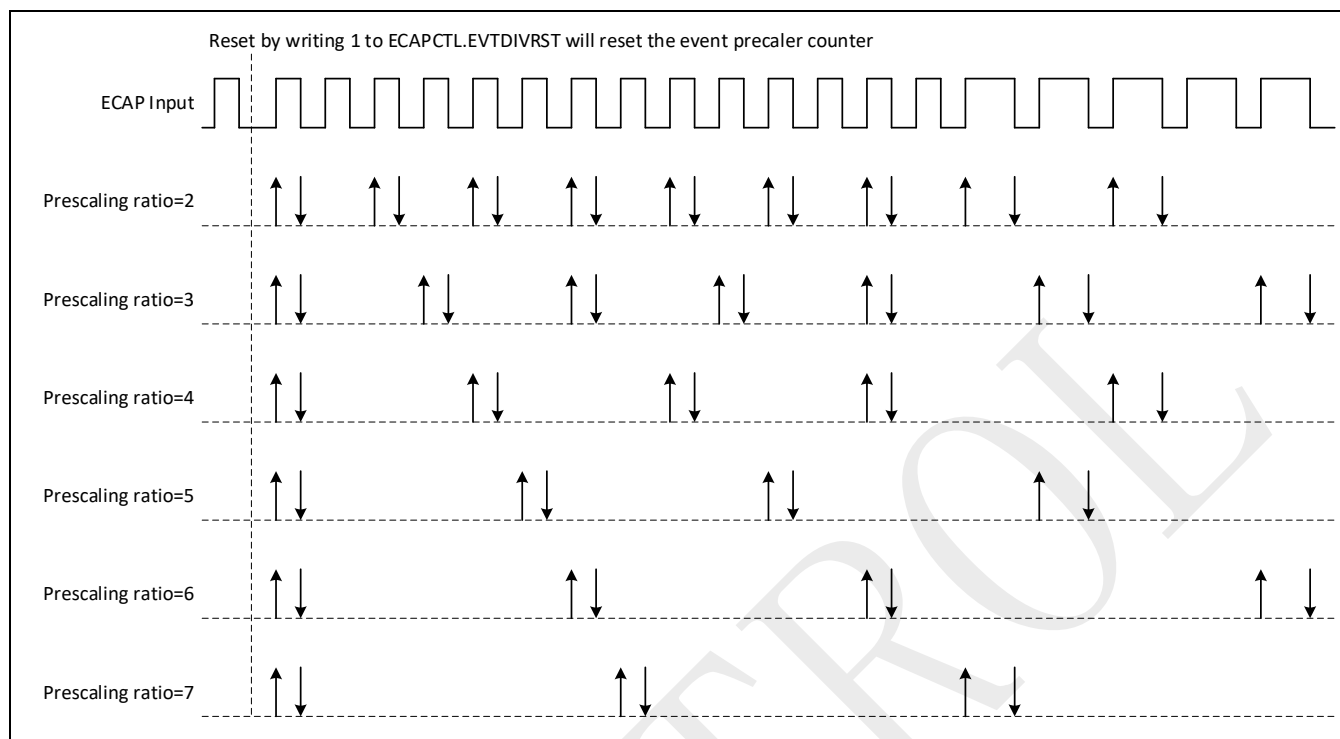


Figure 9-5: Prescale function waveforms



9.4.2 Edge polarity select and qualifier

Each capture event has an independent edge polarity (rising edge/falling edge) selection MUX.

- Each edge (up to 4) is event qualified by the Modulo-4 sequencer.
- The edge event is gated to the corresponding CAPx register by the Mod4 counter.

9.4.3 Dual pin capture

For ease of capture the timing difference of edges from two-pins, optional dual-pin mode is supported. In this case, capture event CEVT0 and CEVT2 are from Pin 0, while capture event CEVT1 and CEVT3 are from Pin 1.

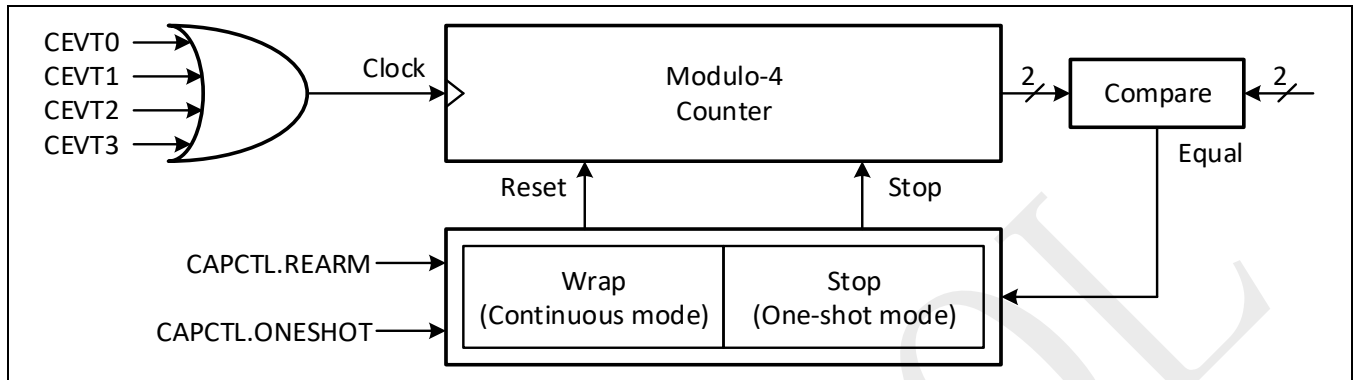
9.4.4 Continuous/one-shot control

The Mod4 (2 bit) counter increases upon edge qualified events (CEVT0-CEVT3). It indicates which event in the sequence is to be detected. A 2-bit wrap threshold register (CAPCTL.STOPWARP) is used to compare with the Mod4 counter output. It controls the length of the capture event sequence as below.

- In continuous mode, the Mod4 counter keeps counting and wraps around when it matches the threshold. For example, if CAPCTL.STOPWARP is set to 2, the Mod4 counter continues to run as 0→1→2→0, and the captured value continue to be written to CAP0-2 in a circular buffer sequence.
- In one-shot mode, the Mod4 counter stops counting when it matches the threshold. Further loads of CAP0-CAP3 registers would be inhibited. For example, if CAPCTL.STOPWARP is set to 3, the Mod4 counter runs as 0→1→2→3, and stops. Captured values are sequentially written to CAP0-3 and keep unchanged later.

Upon writing 1 to CAPCTL.REARM, the Mod4 counter can be re-armed and reset to 0. After that, event capture sequence starts again for both the continuous mode and one-shot mode.

Figure 9-6: Details of the continuous/one-shot block

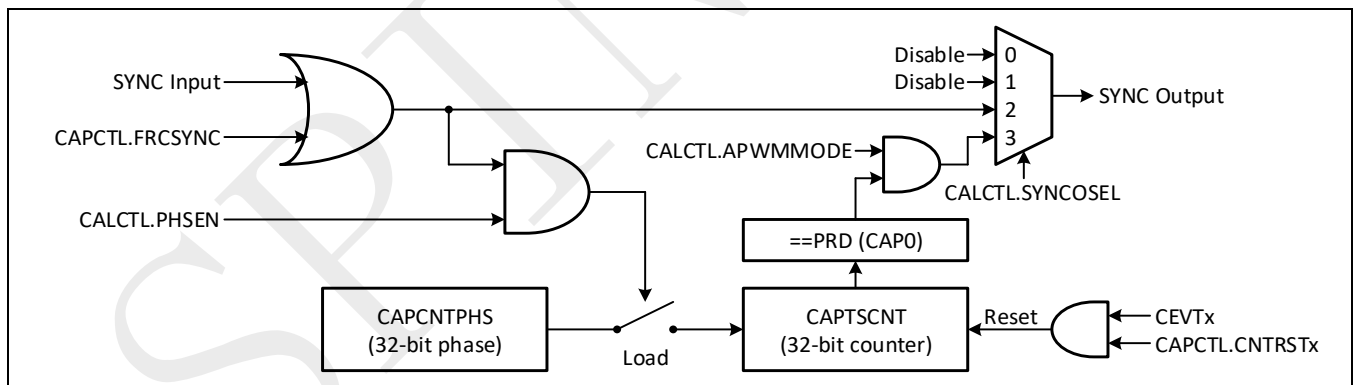


9.4.5 32-bit counter and phase control

The 32-bit counter provides the time-base for event capture, and is clocked via the ECAP clock. A phase register is provided to achieve synchronization with other counters, via a hardware and software forced sync. This is useful in APWM mode when a phase offset between modules is needed.

Upon any of the four capture events, the 32-bit counter value is captured first. After that, it is optional to reset the 32-bit counter. This is useful for time difference capture.

Figure 9-7: Details of the counter and synchronization block



9.4.6 CAP0-CAP3 registers

In capture mode, the four 32-bit registers are fed from the 32-bit time-base counter TSCNT. Upon the captured event, the TSCNT is loaded into the corresponding capture register as indexed by the Mod4 counter. For example, if current Mod4 counter is 1, CAP1 would be loaded upon the next valid capture event.

In APWM mode, CAP0 and CAP1 registers become the active period and compare registers, while CAP2 and CAP3 registers become the shadow registers for CAP0 and CAP1 respectively.

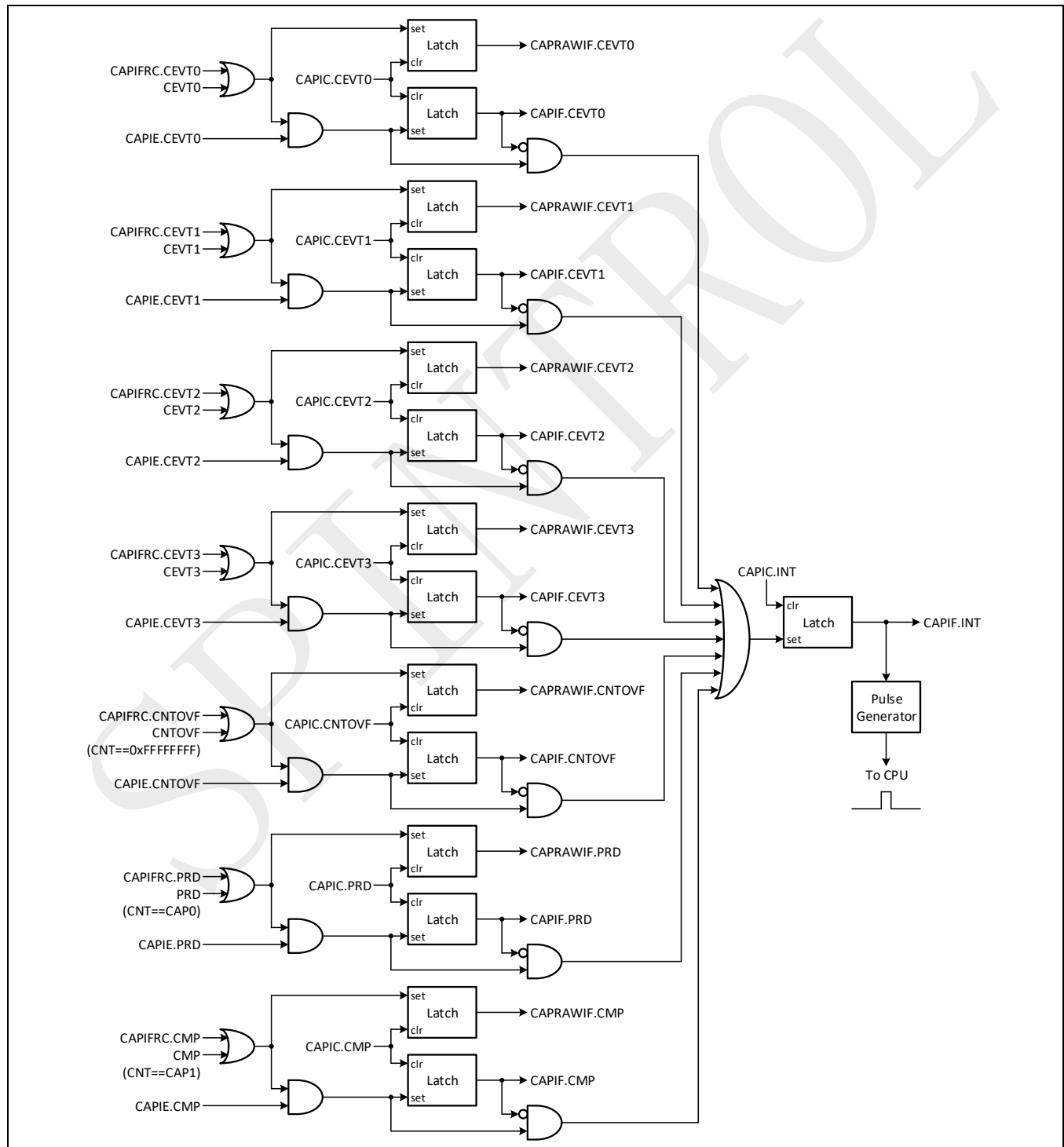
9.4.7 Interrupt control

An interrupt can be generated on capture events (CEVT0-CEVT3) or APWM events (CNT=PRD, CNT=COMP). A counter overflow event (0xFFFFFFFF->0x00000000) is also provided as an interrupt source (CNTOVF).

For capture mode, valid interrupt events includes CEVT0, CEVT1, CEVT2, CEVT3, and CNTOVF.

For APWM mode, valid interrupt events includes CNT=PRD, CNT=COMP, and CNTOVF.

Figure 9-8: Interrupts in ECAP module



Upon any of the overall 7 events, the event flag would be latched into the corresponding bit of CAPRAWIF register. If it is enabled in CAPIE register bit, the event can trigger the final single interrupt request from the ECAP module to the CPU. The individual interrupt flag is latched in CAPIF, while the INT bit of CAPIF is the global interrupt flag of the ECAP module.

The individual interrupt event is valid only when the corresponding flag in CAPIF is 0. Namely, the same event would not trigger the interrupt again until the flag is cleared. Similarly, the ECAP module would no longer issue the interrupt request to the CPU if CAPIF.INT=1. Therefore, in the interrupt service routine, user must clear the global interrupt flag and the serviced event flag in CAPIF via writing 1 to CAPIC register bits.

As a debug feature, the interrupt events can be software forced by CAPIFRC register.

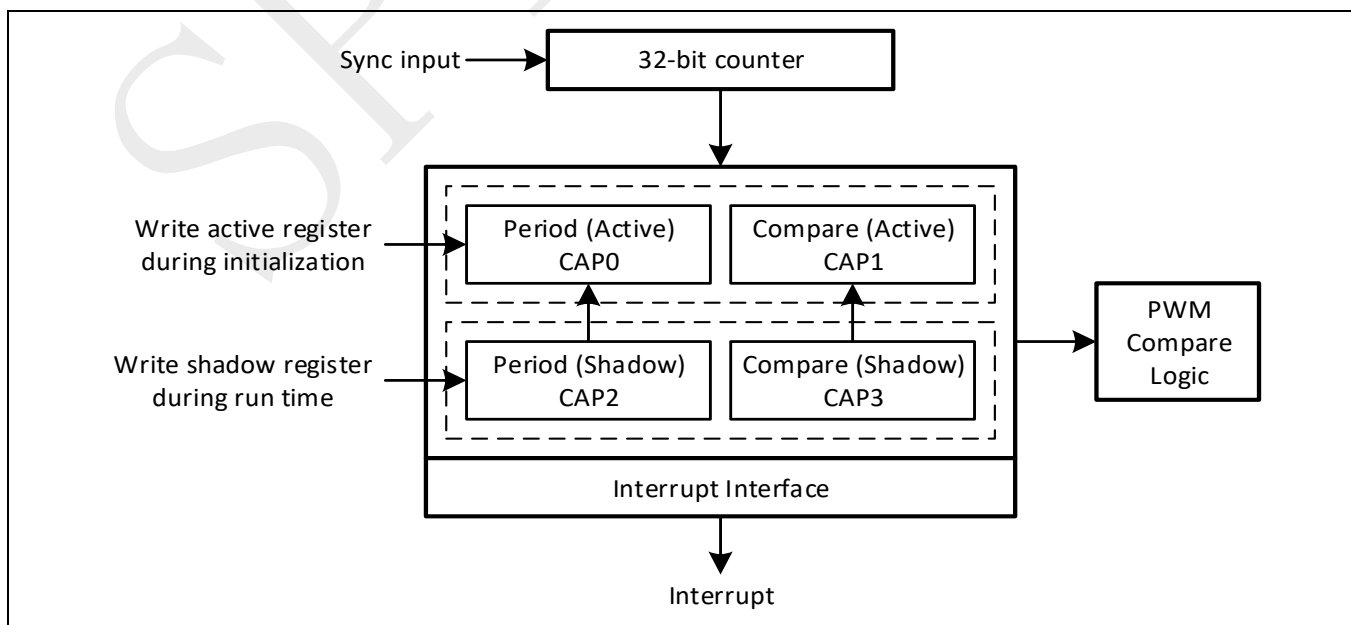
9.5 APWM mode operation

In APWM mode, the 32-bit time-base counter still operates in up-count mode, while the capture registers act as below.

- CAP0 is used as active period (APRD).
- CAP1 is used as active compare (ACMP).
- CAP2 is used as shadow period (SPRD).
- CAP3 is used as shadow compare (SCMP).
- Writing to CAP0 and CAP1 will also update CAP2 and CAP3 with the same value immediately.
- Writing to CAP2 and CAP3 invokes the shadow mode.
 - Load from CAP2 to CAP0 and from CAP3 to CAP1 when TSCNT=CAP0.

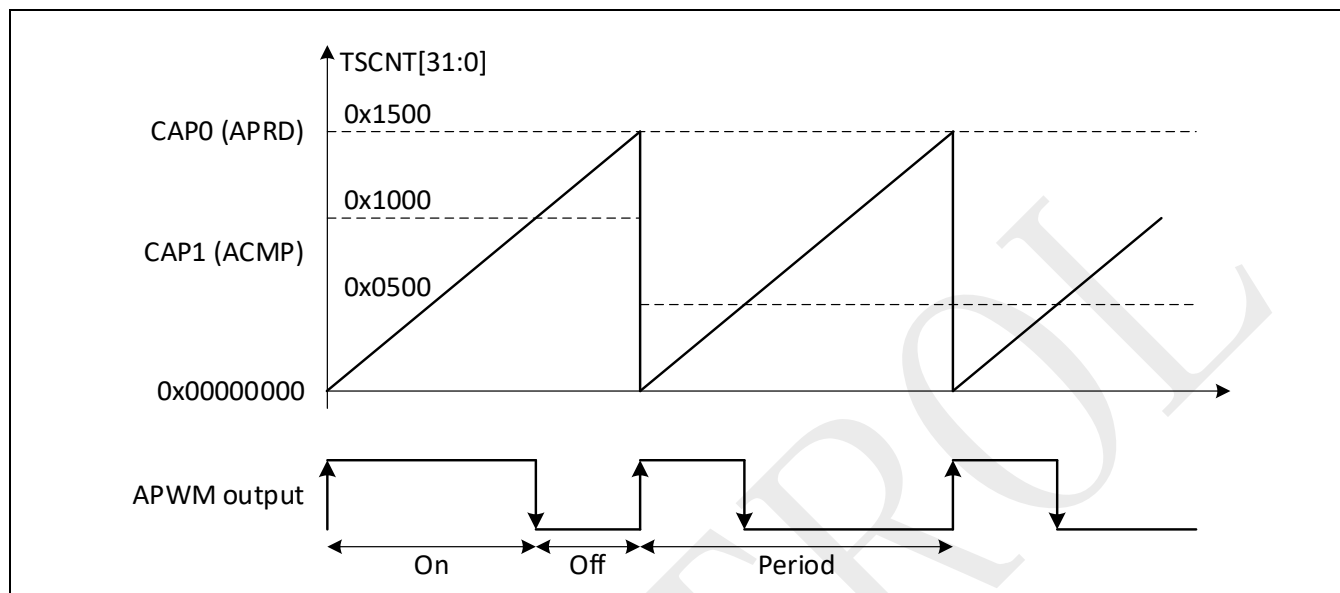
User can write to CAP0 and CAP1 to set the active period and compare threshold during initialization. This automatically copies the initial values into the shadow values. For subsequent updates during run-time, user only needs to use the shadow registers.

Figure 9-9: Active and shadow register in APWM mode



As shown in Figure 9-10, the TSCNT continuously counts from 0 to CAP0 and wraps back to 0. The APWM output toggles to high upon TSCNT=0, and toggles to low upon TSCNT=CAP1.

Figure 9-10: ECAP module APWM mode output waveform (active high)



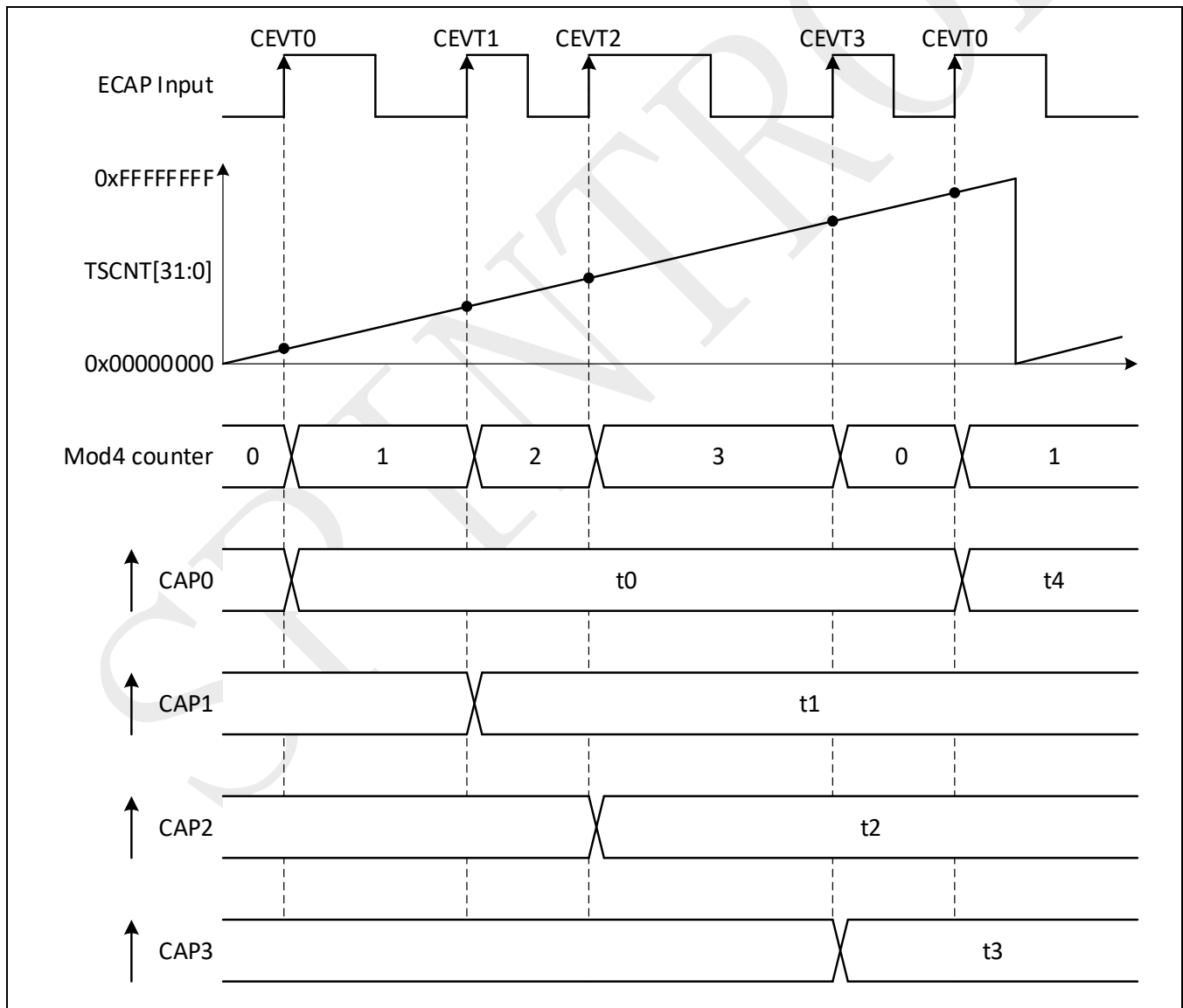
9.6 Application of the ECAP module

9.6.1 Absolute time-stamp for rising edge

Figure 9-11 shows an example of continuous capture operation (Mod4 counter wraps around). In this figure, TSCNT counts-up without resetting on event and capture events are qualified on the rising edge only, this gives period (and frequency) information.

Upon an event, the TSCNT contents (i.e., time-stamp) is captured, then Mod4 counter increases to the next state. When the TSCNT reaches 0xFFFFFFFF (i.e. maximum value), it wraps around to 0x00000000, if this occurs, the CNTOVF (counter overflow) flag is set, and an interrupt (if enabled) occurs. Captured time-stamps are all valid at the point after the 4th event, hence event CEVT3 can be used to trigger an interrupt and the CPU can read data from the CAPx registers.

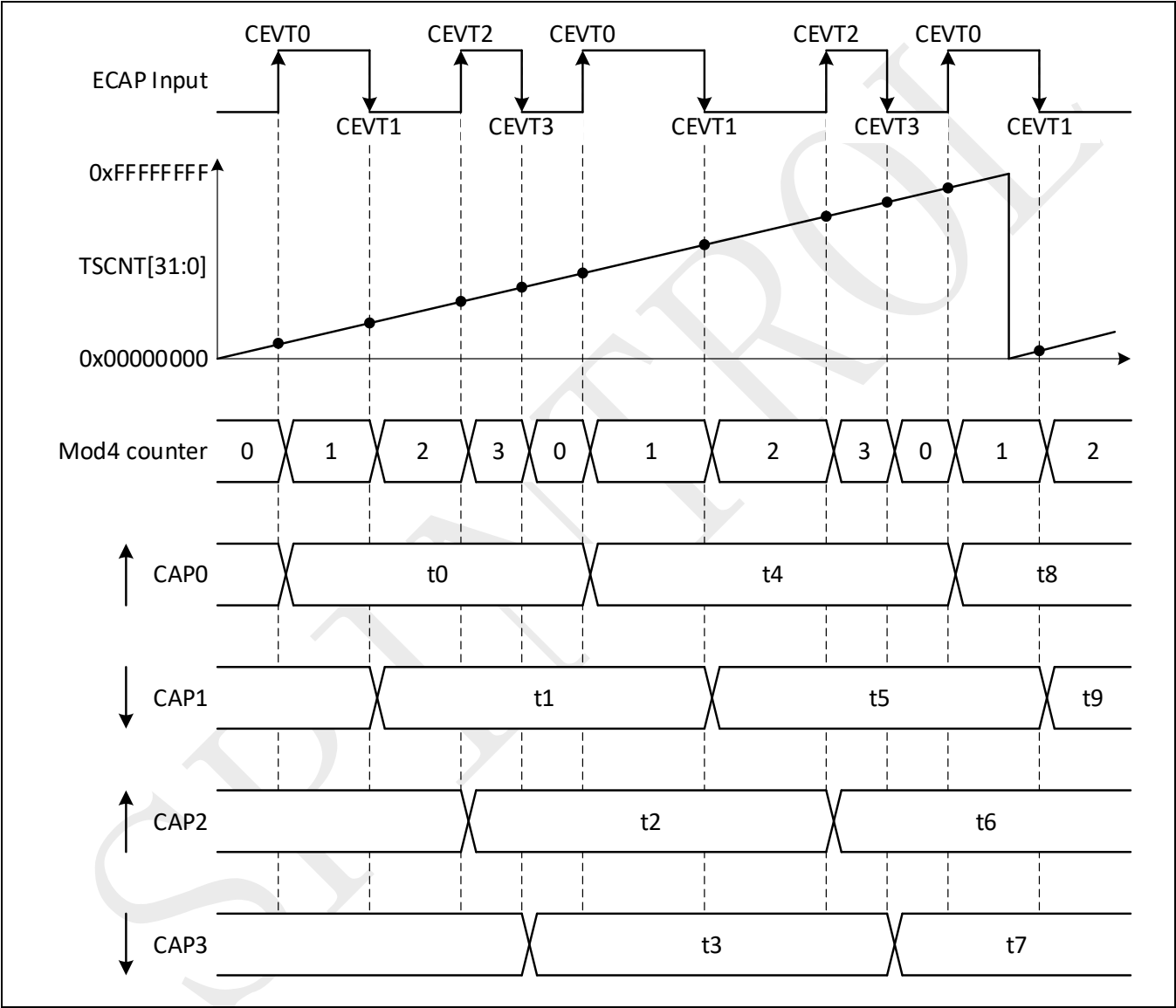
Figure 9-11: Capture sequence for absolute time-stamp and rising edge detect



9.6.2 Absolute time-stamp operation for duty and period

The ECAP operating mode shown in [Figure 9-12](#) is almost the same as in the previous section except that capture events are qualified on both the rising and falling edge. It gives both period and duty cycle information, i.e: $\text{Period1} = t_2 - t_0$, $\text{Period2} = t_4 - t_2$, etc. $\text{Duty Cycle1 (on-time \%)} = (t_1 - t_0) / \text{Period1} \times 100\%$, etc. $\text{Duty Cycle1 (off-time \%)} = (t_2 - t_1) / \text{Period1} \times 100\%$, etc.

Figure 9-12: Capture sequence for period and duty without counter reset

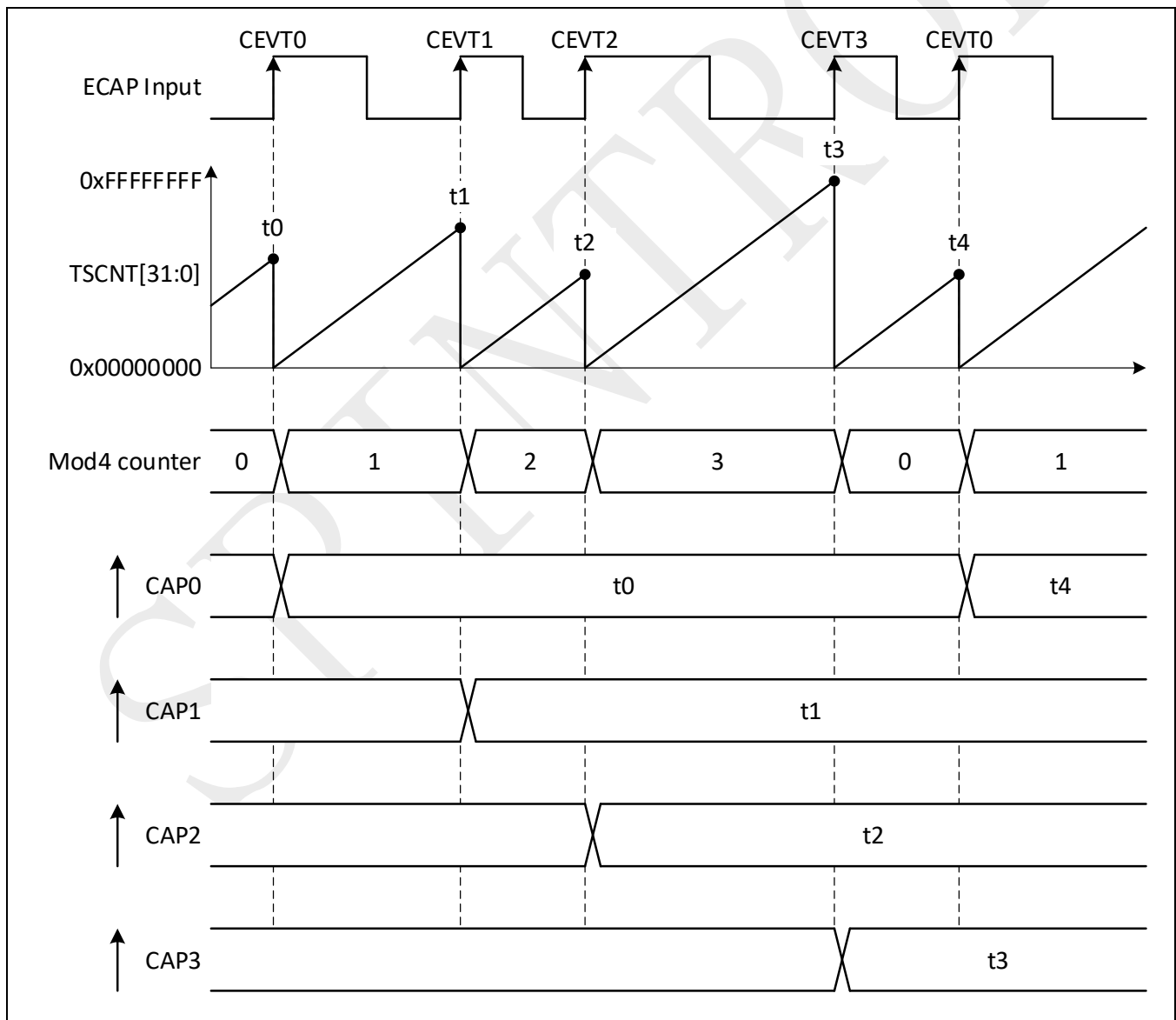


9.6.3 Time difference (Delta) operation with counter reset

Figure 9-13 shows how the ECAP module can be used to collect delta timing data from the waveform. Here continuous capture mode (Mod4 counter wraps around) is used. In delta-time mode, TSCNT is reset back to zero on every valid event. Here capture events are qualified on rising edge only. Upon the valid event, TSCNT contents (i.e. time-stamp) is captured first, and then TSCNT is reset to zero. Meanwhile, the Mod4 counter increases to the next state.

If TSCNT reaches 0xFFFFFFFF (i.e. Max value) before the next event, it wraps around to 0x00000000 and continues, a CntOVF (counter overflow) flag is set, and an interrupt (if enabled) occurs. The advantage of delta-time mode is that the CAPx contents directly give timing information without the need of CPU calculations, i.e. Period1 = t_1 , Period2 = t_2 , etc. As shown in the figure, the CEVT0 event is a good point to read the timing data since t_1 , t_2 , t_3 , t_4 are all valid here.

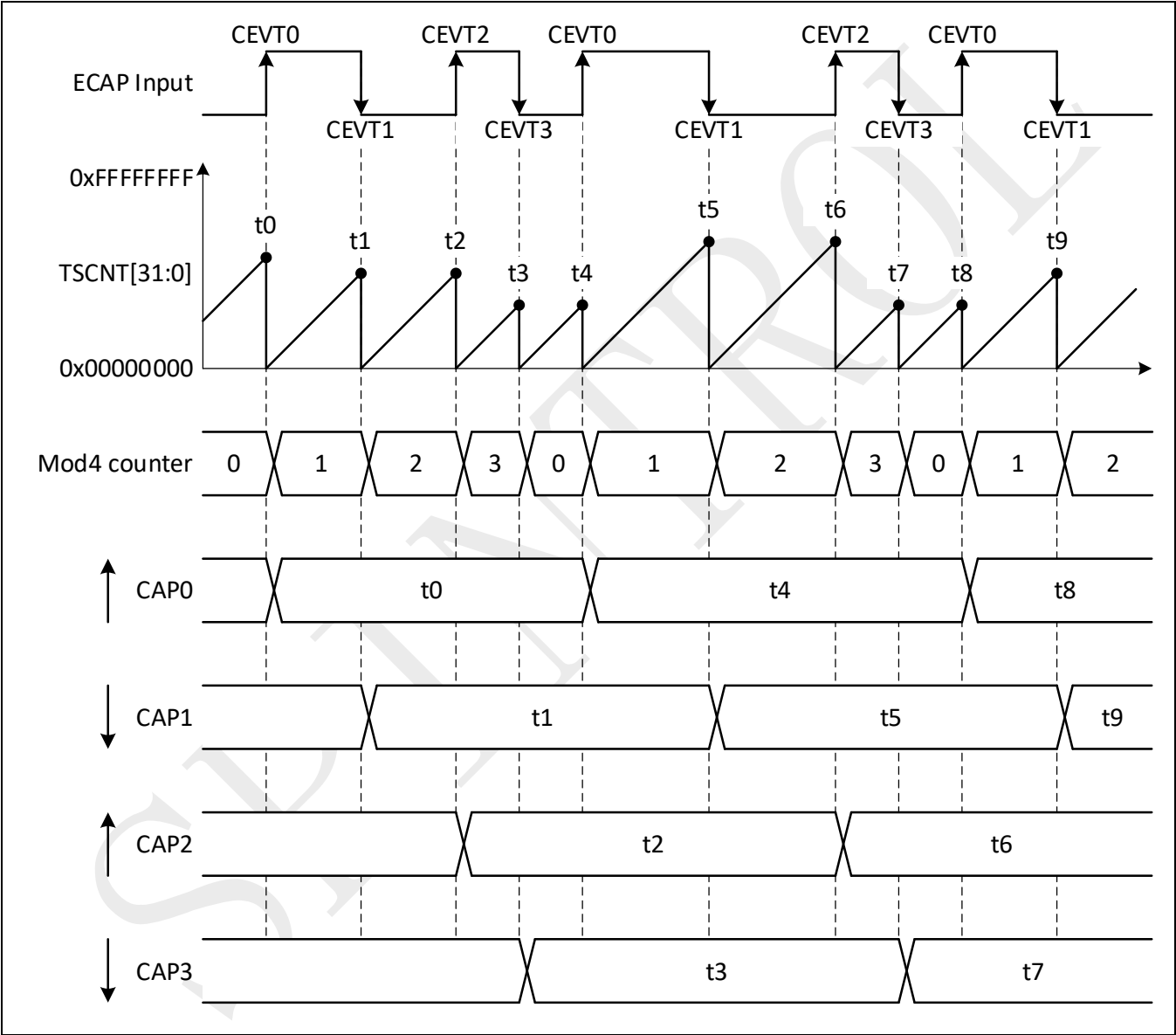
Figure 9-13: Capture period with counter reset



9.6.4 Time difference operation for duty and period

In Figure 9-14 the ECAP operating mode is almost the same as in previous section except the capture events are qualified on both rising or falling edge, this now gives both period and duty cycle information, i.e: $\text{Period1} = t_1 + t_2$, $\text{Period2} = t_3 + t_4$, etc. $\text{Duty Cycle1 (on-time \%)} = t_1 / \text{Period1} \times 100\%$, etc. $\text{Duty Cycle1 (off-time \%)} = t_2 / \text{Period1} \times 100\%$, etc.

Figure 9-14: Capture duty and period with counter reset

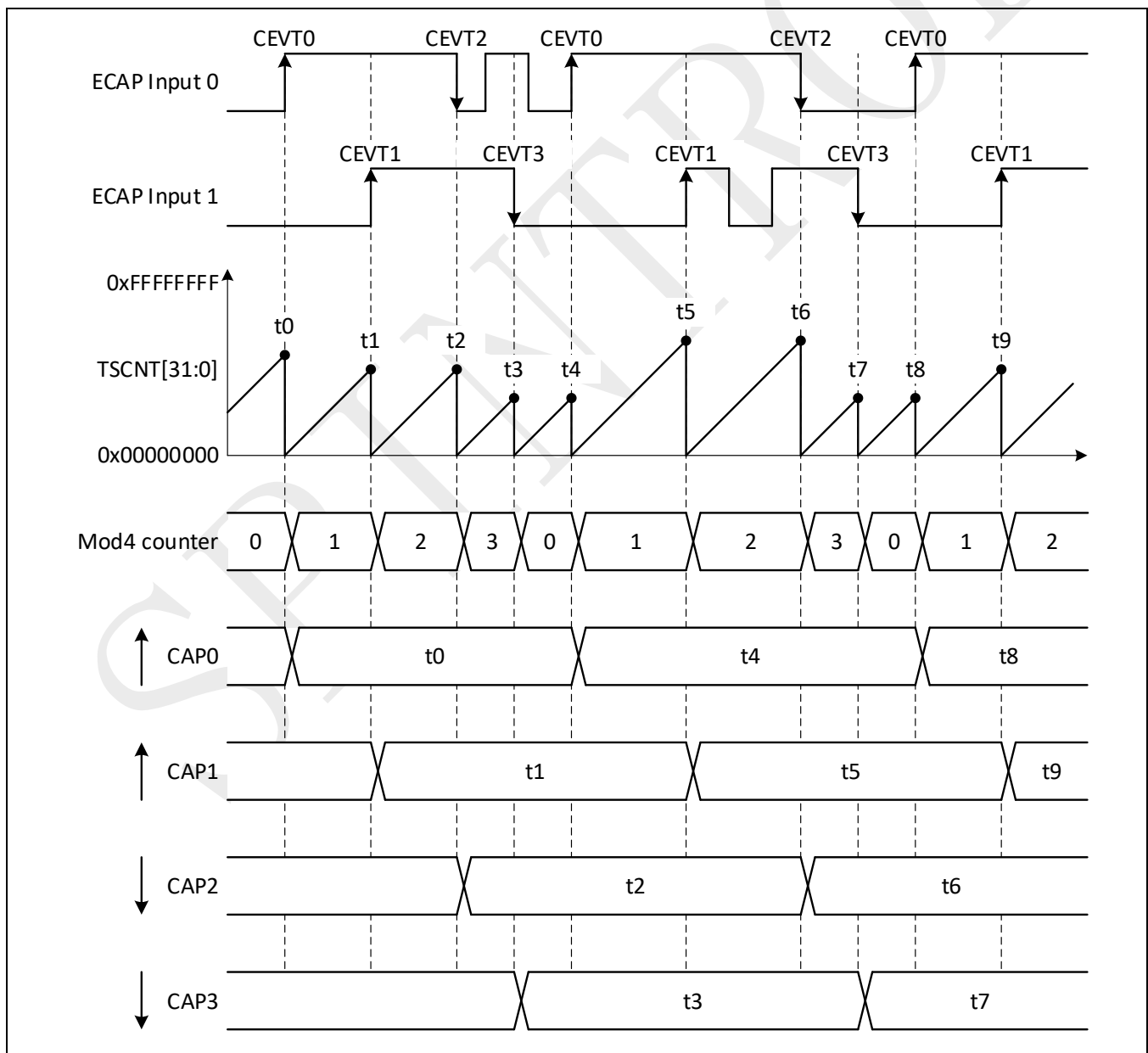


9.6.5 Dual pin capture with counter reset

Figure 9-15 shows the dual pin capture waveform, where CAP0 and CAP2 are used to capture the rising edge and falling edge of pin 0 input, and CAP1 and CAP3 are used to capture the rising edge and falling edge of Pin 1 input. Here delta-time is used. As shown in the figure:

- The timing difference between **adjacent** rising edges of the two pins is captured as t1 in CAP1.
- The timing difference between the **adjacent** rising edge of Pin 1 and falling edge of Pin 0 is captured as t2 in CAP2.
- The timing difference between the **adjacent** falling edge of the two pins is captured as t3 in CAP3.
- The timing difference between the **adjacent** falling edge of Pin 1 and rising edge of Pin 0 is captured as t4 in CAP0.

Figure 9-15: Dual pin capture with counter reset



9.7 Registers

9.7.1 ECAP register map

Table 9-1: ECAP Module Base Address

Peripheral Module	Base Address
ECAP	0x4000 3400

Table 9-2: ECAP Register Map

Register	Offset	Description	Reset Value
<u>CAPSRCCTL</u> *	0x00	Capture Source Input Control Register	0x00010001
<u>CAPSYNCICTL</u> *	0x04	Capture Synchronization Source Input Control Register	0x00000001
<u>CAPTSCNT</u> *	0x08	Time-Stamp Counter Register	0x00000000
<u>CAPCNTPHS</u> *	0x0C	Counter Phase Offset Value Register	0x00000000
<u>CAP0</u> *	0x10	Capture Register 0	0x00000000
<u>CAP1</u> *	0x14	Capture Register 1	0x00000000
<u>CAP2</u> *	0x18	Capture Register 2	0x00000000
<u>CAP3</u> *	0x1C	Capture Register 3	0x00000000
<u>CAPCTL</u> *	0x20	Capture Control Register	0x000C0000
<u>CAPRAWIF</u>	0x24	Capture Interrupt Raw Flag Register	0x00000000
<u>CAPIF</u>	0x28	Capture Interrupt Flag Register	0x00000000
<u>CAPIE</u> *	0x2C	Capture Interrupt Enable Register	0x00000000
<u>CAPIC</u>	0x30	Capture Interrupt Clear Register	0x00000000
<u>CAPIFRC</u> *	0x34	Capture Interrupt Event Force Register	0x00000000
<u>CAPREGKEY</u>	0x38	Capture Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the CAPREGKEY=0x1ACCE551.

9.7.2 ECAP registers

Table 9-3: Capture Source Input Control Register (CAPSRCCTL) Layout

CAPSRCCTL (Capture Source Input Control Register)							
Access: ECAP->CAPSRCCTL		Offset: 0x00		Default: 0x00010001			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
IOSEL1							POL1
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
IOSEL							POL

Table 9-4: Capture Source Input Control Register (CAPSRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:24	RESERVED_31_24	RO	0x0	Reserved.
23:17	IOSEL1	RW	0x0	GPIO number to be selected as 2 nd capture input in dual pin mode
16	POL1	RW	0x1	Source input polarity for the 2 nd pin in dual pin mode 0: Active low 1: Active high
15:8	RESERVED_15_8	RO	0x0	Reserved.
7:1	IOSEL	RW	0x0	GPIO number to be selected as capture input
0	POL	RW	0x1	Source input polarity 0: Active low 1: Active high

Table 9-5: Capture Synchronization Source Input Control Register (CAPSYNCICTL) Layout

CAPSYNCICTL (Capture Synchronization Source Input Control Register)							
Access: ECAP->CAPSYNCICTL		Offset: 0x04		Default: 0x00000001			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
IOSEL							POL

Table 9-6: Capture Synchronization Source Input Control Register (CAPSYNCICTL) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:1	IOSEL	RW	0x0	GPIO number to be selected as sync input
0	POL	RW	0x1	Source input polarity 0: Active low 1: Active high

Table 9-7: Time-Stamp Counter Register (CAPTSCNT) Layout

CAPTSCNT (Time-Stamp Counter Register)							
Access: ECAP->CAPTSCNT		Offset: 0x08		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 9-8: Time-Stamp Counter Register (CAPTSCNT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Active 32-bit counter register that is used as the capture time-base.

Table 9-9: Counter Phase Offset Value Register (CAPCNTPHS) Layout

CAPCNTPHS (Counter Phase Offset Value Register) Offset: 0xC Default: 0x00000000 Access: ECAP->CAPCNTPHS							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 9-10: Counter Phase Offset Value Register (CAPCNTPHS) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Counter phase value register that can be programmed for phase lag/lead.

Table 9-11: Capture Register 0 (CAP0) Layout

CAP0 (Capture Register 0) Offset: 0x10 Default: 0x00000000 Access: ECAP->CAP0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 9-12: Capture Register 0 (CAP0) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	In capture mode -This register is loaded from time-stamp counter (CAPTSCNT) upon a capture event 0. In APWM mode -Writing to this register set the active value of PRD, and also update CAP2 with the same value. This emulates the immediate mode. - Upon CAPTSCNT=PRD (CAP0), this register is updated as CAP2. This emulates the shadow mode.

Table 9-13: Capture Register 1 (CAP1) Layout

CAP1 (Capture Register 1) Offset: 0x14 Default: 0x00000000 Access: ECAP->CAP1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 9-14: Capture Register 1 (CAP1) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	In capture mode -This register is loaded from time-stamp counter (CAPTSCNT) upon a capture event 1. In APWM mode -Writing to this register set the active value of CMP, and also update CAP3 with the same value. This emulates the immediate mode. - Upon CAPTSCNT=PRD (CAP0), this register is updated as CAP3. This emulates the shadow mode.

Table 9-15: Capture Register 2 (CAP2) Layout

CAP2 (Capture Register 2) Offset: 0x18 Default: 0x00000000 Access: ECAP->CAP2							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 9-16: Capture Register 2 (CAP2) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	In capture mode -This register is loaded from time-stamp counter (CAPTSCNT) upon a capture event 2. In APWM mode -Writing to this register set the shadow value of PRD. Upon CAPTSCNT=PRD (CAP0), CAP0 will be updated to the value of this register. This emulates the shadow mode.

Table 9-17: Capture Register 3 (CAP3) Layout

CAP3 (Capture Register 3) Offset: 0x1C Default: 0x00000000 Access: ECAP->CAP3							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 9-18: Capture Register 3 (CAP3) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	In capture mode - This register is loaded from time-stamp counter (CAPTSCNT) upon a capture event 3. In APWM mode - Writing to this register set the shadow value of CMP. Upon CAPTSCNT=PRD (CAP0), CAP1 will be updated to the value of this register. This emulates the shadow mode.

Table 9-19: Capture Control Register (CAPCTL) Layout

CAPCTL (Capture Control Register) Offset: 0x20 Default: 0x000C0000 Access: ECAP->CAPCTL							
31 DUALPIN	30 EVTDIVRST	29 DBGRUN	28 DBGRUN	27 APWMPOL	26 APWMMODE	25 FRCSYNC	24 SYNCOSSEL
23 SYNCOSSEL	22 PHSEN	21 TSCNTRUN	20 REARM	19 STOPWRAP	18 STOPWRAP	17 ONESHOT	16 EVTDIV
15 EVTDIV	14 EVTDIV	13 EVTDIV	12 EVTDIV	11 EVTDIV	10 EVTDIV	9 EVTDIV	8 CAPLDEN
7 CNTRST3	6 CAP3POL	5 CNTRST2	4 CAP2POL	3 CNTRST1	2 CAP1POL	1 CNTRST0	0 CAP0POL

Table 9-20: Capture Control Register (CAPCTL) Description

Bits	Field Name	Type	Reset	Description
31	DUALPIN	RW	0x0	Dual pin capture mode 0: CAP0 ~ CAP3 are used for capture the single pin 1: Use CAP0/CAP2 to capture one pin and CAP1/CAP3 to capture the other
30	EVTDIVRST	W1S	0x0	Event divider reset 0: Write 0 has no effect and always reads back 0 1: Write 1 resets the event divider. This bit is self-cleared
29:28	DBGRUN	RW	0x0	ECAP counter behaviour when CPU is halted Note: Debug operation and CPU fault exceptions both can cause CPU halted. 00: Counter stops immediately when CPU is halted 01: Counter runs until 0 when CPU is halted 10: Counter is unaffected when CPU is halted
27	APWMPOL	RW	0x0	APWM output polarity select. This is applicable only in APWM operating mode. 0: Output is active low (i.e., CMP defines duration for output low) 1: Output is active high (i.e., CMP defines duration for output high)
26	APWMMODE	RW	0x0	CAP/APWM operating mode select 0: ECAP module operates in capture mode. This mode forces the following configuration: - Inhibits counter reset via CNT = PRD event - Inhibits shadow load on CAP0 and CAP1 registers - Permits user to enable CAP0-3 register load - CAPx/APWMx pin operates as a capture input

Bits	Field Name	Type	Reset	Description
				1: ECAP module operates in APWM mode. This mode forces the following configuration: - Resets counter on CNT = PRD event (period boundary) - Permits shadow loading on CAP0 and CAP1 registers - Disables loading of time-stamps into CAP0-3 registers - CAPx/APWMx pin operates as a APWM output
25	FRCSYNC	W1S	0x0	Software-forced counter synchronizing. 0: Write 0 has no effect and always reads back 0 1: Write 1 forces a sync event. This bit is self-cleared.
24:23	SYNCOSEL	RW	0x0	Sync-Out Select 01: Disable sync out signal 10: Select sync-in event to be the sync-out signal (pass through) 11: Select CNT = PRD event to be the sync-out signal
22	PHSEN	RW	0x0	Phase synchronization enable 0: Disable synchronization 1: Enable counter (CAPTSCNT) to be loaded from CAPCNTPHS register upon either a SYNCI signal or a S/W force event
21	TSCNTRUN	RW	0x0	Time stamp counter run/stop control 0: Stop counting 1: Free running
20	REARM	W1S	0x0	One-shot re-arming control. 0: Write 0 has no effect and always reads back 0 1: Write 1 forces events below. This bit is self-cleared. - Resets the MOD4 counter to zero - Unfreezes the MOD4 counter - Enables capture register loads - Reset event prescaler
19:18	STOPWRAP	RW	0x3	Stop value for one-shot mode This is the number (between 0-3) of captures allowed to occur before the CAP(0-3) registers are frozen, i.e., capture sequence is stopped. Wrap value for continuous mode. This is the number (between 0-3) of the capture register in which the circular buffer wraps around and starts again. Notes: Following 2 actions occurs when MOD4=STOPWRAP: - MOD4 counter is stopped (frozen)

Bits	Field Name	Type	Reset	Description
				- Capture register loads are inhibited. In one-shot mode, further interrupt events are blocked until re-armed. 00: Stop after capture event 0 in one-shot mode. Wrap after capture event 0 in continuous mode. 01: Stop after capture event 1 in one-shot mode. Wrap after capture event 1 in continuous mode. 10: Stop after capture event 2 in one-shot mode. Wrap after capture event 2 in continuous mode. 11: Stop after capture event 3 in one-shot mode. Wrap after capture event 3 in continuous mode.
17	ONESHOT	RW	0x0	One-shot mode enable (applicable only in capture mode) 0: Operate in continuous mode 1: Operate in one-shot mode
16:9	EVTDIV	RW	0x0	Event Filter prescale select
8	CAPLDEN	RW	0x0	Enable CAP0-3 loading on a capture event. This bit does not affect the generation of CEVT0-3 events 0: Disable 1: Enable
7	CNTRST3	RW	0x0	Counter reset on capture event 3 0: Do not reset counter 1: Reset counter
6	CAP3POL	RW	0x0	Capture event 3 polarity select 0: Triggered on a falling edge 1: Triggered on a rising edge
5	CNTRST2	RW	0x0	Counter reset on capture event 2 0: Do not reset counter 1: Reset counter
4	CAP2POL	RW	0x0	Capture event 2 polarity select 0: Triggered on a falling edge 1: Triggered on a rising edge
3	CNTRST1	RW	0x0	Counter reset on capture event 1 0: Do not reset counter 1: Reset counter
2	CAP1POL	RW	0x0	Capture event 1 polarity select 0: Triggered on a falling edge 1: Triggered on a rising edge

Bits	Field Name	Type	Reset	Description
1	CNTRST0	RW	0x0	Counter reset on capture event 0 0: Do not reset counter 1: Reset counter
0	CAP0POL	RW	0x0	Capture event 0 polarity select 0: Triggered on a falling edge 1: Triggered on a rising edge

Table 9-21: Capture Interrupt Raw Flag Register (CAPRAWIF) Layout

CAPRAWIF (Capture Interrupt Raw Flag Register) Offset: 0x24 Default: 0x00000000 Access: ECAP->CAPRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6 CMP	5 PRD	4 CNTOVF	3 CEVT3	2 CEVT2	1 CEVT1	0 CEVT0

Table 9-22: Capture Interrupt Raw Flag Register (CAPRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	CMP	RO	0x0	CAPTSCNT=CAP1 (CMP) event raw flag. This flag is active only in APWM mode. 0: Event did not occur 1: Event occurred
5	PRD	RO	0x0	CAPTSCNT=CAP0 (PRD) event raw flag. This flag is only active in APWM mode. 0: Event did not occur 1: Event occurred
4	CNTOVF	RO	0x0	Counter overflow (CAPTSCNT=0xFFFFFFFF) event raw flag. This flag is active in CAP and APWM mode. 0: Event did not occur 1: Event occurred
3	CEVT3	RO	0x0	Capture event 3 raw flag. It is only active in CAP mode. 0: Event did not occur 1: Event occurred
2	CEVT2	RO	0x0	Capture event 2 raw flag. It is active only in CAP mode.

Bits	Field Name	Type	Reset	Description
				0: Event did not occur 1: Event occurred
1	CEVT1	RO	0x0	Capture event 1 raw flag. It is only active in CAP mode. 0: Event did not occur 1: Event occurred
0	CEVT0	RO	0x0	Capture event 0 raw flag. It is only active in CAP mode. 0: Event did not occur 1: Event occurred

Table 9-23: Capture Interrupt Flag Register (CAPIF) Layout

CAPIF (Capture Interrupt Flag Register) Offset: 0x28 Default: 0x00000000 Access: ECAP->CAPIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 INT	6 CMP	5 PRD	4 CNTOVF	3 CEVT3	2 CEVT2	1 CEVT1	0 CEVT0

Table 9-24: Capture Interrupt Flag Register (CAPIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	INT	RO	0x0	Global interrupt flag 0: Interrupt is not triggered 1: Interrupt is triggered and issued to the CPU. Further interrupt pulse to the CPU will not be generated until this bit is cleared.
6	CMP	RO	0x0	CAPTSCNT=CAP1 (CMP) event interrupt flag. This flag is active only in APWM mode. 0: Interrupt is not triggered. 1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.
5	PRD	RO	0x0	CAPTSCNT=CAPO (PRD) event interrupt flag. This flag is only active in APWM mode. 0: Interrupt is not triggered.

Bits	Field Name	Type	Reset	Description
				1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.
4	CNTOVF	RO	0x0	Counter overflow (CAPTSCNT=0xFFFFFFFF) event interrupt flag. This flag is active in CAP and APWM mode. 0: Interrupt is not triggered. 1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.
3	CEVT3	RO	0x0	Capture event 3 interrupt flag. This flag is only active in CAP mode. 0: Interrupt is not triggered. 1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.
2	CEVT2	RO	0x0	Capture event 2 interrupt flag. This flag is active only in CAP mode. 0: Interrupt is not triggered. 1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.
1	CEVT1	RO	0x0	Capture event 1 interrupt flag. This flag is only active in CAP mode. 0: Interrupt is not triggered. 1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.
0	CEVT0	RO	0x0	Capture event 0 interrupt flag. This flag is only active in CAP mode. 0: Interrupt is not triggered. 1: Interrupt is triggered. Further event will not trigger interrupt until this bit is cleared.

Table 9-25: Capture Interrupt Enable Register (CAPIE) Layout

CAPIE (Capture Interrupt Enable Register) Offset: 0x2C Default: 0x00000000 Access: ECAP->CAPIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	CMP	PRD	CNTOVF	CEVT3	CEVT2	CEVT1	CEVT0

Table 9-26: Capture Interrupt Enable Register (CAPIE) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	CMP	RW	0x0	CAPTSCNT=CAP1 (CMP) interrupt enable 0: Disable 1: Enable
5	PRD	RW	0x0	CAPTSCNT=CAPO (PRD) interrupt enable 0: Disable 1: Enable
4	CNTOVF	RW	0x0	Counter overflow interrupt enable 0: Disable 1: Enable
3	CEVT3	RW	0x0	Capture event 3 interrupt enable 0: Disable 1: Enable
2	CEVT2	RW	0x0	Capture event 2 interrupt enable 0: Disable 1: Enable
1	CEVT1	RW	0x0	Capture event 1 interrupt enable 0: Disable 1: Enable
0	CEVT0	RW	0x0	Capture event 0 interrupt enable 0: Disable 1: Enable

Table 9-27: Capture Interrupt Clear Register (CAPIC) Layout

CAPIC (Capture Interrupt Clear Register) Offset: 0x30 Default: 0x00000000 Access: ECAP->CAPIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
INT	CMP	PRD	CNTOVF	CEVT3	CEVT2	CEVT1	CEVT0

Table 9-28: Capture Interrupt Clear Register (CAPIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	INT	W1C	0x0	Global interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
6	CMP	W1C	0x0	CAPTSCNT=CAP1 (CMP) interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
5	PRD	W1C	0x0	CAPTSCNT=CAPO (PRD) interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
4	CNTOVF	W1C	0x0	Counter overflow interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
3	CEVT3	W1C	0x0	Capture event 3 interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
2	CEVT2	W1C	0x0	Capture event 2 interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
1	CEVT1	W1C	0x0	Capture event 1 interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
0	CEVT0	W1C	0x0	Capture event 0 interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.

Table 9-29: Capture Interrupt Event Force Register (CAPIFRC) Layout

CAPIFRC (Capture Interrupt Event Force Register)							
Access: ECAP->CAPIFRC Offset: 0x34 Default: 0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED	CMPWE	PRDWE	CNTOVFWE	CEVT3WE	CEVT2WE	CEVT1WE	CEVT0WE
7	6	5	4	3	2	1	0
RESERVED	CMP	PRD	CNTOVF	CEVT3	CEVT2	CEVT1	CEVT0

Table 9-30: Capture Interrupt Event Force Register (CAPIFRC) Description

Bits	Field Name	Type	Reset	Description
31:15	RESERVED_31_15	RO	0x0	Reserved.
14	CMPWE	W1S	0x0	CMP bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
13	PRDWE	W1S	0x0	PRD bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
12	CNTOVFWE	W1S	0x0	CNTOVF bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
11	CEVT3WE	W1S	0x0	CEVT3 bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
10	CEVT2WE	W1S	0x0	CEVT2 bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
9	CEVT1WE	W1S	0x0	CEVT1 bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
8	CEVT0WE	W1S	0x0	CEVT0 bit write enable. This bit is self-cleared. Write 0 has no effect and always reads back 0.
7	RESERVED_7	RO	0x0	Reserved.
6	CMP	W1S	0x0	CAPTSCNT=CAP1 (CMP) event force. Write 1 to both CMPWE bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.
5	PRD	W1S	0x0	CAPTSCNT=CAP0 (PRD) event force. Write 1 to both PRDWE bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.

Bits	Field Name	Type	Reset	Description
4	CNTOVF	W1S	0x0	Counter overflow event force Write 1 to both CNTOVFWE bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.
3	CEVT3	W1S	0x0	Capture event 3 force Write 1 to both CEVT3 bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.
2	CEVT2	W1S	0x0	Capture event 2 force Write 1 to both CEVT2 bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.
1	CEVT1	W1S	0x0	Capture event 1 force Write 1 to both CEVT1 bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.
0	CEVT0	W1S	0x0	Capture event 0 force Write 1 to both CEVT0 bit and this bit forces the event. This bit is self-cleared. Write 0 has no effect and always reads back 0.

Table 9-31: Capture Register Write-Allow Key Register (CAPREGKEY) Layout

CAPREGKEY (Capture Register Write-Allow Key Register)							
Access: ECAP->CAPREGKEY		Offset: 0x38		Default: 0x1ACCE551			
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 9-32: Capture Register Write-Allow Key Register (CAPREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected ECAP registers

10 13-bit ADC

10.1 13-bit ADC overview

The SPC1169 implements a 13-bit successive approximation A/D converter (SAR-ADC). There are up to 9 external analog inputs to the ADC.

This ADC is easy for the user to create a series of conversions from a single trigger. The basic principle of operation is based on the individual channel configuration for SOC (Start-Of-Conversion).

Functions of the ADC module include:

- 13-bit resolution ADC core with one sample-and-hold (S/H)
 - Up to 2.5 Million samples per second (MSPS)
- Analog full scale input: 3.657 V
- 16 channels with independent:
 - Trigger source
 - Input selection
 - Averaging time and sample/convert time
 - DMA hardware handshake interface (Supports only for channel 0 ~ 7)
 - Gain and offset correction
 - Register to store conversion result
 - Interrupt handling
- Multiple trigger sources for SOC
 - Request from software
 - Request from Timer 0/1/2
 - Request from external pin input
 - Request from PWMs, including: PWM0SOCA ~ PWM0SOCC, PWM1SOCA ~ PWM1SOCC, PWM2SOCA ~ PWM2SOCC, PWM3SOCA ~ PWM3SOCC.
 - Request from EOC (End-Of-Conversion) interrupt of another channel
- 3 post-processing units
 - Reference compare with programmable polarity
 - Out of threshold detection
 - Zero-cross (referred to reference) detection

10.2 13-bit ADC architecture

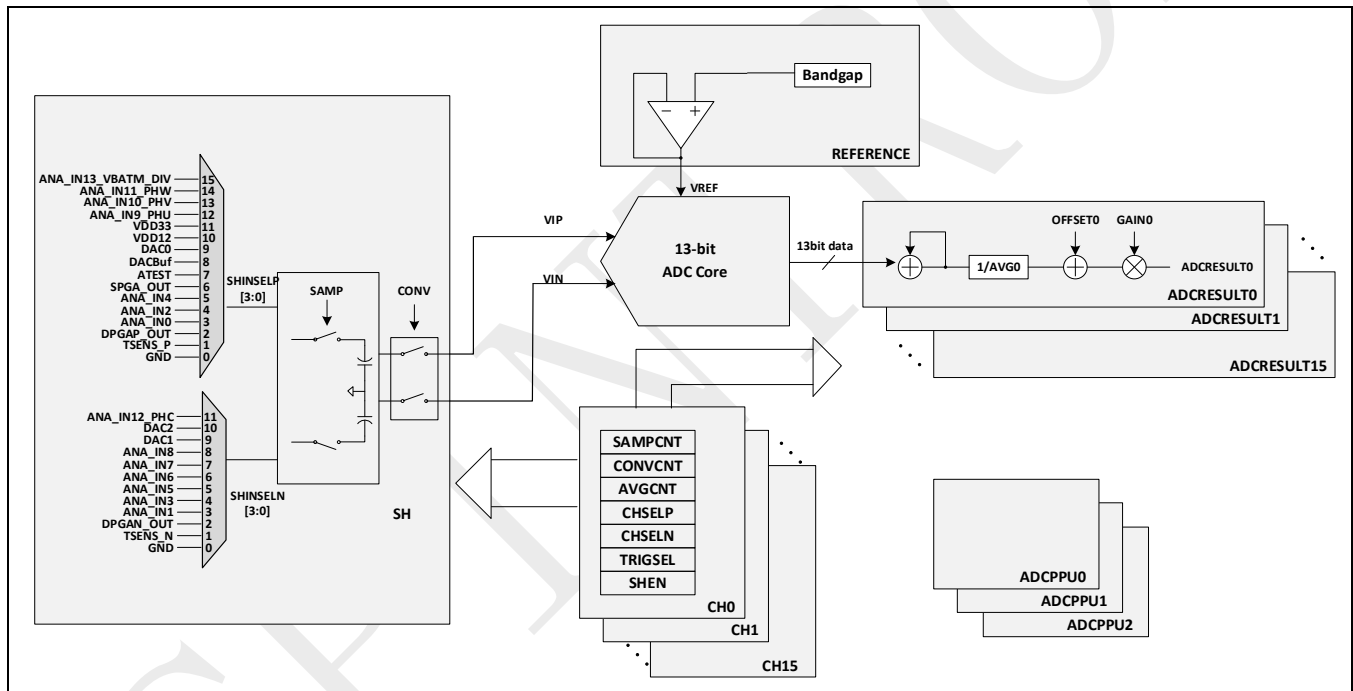
As shown in Figure 10-1, the ADC analog circuit includes:

- 1 ADC sampler (SH0)
- 1 ADC Core
- ADC reference, as well as other analog supporting circuits

The digital circuit includes:

- 16 channel configuration control
- 16 ADC result modules (ADCRESULT) with averaging, gain and offset calibration
- 3 post-processing units (PPU)
- Interface to other on-chip modules (not showed in diagram)

Figure 10-1: ADC block diagram



10.3 Principle of channel based operation

The ADC is channel-based, where a configuration set is used for each channel by ADCCHCTL[x] register to define the corresponding SOC (Start of Conversion) behavior, including:

- The trigger source that starts the conversion defined by ADCCHCTL[x].TRIGSEL
- The input voltage source defined by ADCCHCTL[x].SHINSELP and ADCCHCTL[x].SHINSELN.
- The average times to generate the final result defined by ADCCHCTL[x].AVGCNT
- The conversion window size defined by ADCCHCTL[x].CONVCNT
- The sample window size defined by ADCCHCTL[x].SAMPCNT

Multiple channels can be configured with the same value for the items above as desired. Therefore, it can implemented for scenarios below:

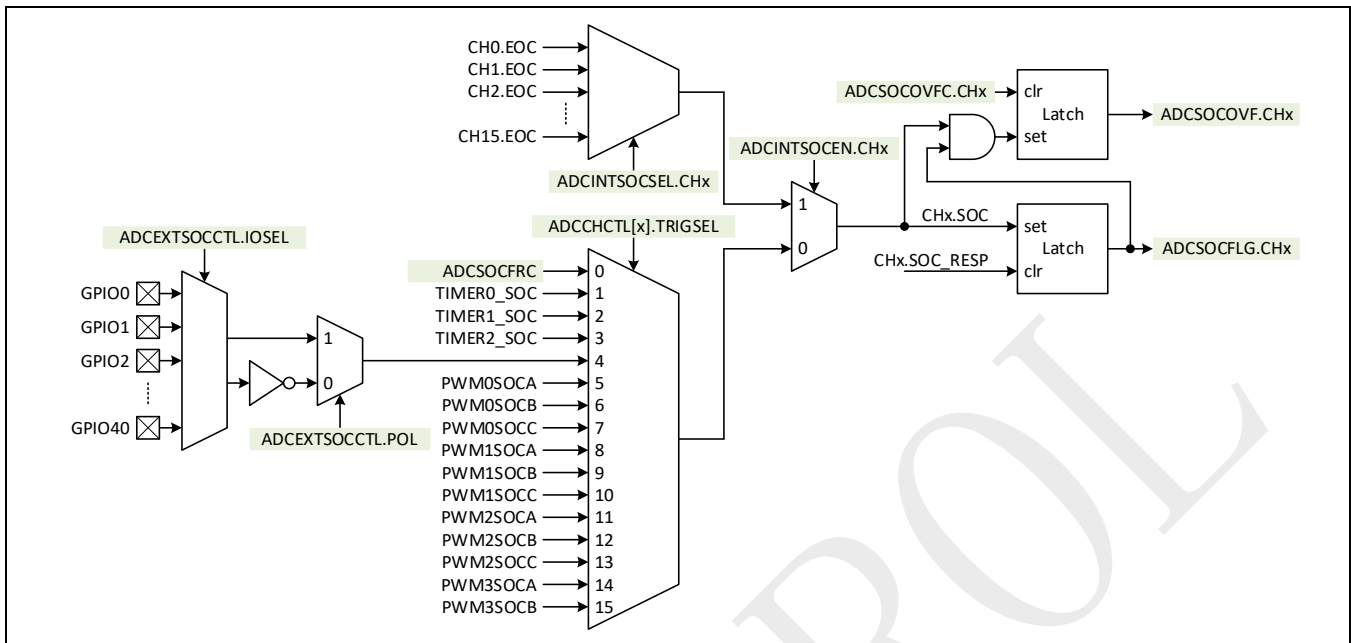
- Individual samples of different inputs with different triggers
- Oversample the same input using a single trigger
- Create a series of conversions of different inputs all from a single trigger.

10.3.1 SOC trigger source

As shown in [Figure 10-2](#), when ADCINTSOCEN.CHx is set as 1, the SOC of channel x will be the EOC (End-Of-Conversion) interrupt request from another channel as selected by ADCINTSOCSEL.CHx. Otherwise, it will be the event as selected by ADCCHCTL[x].TRIGSEL, including

- Request from software
- Request from timer 0/1/2 SOC
- Request from external pin input
- Request from PWMs

Figure 10-2: ADC Channel SOC generation



Corresponding flag bit in ADCSOCFLG register would be set when the SOC is triggered, and cleared when the SOC is responded (i.e. starts to execute). If SOC for channel x is triggered when ADCSOCFLG.CHx is already set, it would be regarded as overflow and latched as ADCSOCOVF.CHx. The overflow flag can be cleared by writing 1 to ADCSOCOVFC.CHx.

10.3.2 Input selection

ADC sampler is enabled by setting ADCCHCTL[x].SHSEL as 1. The detailed voltage inputs for the positive node and the negative node of the sampler are specified as shown in Table 10-1.

Table 10-1: ADC channel input selection

Option	ADCCHCTL[x].SHINSELP	ADCCHCTL[x].SHINSELN
15	ANA_IN13_VBATM_DIV ^[1]	-
14	ANA_IN11_PHW	-
13	ANA_IN10_PHV	-
12	ANA_IN9_PHU	-
11	VDD33	ANA_IN12_PHC
10	VDD12	DAC2
9	DAC0	DAC1
8	DACBUF	ANA_IN8
7	ATEST	ANA_IN7
6	SPGA_OUT	ANA_IN6
5	ANA_IN4	ANA_IN5
4	ANA_IN2	ANA_IN3

Option	ADCCHCTL[x].SHINSELP	ADCCHCTL[x].SHINSELN
3	ANA_IN0	ANA_IN1
2	DPGA_OUTP	DPGA_OUTN
1	TSENS_P	TSENS_N
0	GND	GND

[1] Divided VBATM voltage = VBATM / 13.

10.3.3 Sample and convert window

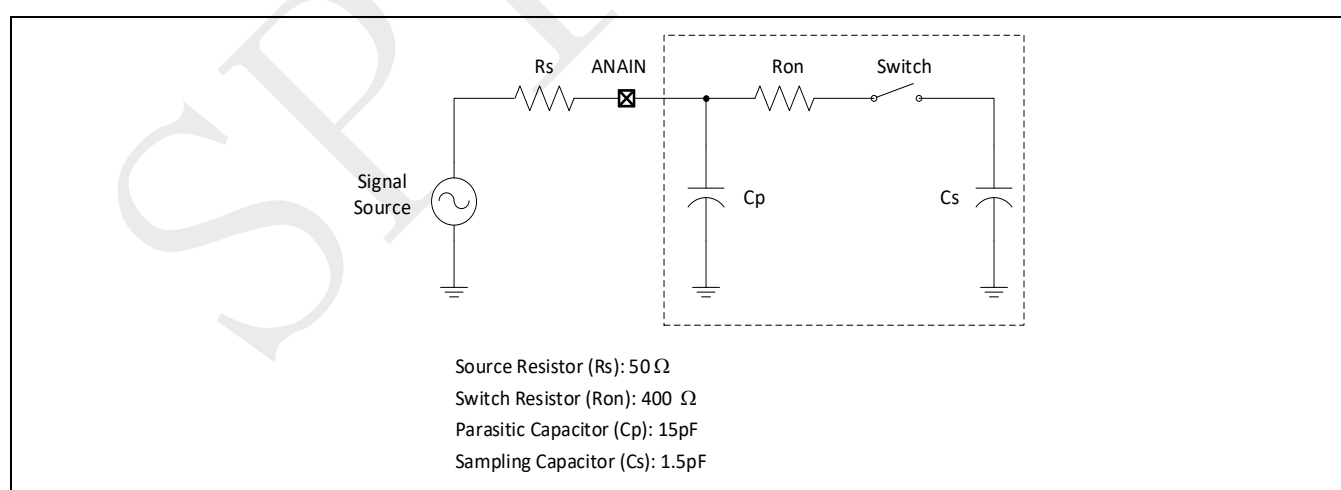
The time required for the sampler to be settled to the expected input voltage varies according to the driving strength of the signal source. Therefore, it is necessary to provide the programmability on the sample and convert time window size. This is achieved by SAMP CNT and CONV CNT in ADCCHCTL[x] register as below:

- Sampling time of channel x is (ADCCHCTL[x].SAMP CNT+1) ADC clock cycles.
- Conversion time of channel x is (ADCCHCTL[x].CONV CNT+1) ADC clock cycles.
- The total time to process analog signal is the sampling time plus the conversion time.

Note: 1. The minimum conversion time is 200ns.
2. The sampling time is recommended to be at least 200ns.

The minimum sampling time is decided by signal source impedance, parasitic resistor and parasitic capacitor in the signal path. As shown in Figure 10-3, the ANAIN pins can be modeled as an RC circuit. A voltage swing from 0 to 3.3 V on ANAIN needs 70ns settling time for 12-bit precision.

Figure 10-3: Analog input pin model

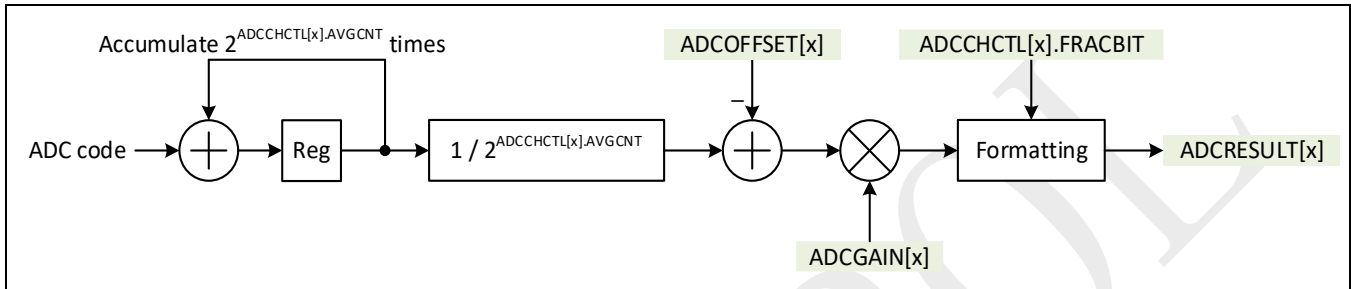


10.3.4 Result average

Sampling system are usually suffered from various noise sources, such as input signal noise, KT/C noise, comparator noise, reference noise, power supply noise and so on. To mitigate its influence, the input

voltage can be sampled and converted multiple times continuously and the code can be accumulated and averaged to get the final result. It would greatly sacrifice the response time and CPU resource if this is implemented by software. Therefore, a hardware average engine is provide as shown in Figure 10-4. A single SOC request leads to continuous sample and conversion for N times, where N can be 1, 2, 4, 8, 16, 32, 64 and 128 as specified by ADCCHCTL[x].AVGCNT. The final result after average can keep the fractional part (7LSBs) or simply truncate it as determined by ADCCHCTL[x].FRACBIT.

Figure 10-4: Averaging control for ADC result



10.3.5 Gain and offset correction

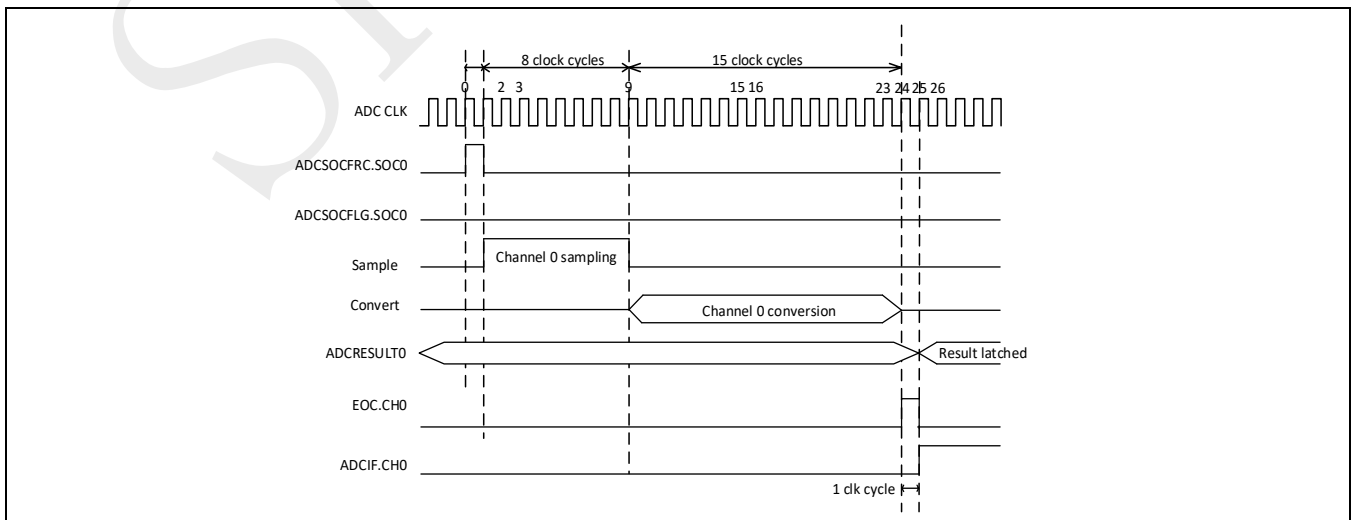
Inherent in any converter, there is an offset error and a full scale gain error. Therefore, hardware gain and offset correction is adopted as shown in Figure 10-4.

During the manufacture test, Spintrol calibrates several ADC settings which are stored in Flash memory. The gain and offset correction parameter would be automatically loaded into SHGAIN and SHOFFSET registers. Users can copy them to the corresponding ADCGAIN[x] and ADCOFFSET[x] for channel x. Users can also do realtime calibration on their own and set independent ADCGAIN[x] and ADCOFFSET[x] for each channel.

10.3.6 Timing diagram

Assuming ADCCHCTL[0].SAMPNT=7 and ADCCHCTL[0].CONVCNT=14, the timing diagram of one-shot sampling and conversion for channel 0 via software triggered SOC is illustrated in Figure 10-3.

Figure 10-5: Timing example



10.4 ADC conversion priority

When multiple SOC requests are pending, priority arbitration is adopted to determine the order in which they are executed.

The default priority arbitration scheme is the round robin. In this scheme, no SOC has an inherent higher priority than another. Priority depends on the round robin pointer (RRPOINTER). The RRPOINTER field of the ADCSOCPRCTL register points to the last SOC executed. The highest priority SOC is assigned to the next value greater than the RRPOINTER value, and wrapped around back to SOC0 after SOC15. The RRPOINTER is reset to 0 by a device reset, or writing 1 to ADCCTL.RST.

An example of the round robin priority arbitration is given in [Figure 10-6](#).

Step1. After reset, SOC0 has the highest priority.

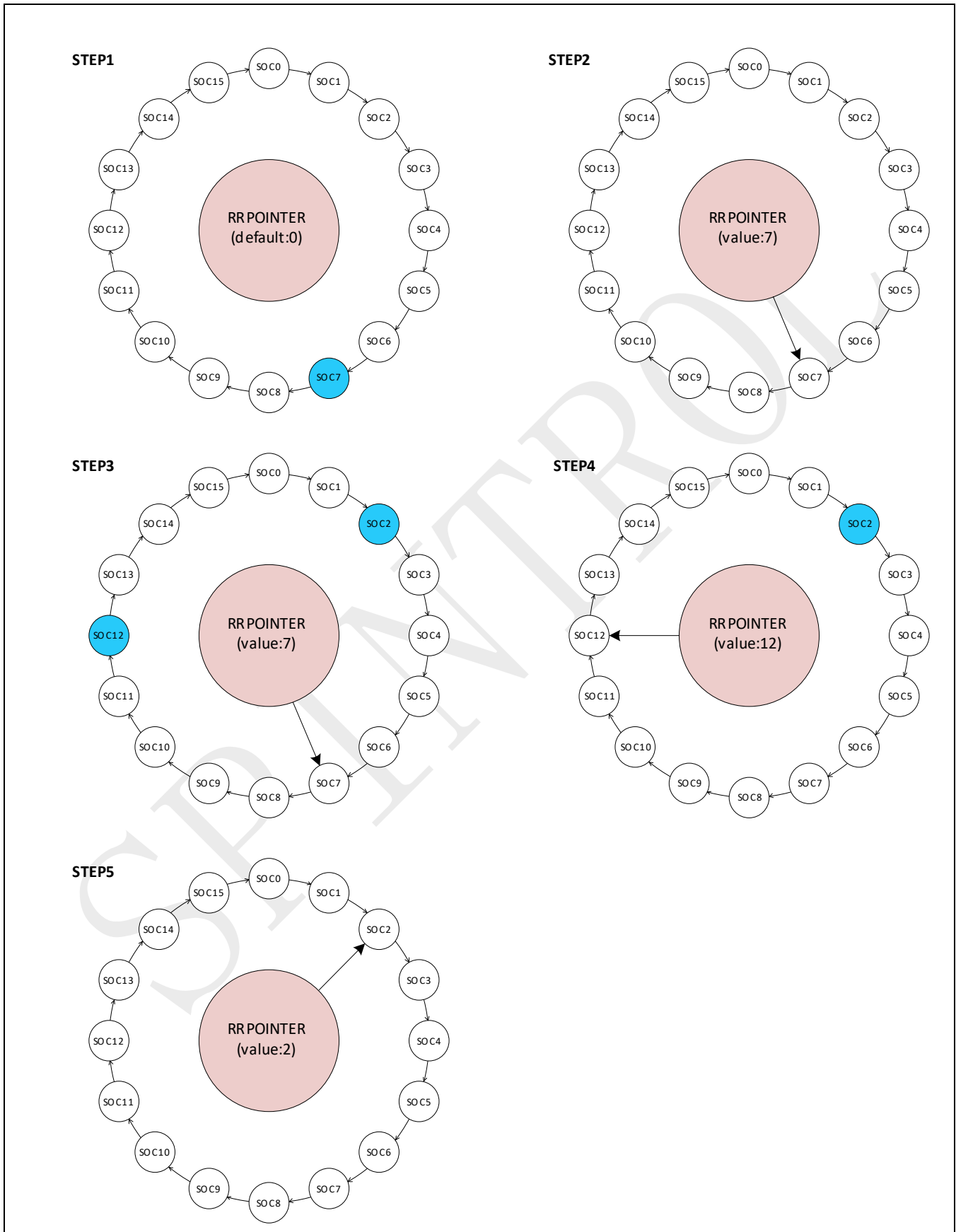
Step2. SOC7 is triggered and responded immediately. RRPOINTER is pointed to SOC7 accordingly and SOC8 has the highest priority now.

Step3. SOC2 & SOC12 are triggered simultaneously.

Step4. Since SOC12 is before SOC2 on round robin wheel, SOC12 is responded first, while SOC2 is pending. RRPOINTER is pointed to SOC12.

Step5. Upon EOC12, SOC2 continues. The RRPOINTER is pointed to SOC2 and SOC3 has the highest priority now.

Figure 10-6: Example for round robin arbitration



The PRIORITY field in the ADCSOCPRCTL register can be used to assign higher priority for a groups of SOC's with the index below PRIORITY. For SOC's within the group, the one with lower index has higher priority. For the SOC's within the other group, round robin arbitration is still applicable. For example, if PRIORITY=4, the priority order from high to low is SOC0, SOC1, SOC2, SOC3, and group SOC4 – SOC15, where SOC4 – SOC15 follows round robin arbitration.

The SOC from the group with higher priority can interrupt the round robin wheel of the other group by the end of current conversion and insert itself as the next one to be responded. When it completes, the round robin wheel will continue where it was interrupted if there is no pending SOC request from the group with higher priority.

An example using high priority SOC's is given in [Figure 10-7](#), where PRIORITY = 4.

Step1. SOC4 is 1st on round robin wheel since PRIORITY is set as 4.

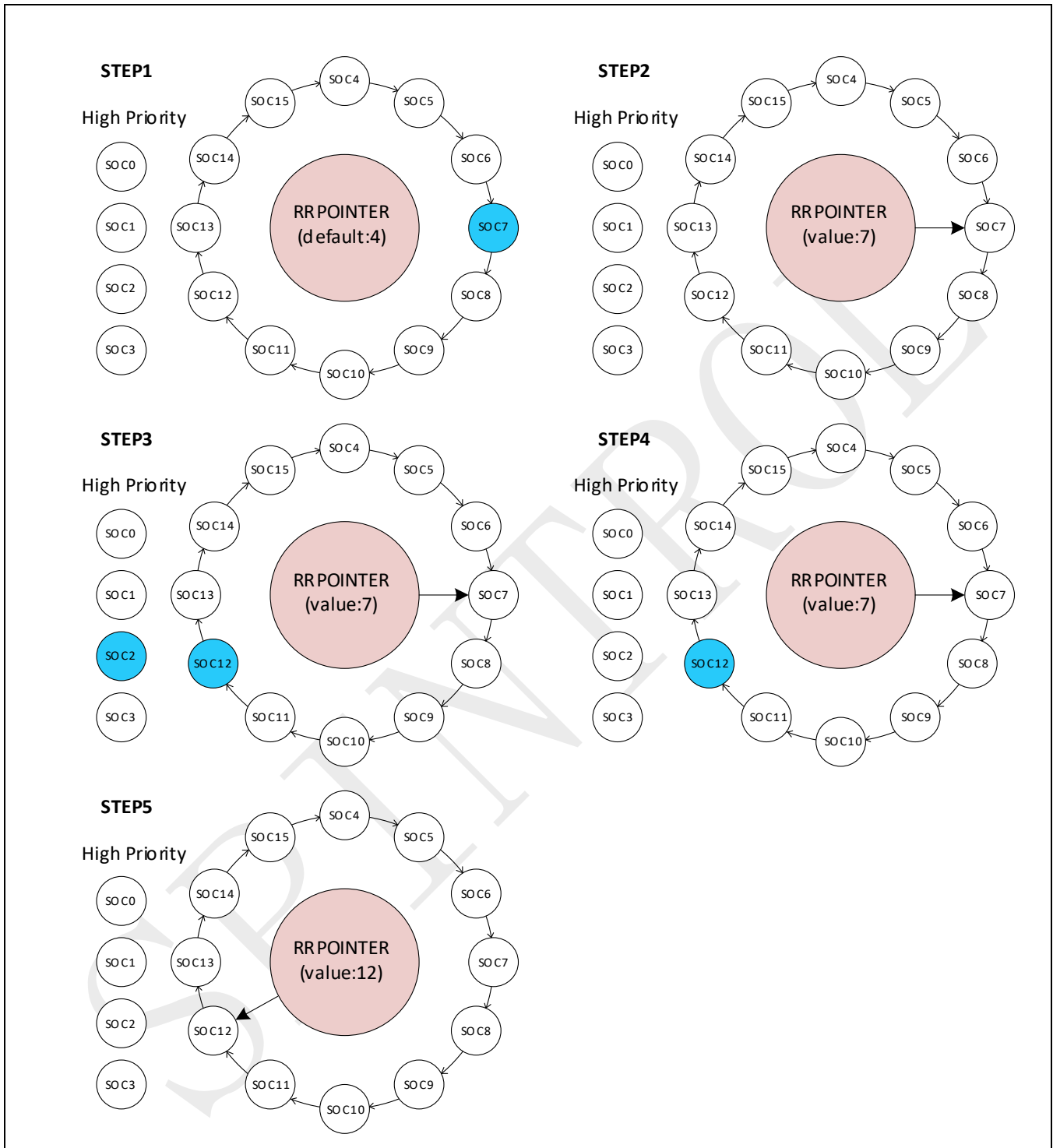
Step2. SOC7 is triggered and responded immediately. RRPOINTER is pointed to SOC7 accordingly and SOC8 has the highest priority on round robin wheel.

Step3. SOC2 & SOC12 are triggered simultaneously.

Step4. SOC2 is responded first since it is in the group with higher priority, while SOC12 is pending. RRPOINTER is still pointed to SOC7.

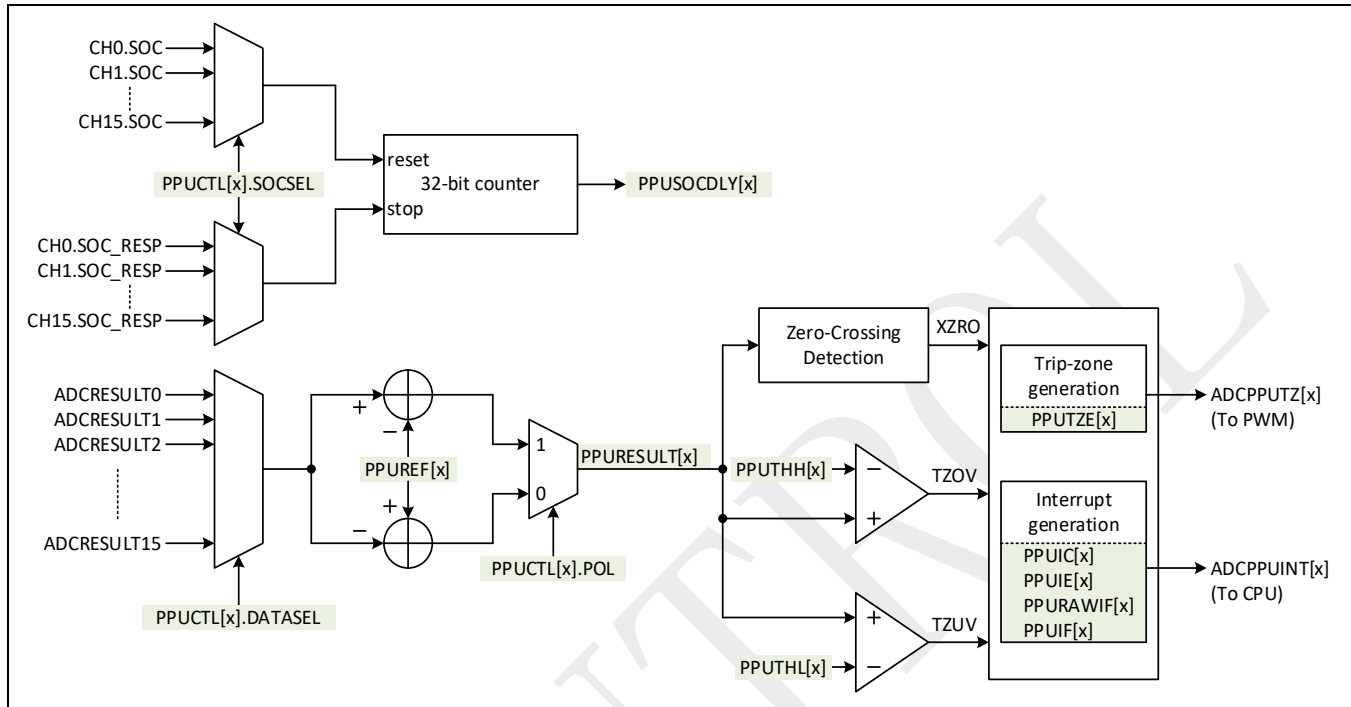
Step5. Upon EOC2, there is no pending SOC's in the group from SOC0 to SOC3. So the pending SOC12 is responded. RRPOINTER is pointed to SOC12. SOC13 is now the next on round robin wheel.

Figure 10-7: Example for group with higher priority



Here, ADCRESULT[y], PPUREF[x] and PPURESULT[x] are all signed numbers. Besides, only the integral part of the ADCRESULT[y] is used regardless of the setting in ADCCHCTL[y].FRACBIT.

Figure 10-9: Post-processing unit topology



10.6.2 Window comparison and zero-cross detection

The PPURESULT[x] would be further compared with a window to check whether the over-voltage or the under-voltage event occurs. The upper boundary of the window is determined by PPUHH[x], while the lower boundary of the window is determined by PPUHL[x].

The zero-cross events of PPURESULT[x] will be also detected, which indicates the corresponding ADCRESULT[y] is crossing the PPUREF[x].

The 3 events above can be individually enabled via PPUTZE[x] register bits to generate the combined trip-zone condition for PWM. They can be also individually enabled via PPUIE[x] register bits to generate the combined interrupt to the CPU.

10.6.3 Sample delay measurement

As discussed in Section, multiple SOC requests may come together but are executed sequentially. Therefore, delay may exist from the time when SOC request is triggered to the time when the input voltage of the corresponding channel is truly sampled and converted. It contributes to the error in a control system. It would be helpful to measure this delay so the software can do some extrapolation and compensation to minimize the error based on it.

As shown in Figure 10-9, this feature is implemented in the PPU with a 32-bit counter. The channel of the SOC request can be independently selected via PPUCTL[x].SOCSEL. Whenever the selected SOC is triggered, the counter starts counting from 0. Whenever the SOC request is responded and the voltage sampling starts, the counter stops and the value is latched to PPUSOCDLY[x]. User can make use of the measured PPUSOCDLY[x] via dedicated algorithm in the software.

10.7 Calculate the expected code

SPC1169 use internal bandgap to generate the reference voltage for the ADC.

The full scale of the ADC quantization is 3.657V, resulting in a differential range from -3.657V ~ 3.657V. However, the actual voltage input for each channel should not exceed the supply voltage (typically 3.3V).

When $ADCCHCTL[x].FRACBIT = 0$, all fractional bits in $ADCRESULT[x]$ would be truncated. Therefore, the code range will be -4096 ~ 4095. For the differential input from AIP and AIN, the expected code would be:

$$ADCRESULT[x] = \frac{(AIP - AIN)}{3.657} \times 4096$$

When $ADCCHCTL[x].FRACBIT = 1$, the 7LSBs for the $ADCRESULT[x]$ is actually the fractional part. Therefore, the nominal code range would be -524288 ~ 524287. For the differential input from AIP and AIN, the expected code would be:

$$ADCRESULT[x] = \frac{(AIP - AIN)}{3.657} \times 524288$$

Result in $SHRAWCODE$ register always keeps the 7LSBs.

10.8 ADC safety feature

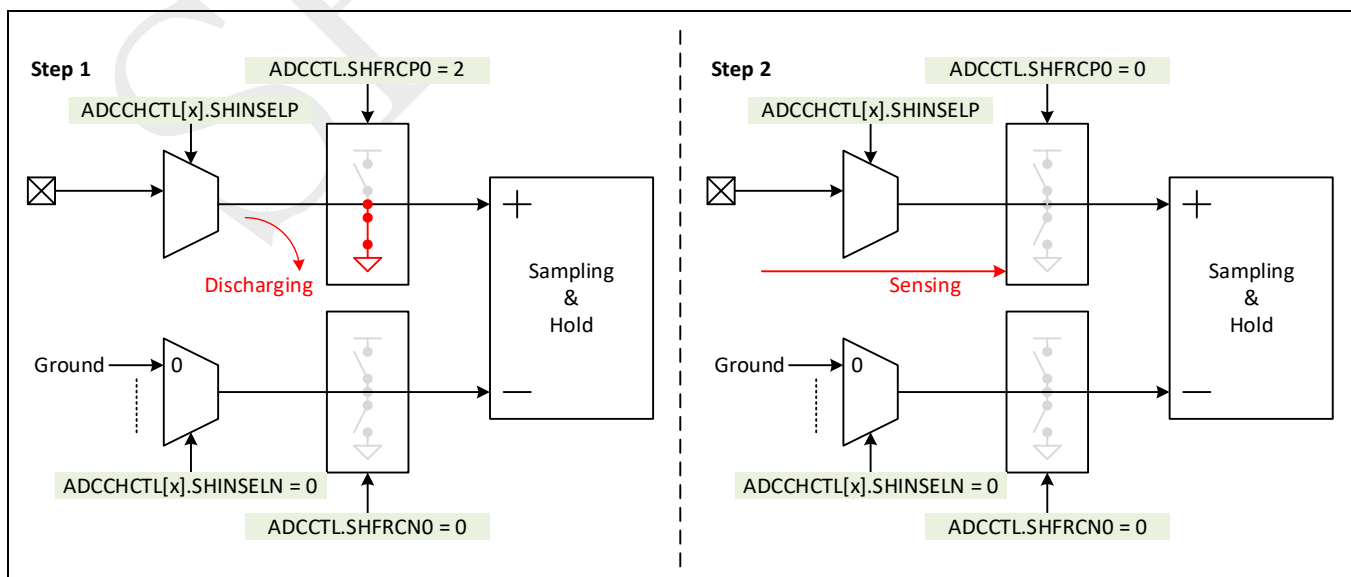
10.8.1 Input floating detection

If the pin for analog input is not reliably connected to the external sensor, the voltage cannot be sensed correctly, leading to potential critical damage in some applications. The ADC input floating detection circuit is provided to figure out such error so as to activate fail-safe procedures.

Two optional schemes are supported to detect the input floating, which are based on fault injection by precharging the node to the supply or discharging the node to the ground respectively.

As shown in [Figure 10-10](#), to detect whether the pin connected to the positive input of the sampler is floating, fault injection by discharging the node can be implemented in two steps.

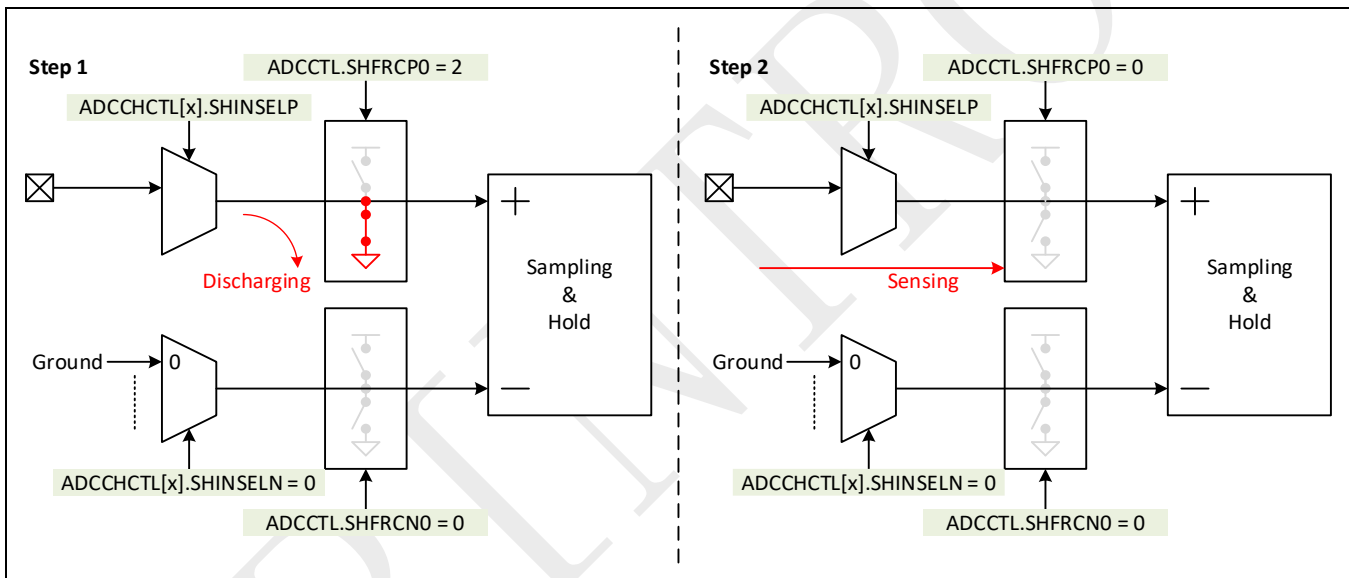
Figure 10-10: Detect the pin floating of ADC input (fault injection by discharging)



- Discharging the input node
 - Connect the positive input of the sampler to the external pin by setting ADCCHCTL[x].SHINSELP.
 - Set ADCCTL.SHFRCP0 as 2 to pull down the positive input of the sampler to the ground.
 - Wait about 10us so that the input node can be fully discharged
- Check the code with normal connection
 - Set ADCCTL.SHFRCP0 as 0 to disable the pull down path.
 - Set ADCCHCTL[x].SHINSELN as 0 to connect the negative input of the sampler to the ground.
 - Trigger SOC for channel x and get the resulting ADCRESULT[x].
 - If the readback is always around 0, it indicates that the input of the sampler does not follow the external input, namely the positive input is floating.

A similar procedure can be adopted to detect whether the pin connected to the negative input of the sampler is floating.

Figure 10-11: Detect the pin floating of ADC input (fault injection by precharging)



As shown in Figure 10-11, to detect whether the pin connected to the positive input of the sampler is floating, fault injection by precharging the node can be implemented in two steps.

- Discharging the input node
 - Connect the positive input of the sampler to the external pin by setting ADCCHCTL[x].SHINSELP.
 - Set ADCCTL.SHFRCP0 as 3 to pull up the positive input of the sampler to the supply.
 - Wait about 10us so that the input node can be fully precharged
- Check the code with normal connection
 - Set ADCCTL.SHFRCP0 as 0 to disable the pull up path.
 - Set ADCCHCTL[x].SHINSELN as 0 to connect the negative input of the sampler to the ground.
 - Trigger SOC for channel x and get the resulting ADCRESULT[x].
 - If the readback is always around 3696 ($3.3 / 3.657 * 4096$), it indicates that the input of the sampler does not follow the external input, namely the positive input is floating.

A similar procedure can be adopted to detect whether pin connected to the negative input of the sampler is floating.

10.8.2 Input short detection

It is possible that the analog input suffers from a short condition. It can be also detected by fault injection using the precharging or discharging circuits in steps below:

- Get the code with normal connection
 - Set ADCCTL.SHFRCP0 and ADCCTL.SHFRCN0 as 0 to disable the pull-up and pull-down to the input of the sampler
 - Select the expected pin and channel via ADCCHCTL[x].SHINSELP and ADCCHCTL[x].SHINSELN
 - Trigger the SOC for channel x and get the 1st ADCRESULT[x].
- Get the code with fault injection by precharging
 - Depending on which node of the sampler is connected to the analog input pin, set ADCCTL.SHFRCP0 or ADCCTL.SHFRCN0 as 3 so as to enable the precharging.
 - Wait about 10us so that the input node can be fully precharged
 - Trigger the SOC for channel x and get the 2nd ADCRESULT[x].
- Get the code with fault injection by charging
 - Depending on which node of the sampler is connected to the analog input pin, set ADCCTL.SHFRCP0 or ADCCTL.SHFRCN0 as 2 so as to enable the discharging.
 - Wait about 10us so that the input node can be fully discharged
 - Trigger the SOC for channel x and get the 3rd ADCRESULT[x].
- The input pin is regarded to be short if the three ADCRESULTs[x] are almost the same.

Note: It can be detected only when the pin is shorted to a low impedance node, whose driving strength is strong enough to override the 8uA precharging and discharging capability provided by the pull-up and pull-down circuits.

10.9 Registers

10.9.1 ADC register map

Table 10-2: ADC Module Base Address

Peripheral Module	Base Address
ADC	0x4000 3000

Table 10-3: ADC Register Map

Register	Offset	Description	Reset Value
ADCIF	0x0	ADC Interrupt Flag Register	0x00000000
ADCIC	0x4	ADC Interrupt Flag Clear Register	0x00000000
ADCIOVF	0x8	ADC Interrupt Overflow Flag Register	0x00000000
ADCIOVFC	0xC	ADC Interrupt Overflow Flag Clear Register	0x00000000
ADCIE *	0x10	ADC Interrupt Enable Register	0x00000000
ADCSOCPRCTL *	0x14	ADC Start of Conversion Priority Control Register	0x00000000
ADCSOCFLG	0x18	ADC SOC Flag Register	0x00000000
ADCSOCFRC	0x1C	ADC SOC Force Register	0x00000000
ADCSOCOVF	0x20	ADC SOC Overflow Register	0x00000000
ADCSOCOVFC	0x24	ADC SOC Overflow Clear Register	0x00000000
ADCINTSOCEN *	0x28	ADC Interrupt Trigger SOC Enable Register	0x00000000
ADCINTSOCSEL *	0x2C	ADC Interrupt Trigger SOC Select Register 0	0x00000000
ADCEXTSOCCTL *	0x34	ADC External SOC Input Control Register	0x00000001
ADCCHCTL *	0x38	ADC Channel n ([0:15]) Control Register	0x00000000
ADCOFFSET *	0x78	ADC Offset Trim Register n ([0:15])	0x00000000
ADCGAIN *	0xB8	ADC Gain Trim Register n ([0:15])	0x00008000
SHOFFSET *	0xF8	ADC SH0 Offset Trim Register	0x00000000
SHGAIN *	0x104	ADC SH0 Gain Trim Register	0x00008000
SHFLG	0x110	ADC Sample Hold Channel EOC Flag Register	0x00000000
SHFLGCLR	0x114	ADC Sample Hold Channel EOC Flag Clear Register	0x00000000
ADCCTL *	0x118	ADC Control Register	0x00000000
SHRAWCODE	0x11C	ADC SH0 Raw Code Register	0x00000000
ADCRESULT	0x128	ADC Result Register n ([0:15])	0x00000000
PPURESULT	0x168	PPU Comparison Result Register n ([0:2])	0x00000000
PPUSOCDLY	0x188	PPU SOC Delay Register n ([0:2])	0x00000000
PPURAWIF	0x1A8	PPU Interrupt Raw Flag Register n ([0:2])	0x00000000
PPUIF	0x1C8	PPU Interrupt Flag Register n ([0:2])	0x00000000
PPUIC	0x1E8	PPU Interrupt Clear Register n ([0:2])	0x00000000
PPUIE *	0x208	PPU Interrupt Enable Register n ([0:2])	0x00000000
PPUTZE *	0x228	PPU Trip-Zone Event Enable Register n ([0:2])	0x00000000
PPUCTL *	0x248	PPU Control Register n ([0:2])	0x00000400
PPUREF *	0x268	PPU Reference Register n ([0:2])	0x00000000

Register	Offset	Description	Reset Value
PPTH_{HL} *	0x288	PPU Trip-Zone Low-Side Threshold Register n ([0:2])	0x00000000
PPTH_{HH} *	0x2A8	PPU Trip-Zone High-Side Threshold Register n ([0:2])	0x00000000
TSENSCTL *	0x2C8	Temperature Sensor Control Register	0x00000000
ADCREGKEY	0x2CC	ADC Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the ADCREGKEY=0x1ACCE551.

10.9.2 ADC registers

Table 10-4: ADC Interrupt Flag Register (ADCIF) Layout

ADCIF (ADC Interrupt Flag Register) Offset: 0x0 Default: 0x00000000 Access: ADCx->ADCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 10-5: ADC Interrupt Flag Register (ADCIF) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH15	RO	0x0	ADC channel 15 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
14	CH14	RO	0x0	ADC channel 14 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
13	CH13	RO	0x0	ADC channel 13 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
12	CH12	RO	0x0	ADC channel 12 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
11	CH11	RO	0x0	ADC channel 11 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
10	CH10	RO	0x0	ADC channel 10 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
9	CH9	RO	0x0	ADC channel 9 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
8	CH8	RO	0x0	ADC channel 8 interrupt flag 0: Interrupt does not happen 1: Interrupt happened

Bits	Field Name	Type	Reset	Description
7	CH7	RO	0x0	ADC channel 7 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
6	CH6	RO	0x0	ADC channel 6 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
5	CH5	RO	0x0	ADC channel 5 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
4	CH4	RO	0x0	ADC channel 4 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
3	CH3	RO	0x0	ADC channel 3 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
2	CH2	RO	0x0	ADC channel 2 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
1	CH1	RO	0x0	ADC channel 1 interrupt flag 0: Interrupt does not happen 1: Interrupt happened
0	CH0	RO	0x0	ADC channel 0 interrupt flag 0: Interrupt does not happen 1: Interrupt happened

Table 10-6: ADC Interrupt Flag Clear Register (ADCIC) Layout

ADCIC (ADC Interrupt Flag Clear Register) Offset: 0x4 Default: 0x00000000 Access: ADCx->ADCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 10-7: ADC Interrupt Flag Clear Register (ADCIC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
15	CH15	W1C	0x0	ADC channel 15 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH15. This bit is self-cleared.
14	CH14	W1C	0x0	ADC channel 14 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH14. This bit is self-cleared.
13	CH13	W1C	0x0	ADC channel 13 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH13. This bit is self-cleared.
12	CH12	W1C	0x0	ADC channel 12 interrupt flag clear 0: Write a 0 has no effect. Always readback 1. 1: Write a 1 clears ADCIF.CH12. This bit is self-cleared.
11	CH11	W1C	0x0	ADC channel 11 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH11. This bit is self-cleared.
10	CH10	W1C	0x0	ADC channel 10 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH10. This bit is self-cleared.
9	CH9	W1C	0x0	ADC channel 9 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH9. This bit is self-cleared.
8	CH8	W1C	0x0	ADC channel 8 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH8. This bit is self-cleared.
7	CH7	W1C	0x0	ADC channel 7 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH7. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
6	CH6	W1C	0x0	ADC channel 6 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH6. This bit is self-cleared.
5	CH5	W1C	0x0	ADC channel 5 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH5. This bit is self-cleared.
4	CH4	W1C	0x0	ADC channel 4 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH4. This bit is self-cleared.
3	CH3	W1C	0x0	ADC channel 3 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH3. This bit is self-cleared.
2	CH2	W1C	0x0	ADC channel 2 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH2. This bit is self-cleared.
1	CH1	W1C	0x0	ADC channel 1 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH1. This bit is self-cleared.
0	CH0	W1C	0x0	ADC channel 0 interrupt flag clear 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears ADCIF.CH0. This bit is self-cleared.

Table 10-8: ADC Interrupt Overflow Flag Register (ADCIOVF) Layout

ADCIOVF (ADC Interrupt Overflow Flag Register) Offset: 0x8 Default: 0x00000000 Access: ADCx->ADCIOVF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 10-9: ADC Interrupt Overflow Flag Register (ADCIOVF) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH15	RO	0x0	ADC channel 15 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
14	CH14	RO	0x0	ADC channel 14 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
13	CH13	RO	0x0	ADC channel 13 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
12	CH12	RO	0x0	ADC channel 12 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
11	CH11	RO	0x0	ADC channel 11 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
10	CH10	RO	0x0	ADC channel 10 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
9	CH9	RO	0x0	ADC channel 9 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
8	CH8	RO	0x0	ADC channel 8 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
7	CH7	RO	0x0	ADC channel 7 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow

Bits	Field Name	Type	Reset	Description
6	CH6	RO	0x0	ADC channel 6 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
5	CH5	RO	0x0	ADC channel 5 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
4	CH4	RO	0x0	ADC channel 4 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
3	CH3	RO	0x0	ADC channel 3 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
2	CH2	RO	0x0	ADC channel 2 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
1	CH1	RO	0x0	ADC channel 1 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow
0	CH0	RO	0x0	ADC channel 0 interrupt overflow flag 0: No overflow 1: ADC channel interrupt overflow

Table 10-10: ADC Interrupt Overflow Flag Clear Register (ADCIOVFC) Layout

ADCIOVFC (ADC Interrupt Overflow Flag Clear Register) Offset: 0xC Default: 0x00000000 Access: ADCx->ADCIOVFC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
CH15	CH14	CH13	CH12	CH11	CH10	CH9	CH8
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 10-11: ADC Interrupt Overflow Flag Clear Register (ADCIOVFC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH15	W1C	0x0	ADC channel 15 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH15. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
14	CH14	W1C	0x0	ADC channel 14 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH14. This bit is self-cleared.
13	CH13	W1C	0x0	ADC channel 13 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH13. This bit is self-cleared.
12	CH12	W1C	0x0	ADC channel 12 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH12. This bit is self-cleared.
11	CH11	W1C	0x0	ADC channel 11 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH11. This bit is self-cleared.
10	CH10	W1C	0x0	ADC channel 10 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH10. This bit is self-cleared.
9	CH9	W1C	0x0	ADC channel 9 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH9. This bit is self-cleared.
8	CH8	W1C	0x0	ADC channel 8 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH8. This bit is self-cleared.
7	CH7	W1C	0x0	ADC channel 7 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH7. This bit is self-cleared.
6	CH6	W1C	0x0	ADC channel 6 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH6. This bit is self-cleared.
5	CH5	W1C	0x0	ADC channel 5 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH5. This bit is self-cleared.
4	CH4	W1C	0x0	ADC channel 4 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH4. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
3	CH3	W1C	0x0	ADC channel 3 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH3. This bit is self-cleared.
2	CH2	W1C	0x0	ADC channel 2 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH2. This bit is self-cleared.
1	CH1	W1C	0x0	ADC channel 1 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH1. This bit is self-cleared.
0	CH0	W1C	0x0	ADC channel 0 interrupt overflow flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the ADCIOVF.CH0. This bit is self-cleared.

Table 10-12: ADC Interrupt Enable Register (ADCIE) Layout

ADCIE (ADC Interrupt Enable Register) Offset: 0x10 Default: 0x00000000 Access: ADCx->ADCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 10-13: ADC Interrupt Enable Register (ADCIE) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH15	RW	0x0	ADC channel 15 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
14	CH14	RW	0x0	ADC channel 14 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
13	CH13	RW	0x0	ADC channel 13 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt

Bits	Field Name	Type	Reset	Description
12	CH12	RW	0x0	ADC channel 12 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
11	CH11	RW	0x0	ADC channel 11 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
10	CH10	RW	0x0	ADC channel 10 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
9	CH9	RW	0x0	ADC channel 9 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
8	CH8	RW	0x0	ADC channel 8 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
7	CH7	RW	0x0	ADC channel 7 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
6	CH6	RW	0x0	ADC channel 6 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
5	CH5	RW	0x0	ADC channel 5 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
4	CH4	RW	0x0	ADC channel 4 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
3	CH3	RW	0x0	ADC channel 3 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
2	CH2	RW	0x0	ADC channel 2 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
1	CH1	RW	0x0	ADC channel 1 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt
0	CH0	RW	0x0	ADC channel 0 interrupt enable 0: Disable ADC channel interrupt 1: Enable ADC channel interrupt

Table 10-14: ADC Start of Conversion Priority Control Register (ADCSOCPRCTL) Layout

ADCSOCPRCTL (ADC Start of Conversion Priority Control Register) Offset: 0x14 Default: 0x00000000 Access: ADCx->ADCSOCPRCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RRPOINTER				PRIORITY			

Table 10-15: ADC Start of Conversion Priority Control Register (ADCSOCPRCTL) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:4	RRPOINTER	RO	0x0	Round Ring Pointer Holds the latest SOCx in Round Ring, and determine the next order of conversion. 0000: SOC0 was the latest SOC. SOC1 is next one in Round Robin. 0001: SOC1 was the latest SOC. SOC2 is next one in Round Robin. 0010: SOC2 was the latest SOC. SOC3 is next one in Round Robin. 0011: SOC3 was the latest SOC. SOC4 is next one in Round Robin. 0100: SOC4 was the latest SOC. SOC5 is next one in Round Robin. 0101: SOC5 was the latest SOC. SOC6 is next one in Round Robin. 0110: SOC6 was the latest SOC. SOC7 is next one in Round Robin. 0111: SOC7 was the latest SOC. SOC8 is next one in Round Robin. 1000: SOC8 was the latest SOC. SOC9 is next one in Round Robin. 1001: SOC9 was the latest SOC. SOC10 is next one in Round Robin. 1010: SOC10 was the latest SOC. SOC11 is next one in Round Robin. 1011: SOC11 was the latest SOC. SOC12 is next one in Round Robin. 1100: SOC12 was the latest SOC. SOC13 is next one in Round Robin.

Bits	Field Name	Type	Reset	Description
				1101: SOC13 was the latest SOC. SOC14 is next one in Round Robin. 1110: SOC14 was the latest SOC. SOC15 is next one in Round Robin. 1111: SOC15 was the latest SOC. SOC0 is next one in Round Robin.
3:0	PRIORITY	RW	0x0	SOC (Start of Conversion) Priority Determines the cutoff point for priority mode and round arbitration of SOC 0000: All SOC priority follow the rule of round mode 0001: SOC0 are high priority, SOC1-SOC15 is in round mode 0010: SOC0-SOC1 are high priority, SOC2-SOC15 is in round mode 0011: SOC0-SOC2 are high priority, SOC3-SOC15 is in round mode 0100: SOC0-SOC3 are high priority, SOC4-SOC15 is in round mode 0101: SOC0-SOC4 are high priority, SOC5-SOC15 is in round mode 0110: SOC0-SOC5 are high priority, SOC6-SOC15 is in round mode 0111: SOC0-SOC6 are high priority, SOC7-SOC15 is in round mode 1000: SOC0-SOC7 are high priority, SOC8-SOC15 is in round mode 1001: SOC0-SOC8 are high priority, SOC9-SOC15 is in round mode 1010: SOC0-SOC9 are high priority, SOC10-SOC15 is in round mode 1011: SOC0-SOC10 are high priority, SOC11-SOC15 is in round mode 1100: SOC0-SOC11 are high priority, SOC12-SOC15 is in round mode 1101: SOC0-SOC12 are high priority, SOC13-SOC15 is in round mode 1110: SOC0-SOC13 are high priority, SOC14-SOC15 is in round mode 1111: SOC0-SOC14 are high priority, SOC15 is in round mode

Table 10-16: ADC SOC Flag Register (ADCSOCFLG) Layout

ADCSOCFLG (ADC SOC Flag Register) Offset: 0x18 Default: 0x00000000 Access: ADCx->ADCSOCFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

Table 10-17: ADC SOC Flag Register (ADCSOCFLG) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	SOC15	RO	0x0	SOC15 (Start of Conversion) flag 0: No sample pending for SOC15 1: Trigger received and pend sample for SOC15. This bit is self-cleared when sample starts.
14	SOC14	RO	0x0	SOC14 (Start of Conversion) flag 0: No sample pending for SOC14 1: Trigger received and pend sample for SOC14. This bit is self-cleared when sample starts.
13	SOC13	RO	0x0	SOC13 (Start of Conversion) flag 0: No sample pending for SOC13 1: Trigger received and pend sample for SOC13. This bit is self-cleared when sample starts.
12	SOC12	RO	0x0	SOC12 (Start of Conversion) flag 0: No sample pending for SOC12 1: Trigger received and pend sample for SOC12. This bit is self-cleared when sample starts.
11	SOC11	RO	0x0	SOC11 (Start of Conversion) flag 0: No sample pending for SOC11 1: Trigger received and pend sample for SOC11. This bit is self-cleared when sample starts.
10	SOC10	RO	0x0	SOC10 (Start of Conversion) flag 0: No sample pending for SOC10 1: Trigger received and pend sample for SOC10. This bit is self-cleared when sample starts.
9	SOC9	RO	0x0	SOC9 (Start of Conversion) flag 0: No sample pending for SOC9 1: Trigger received and pend sample for SOC9. This bit is self-cleared when sample starts.

Bits	Field Name	Type	Reset	Description
8	SOC8	RO	0x0	SOC8 (Start of Conversion) flag 0: No sample pending for SOC8 1: Trigger received and pend sample for SOC8. This bit is self-cleared when sample starts.
7	SOC7	RO	0x0	SOC7 (Start of Conversion) flag 0: No sample pending for SOC7 1: Trigger received and pend sample for SOC7. This bit is self-cleared when sample starts.
6	SOC6	RO	0x0	SOC6 (Start of Conversion) flag 0: No sample pending for SOC6 1: Trigger received and pend sample for SOC6. This bit is self-cleared when sample starts.
5	SOC5	RO	0x0	SOC5 (Start of Conversion) flag 0: No sample pending for SOC5 1: Trigger received and pend sample for SOC5. This bit is self-cleared when sample starts.
4	SOC4	RO	0x0	SOC4 (Start of Conversion) flag 0: No sample pending for SOC4 1: Trigger received and pend sample for SOC4. This bit is self-cleared when sample starts.
3	SOC3	RO	0x0	SOC3 (Start of Conversion) flag 0: No sample pending for SOC3 1: Trigger received and pend sample for SOC3. This bit is self-cleared when sample starts.
2	SOC2	RO	0x0	SOC2 (Start of Conversion) flag 0: No sample pending for SOC2 1: Trigger received and pend sample for SOC2. This bit is self-cleared when sample starts.
1	SOC1	RO	0x0	SOC1 (Start of Conversion) flag 0: No sample pending for SOC1 1: Trigger received and pend sample for SOC1. This bit is self-cleared when sample starts.
0	SOC0	RO	0x0	SOC0 (Start of Conversion) flag 0: No sample pending for SOC0 1: Trigger received and pend sample for SOC0. This bit is self-cleared when sample starts.

Table 10-18: ADC SOC Force Register (ADCSOCFRC) Layout

ADCSOCFRC (ADC SOC Force Register) Offset: 0x1C Default: 0x00000000 Access: ADCx->ADCSOCFRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

Table 10-19: ADC SOC Force Register (ADCSOCFRC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	SOC15	W1S	0x0	SOC15 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC15. This bit is self-cleared.
14	SOC14	W1S	0x0	SOC14 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC14. This bit is self-cleared.
13	SOC13	W1S	0x0	SOC13 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC13. This bit is self-cleared.
12	SOC12	W1S	0x0	SOC12 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC12. This bit is self-cleared.
11	SOC11	W1S	0x0	SOC11 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC11. This bit is self-cleared.
10	SOC10	W1S	0x0	SOC10 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC10. This bit is self-cleared.
9	SOC9	W1S	0x0	SOC9 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC9. This bit is self-cleared.
8	SOC8	W1S	0x0	SOC8 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC8. This bit is self-cleared.
7	SOC7	W1S	0x0	SOC7 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC7. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
6	SOC6	W1S	0x0	SOC6 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC6. This bit is self-cleared.
5	SOC5	W1S	0x0	SOC5 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC5. This bit is self-cleared.
4	SOC4	W1S	0x0	SOC4 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC4. This bit is self-cleared.
3	SOC3	W1S	0x0	SOC3 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC3. This bit is self-cleared.
2	SOC2	W1S	0x0	SOC2 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC2. This bit is self-cleared.
1	SOC1	W1S	0x0	SOC1 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC1. This bit is self-cleared.
0	SOC0	W1S	0x0	SOC0 (Start of Conversion) flag force 0: Write a 0 has no effect and always read back 0 1: Write a 1 forces SOC0. This bit is self-cleared.

Table 10-20: ADC SOC Overflow Register (ADCSOCOVF) Layout

ADCSOCOVF (ADC SOC Overflow Register) Offset: 0x20 Default: 0x00000000 Access: ADCx->ADCSOCOVF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

Table 10-21: ADC SOC Overflow Register (ADCSOCOVF) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	SOC15	RO	0x0	SOC15 (Start of Conversion) overflow flag This indicates an SOC15 interrupt was generated while an existing SOC15 interrupt was already pending. This overflow flag can't stop SOC15 interrupts from being processing. It just show

Bits	Field Name	Type	Reset	Description
				that a trigger was missed. 0: No SOC15 interrupt overflow 1: SOC15 interrupt overflow
14	SOC14	RO	0x0	SOC14 (Start of Conversion) overflow flag This indicates an SOC14 interrupt was generated while an existing SOC14 interrupt was already pending. This overflow flag can't stop SOC14 interrupts from being processing. It just show that a trigger was missed. 0: No SOC14 interrupt overflow 1: SOC14 interrupt overflow
13	SOC13	RO	0x0	SOC13 (Start of Conversion) overflow flag This indicates an SOC13 interrupt was generated while an existing SOC13 interrupt was already pending. This overflow flag can't stop SOC13 interrupts from being processing. It just show that a trigger was missed. 0: No SOC13 interrupt overflow 1: SOC13 interrupt overflow
12	SOC12	RO	0x0	SOC12 (Start of Conversion) overflow flag This indicates an SOC12 interrupt was generated while an existing SOC12 interrupt was already pending. This overflow flag can't stop SOC12 interrupts from being processing. It just show that a trigger was missed. 0: No SOC12 interrupt overflow 1: SOC12 interrupt overflow
11	SOC11	RO	0x0	SOC11 (Start of Conversion) overflow flag This indicates an SOC11 interrupt was generated while an existing SOC11 interrupt was already pending. This overflow flag can't stop SOC11 interrupts from being processing. It just show that a trigger was missed. 0: No SOC11 interrupt overflow 1: SOC11 interrupt overflow
10	SOC10	RO	0x0	SOC10 (Start of Conversion) overflow flag This indicates an SOC10 interrupt was generated while an existing SOC10 interrupt was already pending. This overflow flag can't stop SOC10 interrupts from being processing. It just show that a trigger was missed. 0: No SOC10 interrupt overflow 1: SOC10 interrupt overflow
9	SOC9	RO	0x0	SOC9 (Start of Conversion) overflow flag This indicates an SOC9 interrupt was generated

Bits	Field Name	Type	Reset	Description
				while an existing SOC9 interrupt was already pending. This overflow flag can't stop SOC9 interrupts from being processing. It just show that a trigger was missed. 0: No SOC9 interrupt overflow 1: SOC9 interrupt overflow
8	SOC8	RO	0x0	SOC8 (Start of Conversion) overflow flag This indicates an SOC8 interrupt was generated while an existing SOC8 interrupt was already pending. This overflow flag can't stop SOC8 interrupts from being processing. It just show that a trigger was missed. 0: No SOC8 interrupt overflow 1: SOC8 interrupt overflow
7	SOC7	RO	0x0	SOC7 (Start of Conversion) overflow flag This indicates an SOC7 interrupt was generated while an existing SOC7 interrupt was already pending. This overflow flag can't stop SOC7 interrupts from being processing. It just show that a trigger was missed. 0: No SOC7 interrupt overflow 1: SOC7 interrupt overflow
6	SOC6	RO	0x0	SOC6 (Start of Conversion) overflow flag This indicates an SOC6 interrupt was generated while an existing SOC6 interrupt was already pending. This overflow flag can't stop SOC6 interrupts from being processing. It just show that a trigger was missed. 0: No SOC6 interrupt overflow 1: SOC6 interrupt overflow
5	SOC5	RO	0x0	SOC5 (Start of Conversion) overflow flag This indicates an SOC5 interrupt was generated while an existing SOC5 interrupt was already pending. This overflow flag can't stop SOC5 interrupts from being processing. It just show that a trigger was missed. 0: No SOC5 interrupt overflow 1: SOC5 interrupt overflow
4	SOC4	RO	0x0	SOC4 (Start of Conversion) overflow flag This indicates an SOC4 interrupt was generated while an existing SOC4 interrupt was already pending. This overflow flag can't stop SOC4 interrupts from being processing. It just show that a trigger was missed.

Bits	Field Name	Type	Reset	Description
				0: No SOC4 interrupt overflow 1: SOC4 interrupt overflow
3	SOC3	RO	0x0	SOC3 (Start of Conversion) overflow flag This indicates an SOC3 interrupt was generated while an existing SOC3 interrupt was already pending. This overflow flag can't stop SOC3 interrupts from being processing. It just show that a trigger was missed. 0: No SOC3 interrupt overflow 1: SOC3 interrupt overflow
2	SOC2	RO	0x0	SOC2 (Start of Conversion) overflow flag This indicates an SOC2 interrupt was generated while an existing SOC2 interrupt was already pending. This overflow flag can't stop SOC2 interrupts from being processing. It just show that a trigger was missed. 0: No SOC2 interrupt overflow 1: SOC2 interrupt overflow
1	SOC1	RO	0x0	SOC1 (Start of Conversion) overflow flag This indicates an SOC1 interrupt was generated while an existing SOC1 interrupt was already pending. This overflow flag can't stop SOC1 interrupts from being processing. It just show that a trigger was missed. 0: No SOC1 interrupt overflow 1: SOC1 interrupt overflow
0	SOC0	RO	0x0	SOC0 (Start of Conversion) overflow flag This indicates an SOC0 interrupt was generated while an existing SOC0 interrupt was already pending. This overflow flag can't stop SOC0 interrupts from being processing. It just show that a trigger was missed. 0: No SOC0 interrupt overflow 1: SOC0 interrupt overflow

Table 10-22: ADC SOC Overflow Clear Register (ADCSOCOVFC) Layout

ADCSOCOVFC (ADC SOC Overflow Clear Register) Offset: 0x24 Default: 0x00000000 Access: ADCx->ADCSOCOVFC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SOC15	14 SOC14	13 SOC13	12 SOC12	11 SOC11	10 SOC10	9 SOC9	8 SOC8
7 SOC7	6 SOC6	5 SOC5	4 SOC4	3 SOC3	2 SOC2	1 SOC1	0 SOC0

Table 10-23: ADC SOC Overflow Clear Register (ADCSOCOVFC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	SOC15	W1C	0x0	Clear SOC15 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC15 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
14	SOC14	W1C	0x0	Clear SOC14 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC14 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
13	SOC13	W1C	0x0	Clear SOC13 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC13 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
12	SOC12	W1C	0x0	Clear SOC12 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC12 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
11	SOC11	W1C	0x0	Clear SOC11 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC11 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
10	SOC10	W1C	0x0	Clear SOC10 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC10 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
9	SOC9	W1C	0x0	Clear SOC9 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC9 overflow bit in ADCSOCOVF register simultaneously, then hardware has priority and ADCSOCOVF register will be set. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
8	SOC8	W1C	0x0	Clear SOC8 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC8 overflow bit in ADCSOCOVF register simultaneously, then software has priority and

Bits	Field Name	Type	Reset	Description
				ADCSOCO VF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
7	SOC7	W1C	0x0	Clear SOC7 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC7 overflow bit in ADCSOCO VF register simultaneously, then software has priority and ADCSOCO VF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
6	SOC6	W1C	0x0	Clear SOC6 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC6 overflow bit in ADCSOCO VF register simultaneously, then hardware has priority and ADCSOCO VF register will be set. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
5	SOC5	W1C	0x0	Clear SOC5 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC5 overflow bit in ADCSOCO VF register simultaneously, then hardware has priority and ADCSOCO VF register will be set. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
4	SOC4	W1C	0x0	Clear SOC4 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC4 overflow bit in ADCSOCO VF register simultaneously, then hardware has priority and ADCSOCO VF register will be set. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
3	SOC3	W1C	0x0	Clear SOC3 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC3 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
2	SOC2	W1C	0x0	Clear SOC2 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC2 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
1	SOC1	W1C	0x0	Clear SOC1 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC1 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.
0	SOC0	W1C	0x0	Clear SOC0 overflow flag If this bit receive a request to set from software and hardware try to set respective SOC0 overflow bit in ADCSOCOVF register simultaneously, then software has priority and ADCSOCOVF register will be cleared. 0: Write a 0 has no effect and always read back 0. 1: Write a 1 clears the flag. This bit is self-cleared.

Table 10-24: ADC Interrupt Trigger SOC Enable Register (ADCINTSOCEN) Layout

ADCINTSOCEN (ADC Interrupt Trigger SOC Enable Register) Offset: 0x28 Default: 0x00000000 Access: ADCx->ADCINTSOCEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH15	14 CH14	13 CH13	12 CH12	11 CH11	10 CH10	9 CH9	8 CH8
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 10-25: ADC Interrupt Trigger SOC Enable Register (ADCINTSOCEN) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH15	RW	0x0	ADC interrupt trigger CH15 SOC enable 0: ADCINTSOCSEL1.CH15 is ignored 1: ADCCHCTL15.TRIGSEL is overridden by ADCINTSOCSEL1.CH15
14	CH14	RW	0x0	ADC interrupt trigger CH14 SOC enable 0: ADCINTSOCSEL1.CH14 is ignored 1: ADCCHCTL14.TRIGSEL is overridden by ADCINTSOCSEL1.CH14
13	CH13	RW	0x0	ADC interrupt trigger CH13 SOC enable 0: ADCINTSOCSEL1.CH13 is ignored 1: ADCCHCTL13.TRIGSEL is overridden by ADCINTSOCSEL1.CH13
12	CH12	RW	0x0	ADC interrupt trigger CH12 SOC enable 0: ADCINTSOCSEL1.CH12 is ignored 1: ADCCHCTL12.TRIGSEL is overridden by ADCINTSOCSEL1.CH12
11	CH11	RW	0x0	ADC interrupt trigger CH11 SOC enable 0: ADCINTSOCSEL1.CH11 is ignored 1: ADCCHCTL11.TRIGSEL is overridden by ADCINTSOCSEL1.CH11
10	CH10	RW	0x0	ADC interrupt trigger CH10 SOC enable 0: ADCINTSOCSEL1.CH10 is ignored 1: ADCCHCTL10.TRIGSEL is overridden by ADCINTSOCSEL1.CH10
9	CH9	RW	0x0	ADC interrupt trigger CH9 SOC enable 0: ADCINTSOCSEL1.CH9 is ignored 1: ADCCHCTL9.TRIGSEL is overridden by ADCINTSOCSEL1.CH9

Bits	Field Name	Type	Reset	Description
8	CH8	RW	0x0	ADC interrupt trigger CH8 SOC enable 0: ADCINTSOCSEL1.CH8 is ignored 1: ADCCHCTL8.TRIGSEL is overridden by ADCINTSOCSEL1.CH8
7	CH7	RW	0x0	ADC interrupt trigger CH7 SOC enable 0: ADCINTSOCSEL0.CH7 is ignored 1: ADCCHCTL7.TRIGSEL is overridden by ADCINTSOCSEL0.CH7
6	CH6	RW	0x0	ADC interrupt trigger CH6 SOC enable 0: ADCINTSOCSEL0.CH6 is ignored 1: ADCCHCTL6.TRIGSEL is overridden by ADCINTSOCSEL0.CH6
5	CH5	RW	0x0	ADC interrupt trigger CH5 SOC enable 0: ADCINTSOCSEL0.CH5 is ignored 1: ADCCHCTL5.TRIGSEL is overridden by ADCINTSOCSEL0.CH5
4	CH4	RW	0x0	ADC interrupt trigger CH4 SOC enable 0: ADCINTSOCSEL0.CH4 is ignored 1: ADCCHCTL4.TRIGSEL is overridden by ADCINTSOCSEL0.CH4
3	CH3	RW	0x0	ADC interrupt trigger CH3 SOC enable 0: ADCINTSOCSEL0.CH3 is ignored 1: ADCCHCTL3.TRIGSEL is overridden by ADCINTSOCSEL0.CH3
2	CH2	RW	0x0	ADC interrupt trigger CH2 SOC enable 0: ADCINTSOCSEL0.CH2 is ignored 1: ADCCHCTL2.TRIGSEL is overridden by ADCINTSOCSEL0.CH2
1	CH1	RW	0x0	ADC interrupt trigger CH1 SOC enable 0: ADCINTSOCSEL0.CH1 is ignored 1: ADCCHCTL1.TRIGSEL is overridden by ADCINTSOCSEL0.CH1
0	CH0	RW	0x0	ADC interrupt trigger CH0 SOC enable 0: ADCINTSOCSEL0.CH0 is ignored 1: ADCCHCTL0.TRIGSEL is overridden by ADCINTSOCSEL0.CH0

Table 10-26: ADC Interrupt Trigger SOC Select Register 0 (ADCINTSOCSEL) Layout

ADCINTSOCSEL (ADC Interrupt Trigger SOC Select Register 0) Offset: 0x2C Default: 0x00000000 Access: ADCx->ADCINTSOCSEL							
31	30	29	28	27	26	25	24
CH7				CH6			
23	22	21	20	19	18	17	16
CH5				CH4			
15	14	13	12	11	10	9	8
CH3				CH2			
7	6	5	4	3	2	1	0
CH1				CH0			

Table 10-27: ADC Interrupt Trigger SOC Select Register 0 (ADCINTSOCSEL) Description

Bits	Field Name	Type	Reset	Description
31:28	CH7	RW	0x0	ADC CH7 SOC from which channel interrupt
27:24	CH6	RW	0x0	ADC CH6 SOC from which channel interrupt
23:20	CH5	RW	0x0	ADC CH5 SOC from which channel interrupt
19:16	CH4	RW	0x0	ADC CH4 SOC from which channel interrupt
15:12	CH3	RW	0x0	ADC CH3 SOC from which channel interrupt
11:8	CH2	RW	0x0	ADC CH2 SOC from which channel interrupt
7:4	CH1	RW	0x0	ADC CH1 SOC from which channel interrupt
3:0	CH0	RW	0x0	ADC CH0 SOC from which channel interrupt

Table 10-28: ADC External SOC Input Control Register (ADCEXTSOCCTL) Layout

ADCEXTSOCCTL (ADC External SOC Input Control Register) Offset: 0x34 Default: 0x00000001 Access: ADCx->ADCEXTSOCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 10-29: ADC External SOC Input Control Register (ADCEXTSOCCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number for EXTSOC
0	POL	RW	0x1	EXTSOC polarity 0: Active when low 1: Active when high

Table 10-30: ADC Channel n ([0:15]) Control Register (ADCCHCTL) Layout

ADCCHCTL (ADC Channel n ([0:15]) Control Register) Offset: 0x38 Default: 0x00000000 Access: ADCx->ADCCHCTL							
31 RESERVED	30 RESERVED	29 SHSEL	28 TRIGSEL	27 TRIGSEL	26 TRIGSEL	25 TRIGSEL	24 SHINSELN
23 SHINSELN	22 SHINSELN	21 SHINSELN	20 SHINSELN	19 SHINSELN	18 SHINSELN	17 SHINSELN	16 AVGCNT
15 AVGCNT	14 AVGCNT	13 FRACBIT	12 CONVCNT	11 CONVCNT	10 CONVCNT	9 CONVCNT	8 CONVCNT
7 SAMPCNT	6 SAMPCNT	5 SAMPCNT	4 SAMPCNT	3 SAMPCNT	2 SAMPCNT	1 SAMPCNT	0 SAMPCNT

Table 10-31: ADC Channel n ([0:15]) Control Register (ADCCHCTL) Description

Bits	Field Name	Type	Reset	Description
31:30	RESERVED_31_30	RO	0x0	Reserved.
29	SHSEL	RW	0x0	SH select 0: Disable all SH 1: Enable SH0
28:25	TRIGSEL	RW	0x0	SOC trigger source, which will set bit in ADCSOCFLG It can be overridden by ADCINTSOCSEL 0000: ADCTRIG0 - Software only 0001: ADCTRIG1 - Timer 0 0010: ADCTRIG2 - Timer 1 0011: ADCTRIG3 - Timer 2 0100: ADCTRIG4 - EXTSOC 0101: ADCTRIG5 - PWM0SOCA 0110: ADCTRIG6 - PWM0SOCB 0111: ADCTRIG7 - PWM0SOCC 1000: ADCTRIG8 - PWM1SOCA 1001: ADCTRIG9 - PWM1SOCB 1010: ADCTRIG10 - PWM1SOCC 1011: ADCTRIG11 - PWM2SOCA 1100: ADCTRIG12 - PWM2SOCB 1101: ADCTRIG13 - PWM2SOCC 1110: ADCTRIG14 - PWM3SOCA 1111: ADCTRIG15 - PWM3SOCB
24:21	SHINSELN	RW	0x0	Select negative input of selected SH 0000: Analog ground 0001: T-sensor negative output 0010: Negative output of DPGA 0011: ANA_IN1 pin input 0100: ANA_IN3 pin input 0101: ANA_IN5 pin input 0110: ANA_IN6 pin input

Bits	Field Name	Type	Reset	Description
				0111: ANA_IN7 pin input 1000: ANA_IN8 pin input 1001: DAC1 output 1010: DAC2 output 1011: ANA_IN12 pin input / 3-phase common-node sensed signal
20:17	SHINSELP	RW	0x0	Select positive input of selected SH 0000: Analog ground 0001: T-sensor positive output 0010: Positive output of DPGA 0011: ANA_IN0 pin input 0100: ANA_IN2 pin input 0101: ANA_IN4 pin input 0110: SPGA output 0111: ATEST output 1000: DAC buffer output 1001: DAC0 output 1010: Digital 1.2V power 1011: Analog 3.3V power 1100: ANA_IN9 pin input / U-Phase sensed signal 1101: ANA_IN10 pin input / V-Phase sensed signal 1110: ANA_IN11 pin input / W-Phase sensed signal 1111: ANA_IN13 pin input / VBATMDIV sensed signal
16:14	AVGCNT	RW	0x0	Select average counts for the ADC result 000: No average 001: Each SOC will trigger 2 consecutive ADC operations and then average 010: Each SOC will trigger 4 consecutive ADC operations and then average 011: Each SOC will trigger 8 consecutive ADC operations and then average 100: Each SOC will trigger 16 consecutive ADC operations and then average 101: Each SOC will trigger 32 consecutive ADC operations and then average 110: Each SOC will trigger 64 consecutive ADC operations and then average 111: Each SOC will trigger 128 consecutive ADC operations and then average
13	FRACBIT	RW	0x0	Keep 7 LSBs as fractional part to ADC result due to average 0: ADC result is a 13-bit integer number

Bits	Field Name	Type	Reset	Description
				1: ADC result is 20-bit and the 7 LSBs are the fractional part
12:8	CONVCNT	RW	0x0	Conversion window size, which is (CONVCNT + 1) ADC clock cycles
7:0	SAMPCNT	RW	0x0	Sample window size, which is (SAMPCNT + 1) ADC clock cycles

Table 10-32: ADC Offset Trim Register n ([0:15]) (ADCOFFSET) Layout

ADCOFFSET (ADC Offset Trim Register n ([0:15])) Offset: 0x78 Default: 0x00000000 Access: ADCx->ADCOFFSET							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-33: ADC Offset Trim Register n ([0:15]) (ADCOFFSET) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Offset trimming (signed number) for ADCRESULT0 Should be within [-4096, 4095], where the 7 LSBs are the fractional part.

Table 10-34: ADC Gain Trim Register n ([0:15]) (ADCGAIN) Layout

ADCGAIN (ADC Gain Trim Register n ([0:15])) Offset: 0xB8 Default: 0x00008000 Access: ADCx->ADCGAIN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-35: ADC Gain Trim Register n ([0:15]) (ADCGAIN) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x8000	Gain trimming (VAL/32768) for ADCRESULT0

Table 10-36: ADC SH0 Offset Trim Register (SHOFFSET) Layout

SHOFFSET (ADC SH0 Offset Trim Register) Offset: 0xF8 Default: 0x00000000 Access: ADCx->SHOFFSET							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-37: ADC SH0 Offset Trim Register (SHOFFSET) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Offset trimming (signed number) for SH0 Should be within [-4096, 4095], where the 7 LSBs are the fractional part.

Table 10-38: ADC SH0 Gain Trim Register (SHGAIN) Layout

SHGAIN (ADC SH0 Gain Trim Register) Offset: 0x104 Default: 0x00008000 Access: ADCx->SHGAIN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-39: ADC SH0 Gain Trim Register (SHGAIN) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x8000	Gain trimming (VAL/32768) for SH0

Table 10-40: ADC Sample Hold Channel EOC Flag Register (SHFLG) Layout

SHFLG (ADC Sample Hold Channel EOC Flag Register) Offset: 0x110 Default: 0x00000000 Access: ADCx->SHFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							SHRDY0

Table 10-41: ADC Sample Hold Channel EOC Flag Register (SHFLG) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	SHRDY0	RO	0x0	SHRAWCODE0 ready flag 0: SHRAWCODE0 not ready 1: SHRAWCODE0 ready

Table 10-42: ADC Sample Hold Channel EOC Flag Clear Register (SHFLGCLR) Layout

SHFLGCLR (ADC Sample Hold Channel EOC Flag Clear Register) Offset: 0x114 Default: 0x00000000 Access: ADCx->SHFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							SHRDY0

Table 10-43: ADC Sample Hold Channel EOC Flag Clear Register (SHFLGCLR) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	SHRDY0	W1C	0x0	SHRAWCODE0 ready flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears SHFLG[SHRDY0]. This bit is self-cleared.

Table 10-44: ADC Control Register (ADCCTL) Layout

ADCCTL (ADC Control Register) Offset: 0x118 Default: 0x00000000 Access: ADCx->ADCCTL							
31	30	29	28	27	26	25	24
RESERVED							
23 DMAEN7	22 DMAEN6	21 DMAEN5	20 DMAEN4	19 DMAEN3	18 DMAEN2	17 DMAEN1	16 DMAEN0
15	14	13	12	11	10	9	8
RESERVED							
7 SHFRCN0	6	5	4 SHFRCP0	3 RST	2 EN	1 REFEN	0 RESERVED

Table 10-45: ADC Control Register (ADCCTL) Description

Bits	Field Name	Type	Reset	Description
31:24	RESERVED_31_24	RO	0x0	Reserved.
23	DMAEN7	RW	0x0	ADC result 7 DMA enable 0: Disable 1: Enable
22	DMAEN6	RW	0x0	ADC result 6 DMA enable 0: Disable 1: Enable
21	DMAEN5	RW	0x0	ADC result 5 DMA enable 0: Disable 1: Enable
20	DMAEN4	RW	0x0	ADC result 4 DMA enable 0: Disable 1: Enable
19	DMAEN3	RW	0x0	ADC result 3 DMA enable 0: Disable 1: Enable
18	DMAEN2	RW	0x0	ADC result 2 DMA enable 0: Disable 1: Enable
17	DMAEN1	RW	0x0	ADC result 1 DMA enable 0: Disable 1: Enable
16	DMAEN0	RW	0x0	ADC result 0 DMA enable 0: Disable 1: Enable
15:8	RESERVED_15_8	RO	0x0	Reserved.
7:6	SHFRCN0	RW	0x0	Force SH0 negative input connection 00: Disable 10: Force SH0 negative input tied low 11: Force SH0 negative input tied high

Bits	Field Name	Type	Reset	Description
5:4	SHFRCP0	RW	0x0	Force SH0 positive input connection 00: Disable 10: Force SH0 positive input tied low 11: Force SH0 positive input tied high
3	RST	W1C	0x0	ADC digital finite state machine reset 0: Write a 0 has no effect and always read back 0 1: Write a 1 reset the ADC digital part. This bit is self-cleared
2	EN	RW	0x0	ADC enable 0: Disable 1: Enable
1	REFEN	RW	0x0	ADC reference enable 0: Disable ADC 1: Enable ADC
0	RESERVED_0	RO	0x0	Reserved.

Table 10-46: ADC SH0 Raw Code Register (SHRAWCODE) Layout

SHRAWCODE (ADC SH0 Raw Code Register) Offset: 0x11C Default: 0x00000000 Access: ADCx->SHRAWCODE							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-47: ADC SH0 Raw Code Register (SHRAWCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Raw code (Signed number from -4096 to 4095) for SH0, where the 7 LSBs are the fractional part.

Table 10-48: ADC Result Register n ([0:15]) (ADCRESULT) Layout

ADCRESULT (ADC Result Register n ([0:15])) Offset: 0x128 Default: 0x00000000 Access: ADCx->ADCRESULT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-49: ADC Result Register n ([0:15]) (ADCRESULT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	ADC result (Signed number from -4096 to 4095)

Table 10-50: PPU Comparison Result Register n ([0:2]) (PPURESULT) Layout

PPURESULT (PPU Comparison Result Register n ([0:2])) Offset: 0x168 Default: 0x00000000 Access: ADCx->PPURESULT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-51: PPU Comparison Result Register n ([0:2]) (PPURESULT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Comparison result with extended sign bit

Table 10-52: PPU SOC Delay Register n ([0:2]) (PPUSOCDLY) Layout

PPUSOCDLY (PPU SOC Delay Register n ([0:2])) Offset: 0x188 Default: 0x00000000 Access: ADCx->PPUSOCDLY							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-53: PPU SOC Delay Register n ([0:2]) (PPUSOCDLY) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Delay from trigger to execution for SOC selected by PPUCTL[n].SOCSEL

Table 10-54: PPU Interrupt Raw Flag Register n ([0:2]) (PPURAWIF) Layout

PPURAWIF (PPU Interrupt Raw Flag Register n ([0:2])) Offset: 0x1A8 Default: 0x00000000 Access: ADCx->PPURAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					XZRO	TZOV	TZUV

Table 10-55: PPU Interrupt Raw Flag Register n ([0:2]) (PPURAWIF) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	XZRO	RO	0x0	Latched zero-cross event flag 0: Zero-cross event does not happen 1: Zero-cross event happened
1	TZOV	RO	0x0	Latched over-voltage event flag 0: Over-voltage event does not happen 1: Over-voltage event happened
0	TZUV	RO	0x0	Latched under-voltage event flag 0: Under-voltage event does not happen 1: Under-voltage event happened

Table 10-56: PPU Interrupt Flag Register n ([0:2]) (PPUIF) Layout

PPUIF (PPU Interrupt Flag Register n ([0:2])) Offset: 0x1C8 Default: 0x00000000 Access: ADCx->PPUIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				INT	XZRO	TZOV	TZUV

Table 10-57: PPU Interrupt Flag Register n ([0:2]) (PPUIF) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	INT	RO	0x0	PPU interrupt flag 0: PPU interrupt does not happen 1: PPU interrupt happened and issued to the CPU. Further interrupt will not be generated until this bit is cleared
2	XZRO	RO	0x0	Latched PPU zero-cross interrupt flag 0: PPU zero-cross interrupt does not happen 1: PPU zero-cross interrupt happened. Further event will not trigger interrupt until this bit is cleared
1	TZOV	RO	0x0	Latched PPU over-voltage interrupt flag 0: PPU over-voltage interrupt does not happen 1: PPU over-voltage interrupt happened. Further event will not trigger interrupt until this bit is cleared
0	TZUV	RO	0x0	Latched PPU under-voltage interrupt flag 0: PPU over-voltage interrupt does not happen 1: PPU over-voltage interrupt happened Further event will not trigger interrupt until this bit is cleared

Table 10-58: PPU Interrupt Clear Register n ([0:2]) (PPUIC) Layout

PPUIC (PPU Interrupt Clear Register n ([0:2])) Offset: 0x1E8 Default: 0x00000000 Access: ADCx->PPUIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				INT	XZRO	TZOV	TZUV

Table 10-59: PPU Interrupt Clear Register n ([0:2]) (PPUIC) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	INT	W1C	0x0	PPU interrupt flag clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared.
2	XZRO	W1C	0x0	PPU zero-cross interrupt clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the XZRO flag. This bit is self-cleared.
1	TZOV	W1C	0x0	PPU over-voltage interrupt clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the TZOV flag. This bit is self-cleared.
0	TZUV	W1C	0x0	PPU under-voltage interrupt clear 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the TZUV flag. This bit is self-cleared.

Table 10-60: PPU Interrupt Enable Register n ([0:2]) (PPUIE) Layout

PPUIE (PPU Interrupt Enable Register n ([0:2])) Offset: 0x208 Default: 0x00000000 Access: ADCx->PPUIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					XZRO	TZOV	TZUV

Table 10-61: PPU Interrupt Enable Register n ([0:2]) (PPUIE) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	XZRO	RW	0x0	Enable zero-cross event as interrupt source 0: Disable 1: Enable
1	TZOV	RW	0x0	Enable over-voltage event as interrupt source 0: Disable 1: Enable
0	TZUV	RW	0x0	Enable under-voltage event as interrupt source 0: Disable 1: Enable

Table 10-62: PPU Trip-Zone Event Enable Register n ([0:2]) (PPUTZE) Layout

PPUTZE (PPU Trip-Zone Event Enable Register n ([0:2])) Offset: 0x228 Default: 0x00000000 Access: ADCx->PPUTZE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					XZRO	TZOV	TZUV

Table 10-63: PPU Trip-Zone Event Enable Register n ([0:2]) (PPUTZE) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	XZRO	RW	0x0	Enable zero-cross event as trip-zone event 0: Disable 1: Enable
1	TZOV	RW	0x0	Enable over-voltage event as trip-zone event 0: Disable 1: Enable
0	TZUV	RW	0x0	Enable under-voltage event as trip-zone event 0: Disable 1: Enable

Table 10-64: PPU Control Register n ([0:2]) (PPUCTL) Layout

PPUCTL (PPU Control Register n ([0:2])) Offset: 0x248 Default: 0x00000400 Access: ADCx->PPUCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					POL	DATASEL	
7	6	5	4	3	2	1	0
DATASEL		SOCSEL				CBCEN	EN

Table 10-65: PPU Control Register n ([0:2]) (PPUCTL) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10	POL	RW	0x1	Polarity for comparison 0: PPU result is the reference minus the selected result 1: PPU result is the selected result minus the reference
9:6	DATASEL	RW	0x0	Select the ADC result to compare with the reference
5:2	SOCSEL	RW	0x0	Select the SOC signal to derive its self PPUSOCDLY[n]
1	CBCEN	RW	0x0	PPU cycle-by-cycle clear enable 0: Disable PPU cycle-by-cycle clear 1: Enable PPU cycle-by-cycle clear
0	EN	RW	0x0	PPU enable 0: Disable PPU 1: Enable PPU

Table 10-66: PPU Reference Register n ([0:2]) (PPUREF) Layout

PPUREF (PPU Reference Register n ([0:2])) Offset: 0x268 Default: 0x00000000 Access: ADCx->PPUREF							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-67: PPU Reference Register n ([0:2]) (PPUREF) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Reference with extended sign bit Should be within [-4096, 4095]

Table 10-68: PPU Trip-Zone Low-Side Threshold Register n ([0:2]) (PPUTHL) Layout

PPUTHL (PPU Trip-Zone Low-Side Threshold Register n ([0:2])) Offset: 0x288 Default: 0x00000000 Access: ADCx->PPUTHL							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-69: PPU Trip-Zone Low-Side Threshold Register n ([0:2]) (PPUTHL) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Signed trip-zone threshold within [-8192, 8191]

Table 10-70: PPU Trip-Zone High-Side Threshold Register n ([0:2]) (PPUTHH) Layout

PPUTHH (PPU Trip-Zone High-Side Threshold Register n ([0:2])) Offset: 0x2A8 Default: 0x00000000 Access: ADCx->PPUTHH							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 10-71: PPU Trip-Zone High-Side Threshold Register n ([0:2]) (PPUTHH) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Signed trip-zone threshold within [-8192, 8191]

Table 10-72: Temperature Sensor Control Register (TSENSCTL) Layout

TSENSCTL (Temperature Sensor Control Register) Offset: 0x2C8 Default: 0x00000000 Access: ADCx->TSENSCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6	5	4	3	2 SWAPBJT	1 OUTINV	0 EN
DEMSEL							

Table 10-73: Temperature Sensor Control Register (TSENSCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:3	DEMSEL	RW	0x0	Current branch select for dynamic element match
2	SWAPBJT	RW	0x0	Swap BJT 0: Not swap BJT 1: Swap BJT
1	OUTINV	RW	0x0	Invert T-Sensor output 0 and output 1 0: Normal connect 1: Invert T-Sensor output 0 and output 1
0	EN	RW	0x0	Temperature sensor enable 0: Disable 1: Enable

Table 10-74: ADC Register Write-Allow Key Register (ADCREGKEY) Layout

ADCREGKEY (ADC Register Write-Allow Key Register) Offset: 0x2CC Default: 0x1ACCE551 Access: ADCx->ADCREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 10-75: ADC Register Write-Allow Key Register (ADCREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected ADC registers

11 13-bit Monitoring ADC

11.1 Overview

The SPC1169 implements one 13-bit ADC to monitor TMCU/VDD33/VDD12/VAREF13 for safety consideration.

Functions of the 13-bit Monitoring ADC module include:

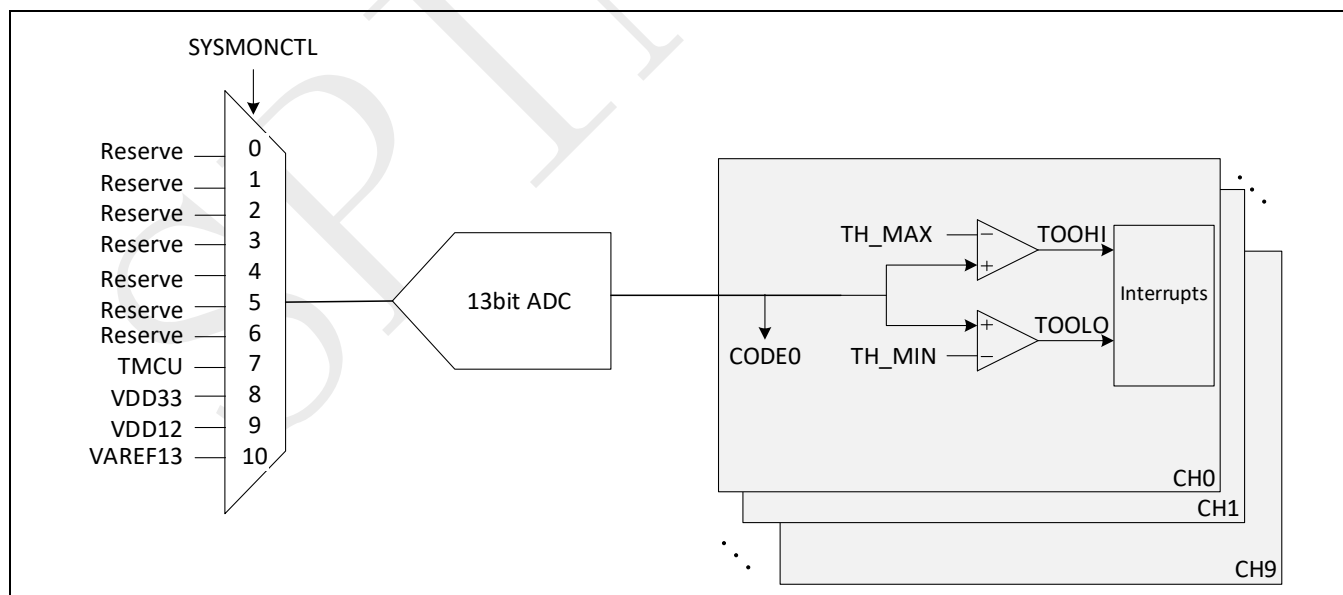
- 13-bit ADC core with built-in sample-and-hold (S/H)
 - 1us conversion time
 - Up to 11-channel, multiplexed inputs
- Single-ended input, input range is 0V to 3.657V
- Individual enable and threshold control for each channel
- Trigger an NMI interrupt if any channel is out of the threshold window

11.2 Architecture and operation

As shown in [Figure 11-1](#), the 13-bit Monitoring ADC circuit includes:

- One sampler and one core
- 13-bit Monitoring ADC reference
- Out of threshold window detection

Figure 11-1: 13-bit Monitoring ADC block diagram



The 13-bit Monitoring ADC is used to monitor the following voltages TMCU, VDD33.. shown in [Table 11-1](#). And the results is saved in corresponding registers (such as VDD33CODE).

The monitoring itmes can be enabled individually. For example, if SYSMONCTL.VDD33EN=1, VDD33 voltage will be monitored, and register VDD33CODE will be updated.

If the resulting code is out of threshold, corresponding error event flag would be latched in NMRAWIF register. And when the event is enabled by setting the NMIE register bit, a non-maskable interrupt will be triggered and the interrupt flag will be set in NMIF. For example, if VDD33CODE is above VDD33TH.MAX, NMRAWIF.VDD33ERR will be set. If NMIE.VDD33ERR=1, NMIF.VDD33ERR will be set simultaneously.

Note: If using a 13-bit analog-to-digital converter to measure the temperature of the MCU, the steps are as follows:

1. Enable the Bandgap of the 13-bit analog-to-digital converter; the corresponding function interface is provided in the SDK driver library.
2. Enable the TSensor on the MCU side.

User can change SYSMONCTL.MONPRD to set the system monitor interval, which is (MONPRD*8+1) system clock cycles. It is recommended to set the interval to be above 10us for voltage settling.

Detailed monitoring items are listed in [Table 11-1](#).

Table 11-1: 13-bit Monitoring ADC monitor items

Step	Item	Physical Item	Note
0	Reserve	Reserve	Reserve
1	Reserve	Reserve	Reserve
2	Reserve	Reserve	Reserve
3	Reserve	Reserve	Reserve
4	Reserve	Reserve	Reserve
5	Reserve	Reserve	Reserve
6	Reserve	Reserve	Reserve
7	TMCU	TSEN1&TSEN0	MCU temperature
8	VDD33	VDD33	VDD33 voltage
9	VDD12	VDD12	VDD12 voltage
10	ADCREF13	ADCREF	13-bit ADC reference

The measured voltage can be derived from the code as

$$V_{\text{meas}} = \text{CODE} * 3.657 / 4096 \approx \text{CODE} / 1120 \text{ (V)}$$

For example, if VDD33CODE is 3696, so the corresponding measured voltage is $V_{\text{DD33}} = 3696/1120 = 3.3\text{V}$. Therefore, VDD33 is 3.3V.

The measured temperature can be derived from the code as:

$$\text{TMCU} = (\text{TMCUOFFSET} + \text{TMCUGAIN} * \text{TMCUCODE}) / 65536 \text{ (}^{\circ}\text{C)}$$

where the parameters are stored in the Flash as shown in [Table 11-2](#).

Table 11-2: Parameters to derive the measured temperature

Parameter	Address	Default value (Signed-number)
TMCUOFFSET	0x1100 1178	0xFEE53EFE (-18530562)
TMCUGAIN	0x1100 117C	0x0003CA01 (248321)

11.3 Registers

Please see [Section 25.10](#) for system monitor related register map and definitions.

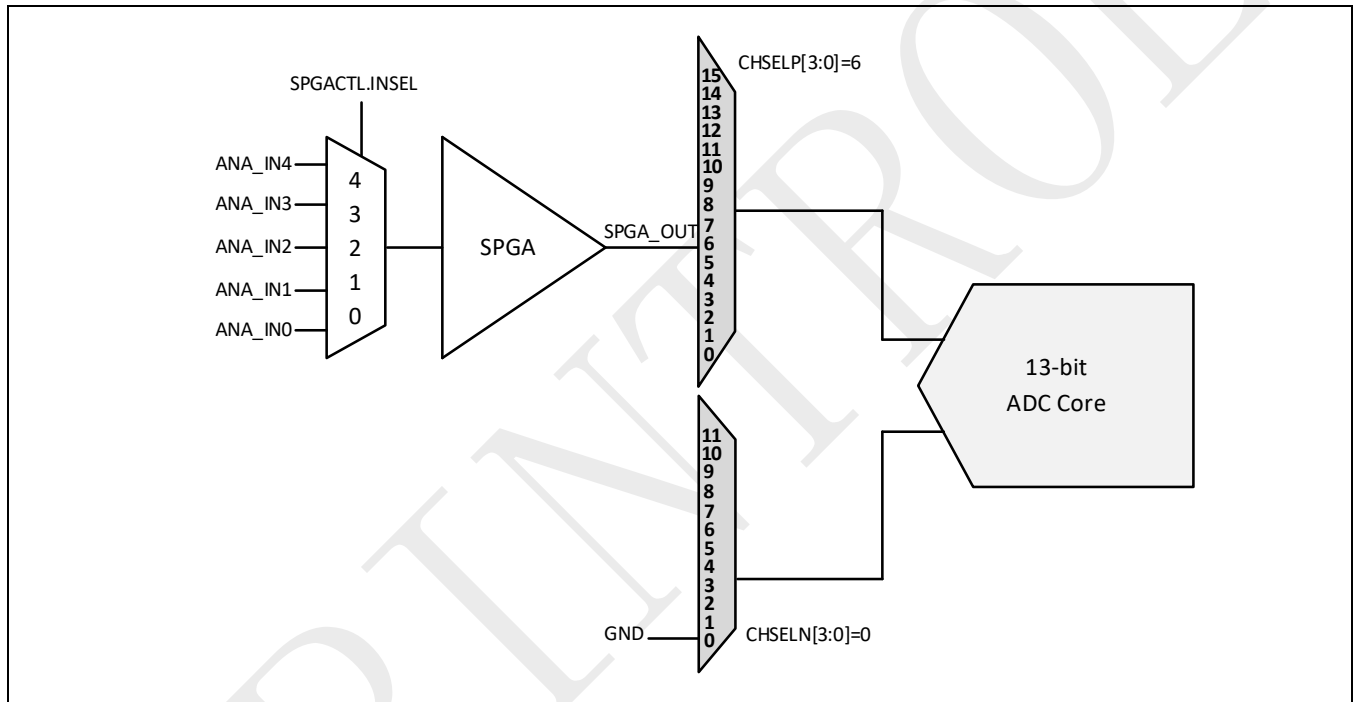
12 SPGA

12.1 SPGA overview

The SPC1169 implements one Single-ended Programmable-Gain Amplifiers (PGA). It can select ANA_IN0 ~ ANA_IN4 as input, and support gain up to 64.

12.2 SPGA architecture

Figure 12-1: SPGA block architecture



12.3 SPGA channel selection

SPGA includes 5 to 1 input MUX. The input selection table is shown as below.

Table 12-1: SPGA MUX selection

SPGACTL.INSEL	Input
0	ANA_IN0
1	ANA_IN1
2	ANA_IN2
3	ANA_IN3
4	ANA_IN4

12.4 SPGA gain selection

The SPGA's gain selection is shown as below:

Table 12-2: SPGA gain selection

SPGACTL.GAIN	Gain
0	1
1	2
2	4
3	8
4	16
5	32
6	48
7	64

SPGA output can be calculated by the following equations:

$$\text{OUT} = \text{GAIN} \times \text{IN}$$

To avoid SPGA in saturation region, SPGA's output range need to be in 0.3V to VDD3-0.3V, and input range within 0V to VDD3-1.3V. The following table shows input and output range for different gain settings. (VDD3=3.3V typically).

Table 12-3: PGA input range in different gain for single-ended mode

SPGACTL.GAIN	GAIN	Input range (V)		Output range (V)
0	1	0.3	~ 2	0.3~2
1	2	0.15	~ 1.5	0.3~3
2	4	0.075	~ 0.75	0.3~3
3	8	0.0375	~ 0.375	0.3~3
4	16	0.01875	~ 0.1875	0.3~3
5	32	0.009375	~ 0.09375	0.3~3
6	48	0.00625	~ 0.0625	0.3~3
7	64	0.0046875	~ 0.046875	0.3~3

12.5 Registers

Please see [Section 13.5](#) for SPGA related register map and definitions.

13 DPGA

13.1 DPGA overview

The SPC1169 implements a Differential Programmable-Gain Amplifiers (DPGA) with two dedicated pins. The DPGA is a current sense amplifier. It is used to measure near ground differential voltage via the 13-bit ADC. Its gain is digitally programmable through internal control registers. The features of the DPGA is as follows:

- Programmable gains: 2, 4, 8, 16, 24, 32, 48 and 64
- Differential input voltage: $\pm 2.7 / \text{Gain}_{\text{DPGA}}$
- Wide common mode input range: $-1.5\text{V} \sim 2\text{V}$ with $\text{Gain}_{\text{DPGA}}(2\times)$
- Settling time: 250 ns to 1.4 μs
- Gain error is within $\pm 1\%$

13.2 DPGA gain and input range

DPGA output range is $0.3\sim 3\text{V}$, so DPGA differential input voltage is $\pm 2.7 / \text{Gain}_{\text{DPGA}}$. DPGA gain setting and differential input range is shown in [Table 13-1](#).

Table 13-1: DPGA gain and differential input range

DPGACTL.GAIN	Gain _{DPGA}	Input range (V)	Output range (V)
0	2	-1.35~1.35	0.3~3
1	4	-0.675~+0.675	0.3~3
2	8	-0.337~+0.337	0.3~3
3	16	-0.168~+0.168	0.3~3
4	24	-0.112~+0.112	0.3~3
5	32	-0.084~+0.084	0.3~3
6	48	-0.056~+0.056	0.3~3
7	64	-0.042~+0.042	0.3~3

13.3 DPGA application

[Figure 13-1](#) shows how the DPGA is used as a low-side current sense amplifier. The motor current is converted to a voltage by a sense resistor R_{sh} . The LP filter is to suppress EMI, recommended value for R_{filt} is 120 Ω , C_{filt} is 1nF. The DPGA is able to measure positive and negative currents through the resistor R_{sh} . The input signal is amplified by the DPGA and sent to the 13-bit ADC. R_{stray} is the stray resistance from sense resistor to ground.

For the input from DPGAIP pin and DPGAIN pin, the output would be:

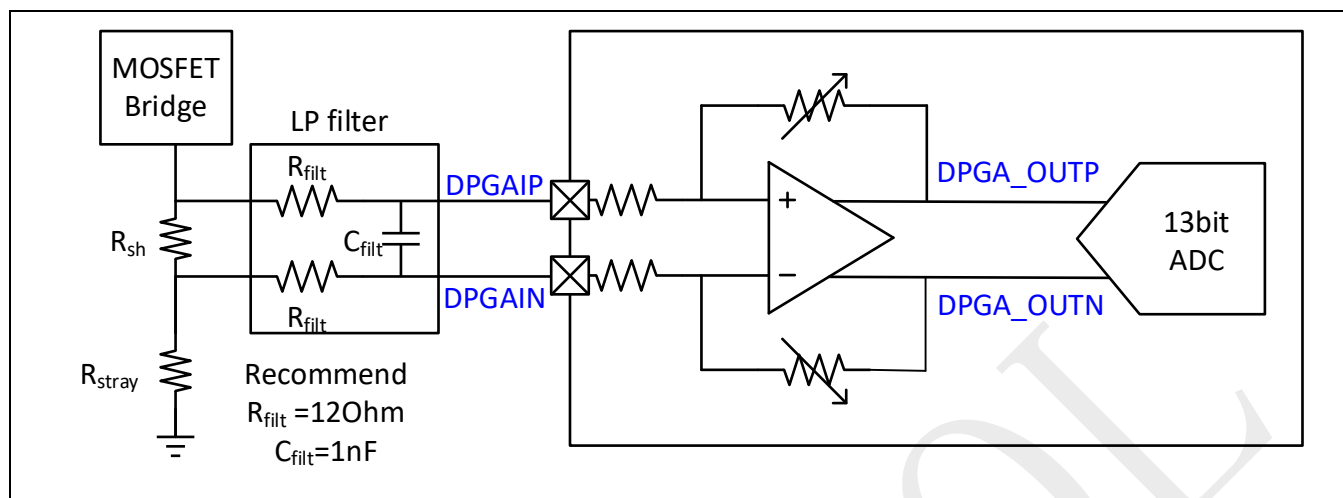
$$V_{\text{DPGA_OUTP}} = V_{\text{CMOUT}} + (V_{\text{DPGAIP}} - V_{\text{DPGAIN}}) \times \text{Gain}_{\text{DPGA}} / 2$$

$$V_{\text{DPGA_OUTN}} = V_{\text{CMOUT}} - (V_{\text{DPGAIP}} - V_{\text{DPGAIN}}) \times \text{Gain}_{\text{DPGA}} / 2$$

where V_{CMOUT} is the output common mode voltage as:

$$V_{\text{CMOUT}} = V_{\text{DVDD33}} / 2$$

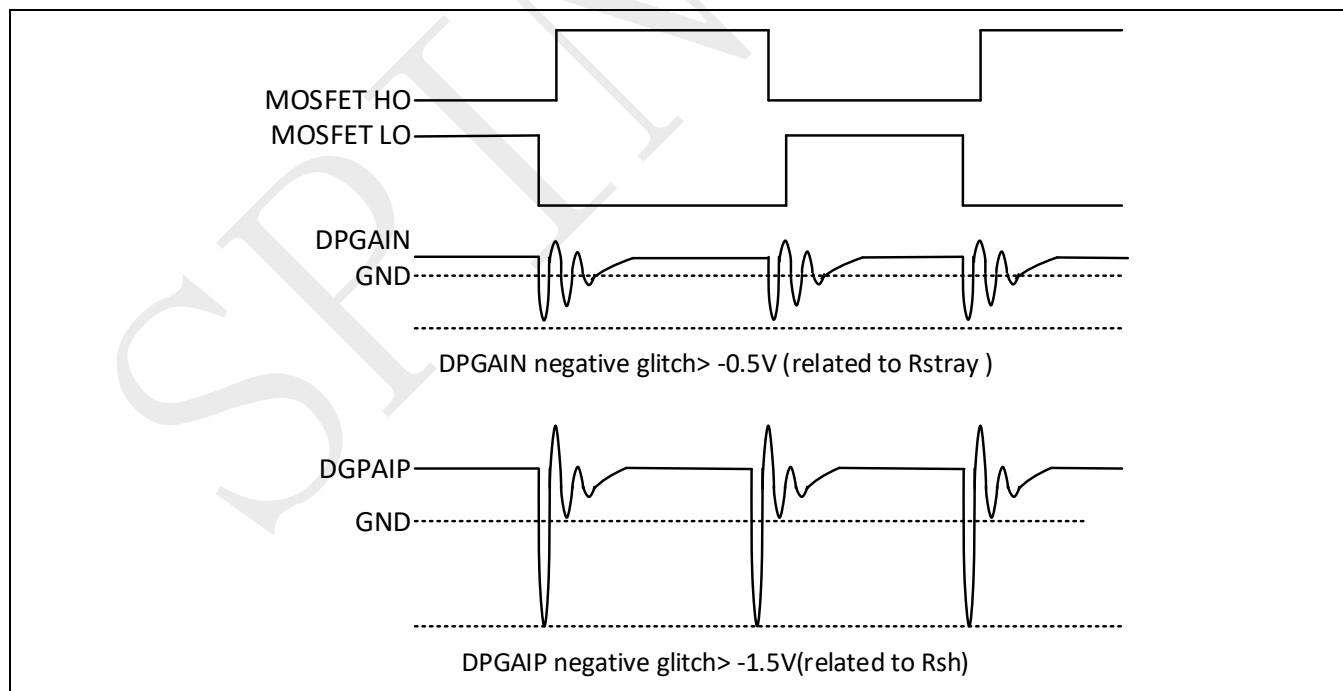
Figure 13-1: DPGA application diagram



When motor bridge driver high-side MOSFET turns off and low-side MOSFET turns on, there are voltage glitches at DPGAIP and DPGAIN due to parasitic inductance.

- DPGAIN negative glitch should be above -0.5V since the DPGA input common mode range is -0.5V~2V.
 - Rstray should be small enough on the PCB.
- DPGAIP negative glitch should be above -1.5V since the DPGA input pin can sustain -1.5V
 - Rsh should be a reasonable value on the PCB.

Figure 13-2: DPGA input range requirement

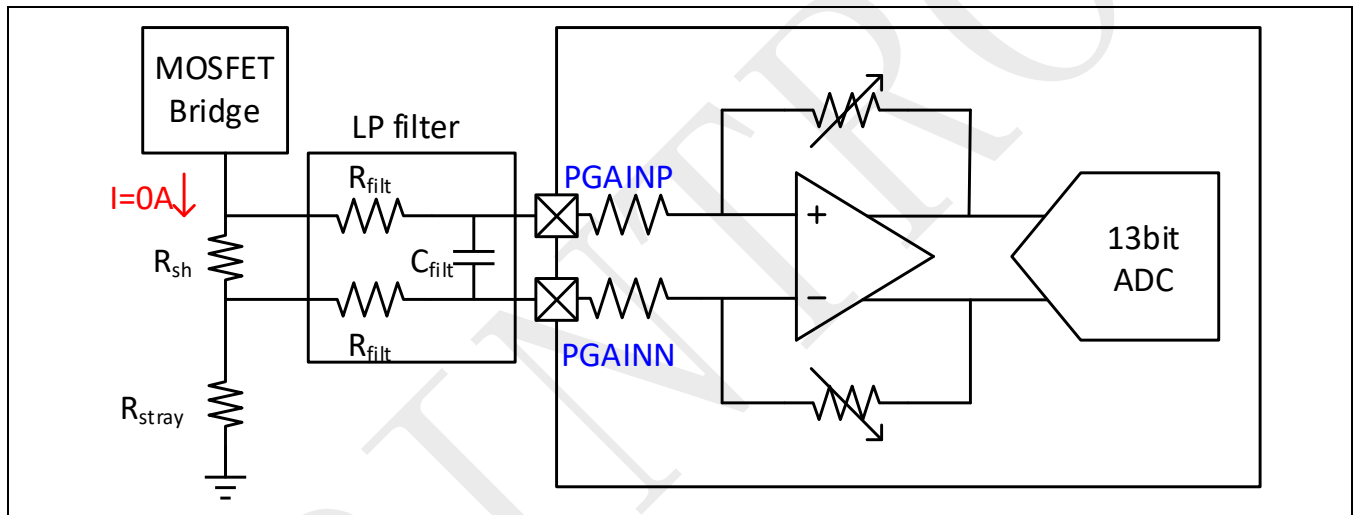


13.4 DPGA offset calibration

In order to perform a correct offset calibration, the complete path from sense resistor through DPGA to the 13-bit ADC should be measured. For this purpose, it should be ensured that there is no current (0A) through the sense resistor, as shown in [Figure 13-3](#). The following steps are the recommended offset calibration:

- Connect DPGA output to ADC input
- Set desired DPGA gain
- Ensure 0A current flowing through the sense resistor
- Perform ADC measurement
- The ADC digital output, is the complete path offset
- Repeat the measurement for other DPGA gain settings if needed

Figure 13-3: DPGA offset calibration path



13.5 Registers

13.5.1 PGA register map

Table 13-2: PGA Module Base Address

Peripheral Module	Base Address
PGA	0x4000 3000

Table 13-3: PGA Register Map

Register	Offset	Description	Reset Value
DPGACTL *	0x2D0	Differential PGA Control Register	0x00000000
SPGACTL *	0x2D4	Single-ended PGA Control Register	0x00000000
PGAREGKEY	0x2F0	PGA Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the PGAREGKEY=0x1ACCE551.

13.5.2 PGA registers

Table 13-4: Differential PGA Control Register (DPGACTL) Layout

DPGACTL (Differential PGA Control Register) Offset: 0x2D0 Default: 0x00000000 Access: PGAx->DPGACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							D2SBUFEN
7	6	5	4	3	2	1	0
PUSTRENGTH	PUIN	PUIP	GAIN			BYP	EN

Table 13-5: Differential PGA Control Register (DPGACTL) Description

Bits	Field Name	Type	Reset	Description
31:9	RESERVED_31_9	RO	0x0	Reserved.
8	D2SBUFEN	RW	0x0	PGA differential to single-ended buffer enable 0: Disable 1: Enable
7	PUSTRENGTH	RW	0x0	PGA pull-up strength 0: 2 mA 1: 10 mA
6	PUIN	RW	0x0	PGA negative input pull-up enable 0: Do not pull-up 1: Pull-up
5	PUIP	RW	0x0	PGA positive input pull-up enable 0: Do not pull-up 1: Pull-up
4:2	GAIN	RW	0x0	PGA differential gain 000: 2x 001: 4x 010: 8x 011: 16x 100: 24x 101: 32x 110: 48x 111: 64x
1	BYP	RW	0x0	PGA bypass 0: Not bypass 1: Bypass

Bits	Field Name	Type	Reset	Description
0	EN	RW	0x0	PGA enable 0: Disable DPGA 1: Enable DPGA

Table 13-6: Single-ended PGA Control Register (SPGACTL) Layout

SPGACTL (Single-ended PGA Control Register) Offset: 0x2D4 Default: 0x00000000 Access: PGAx->SPGACTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	GAIN			INSEL			EN

Table 13-7: Single-ended PGA Control Register (SPGACTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:4	GAIN	RW	0x0	PGA gain 000: 1x 001: 2x 010: 4x 011: 8x 100: 16x 101: 32x 110: 48x 111: 64x
3:1	INSEL	RW	0x0	PGA input select 000: ANA_IN0 001: ANA_IN1 010: ANA_IN2 011: ANA_IN3 100: ANA_IN4 101: T-sensor negative output 110: T-sensor positive output 111: ATEST output
0	EN	RW	0x0	PGA enable 0: Disable SPGA 1: Enable SPGA

Table 13-8: PGA Register Write-Allow Key Register (PGAREGKEY) Layout

PGAREGKEY (PGA Register Write-Allow Key Register) Offset: 0x2F0 Default: 0x1ACCE551 Access: PGAx->PGAREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 13-9: PGA Register Write-Allow Key Register (PGAREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected PGA registers

14 Analog Comparator

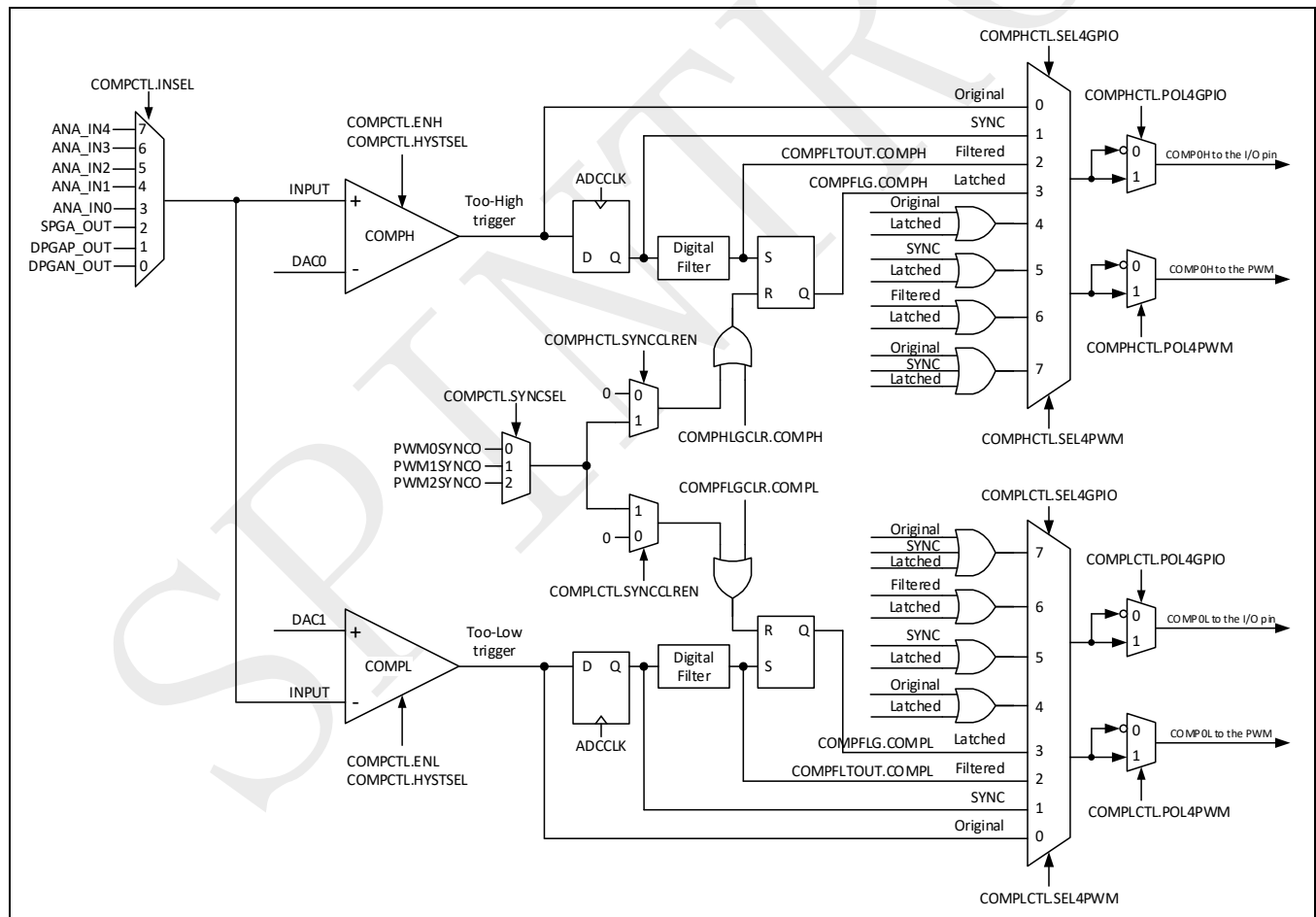
14.1 Comparator overview

In SPC1169, there are 2 high speed rail-to-rail input comparators, check whether the voltage is too high or too low for over-current detection.

14.2 Comparator architecture

Figure 14-1 shows the comparator block diagram, which includes two comparators, one is for monitoring voltage is too high, the other is to check voltage too low, and also includes input MUX to the comparators. Hysteresis can be enabled for each comparator with a typical value of 30 mV. The output of each comparator can be used as the PWM trip-zone event and has a typical delay of 50 ns.

Figure 14-1: Comparator block diagram



14.3 Comparator MUX selection

The input selections for the two comparators are shown in Table 14-1.

Table 14-1: Comparator MUX selection

COMPCTL.INSEL	Input selection
7	ANA_IN4
6	ANA_IN3
5	ANA_IN2
4	ANA_IN1
3	ANA_IN0
2	SPGA_OUT
1	DPGA_OUTP
0	DPGA_OUTN

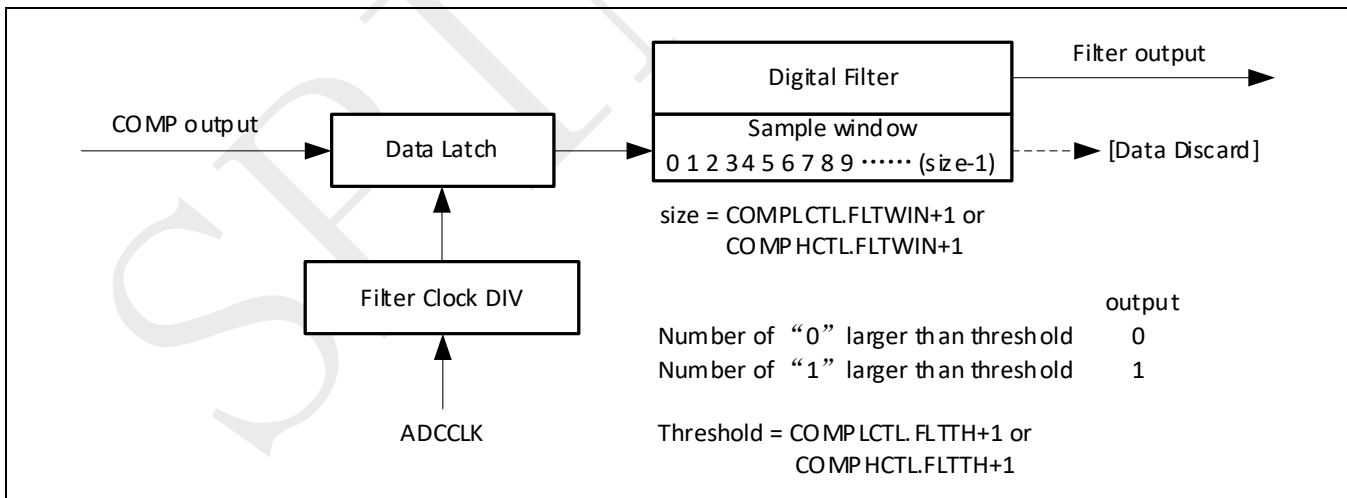
14.4 Digital filter

The comparator's output is sent to the sample window of the digital filter. The sampled clock is from ADCCLK, and can be divided by COMPLCTL.FLTDIV/COMPHCTL.FLTDIV, the window size is defined by COMPLCTL.FLTWIN/COMPHCTL.FLTWIN. The digital filter can count the number of majority value of the sample window. If the number is larger than the pre-defined threshold value COMPLCTL.FLTTH/COMPHCTL.FLTTH, the filter output will change to the majority value; otherwise, the filter output will not be changed.

For proper operation, the threshold values should be greater than COMPLCTL.FLTWIN/2 and COMPHCTL.FLTWIN/2 respectively.

A conceptual model of the digital filter is shown in [Figure 14-2](#).

Figure 14-2: Digital filter behavior



Note: Considering the noise, it is recommended to

1. Add a reasonable capacitor on PCB for the input signal sent to the comparator.
2. Disable the hysteresis to avoid false toggle and lock due to noise.
3. Make use of the digital filter.

14.5 Registers

14.5.1 Comparator register map

Table 14-2: Comparator Module Base Address

Peripheral Module	Base Address
COMP	0x4000 3000

Table 14-3: COMP Register Map

Register	Offset	Description	Reset Value
COMPFLTOUT	0x2F4	Comparator Filter Output Register	0x00000000
COMPFLG	0x2F8	Comparator Flag Register	0x00000000
COMPFLGCLR	0x2FC	Comparator Flag Clear Register	0x00000000
COMPCTL *	0x300	Comparator 0 Control Register	0x00000000
COMPLCTL *	0x304	COMP0L Control Register	0x00000088
COMPHCTL *	0x308	COMP0H Control Register	0x00000088
DAC0CTL *	0x360	DAC0 Control Register	0x00000002
DAC0CODE	0x364	DAC0 Code Register	0x00000000
DAC0CODEA	0x368	DAC0 Active Code Register	0x00000000
RAMP0DLY *	0x36C	RAMP0 Delay Shadow Register	0x00000000
RAMP0DLYA	0x370	RAMP0 Delay Active Register	0x00000000
RAMP0DEC *	0x374	RAMP0 Decrement Shadow Register	0x00000000
RAMP0DECA	0x378	RAMP0 Decrement Active Register	0x00000000
RAMP0MAX *	0x37C	RAMP0 Reload Value Shadow Register	0x00000000
RAMP0MAXA	0x380	RAMP0 Reload Value Active Register	0x00000000
RAMP0CNT	0x384	RAMP0 Count Register	0x00000000
DAC1CTL *	0x388	DAC1 Control Register	0x00000002
DAC1CODE	0x38C	DAC1 Code Register	0x00000000
DAC1CODEA	0x390	DAC1 Active Code Register	0x00000000
DAC2CTL *	0x3B0	DAC2 Control Register	0x00000002
DAC2CODE	0x3B4	DAC2 Code Register	0x00000000
DAC2CODEA	0x3B8	DAC2 Active Code Register	0x00000000
DACBUFCTL *	0x3F8	DAC Buffer Control Register	0x00000000
COMPREGKEY	0x3FC	COMP Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the COMPREGKEY=0x1ACCE551.

14.5.2 Comparator registers

Table 14-4: Comparator Filter Output Register (COMPFLTOUT) Layout

COMPFLTOUT (Comparator Filter Output Register) Offset: 0x2F4 Default: 0x00000000 Access: COMPx->COMPFLTOUT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						COMP	COMPL

Table 14-5: Comparator Filter Output Register (COMPFLTOUT) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	COMP	RO	0x0	COMP filtered output without latch 0: Input is lower than the high-boundary 1: Input is higher than the high-boundary
0	COMPL	RO	0x0	COMPL filtered output without latch 0: Input is higher than the low-boundary 1: Input is lower than the low-boundary

Table 14-6: Comparator Flag Register (COMPFLG) Layout

COMPFLG (Comparator Flag Register) Offset: 0x2F8 Default: 0x00000000 Access: COMPx->COMPFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						COMP	COMPL

Table 14-7: Comparator Flag Register (COMPFLG) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	COMP_H	RO	0x0	Latched COMPFLTOUT[COMP_H] 0: Input has been lower than the high-boundary 1: Input has been higher than the high-boundary
0	COMP_L	RO	0x0	Latched COMPFLTOUT[COMP_L] 0: Input has been higher than the low-boundary 1: Input has been lower than the low-boundary

Table 14-8: Comparator Flag Clear Register (COMPFLGCLR) Layout

COMPFLGCLR (Comparator Flag Clear Register) Offset: 0x2FC Default: 0x00000000 Access: COMPx->COMPFLGCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						COMP_H	COMP_L

Table 14-9: Comparator Flag Clear Register (COMPFLGCLR) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	COMP_H	W1C	0x0	Latched COMP_H flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears COMPFLG[COMP_H]. This bit is self-cleared.
0	COMP_L	W1C	0x0	Latched COMP_L flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears COMPFLG[COMP_L]. This bit is self-cleared.

Table 14-10: Comparator 0 Control Register (COMPCTL) Layout

COMPCTL (Comparator 0 Control Register) Offset: 0x300 Default: 0x00000000 Access: COMPx->COMPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						SYNCSEL	
7	6	5	4	3	2	1	0
RESERVED	INSEL			HYSTSEL		ENH	ENL

Table 14-11: Comparator 0 Control Register (COMPCTL) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:8	SYNCSEL	RW	0x0	Select the synchronous output from PWM[SYNCSEL] 00: PWM0 synchronous output 01: PWM1 synchronous output 10: PWM2 synchronous output 11: PWM3 synchronous output
7	RESERVED_7	RO	0x0	Reserved.
6:4	INSEL	RW	0x0	COMP0 input select 000: DPGA negative output 001: DPGA positive output 010: SPGA output 011: ANA_IN0 100: ANA_IN1 101: ANA_IN2 110: ANA_IN3 111: ANA_IN4
3:2	HYSTSEL	RW	0x0	COMP0 hysteresis select 00: 0 mV 01: 12 mV 10: 24 mV 11: 36 mV
1	ENH	RW	0x0	COMP0H enable 0: Disable COMPH 1: Enable COMPH
0	ENL	RW	0x0	COMP0L enable 0: Disable COMPL 1: Enable COMPL

Table 14-12: COMP0L Control Register (COMPLCTL) Layout

COMPLCTL (COMPOL Control Register) Offset: 0x304 Default: 0x00000088 Access: COMPx->COMPLCTL											
31 30		29	28 27		26 25		24				
RESERVED		FLTRST			FLTTH						
23 22		21	20 19		18 17		16				
		FLTWIN				FLTDIV					
15 14		13	12 11		10 9		8				
		FLTDIV						SYNCCLREN			
7	6 5		4	3		2 1		0			
POL4GPIO	SEL4GPIO			POL4PWM		SEL4PWM					

Table 14-13: COMP0L Control Register (COMPLCTL) Description

Bits	Field Name	Type	Reset	Description
31:30	RESERVED_31_30	RO	0x0	Reserved.
29	FLTRST	W1S	0x0	Reset all registers in the filter with the input 0: Write 0 has no effect. Always read back 0. 1: Write 1 will fill all registers in the filter with the input
28:24	FLTTH	RW	0x0	Filter's threshold is (FLTTH+1) 1's/0's for output to toggle
23:19	FLTWIN	RW	0x0	Filter's window size is (FLTWIN+1) samples
18:9	FLTDIV	RW	0x0	Filter's clock dividing ratio is (FLTDIV+1)
8	SYNCCLREN	RW	0x0	Enable latched COMP0L status clear by PWMSYNC 0: PWMSYNC does not affect the latched COMPL status 1: PWMSYNC will clear the latched COMPL status
7	POL4GPIO	RW	0x1	COMP0L output polarity for GPIO 0: Output 0 when the comparator input is lower than the reference 1: Output 1 when the comparator input is lower than the reference
6:4	SEL4GPIO	RW	0x0	COMP0L output select for GPIO 000: Original output 001: Synchronous output with ADC clock 010: Digital filter output 011: Latched digital filter output 100: Original output or Latched digital filter output 101: Synchronous output with ADC clock or Latched digital filter output 110: Digital filter output or the latched one

Bits	Field Name	Type	Reset	Description
				111: Original output or Synchronous output with ADC clock or Latched digital filter output
3	POL4PWM	RW	0x1	COMP0L output polarity for PWM 0: Output 0 when the comparator input is lower than the reference 1: Output 1 when the comparator input is lower than the reference
2:0	SEL4PWM	RW	0x0	COMP0L output select for PWM 000: Original output 001: Synchronous output with ADC clock 010: Digital filter output 011: Latched digital filter output 100: Original output or Latched digital filter output 101: Synchronous output with ADC clock or Latched digital filter output 110: Digital filter output or the latched one 111: Original output or Synchronous output with ADC clock or Latched digital filter output

Table 14-14: COMP0H Control Register (COMPCTL) Layout

COMPCTL (COMP0H Control Register) Offset: 0x308 Default: 0x00000088 Access: COMPx->COMPCTL							
31	30	29	28	27	26	25	24
RESERVED		FLTRST	FLTTH				
23	22	21	20	19	18	17	16
FLTWIN					FLTDIV		
15	14	13	12	11	10	9	8
FLTDIV							SYNCCLEN
7	6	5	4	3	2	1	0
POL4GPIO	SEL4GPIO			POL4PWM	SEL4PWM		

Table 14-15: COMP0H Control Register (COMPCTL) Description

Bits	Field Name	Type	Reset	Description
31:30	RESERVED_31_30	RO	0x0	Reserved.
29	FLTRST	W1S	0x0	Reset all registers in the filter with the input 0: Write 0 has no effect. Always read back 0. 1: Write 1 will fill all registers in the filter with the input
28:24	FLTTH	RW	0x0	Filter's threshold is (FLTTH+1) 1's/0's for output to toggle
23:19	FLTWIN	RW	0x0	Filter's window size is (FLTWIN+1) samples

Bits	Field Name	Type	Reset	Description
18:9	FLTDIV	RW	0x0	Filter's clock dividing ratio is (FLTDIV+1)
8	SYNCCLREN	RW	0x0	Enable latched COMP0H status clear by PWMSYNC 0: PWMSYNC does not affect the latched COMPH status 1: PWMSYNC will clear the latched COMPH status
7	POL4GPIO	RW	0x1	COMP0H output polarity for GPIO 0: Output 0 when the comparator input is higher than the reference 1: Output 1 when the comparator input is higher than the reference
6:4	SEL4GPIO	RW	0x0	COMP0H output select for GPIO 000: Original output 001: Synchronous output with ADC clock 010: Digital filter output 011: Latched digital filter output 100: Original output or Latched digital filter output 101: Synchronous output with ADC clock or Latched digital filter output 110: Digital filter output or the latched one 111: Original output or Synchronous output with ADC clock or Latched digital filter output
3	POL4PWM	RW	0x1	COMP0H output polarity for PWM 0: Output 0 when the comparator input is higher than the reference 1: Output 1 when the comparator input is higher than the reference
2:0	SEL4PWM	RW	0x0	COMP0H output select for PWM 000: Original output 001: Synchronous output with ADC clock 010: Digital filter output 011: Latched digital filter output 100: Original output or Latched digital filter output 101: Synchronous output with ADC clock or Latched digital filter output 110: Digital filter output or the latched one 111: Original output or Synchronous output with ADC clock or Latched digital filter output

Table 14-16: DAC0 Control Register (DAC0CTL) Layout

DAC0CTL (DAC0 Control Register) Offset: 0x360 Default: 0x00000002 Access: COMPx->DAC0CTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
DBGRUN		RESERVED					
7	6	5	4	3	2	1	0
RAMPLOAD	RAMPEN	RESERVED		SYNCSEL		CODELOAD	EN

Table 14-17: DAC0 Control Register (DAC0CTL) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:14	DBGRUN	RW	0x0	Behavior on CPU halted or lockup 00: Stop immediately 01: Stop at PWM synchronous event 10: Free running
13:8	RESERVED_13_8	RO	0x0	Reserved.
7	RAMPLOAD	RW	0x0	DAC Ramp load mode 0: Register shadowing is applied for ramping counter load from RAMPMAX 1: RAMP counter is directly loaded from the latest RAMPMAX
6	RAMPEN	RW	0x0	DAC Ramping enable 0: Disable ramping 1: Enable ramping. DAC0CODEA is equivalent to the 10 MSBs of RAMPOCNT
5:4	RESERVED_5_4	RO	0x0	Reserved.
3:2	SYNCSEL	RW	0x0	PWMSYNC is selected from PWM[SYNCSEL] 00: PWM0 synchronous output 01: PWM1 synchronous output 10: PWM2 synchronous output 11: PWM3 synchronous output
1	CODELOAD	RW	0x1	DAC code load mode 0: DAC code is updated upon PWMSYNC event 1: DAC code is immediately updated by writing to DAC0CODE
0	EN	RW	0x0	DAC enable 0: Disable 1: Enable

Table 14-18: DAC0 Code Register (DAC0CODE) Layout

DAC0CODE (DAC0 Code Register) Offset: 0x364 Default: 0x00000000 Access: COMPx->DAC0CODE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 14-19: DAC0 Code Register (DAC0CODE) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RW	0x0	DAC code When DAC0CTL[CODELOAD]=0, write to this field only affects the shadow code. When DAC0CTL[CODELOAD]=1, write to this field affects both the shadow code and the active code.

Table 14-20: DAC0 Active Code Register (DAC0CODEA) Layout

DAC0CODEA (DAC0 Active Code Register) Offset: 0x368 Default: 0x00000000 Access: COMPx->DAC0CODEA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 14-21: DAC0 Active Code Register (DAC0CODEA) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RO	0x0	Active DAC code

Table 14-22: RAMP0 Delay Shadow Register (RAMP0DLY) Layout

RAMP0DLY (RAMP0 Delay Shadow Register) Offset: 0x36C Default: 0x00000000 Access: COMPx->RAMP0DLY							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-23: RAMP0 Delay Shadow Register (RAMP0DLY) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Delay from PWMSYNC to start of ramping

Table 14-24: RAMP0 Delay Active Register (RAMP0DLYA) Layout

RAMP0DLYA (RAMP0 Delay Active Register) Offset: 0x370 Default: 0x00000000 Access: COMPx->RAMP0DLYA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-25: RAMP0 Delay Active Register (RAMP0DLYA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Delay from PWMSYNC to start of ramping

Table 14-26: RAMP0 Decrement Shadow Register (RAMP0DEC) Layout

RAMP0DEC (RAMP0 Decrement Shadow Register) Offset: 0x374 Default: 0x00000000 Access: COMPx->RAMP0DEC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-27: RAMP0 Decrement Shadow Register (RAMP0DEC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Step size for decremental ramping

Table 14-28: RAMP0 Decrement Active Register (RAMP0DECA) Layout

RAMP0DECA (RAMP0 Decrement Active Register) Offset: 0x378 Default: 0x00000000 Access: COMPx->RAMP0DECA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-29: RAMP0 Decrement Active Register (RAMP0DECA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Step size for decremental ramping

Table 14-30: RAMP0 Reload Value Shadow Register (RAMP0MAX) Layout

RAMP0MAX (RAMP0 Reload Value Shadow Register) Offset: 0x37C Default: 0x00000000 Access: COMPx->RAMP0MAX							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-31: RAMP0 Reload Value Shadow Register (RAMP0MAX) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Reload value when reset ramping

Table 14-32: RAMP0 Reload Value Active Register (RAMP0MAXA) Layout

RAMP0MAXA (RAMP0 Reload Value Active Register) Offset: 0x380 Default: 0x00000000 Access: COMPx->RAMP0MAXA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-33: RAMP0 Reload Value Active Register (RAMP0MAXA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Reload value when reset ramping

Table 14-34: RAMP0 Count Register (RAMP0CNT) Layout

RAMP0CNT (RAMP0 Count Register) Offset: 0x384 Default: 0x00000000 Access: COMPx->RAMP0CNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-35: RAMP0 Count Register (RAMP0CNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Count number of RAMP0

Table 14-36: DAC1 Control Register (DAC1CTL) Layout

DAC1CTL (DAC1 Control Register) Offset: 0x388 Default: 0x00000002 Access: COMPx->DAC1CTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
DBGRUN		RESERVED					
7	6	5	4	3	2	1	0
RAMPLOAD	RAMPEN	RESERVED		SYNCSEL		CODELOAD	EN

Table 14-37: DAC1 Control Register (DAC1CTL) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:14	DBGRUN	RW	0x0	Behavior on CPU halted or lockup 00: Stop immediately 01: Stop at PWM synchronous event 10: Free running
13:8	RESERVED_13_8	RO	0x0	Reserved.
7	RAMPLOAD	RW	0x0	DAC Ramp load mode 0: Register shadowing is applied for ramping counter load from RAMPMAX 1: RAMP counter is directly loaded from the latest RAMPMAX

Bits	Field Name	Type	Reset	Description
6	RAMPEN	RW	0x0	DAC Ramping enable 0: Disable ramping 1: Enable ramping. DAC1CODEA is equivalent to the 10 MSBs of RAMP1CNT
5:4	RESERVED_5_4	RO	0x0	Reserved.
3:2	SYNCSEL	RW	0x0	PWMSYNC is selected from PWM.SYNCSEL 00: PWM0 synchronous output 01: PWM1 synchronous output 10: PWM2 synchronous output 11: PWM3 synchronous output
1	CODELOAD	RW	0x1	DAC code load mode 0: DAC code is updated upon PWMSYNC event 1: DAC code is immediately updated by writing to DAC1CODE
0	EN	RW	0x0	DAC enable 0: Disable 1: Enable

Table 14-38: DAC1 Code Register (DAC1CODE) Layout

DAC1CODE (DAC1 Code Register) Offset: 0x38C Default: 0x00000000 Access: COMPx->DAC1CODE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 14-39: DAC1 Code Register (DAC1CODE) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RW	0x0	DAC code When DAC1CTL.CODELOAD=0, write to this field only affects the shadow code. When DAC1CTL.CODELOAD=1, write to this field affects both the shadow code and the active code.

Table 14-40: DAC1 Active Code Register (DAC1CODEA) Layout

DAC1CODEA (DAC1 Active Code Register) Offset: 0x390 Default: 0x00000000 Access: COMPx->DAC1CODEA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 14-41: DAC1 Active Code Register (DAC1CODEA) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RO	0x0	Active DAC code

Table 14-42: RAMP1 Delay Shadow Register (RAMP1DLY) Layout

RAMP1DLY (RAMP1 Delay Shadow Register) Offset: 0x394 Default: 0x00000000 Access: COMPx->RAMP1DLY							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-43: RAMP1 Delay Shadow Register (RAMP1DLY) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Delay from PWMSYNC to start of ramping

Table 14-44: RAMP1 Delay Active Register (RAMP1DLYA) Layout

RAMP1DLYA (RAMP1 Delay Active Register) Offset: 0x398 Default: 0x00000000 Access: COMPx->RAMP1DLYA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-45: RAMP1 Delay Active Register (RAMP1DLYA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Delay from PWMSYNC to start of ramping

Table 14-46: RAMP1 Decrement Shadow Register (RAMP1DEC) Layout

RAMP1DEC (RAMP1 Decrement Shadow Register) Offset: 0x39C Default: 0x00000000 Access: COMPx->RAMP1DEC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-47: RAMP1 Decrement Shadow Register (RAMP1DEC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Step size for decremental ramping

Table 14-48: RAMP1 Decrement Active Register (RAMP1DECA) Layout

RAMP1DECA (RAMP1 Decrement Active Register) Offset: 0x3A0 Default: 0x00000000 Access: COMPx->RAMP1DECA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-49: RAMP1 Decrement Active Register (RAMP1DECA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Step size for decremental ramping

Table 14-50: RAMP1 Reload Value Shadow Register (RAMP1MAX) Layout

RAMP1MAX (RAMP1 Reload Value Shadow Register) Offset: 0x3A4 Default: 0x00000000 Access: COMPx->RAMP1MAX							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-51: RAMP1 Reload Value Shadow Register (RAMP1MAX) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	Reload value when reset ramping

Table 14-52: RAMP1 Reload Value Active Register (RAMP1MAXA) Layout

RAMP1MAXA (RAMP1 Reload Value Active Register) Offset: 0x3A8 Default: 0x00000000 Access: COMPx->RAMP1MAXA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-53: RAMP1 Reload Value Active Register (RAMP1MAXA) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Reload value when reset ramping

Table 14-54: RAMP1 Count Register (RAMP1CNT) Layout

RAMP1CNT (RAMP1 Count Register) Offset: 0x3AC Default: 0x00000000 Access: COMPx->RAMP1CNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 14-55: RAMP1 Count Register (RAMP1CNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Count number of RAMP1

Table 14-56: DAC2 Control Register (DAC2CTL) Layout

DAC2CTL (DAC2 Control Register) Offset: 0x3B0 Default: 0x00000002 Access: COMPx->DAC2CTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				SYNCSEL		CODELOAD	EN

Table 14-57: DAC2 Control Register (DAC2CTL) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3:2	SYNCSEL	RW	0x0	PWM synchronous output select 00: PWM0 synchronous output 01: PWM1 synchronous output 10: PWM2 synchronous output 11: PWM3 synchronous output
1	CODELOAD	RW	0x1	DAC code load mode 0: DAC code is updated upon PWMSYNC event 1: DAC code is immediately updated by writing to DAC2CODE
0	EN	RW	0x0	DAC enable 0: Disable 1: Enable

Table 14-58: DAC2 Code Register (DAC2CODE) Layout

DAC2CODE (DAC2 Code Register) Offset: 0x3B4 Default: 0x00000000 Access: COMPx->DAC2CODE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 14-59: DAC2 Code Register (DAC2CODE) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RW	0x0	DAC code When DAC2CTL.CODELOAD=0, write to this field only affects the shadow code. When DAC2CTL.CODELOAD=1, write to this field affects both the shadow code and the active code.

Table 14-60: DAC2 Active Code Register (DAC2CODEA) Layout

DAC2CODEA (DAC2 Active Code Register) Offset: 0x3B8 Default: 0x00000000 Access: COMPx->DAC2CODEA							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 14-61: DAC2 Active Code Register (DAC2CODEA) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RO	0x0	Active DAC code

Table 14-62: DAC Buffer Control Register (DACBUFCTL) Layout

DACBUFCTL (DAC Buffer Control Register) Offset: 0x3F8 Default: 0x00000000 Access: COMPx->DACBUFCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					INSEL	OE	EN

Table 14-63: DAC Buffer Control Register (DACBUFCTL) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	INSEL	RW	0x0	Select the DAC module 0: DAC0 1: DAC1
1	OE	RW	0x0	Enable output to GPIO 0: Disable 1: Enable
0	EN	RW	0x0	DAC buffer enable 0: Disable 1: Enable

Table 14-64: COMP Register Write-Allow Key Register (COMPREGKEY) Layout

COMPREGKEY (COMP Register Write-Allow Key Register) Offset: 0x3FC Default: 0x1ACCE551 Access: COMPx->COMPREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 14-65: COMP Register Write-Allow Key Register (COMPREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected COMP registers

15 Phase Comparator

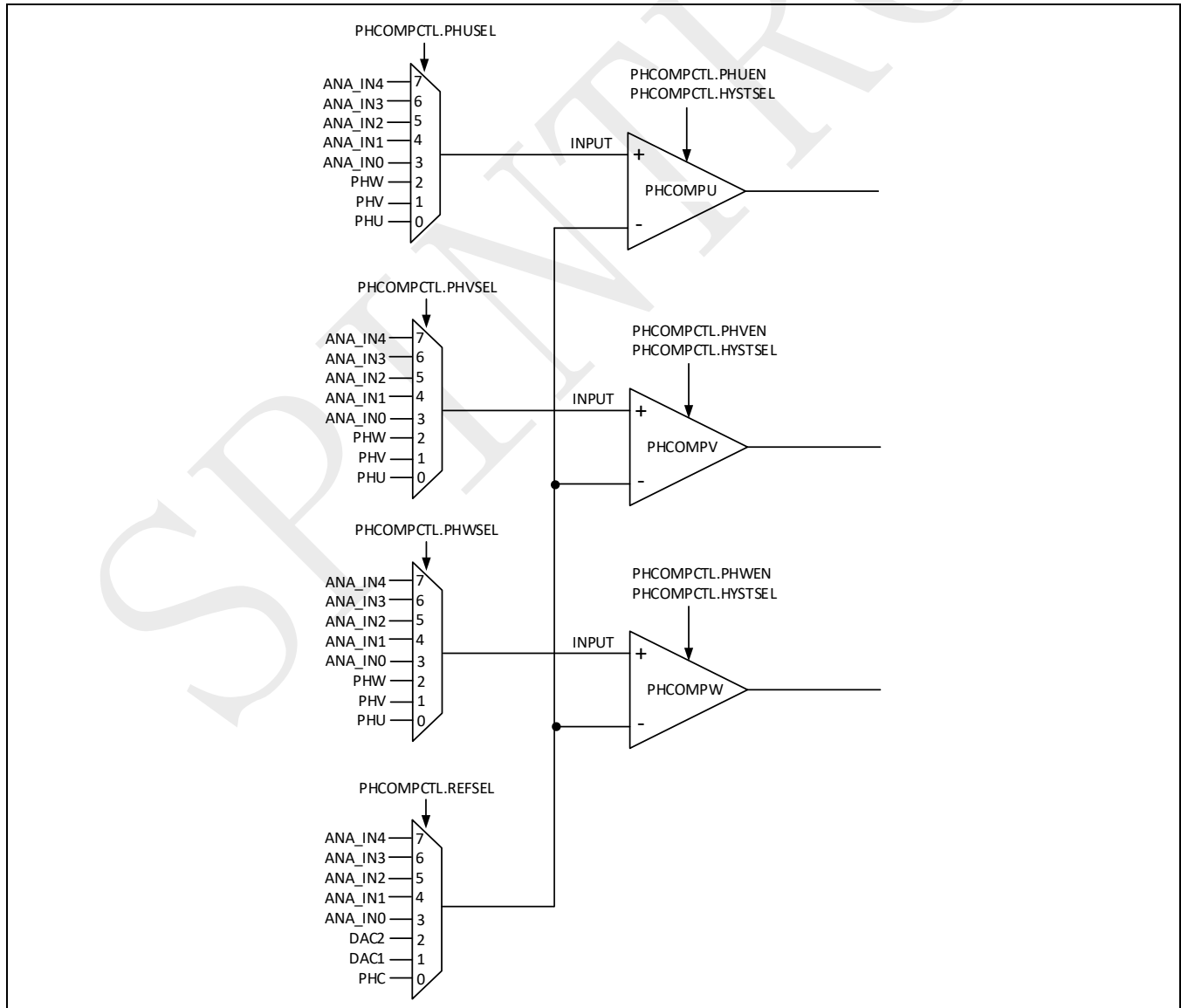
15.1 Overview

SPC1169 include 3 high speed rail-to-rail phase comparators, which can compare three phases voltage (PHU/PHV/PHW) to the common mode voltage (PHC). Phase comparator has 8-to-1 input mux, it can also used as normal comparator to compare analog ADC input to reference voltage (ADC input or DAC output).

15.2 Architecture

Figure 15-1 shows phase comparator block diagram, which includes three phase comparators, and also includes input MUX to the comparators. Hysteresis can be enabled for each comparator with a typical value of 30 mV. The output of each phase comparator has a typical delay of 200ns.

Figure 15-1: Phase Comparator block diagram

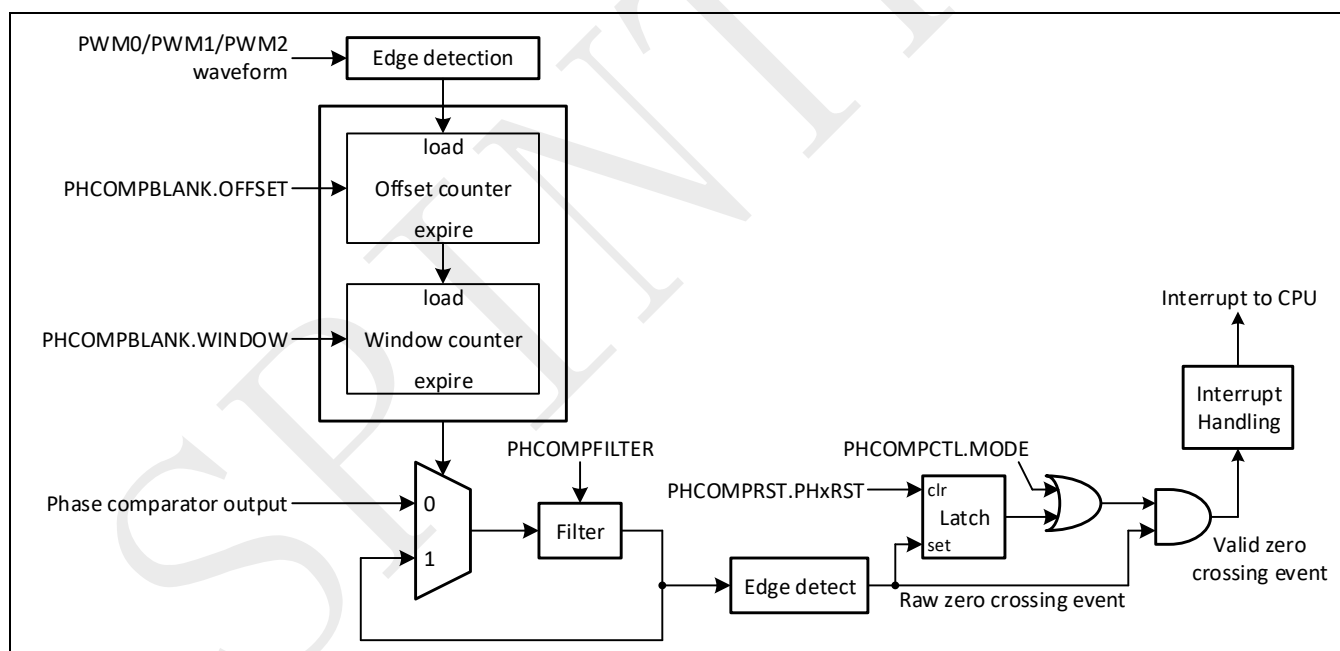


15.3 Post processing

As shown in [Figure 15-2](#), the raw output of the phase comparator will pass through a series of post processing, which operates in HCLK clock domain.

- Blanking with programmable offset and the size of the window.
This is to get rid of the interference due to power switching by PWM toggling.
- Digital filtering with programmable number of clock scaling
- Finite state machine to get rid of the first false-detection of zero-crossing event due to PWM toggling.
 - Phase comparator mode (PHCOMPCTL.MODE=0)
After writing a 1 to the corresponding PHxRST bit in PHCOMPRST register to reset the finite state machine, the first detected zero-crossing event of the filter output is ignored. All succeeding zero-cross events will be sent to the interrupt handling block until the state machine is reset again.
 - Normal comparator mode (PHCOMPCTL.MODE=1)
The finite state machine is by-passed and all detected zero-crossing events are regarded as valid.
- Interrupt and flag handling

Figure 15-2: Phase comparator output post-process

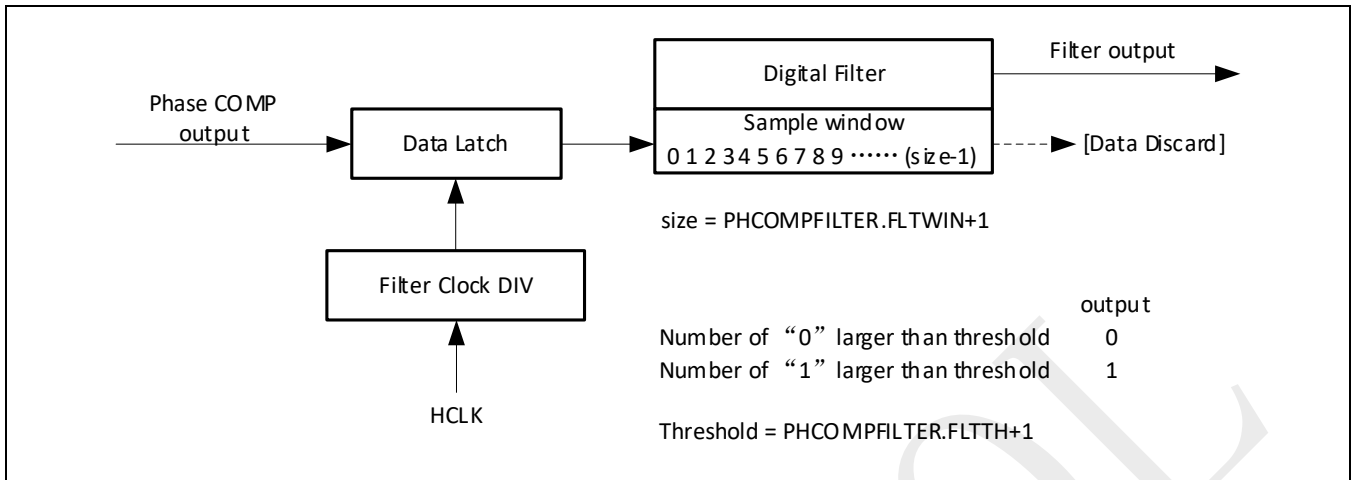


The detail of digital filter block is shown in [Figure 15-3](#).

The phase comparator's output is sent to the sample window of the digital filter. The sampled clock is from HCLK, and can be divided by PHCOMPFILTER.FLTDIV, the window size is defined by PHCOMPFILTER.FLTWIN. The digital filter can count the number of majority value of the sample window. If the number is larger than the pre-defined threshold value PHCOMPFILTER.FLTTH, the filter output will change to the majority value; otherwise, the filter output will not be changed.

For proper operation, the threshold values should be greater than PHCOMPFILTER.FLTWIN/2.

Figure 15-3: Digital filter behavior



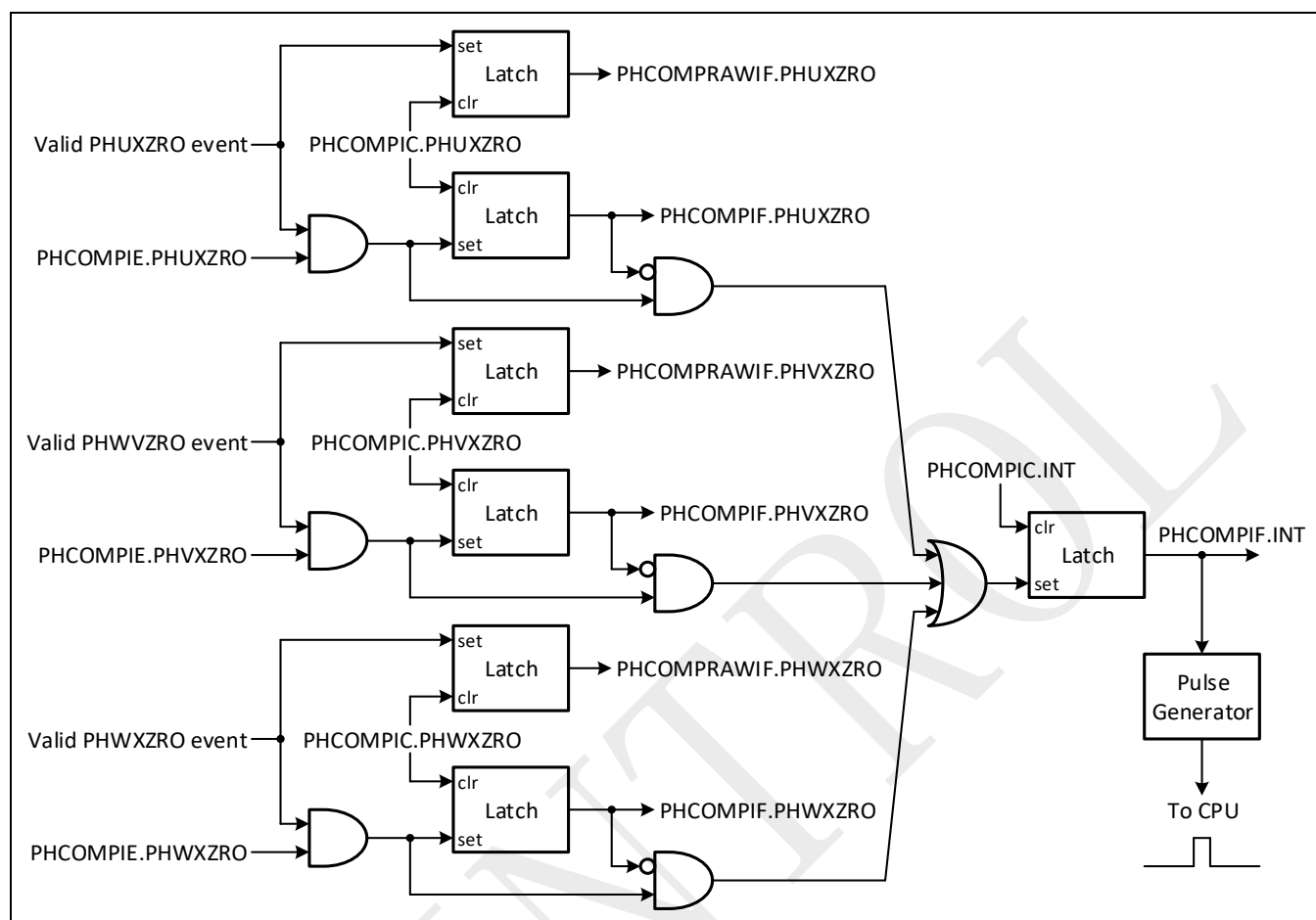
Note: Considering the noise, it is recommended to

1. Add a reasonable capacitor on PCB for the input signal sent to the comparator.
2. Disable the hysteresis to avoid false toggle and lock due to noise.
3. Make use of the digital filter.

The detail of the interrupt handling block is shown in [Figure 15-4](#).

- The valid zero crossing events from the three phases trigger the corresponding event flag bit.
- When enabled, the valid zero crossing event triggers the corresponding interrupt flag bit, and triggers the interrupt. Once the event flag is set, further event will not trigger interrupt until it is cleared.
- Once an interrupt request is issued to the CPU, the combined interrupt flag is also it. It will prevent future interrupt pulse generation until the flag is cleared.

Figure 15-4: Phase comparator interrupt generation



15.4 Registers

15.4.1 Phase Comparator register map

Table 15-1: Phase Comparator Module Base Address

Peripheral Module	Base Address
PHCOMP	0x4000 0500

Table 15-2: PHCOMP Register Map

Register	Offset	Description	Reset Value
PHCOMPSTS	0x0	Phase Comparison Status Register	0x00000000
PHCOMPRAWIF	0x4	Phase Comparison Interrupt Raw Flag Register	0x00000000
PHCOMPPIF	0x8	Phase Comparison Interrupt Flag Register	0x00000000
PHCOMPIC	0xC	Phase Comparison Interrupt Clear Register	0x00000000
PHCOMPIE*	0x10	Phase Comparison Interrupt Enable Register	0x00000000
PHCOMPCTL*	0x14	Phase Comparison Control Register	0x00000420
PHCOMPBLANK*	0x18	Phase Comparison Blanking Control	0x00000000
PHCOMPFILTER*	0x1C	Phase Comparison Filter Control	0x00000000
PHCOMPRST	0x20	Phase Comparison FSM Reset Register	0x00000000
PHCOMPREGKEY	0x24	Phase Comparison Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the COMPREGKEY=0x1ACCE551.

15.4.2 Phase Comparator registers

Table 15-3: Phase Comparison Status Register (PHCOMPSTS) Layout

PHCOMPSTS (Phase Comparison Status Register) Offset: 0x0 Default: 0x00000000 Access: PHCOMPx->PHCOMPSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		PHWRAW	PHVRAW	PHURAW	PHWFLT	PHVFLT	PHUFLT

Table 15-4: Phase Comparison Status Register (PHCOMPSTS) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5	PHWRAW	RO	0x0	W-phase realtime original output status 0: Low 1: High
4	PHVRAW	RO	0x0	V-phase realtime original output status 0: Low 1: High
3	PHURAW	RO	0x0	U-phase realtime original output status 0: Low 1: High
2	PHWFLT	RO	0x0	W-phase realtime filtered output status 0: Low 1: High
1	PHVFLT	RO	0x0	V-phase realtime filtered output status 0: Low 1: High
0	PHUFLT	RO	0x0	U-phase realtime filtered output status 0: Low 1: High

Table 15-5: Phase Comparison Interrupt Raw Flag Register (PHCOMPRAWIF) Layout

PHCOMPRAWIF (Phase Comparison Interrupt Raw Flag Register) Offset: 0x4 Default: 0x00000000 Access: PHCOMPx->PHCOMPRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		PHWZCH	PHWZCL	PHVZCH	PHVZCL	PHUZCH	PHUZCL

Table 15-6: Phase Comparison Interrupt Raw Flag Register (PHCOMPRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5	PHWZCH	RO	0x0	W-phase zero-cross to high event flag 0: Event does not occur 1: Event occurred
4	PHWZCL	RO	0x0	W-phase zero-cross to low event flag 0: Event does not occur 1: Event occurred
3	PHVZCH	RO	0x0	V-phase zero-cross to high event flag 0: Event does not occur 1: Event occurred
2	PHVZCL	RO	0x0	V-phase zero-cross to low event flag 0: Event does not occur 1: Event occurred
1	PHUZCH	RO	0x0	U-Phase zero-cross to high event flag 0: Event does not occur 1: Event occurred
0	PHUZCL	RO	0x0	U-Phase zero-cross to low event flag 0: Event does not occur 1: Event occurred

Table 15-7: Phase Comparison Interrupt Flag Register (PHCOMPIF) Layout

PHCOMPIF (Phase Comparison Interrupt Flag Register) Offset: 0x8 Default: 0x00000000 Access: PHCOMPx->PHCOMPIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	INT	PHWZCH	PHWZCL	PHVZCH	PHVZCL	PHUZCH	PHUZCL

Table 15-8: Phase Comparison Interrupt Flag Register (PHCOMPIF) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	INT	RO	0x0	Global phase comparison interrupt flag 0: Interrupt does not occur 1: Interrupt occurred and issued to the CPU. Further interrupt will not be generated until this bit is cleared
5	PHWZCH	RO	0x0	W-phase zero-cross to high interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
4	PHWZCL	RO	0x0	W-phase zero-cross to low interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
3	PHVZCH	RO	0x0	V-phase zero-cross to high interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
2	PHVZCL	RO	0x0	V-phase zero-cross to low interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
1	PHUZCH	RO	0x0	V-phase zero-cross to high interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
0	PHUZCL	RO	0x0	U-Phase zero-cross to low interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Table 15-9: Phase Comparison Interrupt Clear Register (PHCOMPIC) Layout

PHCOMPIC (Phase Comparison Interrupt Clear Register) Offset: 0xC Default: 0x00000000 Access: PHCOMPx->PHCOMPIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	INT	PHWZCH	PHWZCL	PHVZCH	PHVZCL	PHUZCH	PHUZCL

Table 15-10: Phase Comparison Interrupt Clear Register (PHCOMPIC) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	INT	W1C	0x0	Global phase comparison interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.
5	PHWZCH	W1C	0x0	W-Phase zero-cross to high interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.
4	PHWZCL	W1C	0x0	W-Phase zero-cross to low interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.
3	PHVZCH	W1C	0x0	V-phase zero-cross to high interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.
2	PHVZCL	W1C	0x0	V-phase zero-cross to low interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.
1	PHUZCH	W1C	0x0	V-phase zero-cross to high interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.
0	PHUZCL	W1C	0x0	U-Phase zero-cross to low interrupt clear 0: Write a 0 has no effect. Always readback 0 1: Write a 1 clears the flag. This bit is self-cleared.

Table 15-11: Phase Comparison Interrupt Enable Register (PHCOMPIE) Layout

PHCOMPIE (Phase Comparison Interrupt Enable Register) Offset: 0x10 Default: 0x00000000 Access: PHCOMPx->PHCOMPIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		PHWZCH	PHWZCL	PHVZCH	PHVZCL	PHUZCH	PHUZCL

Table 15-12: Phase Comparison Interrupt Enable Register (PHCOMPIE) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5	PHWZCH	RW	0x0	W-Phase zero-cross to high interrupt enable 0: Disable 1: Enable
4	PHWZCL	RW	0x0	W-Phase zero-cross to low interrupt enable 0: Disable 1: Enable
3	PHVZCH	RW	0x0	V-phase zero-cross to high interrupt enable 0: Disable 1: Enable
2	PHVZCL	RW	0x0	V-phase zero-cross to low interrupt enable 0: Disable 1: Enable
1	PHUZCH	RW	0x0	U-phase zero-cross to high interrupt enable 0: Disable 1: Enable
0	PHUZCL	RW	0x0	U-Phase zero-cross to low interrupt enable 0: Disable 1: Enable

Table 15-13: Phaes Comparison Control Register (PHCOMPCTL) Layout

PHCOMPCTL (Phaes Comparison Control Register) Offset: 0x14 Default: 0x00000420 Access: PHCOMPx->PHCOMPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED		PWM3BLANK	PWM2BLANK	PWM1BLANK	PWM0BLANK	MODE	HYSTSEL
15	14	13	12	11	10	9	8
HYSTSEL	PHREFSEL			PHWSEL			PHWEN
7	6	5	4	3	2	1	0
PHVSEL			PHVEN	PHUSEL			PHUEN

Table 15-14: Phaes Comparison Control Register (PHCOMPCTL) Description

Bits	Field Name	Type	Reset	Description
31:22	RESERVED_31_22	RO	0x0	Reserved.
21	PWM3BLANK	RW	0x0	Trigger blanking due to PWM3 toggling 0: Disable 1: Enable
20	PWM2BLANK	RW	0x0	Trigger blanking due to PWM2 toggling 0: Disable 1: Enable
19	PWM1BLANK	RW	0x0	Trigger blanking due to PWM1 toggling 0: Disable 1: Enable
18	PWM0BLANK	RW	0x0	Trigger blanking due to PWM0 toggling 0: Disable 1: Enable
17	MODE	RW	0x0	Operation mode 0: Phase comparator mode for back EMF sensing 1: Normal voltage comparator
16:15	HYSTSEL	RW	0x0	Comparitor hysteresis select 00: 0 mV 01: 12 mV 10: 24 mV 11: 36 mV
14:12	PHREFSEL	RW	0x0	Phase comparator reference select 000: Common-node sensed signal 001: DAC1 output 010: DAC2 output 011: ANA_IN0 pin input 100: ANA_IN1 pin input 101: ANA_IN2 pin input 110: ANA_IN3 pin input 111: ANA_IN4 pin input

Bits	Field Name	Type	Reset	Description
11:9	PHWSEL	RW	0x2	W-Phase comparitor input channel select 000: U-Phase sensed signal 001: V-Phase sensed signal 010: W-Phase sensed signal 011: ANA_IN0 pin input 100: ANA_IN1 pin input 101: ANA_IN2 pin input 110: ANA_IN3 pin input 111: ANA_IN4 pin input
8	PHWEN	RW	0x0	W-phase comparion enable 0: Disable 1: Enable
7:5	PHVSEL	RW	0x1	V-Phase comparitor input channel select 000: U-Phase sensed signal 001: V-Phase sensed signal 010: W-Phase sensed signal 011: ANA_IN0 pin input 100: ANA_IN1 pin input 101: ANA_IN2 pin input 110: ANA_IN3 pin input 111: ANA_IN4 pin input
4	PHVEN	RW	0x0	V-phase comparitor enable 0: Disable 1: Enable
3:1	PHUSEL	RW	0x0	U-Phase comparitor input channel select 000: U-Phase sensed signal 001: V-Phase sensed signal 010: W-Phase sensed signal 011: ANA_IN0 pin input 100: ANA_IN1 pin input 101: ANA_IN2 pin input 110: ANA_IN3 pin input 111: ANA_IN4 pin input
0	PHUEN	RW	0x0	U-phase comparitor enable 0: Disable 1: Enable

Table 15-15: Phase Comparison Blanking Control (PHCOMPBLANK) Layout

PHCOMPBLANK (Phase Comparison Blanking Control) Offset: 0x18 Default: 0x00000000 Access: PHCOMPx->PHCOMPBLANK							
31	30	29	28	27	26	25	24
WINSIZE							
23	22	21	20	19	18	17	16
WINSIZE							
15	14	13	12	11	10	9	8
OFFSET							
7	6	5	4	3	2	1	0
OFFSET							

Table 15-16: Phase Comparison Blanking Control (PHCOMPBLANK) Description

Bits	Field Name	Type	Reset	Description
31:16	WINSIZE	RW	0x0	Blanking window size
15:0	OFFSET	RW	0x0	Blanking window offset referred to the toggling of all PWM outputs

Table 15-17: Phase Comparison Filter Control (PHCOMPFILTER) Layout

PHCOMPFILTER (Phase Comparison Filter Control) Offset: 0x1C Default: 0x00000000 Access: PHCOMPx->PHCOMPFILTER							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				FLTTH			
15	14	13	12	11	10	9	8
FLTTH	FLTWIN				FLTDIV		
7	6	5	4	3	2	1	0
FLTDIV							

Table 15-18: Phase Comparison Filter Control (PHCOMPFILTER) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:15	FLTTH	RW	0x0	Set the filter's threshold to (FLTTH+1) 1's/0's for output to toggle
14:10	FLTWIN	RW	0x0	Set the filter's window size to (FLTWIN+1) samples
9:0	FLTDIV	RW	0x0	Set the filter's clock dividing ratio to (FLTDIV+1)

Table 15-19: Phase Comparison FSM Reset Register (PHCOMPRST) Layout

PHCOMPRST (Phase Comparison FSM Reset Register) Offset: 0x20 Default: 0x00000000 Access: PHCOMPx->PHCOMPRST							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					PHWRST	PHVRST	PHURST

Table 15-20: Phase Comparison FSM Reset Register (PHCOMPRST) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	PHWRST	W1C	0x0	Reset W-phase comparison state machine In BEMF phase comparator mode, the 1st zero-cross event will be ignored after the reset pulse 0: Write 0 has no effect. Always reads back 0. 1: Write 1 resets the FSM. This bit is self-cleared
1	PHVRST	W1C	0x0	Reset V-phase comparison state machine In BEMF phase comparator mode, the 1st zero-cross event will be ignored after the reset pulse 0: Write 0 has no effect. Always reads back 0. 1: Write 1 resets the FSM. This bit is self-cleared
0	PHURST	W1C	0x0	Reset U-phase comparison state machine In BEMF phase comparator mode, the 1st zero-cross event will be ignored after the reset pulse 0: Write 0 has no effect. Always reads back 0. 1: Write 1 resets the FSM. This bit is self-cleared

Table 15-21: Phase Comparison Register Write-Allow Key Register (PHCOMPREGKEY) Layout

PHCOMPREGKEY (Phase Comparison Register Write-Allow Key Register) Offset: 0x24 Default: 0x1ACCE551 Access: PHCOMPx->PHCOMPREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 15-22: Phase Comparison Register Write-Allow Key Register (PHCOMPREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected registers

16 Buffered 10-bit DAC

16.1 Buffered DAC overview

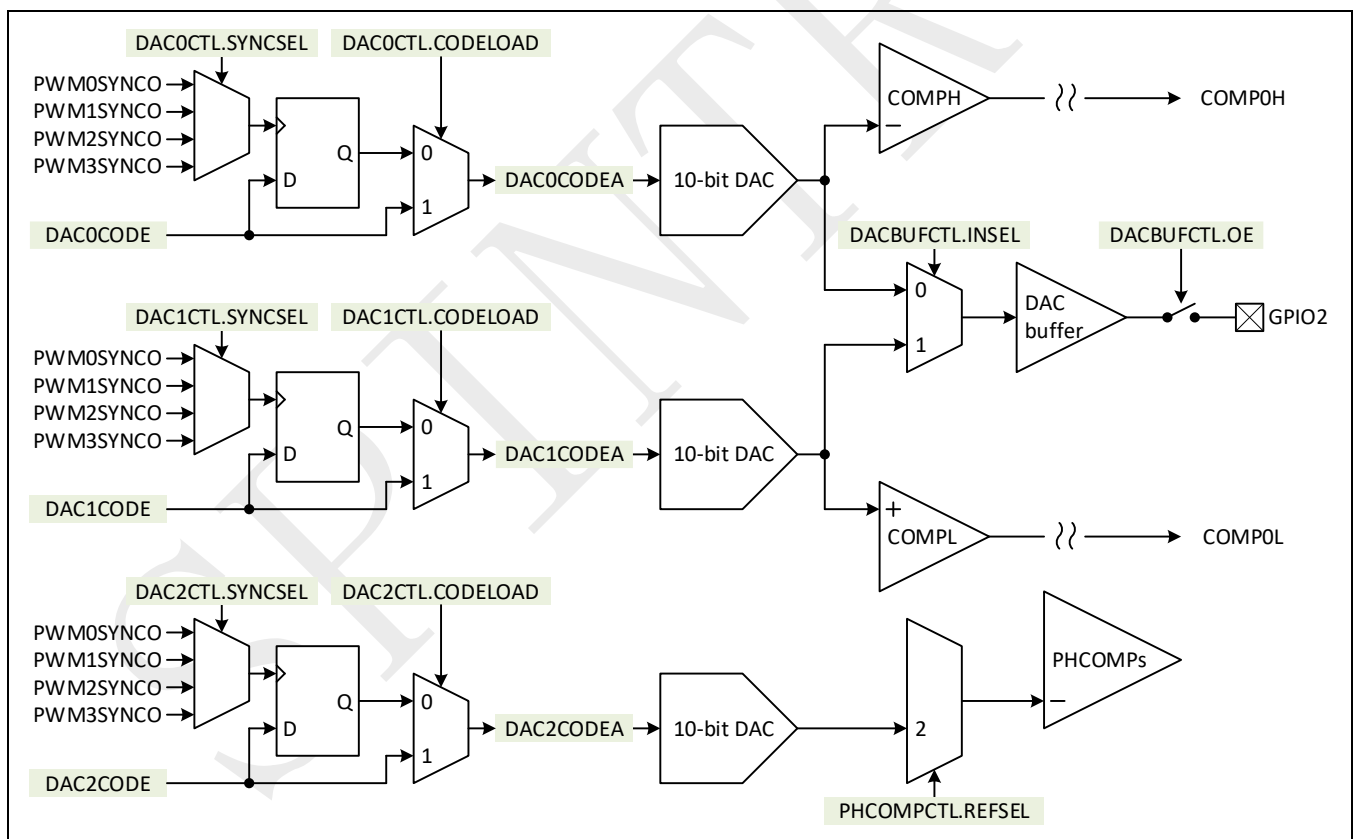
The buffered DAC module consists of three internal 10-bit DACs and an analog output buffer that is capable of driving external load. As shown in Figure 16-1, DAC0 and DAC1 can be sent to the DAC buffer, while DAC2 are used as internal reference for phase comparator. The buffered DAC is a general purpose DAC that can be configured to generate a DC voltage and AC waveforms such as triangle, square waves and so forth. Each DAC output is divided from DVDD33 pin with 10-bit resolution. Given the active code as DACxCODEA for DACx, the output is:

$$V_{DACx} = V_{DVDD3} / 1024 * DACxCODEA$$

16.2 Buffered DAC architecture

The block diagram for the buffered DAC is shown in Figure 16-1.

Figure 16-1: Buffered DAC block diagram



Buffered DAC module has two sets of DAC code registers: DACxCODE and DACxCODEA. DACxCODEA is a read-only active code that directly controls the buffered DAC value, and DACxCODE is a writable shadow register with two operation modes below:

- Immediate load (DACxCTL.CODELOAD = 1)
 - Write to DACxCODE also update the DACxCODEA
- Shadow load (DACxCTL.CODELOAD = 0)

- DACxCODEA is updated as DACxCODE upon the PWMMySYNCO defined by DACxCTL.SYNCSEL.

The high and low thresholds of the comparator are generated by two DACs, selected from the [Table 14-1](#).

SPC1169 implements one internal DAC buffer, each DAC's output can be sent to the DAC buffer depending on the DAC buffer MUX setting (DACBUFCTL.INSEL), shown as [Table 16-1](#).

Table 16-1: DAC buffer MUX

DACBUFCTL.INSEL	DAC buffer output
0	DAC0
1	DAC1

DAC buffer output can be directly sent to the ADC through ADC input MUX (See in ADC input selection: SHINSELP=8). DAC buffer output can also be sent to the ANA_IN2 pin, depends on the setting DACBUFCTL.OE.

16.3 Ramp generator

The ramp generator can produce a falling ramp reference (REFP voltage) for the too high compactor, see [Figure 14-1](#). DAC0 and DAC1 have ramp capability.

[Figure 16-2](#) shows ramp generator diagram. There's two sets of registers for ramp, one is read-only active registers, named RAMPxDLYA, RAMPxDECA, and RAMPxMAXA, which can directly control ramp function. The other is a set of writable shadow registers called RAMPxDLY, RAMPxDEC, and RAMPxMAX. Those shadow registers will load to read-only registers only under certain conditions, for example rising edge of RAMPEN or PWMSYNC, as shown in [Figure 16-2](#).

The most significant 10bits of RAMPxCNT will be sent to register CODEA, which controls 10-bit DAC output. RAMPxMAXA is initial DAC value, RAMPxDLYA is delay counter. For every PWMCLK cycle, RAMPxDLYA will decrease one until it reaches zero. RAMPxDECA controls DAC decrease step for each time. If RAMPxDLYA is zero, RAMPxCNT value will subtract RAMPxDECA.

When ramp generator is enabled (RAMPEN=1), RAMPxCNT register will first load from RAMPxMAX, and the register remains stable until the PWMSYNC is triggered. Then for every PWMCLK cycle, RAMPxDLYA will decrease one. When RAMPxDLYA reaches 0, RAMPxCNT value will subtract RAMPxDECA. Afterwards RAMPxDLYA will load initial value for the next PWMCLK cycle and repeat. Therefore, DAC output will decrease RAMPxDECA step for each (RAMPxDLYA+1) PWMCLK cycles.

When RAMPLOAD=0, ramp generator block is in shadow mode. This means that while COMPH is asserted, RAMPxMAX value will not load to RAMPxMAXA immediately. RAMPxCNT will be controlled by the previous RAMPxMAXA value, and updated RAMPxMAX value will load to RAMPxMAXA only at the next PWMSYNC rising edge. If RAMPLOAD=1, ramp generator block is in direct mode, which means that when COMPH is asserted, RAMPxMAX value will directly load to RAMPxMAXA and control RAMPxCNT.

Figure 16-2: Ramp generator diagram

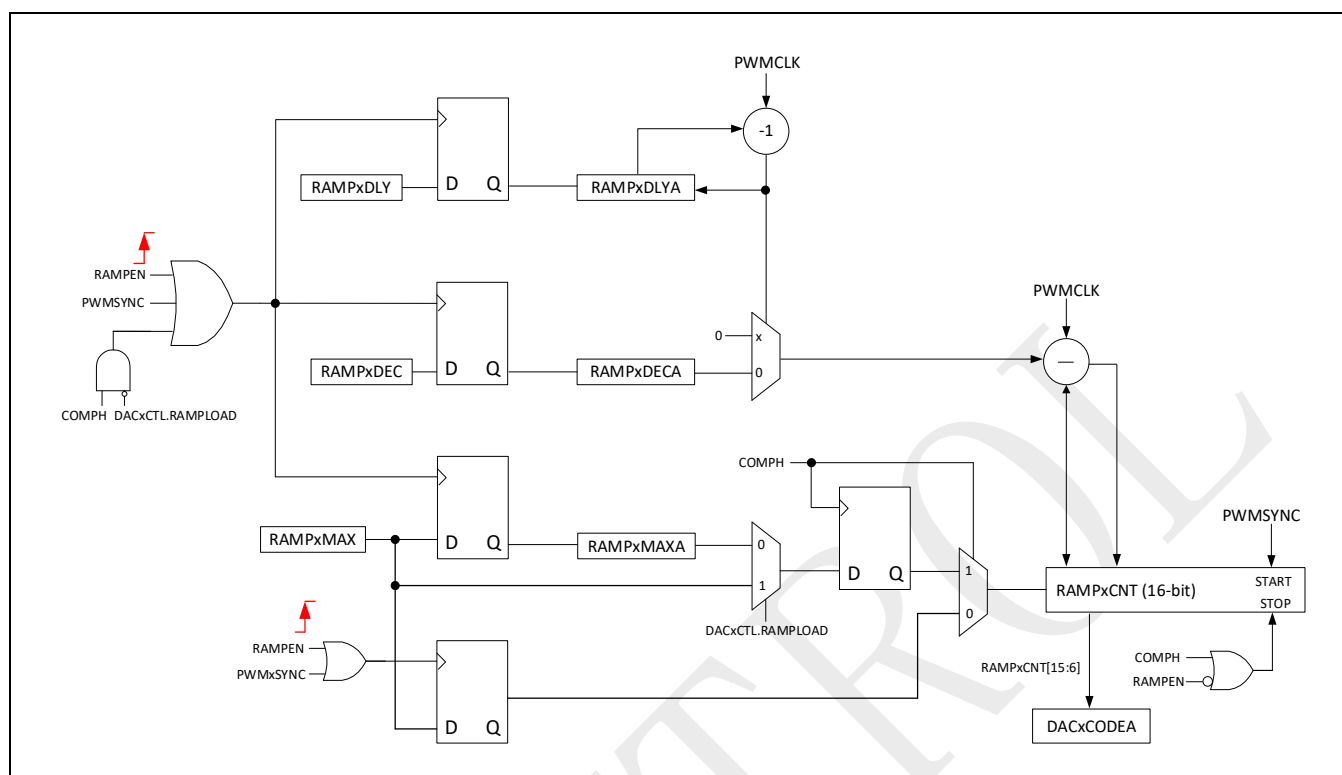
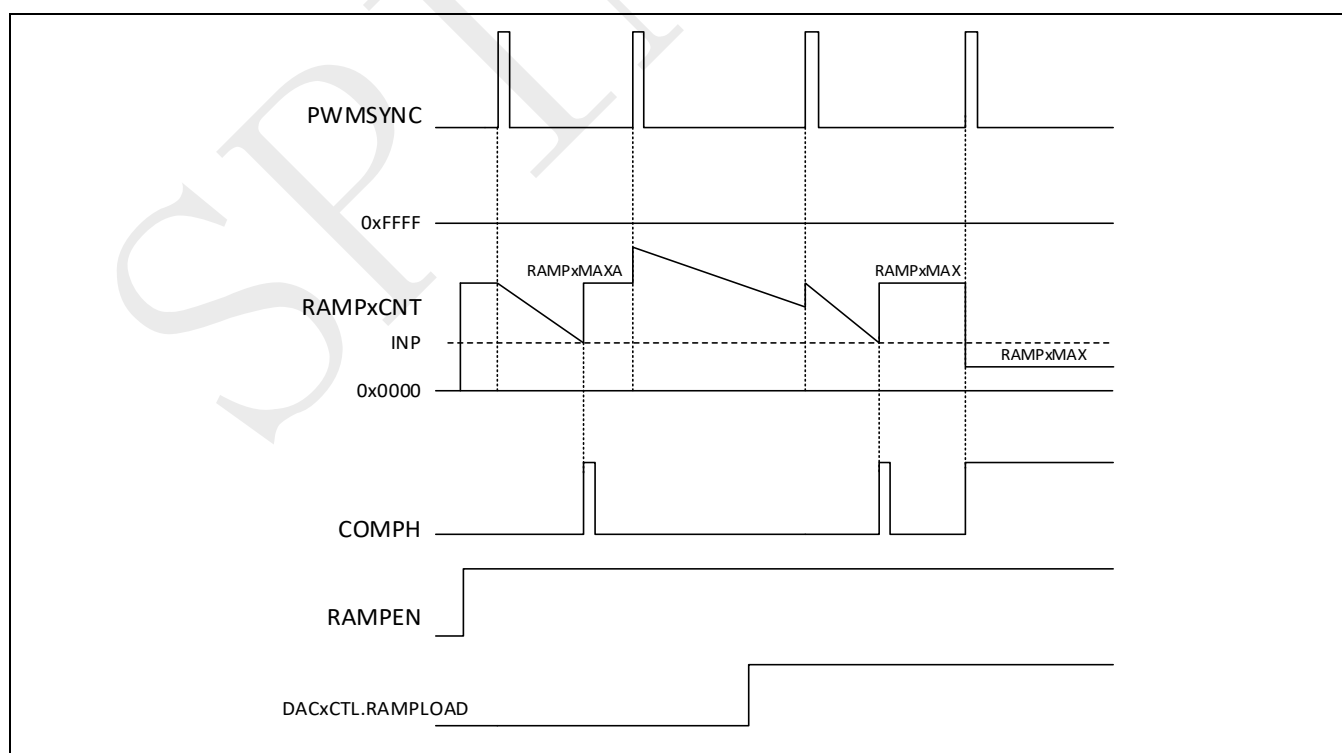


Figure 16-3 shows an example for DAC output (same as RAMPxCNT) during PWMSYNC triggered and COMPH asserted. Suppose the positive input to the too high comparator is fixed (INP in Figure 16-3).

Figure 16-3: Ramp generator waveform



16.4 Registers

Please see [Section 14.5](#) for DAC related register map and definitions.

SPIN TROL

17 D2S buffer

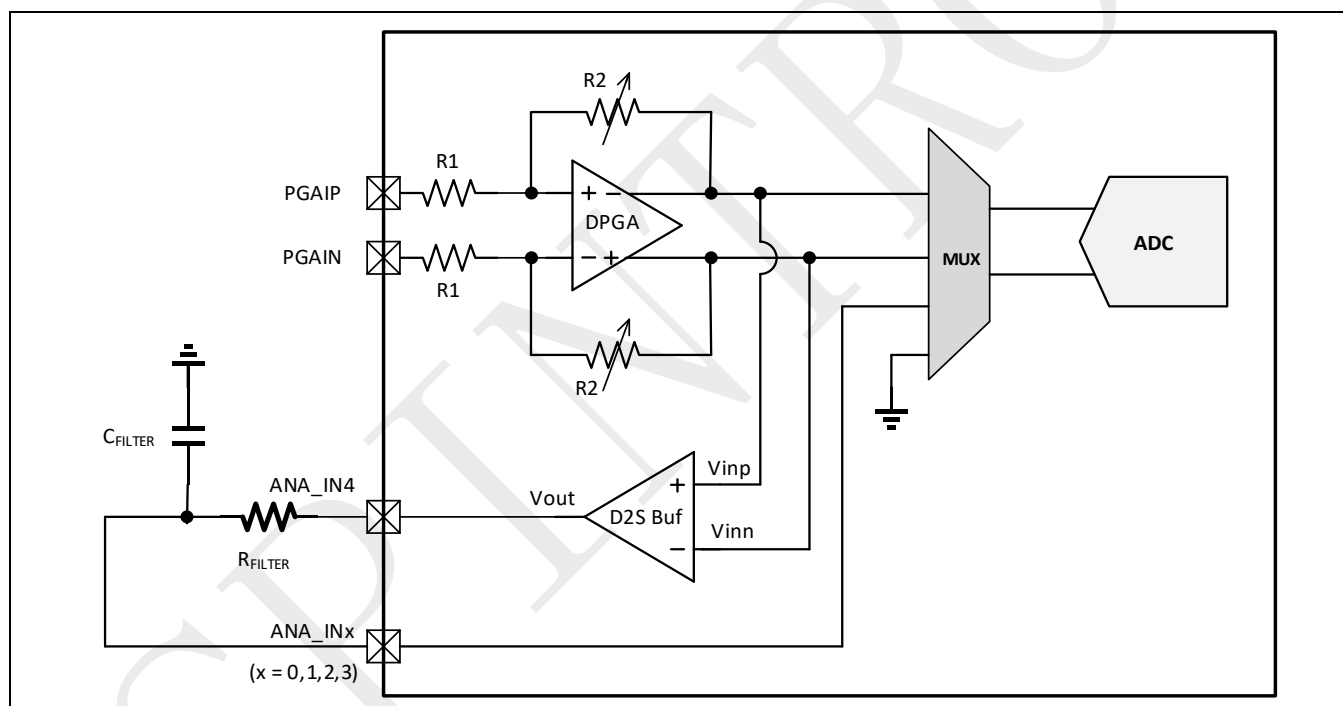
17.1 Overview

The SPC1169 implement one D2S (differential to single-ended) buffer, which can convert DPGA's differential output to singled-ended output, and then send it to ANA_IN4 pin. Through external RC filter (typically, $R = 1k\Omega$, $C = 1\mu F$), the singled-ended output can be sent back to ANA_INx ($x = 0\sim 3$) pin and measured by 13-bit ADC. Thus, 13-bit ADC can get the DC average voltage of sensing resistor.

17.2 Function description

Figure 17-1 shows the application diagram of D2S buffer.

Figure 17-1: D2S buffer application diagram



[1] Typically, $R_{\text{FILTER}} = 1k\Omega$, $C_{\text{FILTER}} = 1\mu F$.

D2S buffer gain is fixed at 0.5 and the equation for D2S buffer input and output is:

$$V_{\text{out}} = (V_{\text{inp}} - V_{\text{inn}}) \times 0.5 + V_{\text{DD3}}/2$$

Typically, $V_{\text{DD3}} = 3.3V$, so $V_{\text{out}} = (V_{\text{inp}} - V_{\text{inn}}) \times 0.5 + 1.65$. And DS2 buffer output range is from 0.3V to $V_{\text{DD3}} - 0.3V$.

Note: User can call `D2S_Enable()`/`D2S_Disable()` functions in SDK to enable/disable D2S buffer.

18 Temperature sensor

18.1 T-sensor overview

The SPC1169 implements an internal temperature sensor to measure the junction temperature of the device. The sensor output can be sampled with the 13-bit ADC, which was shown in [Figure 10-1](#).

18.2 Transfer function

The temperature sensor output and the resulting ADC values increase with increasing junction temperature.

The following equation shows the relationship between the temperature and the ADC code:

$$\text{Temperature} = (\text{ADC_CODE} - T_{\text{OFFSET}}) * T_{\text{SLOPE}} + 25$$

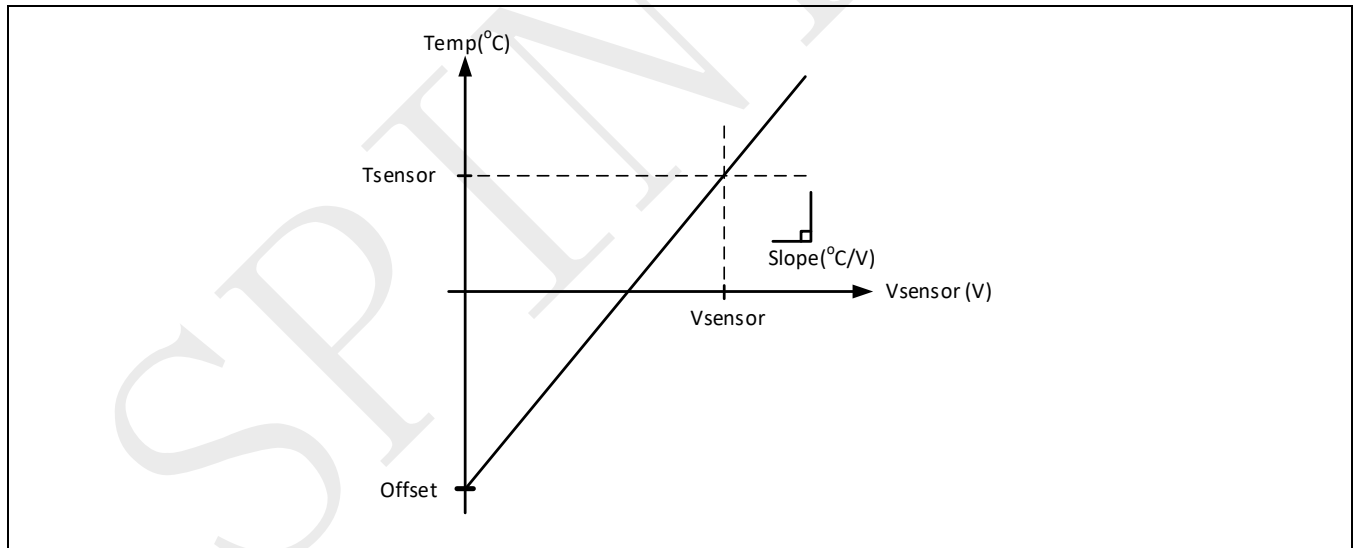
Where $T_{\text{SLOPE}} = 3.808\text{ }^{\circ}\text{C}$, $T_{\text{OFFSET}} = 81$.

The relationship between the temperature and the sensor output voltage (Vsensor) is shown in [Figure 18-1](#), the transfer function of the sensor is defined as:

$$\text{Temperature} = \text{Slope} \times \text{Vsensor} + \text{Offset}$$

Where Slope = $4238.97\text{ }^{\circ}\text{C/V}$, Offset = $-283.36\text{ }^{\circ}\text{C}$.

Figure 18-1: Temperature sensor transfer function



So the measure temperature step is shown below:

- Step 1. In target temperature T, measure the sensor output as Vsensor. The Vsensor should be calibrated using ADC gain and offset trim value.
- Step 2. Using the above formula to calculate the target temperature.

18.3 Registers

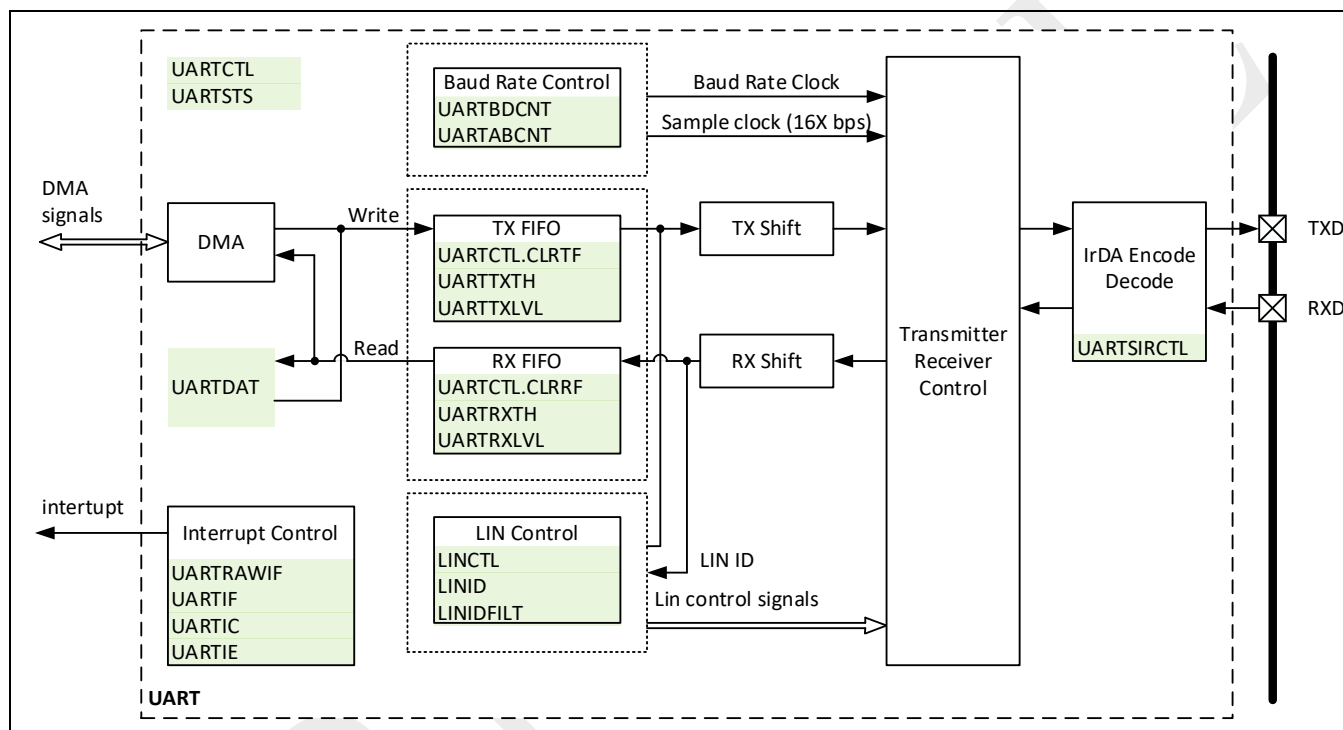
Please see [Table 10-72](#) and [Table 10-73](#) for T-sensor related register definitions.

19 UART

19.1 UART overview

The SPC1169 implements two Universal Asynchronous Receiver/Transmitter (UART) modules. The port contains a UART, a slow serial infrared Transmit encoder and Receive decoder that conform to the IrDA serial infrared specification, LIN control module. The block diagram is shown in [Figure 19-1](#).

Figure 19-1: UART block diagram



The UART port performs serial-to-parallel conversion on data bytes received from a peripheral device and parallel-to-serial conversion on data bytes received from CPU.

The CPU can read a complete UART status from UARTSTS. Status information includes the type and condition of transfer operations and error conditions (parity, overrun, framing or break interrupt) associated with UART.

The UART have a Transmit FIFO (64-bytes depth) and a Receive FIFO (64-bytes depth). Transmit FIFO holds data from the CPU until it is transmitted on the serial link; Receive FIFO buffers data from the serial link until it is read by the CPU. Write register UARTDAT will write new data to Transmit FIFO; Read register UARTDAT will get received data from Receive FIFO.

The UART includes a programmable baud-rate generator that can divide the input module clock by 16 to $(2^{16}-1)$, which produces a baud rate clock and a sample clock (16X bps) that can be used to drive the internal transmit and receive logic. This module also support auto baud rate detect. The UART port supports baud rates of 9600, 19.2K, 38.4K, 57.6K, 115.2K, 230K, 460K, 921K, 1.8M, 3.6M and so on.

Software can program interrupts to meet its requirements, which minimizes the number of computations required to handle the communications link. UART operates in an environment that is either controlled by software (polling or interrupt) or DMA driven.

The UART also support LIN control functions.

19.2 UART features

The UART has the following features:

- Independently controlled transmit, receive, line status and data-set interrupts
- Programmable serial interface:
 - 5 – 8 data bits
 - Even, odd or no parity detection
 - One, one-and-a-half, or two stop bits generation
 - Baud-rate generation up to 12.5 Mbps
 - False start-bit detection
 - Break signal send and detection
- Auto baud-rate detection
- Receive data 16X oversample
- Non-Return to Zero (NRZ) encoding/decoding function
- 64-bytes Transmit FIFO
- 64-bytes Receive FIFO
- Programmable FIFO trigger threshold
- Complete status-reporting capability
- Ability to generate and detect line breaks
- Internal diagnostic capabilities that include:
 - Loopback controls for communications link fault isolation
 - Break, parity and framing-error simulation
- Serial Infrared asynchronous interface that conform to the Infrared Data Association(IrDA) specification
- LIN function support
- DMA support

19.3 UART signal descriptions

Table 19-1 lists and describes each external signal that is connected to a UART module. The pins transmit digital CMOS-level signals and are connected to through GPIOs. Please refer to the GPIO pin configuration registers for details.

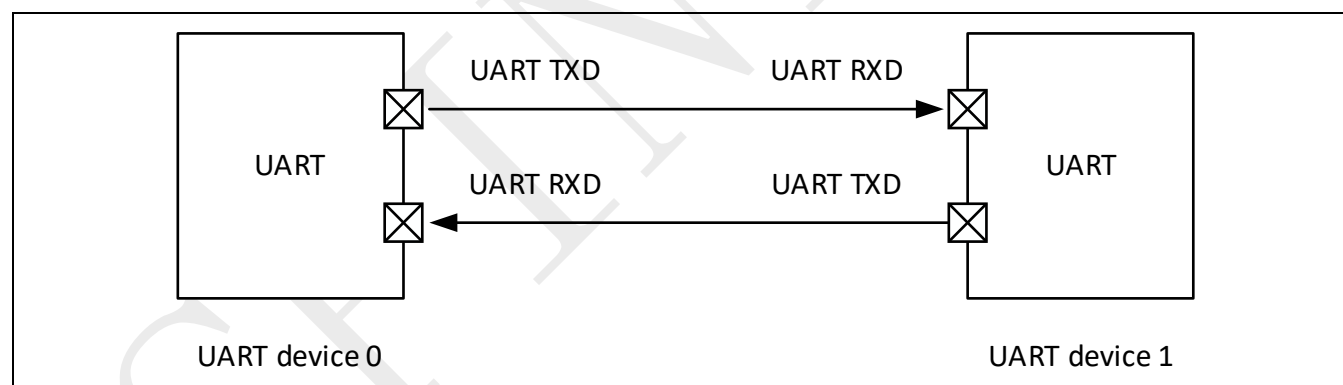
Table 19-1: UART signal descriptions

Name	Type	Descriptions
RXD	Input	Serial Input Serial data input to the Receive Shift register; in Infrared mode, it is connected to the infrared receiver input; in lin mode, it is connected to lin phy RXD pin.
TXD	Output	Serial Output Serial data output to the communication-link peripheral or data set. The TXD signal is set to the logic 1 state upon a reset operation; in infrared mode, It is connected to the output of the infrared transmitter; in lin mode, it is connected to lin phy TXD pin.

19.3.1 UART device connection

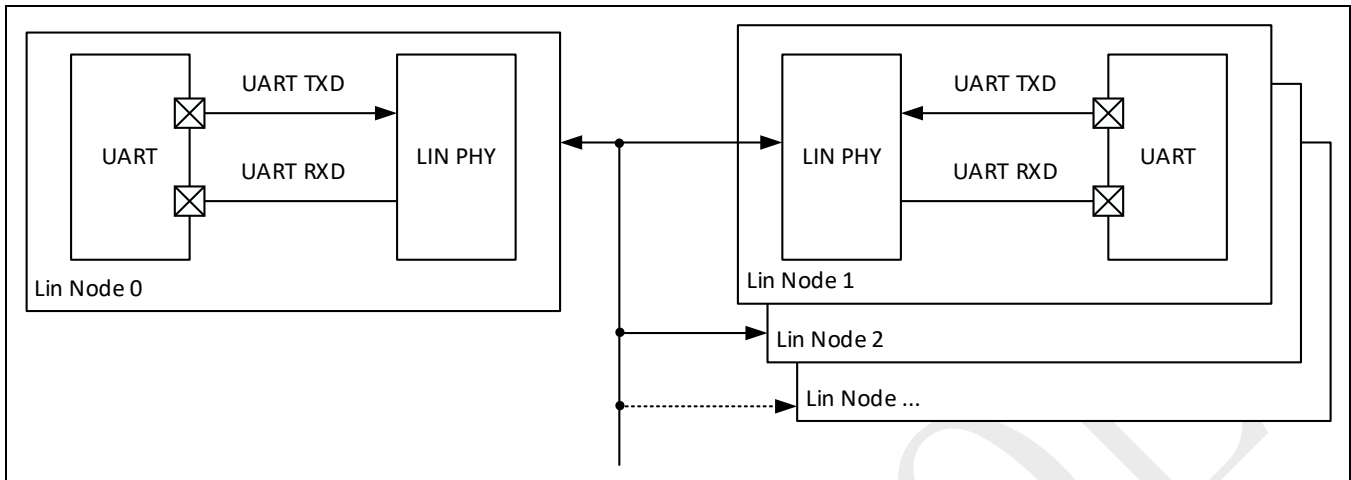
When UART connect with other modules, the connection diagram is shown in Figure 19-2.

Figure 19-2: UART device connection diagram



19.3.2 LIN device connection

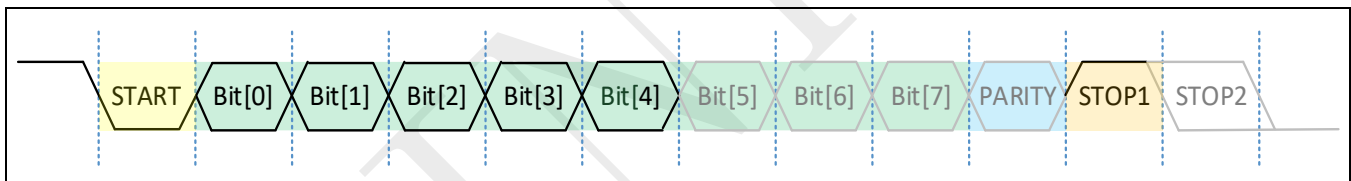
When UART worked under LIN mode, the connection diagram is shown in Figure 19-3. For more LIN knowledges, please reference LIN specification.

Figure 19-3: UART LIN device connection diagram

For details about LIN PHY, please refer to [Chapter 19.5](#).

19.4 UART operation

[Figure 19-4](#) shows the format of a UART data frame.

Figure 19-4: UART frame

The receive-data sample-counter frequency is 16 times the value of the bit frequency (sample clock is 16X bps). The 16X sample clock is created by the baud-rate generator. Each bit is sampled three times in the middle. Shaded bits in [Figure 19-4](#) can be programmed by software.

Each frame is between 7 and 12 bits long, depending on the size of the data bit programmed, whether parity is enabled and stop bit programmed. A frame begins by transmitting a start bit that is represented by a high to low transition. The start bit is followed by 5~8 bits of data that begin with the Least Significant bit (LSB). The data bits are followed by an optional parity bit. The data frame ends with 1/1.5/2 stop bit.

If frame parity enabled, frame parity bit is set when:

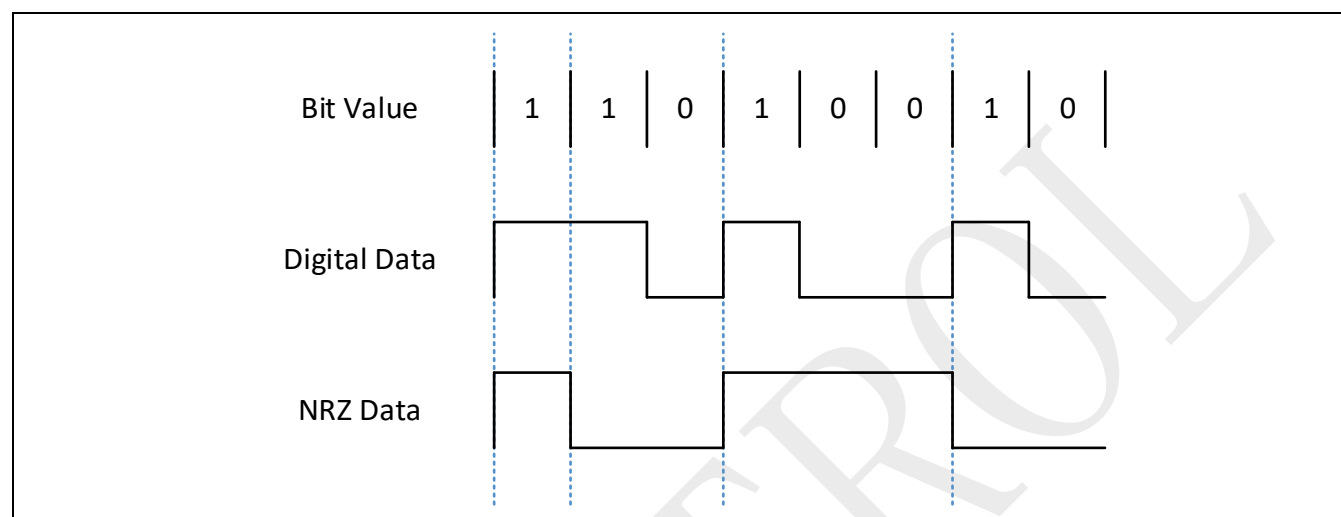
- even parity is enabled and the data byte has an odd number of ones;
- or odd parity is enabled and the data byte has an even number of ones;
- or set parity bit sticky high.

When UARTCTL.STOPBIT is set for 2-bits stop and UARTCTL.DATABIT is set for 5-bits data, the trully stop bit width is 1.5 bits width.

The UART module has two FIFOs: one Transmit and one Receive. The Transmit FIFO is 64 bytes deep and 8 bits wide. The Receive FIFO is 64 bytes deep and 11 bits wide. Three bits are used for tracking errors, will reflect in UARTSTS when read Receive FIFO. Read UARTDAT will only get another 8 bits data.

The UART can use NRZ coding to represent individual bit values. To enable NRZ coding, set the UARCTL.NRZMODE bit. A bit value of 1 is represented by a line transition, and 0 is represented by no line transition. Figure 19-5 shows the data byte 0b0100_1011 in NRZ coding. The LSB in the byte is transmitted first.

Figure 19-5: Example NRZ bit Encoding – 0b0100_1011



19.4.1 Reset

The UART is disabled on reset. To enable it, software must program the GPIO Pin Configuration registers, and then set the UARCTL.TXEN and UARCTL.RXEN. When the UART is enabled, the receiver waits for a frame start bit and the transmitter sends data if Transmit FIFO not empty. Transmit data can be written to the Transmit FIFO before the UART unit is enabled.

When the UART is disabled by clear UARCTL.TXEN and UARCTL.RXEN, the transmitter or receiver finishes the current byte and stops transmitting or receiving more data. Data in the FIFO is not cleared and transmission resumes when the UART is enabled. User can clean FIFO by write 1 to UARCTL.CLRTF and UARCTL.CLRRF.

19.4.2 FIFO operation

The UART has a Transmit FIFO and a Receive FIFO, with each FIFO holding 64 bytes of data. There is two one method for moving data into and out of the FIFOs: access UARTDAT directly and access UARTDAT by DMA.

19.4.2.1 FIFO polling mode operation

CPU access UARTDAT directly is called polling mode for FIFO operation. In polling mode, software checks receiver and transmitter status via UARTSTS, UARTRAWIF. The CPU polls the following bits for receive and transmit data service:

- Receive Data Service – the CPU checks the UARTRAWIF.RXREQ, which is set when one or more bytes remain in the Receive FIFO.

- Transmit Data Service – the CPU checks the UARTRAWIF.TXREQ, which is set when the transmitter needs data.

The CPU can also check the UARTSTS.TXEMPTY, which is set when the Transmit FIFO is empty.

For more status or raw flag bits functions, please reference register map.

19.4.2.2 FIFO DMA mode operation

Under this mode, FIFO are controlled by DMA.

When Transmit FIFO data request event occurred, DMA need move data to Transmit FIFO; And when Receive FIFO data request event occurred, DMA need read data and move to usr specified memory space.

See DMA control chapters for more knowledges about DMA.

When timeout occurred, usr need clear trailing bytes, and need do these action based on DMA end of transfer chain (UARTRAWIF.DMAEOC) flag.

19.4.3 Timeout and removing trailing bytes

A receive timeout occurs when all of the following conditions exist:

- Receive FIFO level lower than its threshold.
- The most recently received byte was received more than four continuous byte times ago.

When read one byte from the Receive FIFO or a new start bit is received, the timeout counter will reset.

When timeout occurred, the bytes in Receive FIFO called trailing bytes. And CPU must remove trailing bytes. When servicing a byte timeout interrupt, the CPU uses the following procedure:

1. Disable the receiver timeout interrupt via UARTIE.RXTIMEOUT.
2. Read Receive FIFO by UARTDAT.
3. Read the UARTSTS and check for errors.
4. Loop back to step 2. If the UARTSTS.RXNE is clear, go to step 5.
5. No more data in Receive FIFO now. Re-enable UARTIE.RXTIMEOUT.
6. Done.

19.4.4 Programmable baud-rate generator

The UART contains a programmable baud-rate generator that can take a fixed-input clock and divide it down to generate the preferred baud rate. The baud rate is calculated by taking the UART peripheral clock (CLK_UART).

The baud-rate generator will generate baud rate clock and a sample clock used for sample RXD data. The sample clock is 16X bps clock. UARTBDCNT store the divisor in a 16-bit binary format. Load these divisor latches during initialization to ensure that the baud-rate generator operates properly. The UARTBDCNT value can't be set lower than 16, and hardware will force set 16 if CPU write value lower than 16. The baud rate of the data shifted into or out of a UART is given by the formula:

$$\text{BaudRate} = \frac{\text{CLK_UART}}{\text{UARTBDCNT}}$$

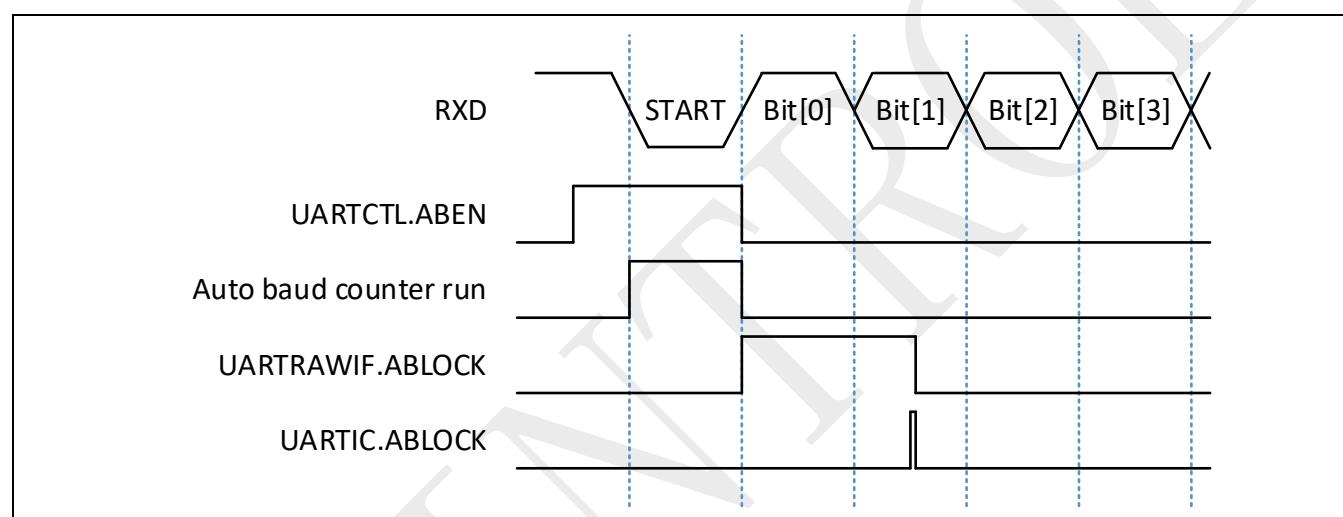
For example: if the Divisor is 1302 and UART clock is 50 MHz, the baud rate is 38402 bps.

Note: Changing the baud rate is not permitted when actively transmitting or receiving data.

19.4.5 Auto-baud-rate detection

The UART supports auto-baud-rate detection.

Figure 19-6: Example of auto baud



As shown in Figure 19-6, when enabled (UARTCTL.ABEN), the UART counts the numbers of clock cycles (CLK_UART) within the start-bit pulse. This number is then written into the UARTABCNT and UARTBDCNT (UARTCTL.ABUP should be 0). The first bit of received byte should be 1.

When the UARTABCNT is written by hardware, an auto-baud-lock interrupt is generated (if enabled), UARTRAWIF.ABLOCK is set. If preferred, the CPU can read the UARTABCNT and use this information to program the UARTBDCNT with a baud rate calculated by the CPU (UARTCTL.ABUP is 1). After the baud rate has been programmed, the CPU verifies that the predetermined bytes (usually AT or at) are being received correctly.

When the baud rate is detected, the auto-baud circuitry disables itself by clearing the UARTCTL.ABEN. To re-enable auto-baud detection, set the UARTCTL.ABEN field again.

Note: Changing the baud rate is not permitted when actively transmitting or receiving data. Auto-baud-rate detection is not supported in IrDA (serial infrared) mode or LIN mode.

19.4.6 Serial infrared asynchronous interface

The Slow Serial Infrared (SIR) interface is used with the UART to support two-way wireless communication that uses infrared transmission. The SIR interface provides a Transmit encoder and Receive decoder to support a physical link that conforms to the IrDA Serial Infrared Specification.

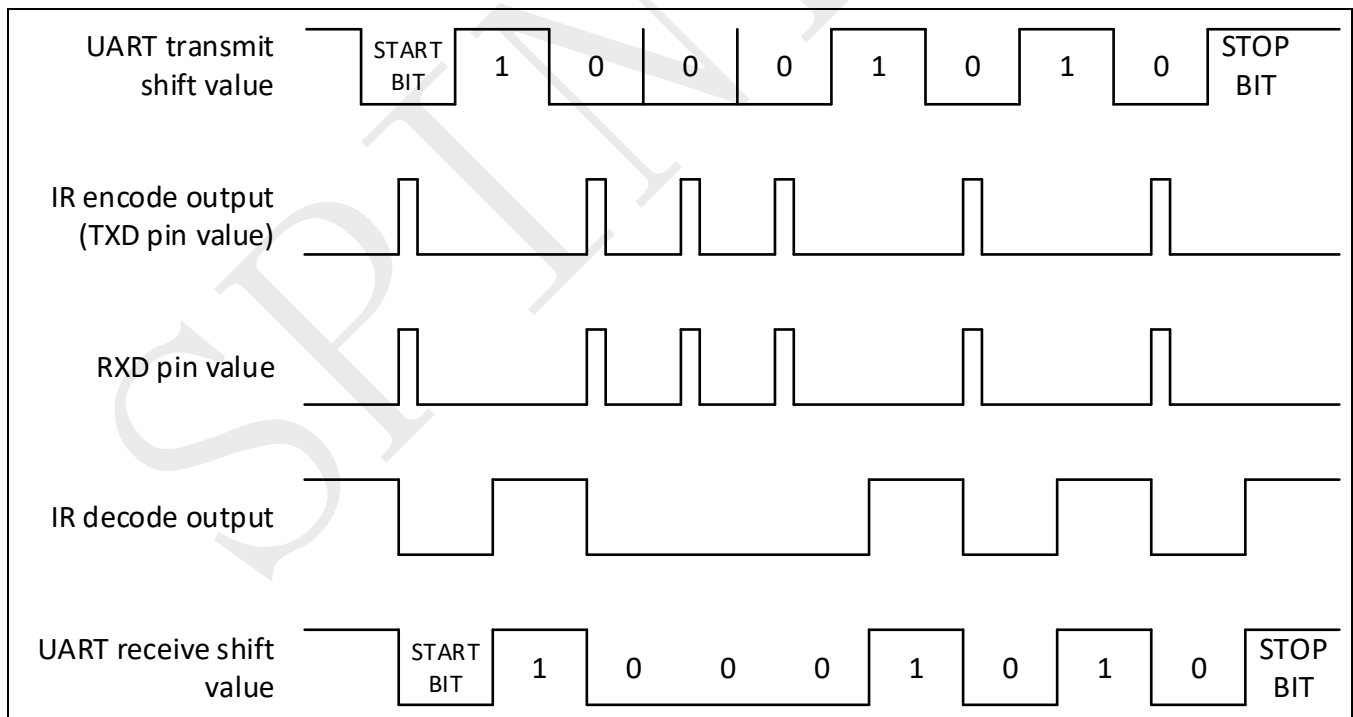
The SIR interface does not contain the actual IR LED driver or the receive amplifier. The I/O pins attached to the SIR interface have only digital signals. SIR supports two-way communication, and full-duplex communication is not used because reflections from the transmit LED will enter the receiver.

Set `UARTSIRCTL.TXSIREN`, `UARTSIRCTL.RXSIREN` will separately enable TXD, RXD infrared encode or decode functions.

Note: This SIR interface should be turned on and off only while the UART module is completely idle and disabled. On-the-fly changes are likely to produce problems and are not recommended. UART default set active level is 3/16 bit of frame byte time, set `UARTSIRCTL.XMODE`, `UARTSIRCTL.TXPW` if default active drive time not suitable.

The encoding scheme sends a pulse 3/16 of a bit wide in the middle of every zero-value bit and sends no pulses for bits with a value of one. The pulse for each zero-value bit must occur, even for consecutive bits with no edge between them. [Figure 19-7](#) shows an example of Transmit and Receive operation.

Figure 19-7: IR transmit and receive example



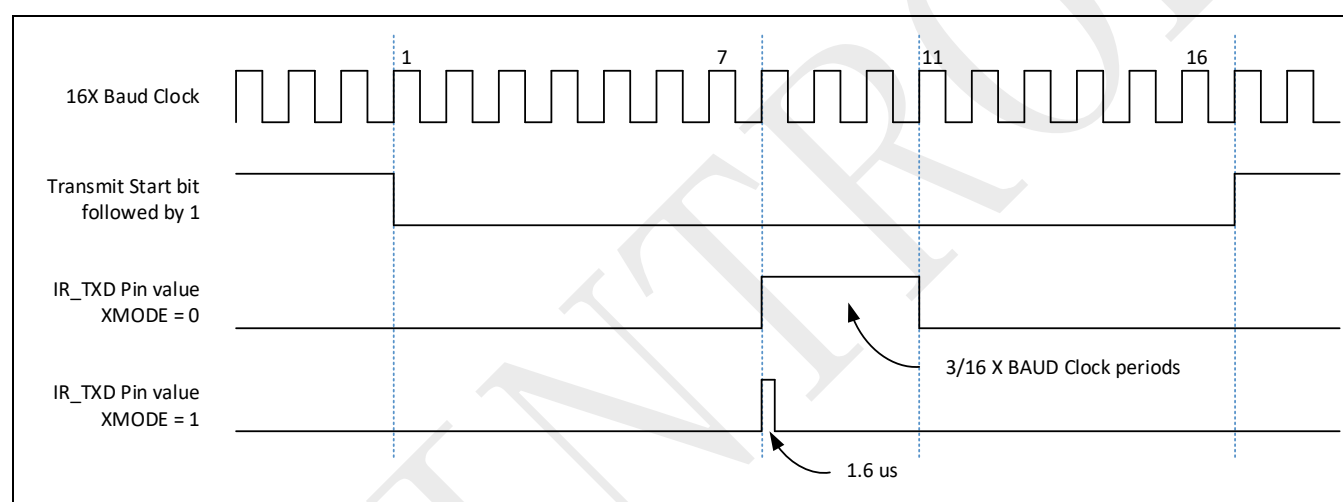
The top line in [Figure 19-7](#) shows an asynchronous transmission as it is sent from the UART. The second line shows the pulses generated by the IR encoder at the TXD pin. A 3/16 bit pulse is generated in the middle of the START bit and any data bit that is a zero. The third line shows the values received at the RXD input pin. The fourth line shows the receive-decoder output. The Receive decoder drives the receive

data line low when it detects a pulse. The bottom line shows how the UART receiver interrupts the decoder action. This last line is the same as the first, but it is shifted half a bit period.

Spintronic strongly recommends setting the UARTISR.XMODE for the transmit-pulse width. When the UARTISR.XMODE is clear, each zero bit has a pulse width of 3/16 of a bit time. The UARTISR.XMODE field must be set to 0x1 so a pulse of specified time (UARTSIRCTL.TXPW) is generated in the middle of each zero bit. The minimum value of UARTSIRCTL.TXPW is 1, written value below 1 will be reset as 1 by hardware. The shorter infrared pulse generated when the UARTISR.XMODE field is set reduces the LED power consumption. At 2400 bps, the LED is normally on for 78us for each zero bit that is transmitted. When the UARTISR.XMODE field is set, UARTSIRCTL.TXPW is 80 and CLK_UART is 50MHZ, the LED is on for only 1.6us.

Figure 19-8 shows an example of XMODE operation.

Figure 19-8: XMODE example



To prevent transmitter LED reflection feedback to the receiver, disable the IR Receive decoder when the IR Transmit encoder transmits data, and disable the IR Transmit encoder Receive decoder when the IR Receive decoder receives data.

Note: This restriction specifically is to ensure no feedback; no filtering is performed to remove this feedback data.

19.5 LIN operation

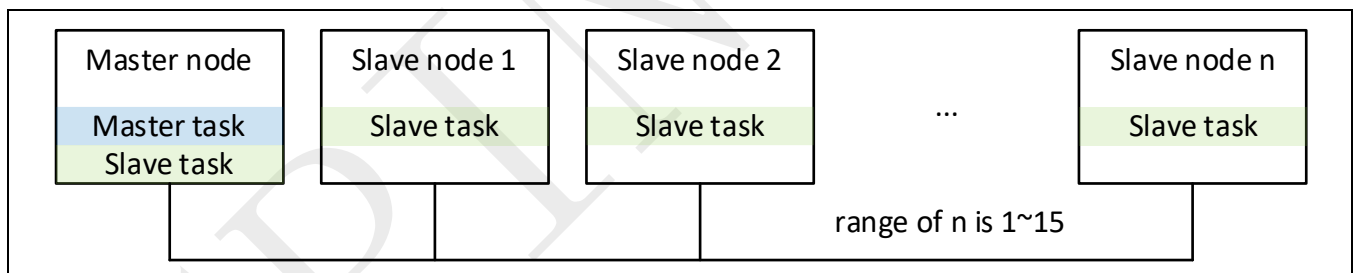
Local Interconnect Network (LIN) is a concept for low cost automotive networks, which complements the existing portfolio of automotive multiplex networks. UART LIN function is based on UART, and add some extra functions for LIN protocol. Set LINCTL.LINEN enable LIN functions. UART LIN support below functions:

- Support master or slave mode
- Hardware send frame head
- Programmable break field length
- Programmable break delimiter length
- Automatically protected ID parity bit calculation
- Support LIN ID receive filter
- Support LIN bit error, check sum error, ID matched interrupts

19.5.1 LIN protocol

As shown in Figure 19-9, LIN bus topological graph include one master node and up to 15 slave node. Each node have a slave task. Only master node have a master task, which take charge of sending frame head. Slave node only have slave task, and send frame response.

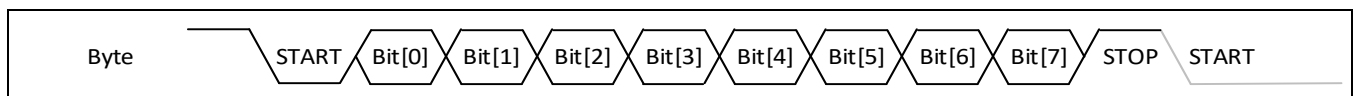
Figure 19-9: LIN bus topology



19.5.2 Frame format

LIN byte field is defined as 1 start bit, 8 data bits, and 1 stop bit. And no parity bit. As shown in Figure 19-10.

Figure 19-10: LIN byte field

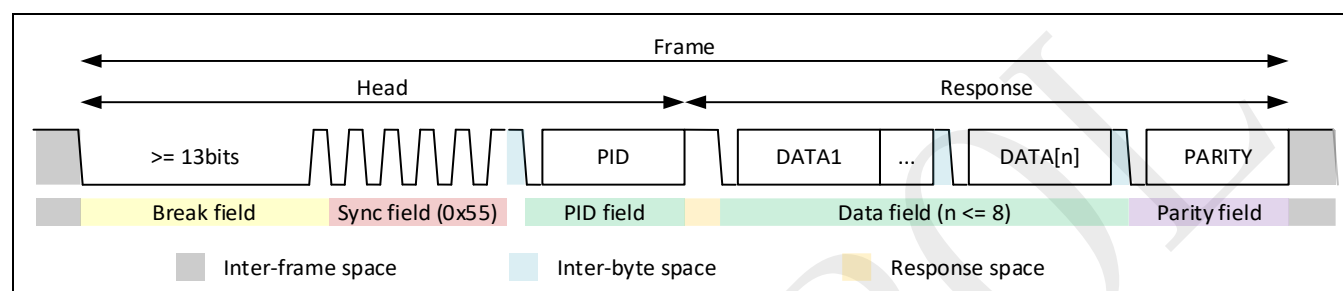


UART LIN can be configured as master node or slave node (LINCTL.MASTER). And master node can do slave task.

LIN frame and transfer is defined as Figure 19-11 and Figure 19-12. Master task send frame head, and selected slave task send response. LIN frame consist of frame head and response. Except break field, all other fields is byte field defined as Figure 19-10 shown.

Master task first send break field which length is bigger than 13 bits (LINCTL.SBRKLEN), and ended with 1 break delimiter (which length is bigger than 1bits (LINCTL.SBRKDEL)). Slave tasks only need detect 11 consecutive dominant bit and treat it as detected break field, then wait break delimiter.

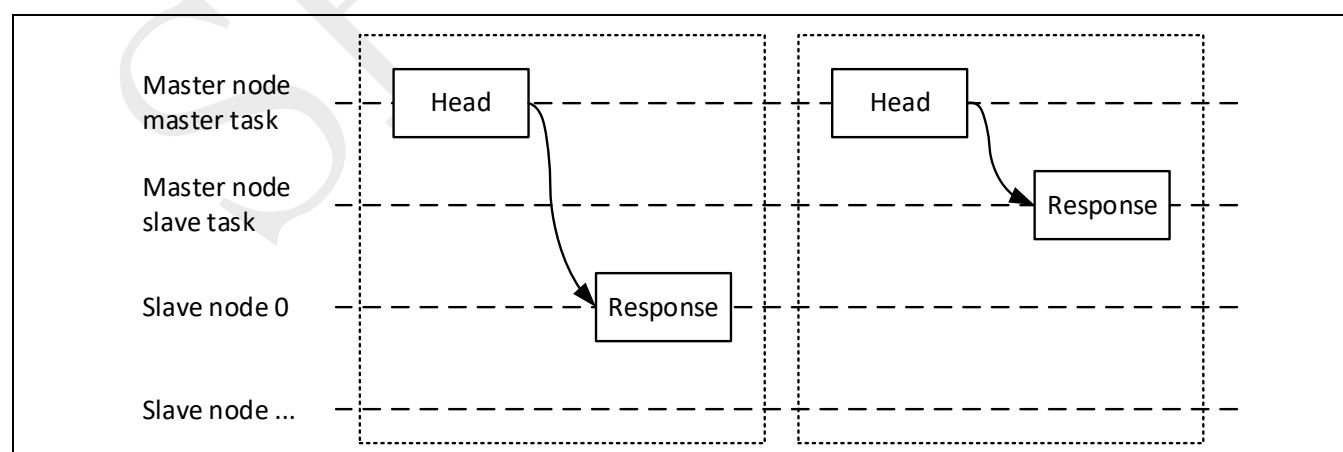
Figure 19-11: LIN frame



Then master task send sync field (0x55) and protected ID (PID), finish frame head sending. Slave receive sync field data and adjust its baud rate (From the description above, it can be inferred that only after correctly identifying the "Break field," can the baud rate adjustment phase be entered. In this phase, the formula for calculating the baud rate magnitude is: **(Time of the falling edge of the 7th bit - Time of the falling edge of the start bit) / 8**. If the automatic baud rate function is enabled, upon detecting the falling edge of the 7th bit of the "Sync field," an automatic baud rate lock interrupt will be generated, and UARTRWIF.ABLOCK will be set.), and receive PID. Then each slave task handle frame response based on PID, which function is predefined by LIN system designer. Only one slave task can send frame response, and other slave tasks can receive this response or do nothing. Frame response data length is controlled by LINCTL.RESLEN.

If slave task handle response, need set LINCTL.RESP value to 0x2; if slave task only receive other response, need set LINCTL.RESP value to 0x1; default is do nothing, LINCTL.RESP is 0;

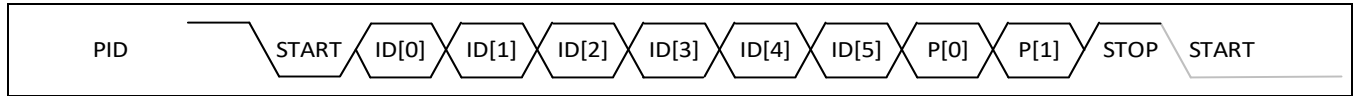
Figure 19-12: LIN transfer



19.5.3 Protected ID

LIN protected ID byte field is defined as [Figure 19-13](#).

Figure 19-13: LIN protected ID byte field



Six bits are reserved for the frame identifier (LINID.ID), values in the range 0 to 63 can be used. The frame identifiers are split in three categories:

- 0 to 59 (0x3B) are used for signal carrying frames;
- 60 (0x3C) and 61 (0x3D) are used to carry diagnostic and configuration data;
- 62 (0x3E) and 63 (0x3F) are reserved for future protocol enhancements

User can set LINIDFILT register for ID receive filter.

The parity bit P[1:0] is calculated on the frame identifier bits by hardware as below formula:

$$P[0] = ID[0] \oplus ID[1] \oplus ID[2] \oplus ID[4]$$

$$P[1] = \sim(ID[1] \oplus ID[3] \oplus ID[4] \oplus ID[5])$$

19.5.4 Checksum calculation

Frame response send 1~8 bytes data field and followed on parity field. The parity or called check sum can be calculated by two method (LINCTL.CHSUM):

- Classic checksum
Only data field bytes are calculated and used for master request frame, slave response frame and communication with LIN 1.x slaves
- Enhanced checksum
PID and data field bytes are calculated and used for communication with LIN 2.x slaves

The caculation method is eight bit sum with carry, which is equivalent to sum all values and subtract 255 every time the sum is greater or equal to 256.

Note: There is no support for automatic switching from enhanced checksum to the classic checksum. Even for frame whose identifier is 0x3C or 0x3D, user should manually configure via LINCTL.CHSUM to use the classic checksum.

19.6 Interrupt

UARTRAWIF will show the interrupt event raw flags, even if interrupt not enabled. Write UARTIC bit 1 will clear the associated bit in raw flag registers and writing 0 has no effect.

UARTRAWIF.RXREQ and UARTRAWIF.TXREQ have no clear bit in UARTIC, because they are cleared automatically by FIFO hardware logic.

If one bit of UARTIE enabled, associated bit of UARTIF will be set when associate bit in UARTRAWIF is set. Writing 1 to each bit of UARTIC will also clear associate bit in UARTIF.

19.6.1 Receive interrupt

When receive frame error or detect break signal during receive bytes and associate interrupt is enabled in UARTIE, UART will trigger frame error (UARTIF.RXFRMERR), parity error (UARTIF.RXPRTYERR) or receive break (UARTIF.RXBRK) interrupts to CPU.

However, the status bit of these events in UARTSTS will always show based on the newly popped data from Receive FIFO by read UARTDAT. These status bits is stroed in FIFO, and reflect when read FIFO.

19.6.2 Receive timeout interrupt

A receive timeout interrupt occurs when the receive timeout interrupt (UARTIE.RXTIMEOUT) is enabled and timeout event occurred.

19.6.3 Receive auto baud counter lock interrupt

When auto baud is enabled, its interrupt enabled and detect first byte start bit finished, UART will trigger interrupt and set UARTIF.ABLOCK bit.

19.6.4 Receive FIFO interrupt

For a receive interrupt to occur, the receive interrupts must be enabled.

UARTIE.RXREQ means Receive FIFO request when level of Receive FIFO (UARTRXLVL) is equal to or above its threshold (UARTRXTH), UART will send interrupt to CPU.

UARTIE.RXOVF, UARTIE.RXUDF means Receive FIFO access overflow or underflow occurred.

19.6.5 Transmit FIFO interrupt

Transmit FIFO have write Transmit FIFO overflow interrupt when CPU write UARTDA and Transmit FIFO is full.

And also have a Transmit FIFO request interrupt when level of Transmit FIFO (UARTTXLVL) is below its threshold (UARTTXTH).

19.6.6 LIN error interrupt

When UART worked as LIN mode, will enable LIN bit error check, check sum check and LIN ID check (UARTIE.LINBITERR, UARTIE.LINCSEERR, UARTIE.LINIDMATCH).

19.6.7 DMA interrupt

When DMA finished its transfer, will generate an end of chain interrupt (UARTIF.DMAEOC) if associated interrupt is enabled.

SPINTROL

19.7 Registers

19.7.1 UART register map

Table 19-2: UART Module Base Address

Peripheral Module	Base Address
UART0	0x4000 1400
UART1	0x4000 1600

Table 19-3: UART Register Map

Register	Offset	Description	Reset Value
UARTDAT	0x0	UART Data Register	0x00000000
UARTCTL *	0x4	UART Control Register	0x00000000
LINCTL *	0x8	LIN Control Register	0x00E00000
LINID	0xC	LIN ID Register	0x00000000
LINIDFILT *	0x10	LIN ID Filter Register	0x00000000
BUSIDLETH *	0x14	Bus Idle Threshold Register	0xFFFFFFFF
UARTBDCNT *	0x18	UART Baud Count Register	0x00000010
UARTABCNT	0x1C	UART Auto-Baud Count Register	0x00000000
UARTTOTH *	0x20	UART Timeout Threshold Register	0x00000030
UARTRXLVL	0x24	UART Receive FIFO Level Register	0x00000000
UARTTXLVL	0x28	UART Transmit FIFO Level Register	0x00000000
UARTRXTH *	0x2C	UART Receive FIFO Threshold Register	0x00000000
UARTTXTH *	0x30	UART Transmit FIFO Threshold Register	0x00000000
UARTSIRCTL *	0x34	UART Infrared Control Register	0x0000010C
UARTSTS	0x40	UART Status Register	0x0000000C
UARTRAWIF	0x44	UART Interrupt Raw Flag Register	0x00000000
UARTIF	0x48	UART Interrupt Flag Register	0x00000000
UARTIC	0x4C	UART Interrupt Clear Register	0x00000000
UARTIE *	0x50	UART Interrupt Enable Register	0x00000000
UARTDGCTL *	0x54	UART RXD Deglitch Control	0x00000000
UARTREGKEY	0x58	Uart Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the UARTREGKEY=0x1ACCE551.

19.7.2 UART registers

Table 19-4: UART Data Register (UARTDAT) Layout

UARTDAT (UART Data Register) Offset: 0x0 Default: 0x00000000 Access: UARTx->UARTDAT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 19-5: UART Data Register (UARTDAT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Data transmitted or received on the UART bus. A write to the register accesses the transmit hold register to set the data to be transmitted. A read to the register accesses the receive buffer register to get the received data

Table 19-6: UART Control Register (UARTCTL) Layout

UARTCTL (UART Control Register) Offset: 0x4 Default: 0x00000000 Access: UARTx->UARTCTL							
31	30	29	28	27	26	25	24
SWAPPIN	RFEMODE	IDLECNTEN	RESERVED				
23	22	21	20	19	18	17	16
RESERVED					SETBRK	CLRTF	CLRRF
15	14	13	12	11	10	9	8
STOPBIT	PRTYEN	PRTYBIT		DATABIT		NRZMODE	
7	6	5	4	3	2	1	0
ABUP	ABEN	BUS32	LPBK	TXDMAEN	RXDMAEN	TXEN	RXEN

Table 19-7: UART Control Register (UARTCTL) Description

Bits	Field Name	Type	Reset	Description
31	SWAPPIN	RW	0x0	Swap TXD and RXD pin function 0: Transmit data via TXD and receive data via RXD 1: Transmit data via RXD and receive data via TXD
30	RFEMODE	RW	0x0	Receive FIFO error report mode 0: Report frame error, parity error and break indicator only when the FIFO is not full and data is writable 1: Report frame error, parity error and break indicator upon event regardless of FIFO status
29	IDLECNTEN	RW	0x0	Bus idle counter enable 0: Disable 1: Enable
28:19	RESERVED_28_19	RO	0x0	Reserved.
18	SETBRK	W1S	0x0	Set Break Causes a break condition to be transmitted to the receiving UART. Acts only on the TXD pin and has no effect on the transmit logic. In FIFO mode, wait until the transmitter is idle (UARTSTS.TXDONE=1) to set and clear SB. 0: No effect on TXD output 1: This bit is self cleared when the UART starts to transmit the break condition. If software write this bit again before the current break condition is stop transmitting, TXD pin will keep low for the new frame. i.e. The receiver sees single break condition with bus is holding low for multiple frame duration.
17	CLRTF	W1C	0x0	Clear the transmit FIFO The transmit shift register is not affected by this bit, and it completes the current transmission. 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the transmit FIFO and sets TXDREQ flag. This bit is self-cleared
16	CLRRF	W1C	0x0	Clear the receive FIFO 0: Write a 0 has no effect and always read back 0 1: Write a 1 clears the receive FIFO and RXDAV flag This bit is self-cleared
15	STOPBIT	RW	0x0	Stop bits 0: 1 stop bit

Bits	Field Name	Type	Reset	Description
				1: 1.5 stop bits for 5-bit character 2 stop bits for 6/7/8 bit character
14	PRTYEN	RW	0x0	Parity enable 0: Disable parity bit for UART mode and disable ID parity check for LIN 1: Enable parity bit for UART mode and enable ID parity check for LIN
13:12	PRTYBIT	RW	0x0	Parity bit mode 00: Sends or checks for even parity 01: Sends or checks for odd parity 10: Sends or checks for sticky high parity 11: Sends or checks for sticky low parity
11:10	DATABIT	RW	0x0	Data bits of each character 00: 5-bit character 01: 6-bit character 10: 7-bit character 11: 8-bit character
9:8	NRZMODE	RW	0x0	NRZ coding mode. This bit is ignored in serial infra-red (SIR) mode and general NRZ level is always used. 00: NRZ level coding. High level for 1 and low level for 0. 10: NRZ space coding. Keep constant on 1 and toggle on 0. 11: NRZ mark coding. Keep constant on 0 and toggle on 1.
7	ABUP	RW	0x0	Auto baud count update option 0: Baud counter is updated by the hardware 1: Baud counter is updated by the software
6	ABEN	W1S	0x0	Auto baud enable 0: Disable auto baud 1: Enable auto baud
5	BUS32	RW	0x0	32-Bit bus width 0: 8-bit peripheral bus 1: 32-bit peripheral bus
4	LPBK	RW	0x0	Loopback mode for diagnostic testing as below (1) Set the transmitter serial output to logic 1 state (2) Disconnect the receiver serial input from pin. (3) Transmitted data is received immediately. The output of the Transmit Shift Register is looped back into the Receive Shift Register input. (4) Support transferring a break signal from the 0: Normal UART operation 1: Loopback-mode UART operation

Bits	Field Name	Type	Reset	Description
3	TXDMAEN	RW	0x0	Transmit DMA requests enable 0: Disable DMA for UART transmitter 1: Enable DMA for UART transmitter
2	RXDMAEN	RW	0x0	Receive DMA requests enable 0: Disable DMA for UART receiver 1: Enable DMA for UART receiver
1	TXEN	RW	0x0	UART transmit enable 0: Disable UART transmit 1: Enable UART transmit
0	RXEN	RW	0x0	UART receive enable 0: Disable UART receive 1: Enable UART receive

Table 19-8: LIN Control Register (LINCTL) Layout

LINCTL (LIN Control Register) Offset: 0x8 Default: 0x00E00000 Access: UARTx->LINCTL										
31 30 29 28 27 26 25							24			
RESERVED							TXCHKSUM			
23	22		21	20	19	18		17	16	
CHKSUM	ATXCHKSUM		ATOTH	RFMODE	RESERVED	RESPSPACE				
15	14		13	12		11		10		9
RESERVED	BYTESPACE			RESPLEN						8
										RESP
7	6		5	4	3		2		1	0
RESP	SBRKDEL			SBRKLEN				MASTER	LINEN	

Table 19-9: LIN Control Register (LINCTL) Description

Bits	Field Name	Type	Reset	Description
31:25	RESERVED_31_25	RO	0x0	Reserved.
24	TXCHKSUM	W1S	0x0	One-shot checksum transmit request 0: Write 0 has no effect and read back 0 after checksum is transmitted 1: Write 1 will request to transmit the checksum If the data field is already finished, the checksum will be transmitted immediately If the data field is not finished, the request is pended and the checksum will be transmitted right after the data field is done
23	CHKSUM	RW	0x1	Checksum mode 0: Classic checksum is used so the checksum contains the modulo-256 sum with carry over all data bytes 1: Enhanced checksum is used so the checksum

Bits	Field Name	Type	Reset	Description
				contains the modulo-256 sum with carry over all data bytes and the protected ID.
22	ATXCHKSUM	RW	0x1	Auto transmit checksum 0: Checksum is transmitted by manual request via writing 1 to TXCHKSUM bit 1: Checksum is transmitted automatically after the data field
21	ATOTH	RW	0x1	Auto timeout threshold 0: The threshold is manually specified via UARTTOTH 1: The threshold is automatically set according to LIN standard, namely the minimum required transmission time plus 40% margin
20	RFMODE	RW	0x0	Receive FIFO operation mode 0: The frame data as well as the checksum byte will be saved in the RX FIFO regardless whether the checksum is correct This is useful to debug the checksum error 1: The frame data will saved in the RX FIFO only when the checksum is correct
19	RESERVED_19	RO	0x0	Reserved.
18:16	RESPSPACE	RW	0x0	Response space (number of bauds)
15	RESERVED_15	RO	0x0	Reserved.
14:13	BYTESPACE	RW	0x0	Inter-byte space (number of bauds)
12:9	RESPLEN	RW	0x0	Response data length 0000: The repsonse data field length is decide by LINID[5:4] 11: 8-byte; 10: 4-byte; Other: 2-byte 0001: The repsonse data field has 1-byte 0010: The repsonse data field has 2-byte 0011: The repsonse data field has 3-byte 0100: The repsonse data field has 4-byte 0101: The repsonse data field has 5-byte 0110: The repsonse data field has 6-byte 0111: The repsonse data field has 7-byte 1000: The repsonse data field has 8-byte
8:7	RESP	RW	0x0	Response to take 00: Do not response in the frame 01: Receive data in the frame 10: Transmit data in the frame 11: Transmit and receive data in the frame (For debug purpose)
6:5	SBRKDEL	RW	0x0	Synchronize break delimiter 00: Synch break delimiter is 1-bit recessive 01: Synch break delimiter is 2-bit recessive

Bits	Field Name	Type	Reset	Description
				10: Synch break delimiter is 3-bit recessive 11: Synch break delimiter is 4-bit recessive
4:2	SBRKLEN	RW	0x0	Synchronize break length 000: Synch break is 13-bit dominant 001: Synch break is 14-bit dominant 010: Synch break is 15-bit dominant 011: Synch break is 16-bit dominant 100: Synch break is 17-bit dominant 101: Synch break is 18-bit dominant 110: Synch break is 19-bit dominant 111: Synch break is 20-bit dominant
1	MASTER	RW	0x0	LIN node role 0: Slave node 1: Mater node
0	LINEN	RW	0x0	LIN mode enable 0: Disable LIN mode 1: Enable LIN mode

Table 19-10: LIN ID Register (LINID) Layout

LINID (LIN ID Register) Offset: 0xC Default: 0x00000000 Access: UARTx->LINID							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
ID							

Table 19-11: LIN ID Register (LINID) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	ID	RW	0x0	LIN ID Write to this register set the ID to be transmitted Read from this register get the received ID

Table 19-12: LIN ID Filter Register (LINIDFILT) Layout

LINIDFILT (LIN ID Filter Register) Offset: 0x10 Default: 0x00000000 Access: UARTx->LINIDFILT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
MASK							
7	6	5	4	3	2	1	0
REFID							

Table 19-13: LIN ID Filter Register (LINIDFILT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:8	MASK	RW	0x0	LIN ID mask It indicates whether the corresponding bit in REFID is used for comparison 0: Not compare; 1: Compare
7:0	REFID	RW	0x0	LIN ID reference for the compare

Table 19-14: Bus Idle Threshold Register (BUSIDLETH) Layout

BUSIDLETH (Bus Idle Threshold Register) Offset: 0x14 Default: 0xFFFFFFFF Access: UARTx->BUSIDLETH							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 19-15: Bus Idle Threshold Register (BUSIDLETH) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0xFFFFFFFF	Number of AHB clock cycles to report bus idle (no activity on the RXD)

Table 19-16: UART Baud Count Register (UARTBDCNT) Layout

UARTBDCNT (UART Baud Count Register) Offset: 0x18 Default: 0x00000010 Access: UARTx->UARTBDCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 19-17: UART Baud Count Register (UARTBDCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x10	UART baud rate will be UART clock / UARTBDCNT. The minimum valid value is 16 and write a value below 16 will be reset to 16.

Table 19-18: UART Auto-Baud Count Register (UARTABCNT) Layout

UARTABCNT (UART Auto-Baud Count Register) Offset: 0x1C Default: 0x00000000 Access: UARTx->UARTABCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 19-19: UART Auto-Baud Count Register (UARTABCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RO	0x0	Number of UART clock cycles within a start-bit pulse

Table 19-20: UART Timeout Threshold Register (UARTTOTH) Layout

UARTTOTH (UART Timeout Threshold Register) Offset: 0x20 Default: 0x00000030 Access: UARTx->UARTTOTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

Table 19-21: UART Timeout Threshold Register (UARTTOTH) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	VAL	RW	0x30	Number of UART baud cycles for the specified condition to keep before the timeout event is generated The minimum allowed number is 1. Write a value as 0 will be reset to 1 For UART mode, the condition is that there is unread received data in the FIFO and the RXD bus keeps high For LIN mode, the condition is that the received data length is below the expectation when the response field is timeout

Table 19-22: UART Receive FIFO Level Register (UARTRXLVL) Layout

UARTRXLVL (UART Receive FIFO Level Register) Offset: 0x24 Default: 0x00000000 Access: UARTx->UARTRXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	VAL						

Table 19-23: UART Receive FIFO Level Register (UARTRXLVL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:0	VAL	RO	0x0	Number of bytes (0-64) remaining in the Receive FIFO

Table 19-24: UART Transmit FIFO Level Register (UARTTXLVL) Layout

UARTTXLVL (UART Transmit FIFO Level Register) Offset: 0x28 Default: 0x00000000 Access: UARTx->UARTTXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6	5	4	3 VAL	2	1	0

Table 19-25: UART Transmit FIFO Level Register (UARTTXLVL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:0	VAL	RO	0x0	Number of bytes (0-64) remaining in the Transmit FIFO

Table 19-26: UART Receive FIFO Threshold Register (UARTRXTH) Layout

UARTRXTH (UART Receive FIFO Threshold Register) Offset: 0x2C Default: 0x00000000 Access: UARTx->UARTRXTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 RESERVED	6	5	4	3 VAL	2	1	0

Table 19-27: UART Receive FIFO Threshold Register (UARTRXTH) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5:0	VAL	RW	0x0	Trigger receive data available interrupt when UARTRFLVL>UARTRFTH

Table 19-28: UART Transmit FIFO Threshold Register (UARTTXTH) Layout

UARTTXTH (UART Transmit FIFO Threshold Register) Offset: 0x30 Default: 0x00000000 Access: UARTx->UARTTXTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED		VAL					

Table 19-29: UART Transmit FIFO Threshold Register (UARTTXTH) Description

Bits	Field Name	Type	Reset	Description
31:6	RESERVED_31_6	RO	0x0	Reserved.
5:0	VAL	RW	0x0	Trigger transmit data request interrupt when UARTEFLVL<=UARTEFTH

Table 19-30: UART Infrared Control Register (UARTSIRCTL) Layout

UARTSIRCTL (UART Infrared Control Register) Offset: 0x34 Default: 0x0000010C Access: UARTx->UARTSIRCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		TXPW					
7	6	5	4	3	2	1	0
TXPW			XMODE	TXPOL	RXPOL	TXSIREN	RXSIREN

Table 19-31: UART Infrared Control Register (UARTSIRCTL) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13:5	TXPW	RW	0x8	Transmit pulse width The minimum allowed value is 1. Write a 0 will be reset as 1 by hardware.
4	XMODE	RW	0x0	Transmit pulse width select 0: Transmit pulse width has a constant duty cycle as 3/16 bit time 1: Transmit pulse width has a constant width as TXPW UART clock cycles

Bits	Field Name	Type	Reset	Description
3	TXPOL	RW	0x1	Transmit data polarity 0: SIR encoder generates a negative pulse for a data bit of zero 1: SIR encoder generates a positive pulse for a data bit of zero
2	RXPOL	RW	0x1	Receive data polarity 0: SIR decoder takes negative pulses as zeros 1: SIR decoder takes positive pulses as zeros
1	TXSIREN	RW	0x0	Transmitter SIR enable 0: Transmitter is in UART mode. All clocking to the IrDA encoder is blocked and the UART's TXD signal is connected directly to the device pin. 1: Transmitter is in infrared mode. The normal TXD output from the UART is processed by the IrDA encoder before it is fed to the device pin.
0	RXSIREN	RW	0x0	Receiver SIR enable 0: Receiver is in UART mode. All clocking to the IrDA decoder is blocked and the RXD pin is fed directly to the UART. 1: Receiver is in infrared mode. The signal from the RXD pin is processed by IrDA decoder before it is fed to the UART.

Table 19-32: UART Status Register (UARTSTS) Layout

UARTSTS (UART Status Register) Offset: 0x40 Default: 0x0000000C Access: UARTx->UARTSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						TXBUSY	RXBUSY
7	6	5	4	3	2	1	0
RXERR	RXBRK	RXFRMERR	RXPRTYERR	TXNF	TXEMPTY	RXFULL	RXNE

Table 19-33: UART Status Register (UARTSTS) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	TXBUSY	RO	0x0	Indicate the UART is transmitting 0: The UART transmitter is in idle state 1: The UART transmitter is busy
8	RXBUSY	RO	0x0	Indicate the UART is receiving 0: The UART receiver is in idle state 1: The UART receiver is busy
7	RXERR	RO	0x0	Indicate there is parity error, frame error or break condition for the un-read data in the receive FIFO 0: Data in receive FIFO are all error-free 1: There is error for un-read data in the receive FIFO
6	RXBRK	RO	0x0	Break condition indicator for the data just read out from the receive FIFO. For 32-bit bus, this bit is the logic OR of the break condition for all bytes read. 0: The all "0" byte read from UARTDAT is a normal data 1: The all "0" byte read from UARTDAT is due to a break condition
5	RXFRMERR	RO	0x0	Frame error indicator for the data just read out from the receive FIFO. For 32-bit bus, this bit is the logic OR of the frame error for all bytes read. 0: No frame error for the data read out via UARTDAT 1: Frame error for the data read out via UARTDAT
4	RXPRTYERR	RO	0x0	Parity error indicator for the data just read out from the receive FIFO. For 32-bit bus, this bit is the logic OR of the parity error for all bytes read. 0: No parity error for the data read out via UARTDAT 1: Parity error for the data read out via UARTDAT
3	TXNF	RO	0x1	Transmit FIFO not full (i.e. UARTTFLVL<64) 0: Transmit FIFO is full 1: Transmit FIFO contains empty locations

Bits	Field Name	Type	Reset	Description
2	TXEMPTY	RO	0x1	Transmit FIFO empty 0: Transmit FIFO contains one or more data entries 1: Transmit FIFO is completely empty. But there may be still data in the transmit shift out register and the transmit is going on.
1	RXFULL	RO	0x0	Receive FIFO full 0: Receive FIFO contains empty locations 1: Receive FIFO is completely full
0	RXNE	RO	0x0	Receive FIFO not empty (i.e. UARTRFLVL>0) 0: Receive FIFO is empty 1: Receive FIFO contains one or more data entries

Table 19-34: UART Interrupt Raw Flag Register (UARTRAWIF) Layout

UARTRAWIF (UART Interrupt Raw Flag Register) Offset: 0x44 Default: 0x00000000 Access: UARTx->UARTRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCERR	LINBITERR	LINIDMATCH	ABLOCK	TXREQ
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RXREQ	RXOVF	RXUDF

Table 19-35: UART Interrupt Raw Flag Register (UARTRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:17	RESERVED_31_17	RO	0x0	Reserved.
16	RESERVED_16	RO	0x0	Reserved.
15	RESERVED_15	RO	0x0	Reserved.
14	BUSIDLE	RO	0x0	LIN bus idle raw flag 0: Event does not occur 1: Event occurred
13	LINSBERR	RO	0x0	LIN received sync byte error raw flag 0: Event does not occur 1: Event occurred
12	LINCERR	RO	0x0	LIN checksum error interrupt raw flag 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
11	LINBITERR	RO	0x0	LIN bit error interrupt raw flag 0: Event does not occur 1: Event occurred
10	LINIDMATCH	RO	0x0	LIN received ID match interrupt raw flag 0: Event does not occur 1: Event occurred
9	ABLOCK	RO	0x0	Auto-baud lock interrupt raw flag 0: Event does not occur 1: Event occurred
8	TXREQ	RO	0x0	Transmit data write request interrupt raw flag 0: Read a 0 indicates there is enough data in the transmit FIFO It will be set to 1 once $UARTTXLVL \leq UARTTXTH$ 1: Read a 1 indicates the transmit FIFO requests more data It will be cleared to 0 once $UARTTXLVL > UARTTXTH$
7	TXOVF	RO	0x0	Transmit overflow interrupt raw flag It indicates an attempt to write UARTDAT when transmit FIFO is full 0: Event occurred 1: Event does not occur
6	RXTIMEOUT	RO	0x0	Receive time out interrupt raw flag For LIN mode, it indicates timeout for the response field 0: Event occurred 1: Event does not occur
5	RXBRK	RO	0x0	Receive break condition interrupt raw flag This bit is set once a break condition is received 0: Event occurred 1: Event does not occur
4	RXFRMERR	RO	0x0	Receive frame error interrupt raw flag This bit is set once a frame error (stop bit is low) is received 0: Event occurred 1: Event does not occur
3	RXPRTYERR	RO	0x0	Receive parity error interrupt raw flag This bit is set once a parity error is received In LIN mode, it indicates parity error for the PID 0: Event occurred 1: Event does not occur

Bits	Field Name	Type	Reset	Description
2	RXREQ	RO	0x0	Receive data read request interrupt raw flag 0: Read a 0 indicates there are not enough data in the receive FIFO It will be set to 1 once UARTRXLVL>UARTRXTH 1: Read a 1 indicates there are enough data in the receive FIFO It will be cleared to 0 once UARTRXLVL<=UARTRXTH
1	RXOVF	RO	0x0	Receive overflow interrupt raw flag It indicates a new character is received when the receive FIFO is full and the new data is dropped. 0: Event occurred 1: Event does not occur
0	RXUDF	RO	0x0	Receive underflow interrupt raw flag It indicates an attempt to read UARTDAT when the receive FIFO is empty 0: Event occurred 1: Event does not occur

Table 19-36: UART Interrupt Flag Register (UARTIF) Layout

UARTIF (UART Interrupt Flag Register) Offset: 0x48 Default: 0x00000000 Access: UARTx->UARTIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						INT	RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCERR	LINBITERR	LINIDMATCH	ABLOCK	TXREQ
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RXREQ	RXOVF	RXUDF

Table 19-37: UART Interrupt Flag Register (UARTIF) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	INT	RO	0x0	Global interrupt flag 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further interrupt pulse to the CPU will not be generated until this bit is cleared.
16	RESERVED_16	RO	0x0	Reserved.
15	RESERVED_15	RO	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
14	BUSIDLE	RO	0x0	Bus idle interrupt flag 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
13	LINSBERR	RO	0x0	LIN sync byte error interrupt flag 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
12	LINCERR	RO	0x0	LIN checksum error interrupt flag This bit is always 0 if UARTIE_LINCERR=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
11	LINBITERR	RO	0x0	LIN bit error interrupt flag This bit is always 0 if UARTIE_LINBITERR=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
10	LINIDMATCH	RO	0x0	LIN received ID match interrupt flag This bit is always 0 if UARTIE_LINIDMATCH=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
9	ABLOCK	RO	0x0	Auto-baud lock interrupt flag This bit is always 0 if UARTIE_ABLOCK=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
8	TXREQ	RO	0x0	Transmit data write request interrupt flag This bit is always 0 if UARTIE_TXREQ=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
7	TXOVF	RO	0x0	Transmit overflow interrupt flag This bit is always 0 if UARTIE_RXOVF=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU.

Bits	Field Name	Type	Reset	Description
				Further event will not trigger interrupt until this bit is cleared.
6	RXTIMEOUT	RO	0x0	Time out detected interrupt flag This bit is always 0 if UARTIE_RXTIMEOUT=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
5	RXBRK	RO	0x0	Receive break condition interrupt flag This bit is always 0 if UARTIE_RXBRK=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
4	RXFRMERR	RO	0x0	Receive frame error interrupt flag. This bit is always 0 if UARTIE_RXFRMERR=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
3	RXPRTYERR	RO	0x0	Receive parity error interrupt flag. This bit is always 0 if UARTIE_RXPTYERR=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
2	RXREQ	RO	0x0	Receive data read request interrupt flag This bit is always 0 if UARTIE_RXREQ=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
1	RXOVF	RO	0x0	Receive overflow interrupt flag This bit is always 0 if UARTIE_RXOVF=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.
0	RXUDF	RO	0x0	Receive underflow interrupt flag This bit is always 0 if UARTIE_RXUDF=0 0: Interrupt is never triggered 1: Interrupt is triggered and issued to the CPU. Further event will not trigger interrupt until this bit is cleared.

Table 19-38: UART Interrupt Clear Register (UARTIC) Layout

UARTIC (UART Interrupt Clear Register) Offset: 0x4C Default: 0x00000000 Access: UARTx->UARTIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED						INT	RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCSEERR	LINBITERR	LINIDMATCH	ABLOCK	RESERVED
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RESERVED	RXOVF	RXUDF

Table 19-39: UART Interrupt Clear Register (UARTIC) Description

Bits	Field Name	Type	Reset	Description
31:18	RESERVED_31_18	RO	0x0	Reserved.
17	INT	W1C	0x0	Global interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
16	RESERVED_16	W1C	0x0	Reserved.
15	RESERVED_15	W1C	0x0	Reserved.
14	BUSIDLE	W1C	0x0	Bus idle interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
13	LINSBERR	W1C	0x0	LIN sync byte error interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
12	LINCSEERR	W1C	0x0	LIN checksum error interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
11	LINBITERR	W1C	0x0	LIN bit error interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
10	LINIDMATCH	W1C	0x0	LIN received ID match interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
9	ABLOCK	W1C	0x0	Auto-baud lock interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
8	RESERVED_8	RO	0x0	Reserved.
7	TXOVF	W1C	0x0	Transmit overflow interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.

Bits	Field Name	Type	Reset	Description
6	RXTIMEOUT	W1C	0x0	Receive timeout interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
5	RXBRK	W1C	0x0	Receive break condition interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
4	RXFRMERR	W1C	0x0	Receive frame error interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
3	RXPRTYERR	W1C	0x0	Receive parity error interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
2	RESERVED_2	RO	0x0	Reserved.
1	RXOVF	W1C	0x0	Receive overflow interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.
0	RXUDF	W1C	0x0	Receive underflow interrupt clear 0: Write 0 has no effect and always read back 0 1: Write 1 clears the flag. This bit is self cleared.

Table 19-40: UART Interrupt Enable Register (UARTIE) Layout

UARTIE (UART Interrupt Enable Register) Offset: 0x50 Default: 0x00000000 Access: UARTx->UARTIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							RESERVED
15	14	13	12	11	10	9	8
RESERVED	BUSIDLE	LINSBERR	LINCERR	LINBITERR	LINIDMATCH	ABLOCK	TXREQ
7	6	5	4	3	2	1	0
TXOVF	RXTIMEOUT	RXBRK	RXFRMERR	RXPRTYERR	RXREQ	RXOVF	RXUDF

Table 19-41: UART Interrupt Enable Register (UARTIE) Description

Bits	Field Name	Type	Reset	Description
31:17	RESERVED_31_17	RO	0x0	Reserved.
16	RESERVED_16	RW	0x0	Reserved.
15	RESERVED_15	RW	0x0	Reserved.
14	BUSIDLE	RW	0x0	Bus idle interrupt enable 0: Disable 1: Enable
13	LINSBERR	RW	0x0	LIN sync byte error interrupt enable When UARTCTL.ABEN is set to 1 for the auto

Bits	Field Name	Type	Reset	Description
				baud detection, this interrupt should be disabled to avoid false alarm. 0: Disable 1: Enable
12	LINCERR	RW	0x0	LIN checksum error interrupt enable 0: Disable 1: Enable
11	LINBITERR	RW	0x0	LIN bit error interrupt enable 0: Disable 1: Enable
10	LINIDMATCH	RW	0x0	LIN received ID match interrupt enable 0: Disable 1: Enable
9	ABLOCK	RW	0x0	Auto-baud lock interrupt enable 0: Disable 1: Enable
8	TXREQ	RW	0x0	Transmit data write request interrupt enable 0: Disable 1: Enable
7	TXOVF	RW	0x0	Transmit overflow interrupt enable 0: Disable 1: Enable
6	RXTIMEOUT	RW	0x0	Receive timeout interrupt enable 0: Disable 1: Enable
5	RXBRK	RW	0x0	Receive break condition interrupt enable 0: Disable 1: Enable
4	RXFRMERR	RW	0x0	Receive frame error interrupt enable 0: Disable 1: Enable
3	RXPRTYERR	RW	0x0	Receive parity error interrupt enable 0: Disable 1: Enable
2	RXREQ	RW	0x0	Receive data read request interrupt enable 0: Disable 1: Enable
1	RXOVF	RW	0x0	Receive overflow interrupt enable 0: Disable 1: Enable
0	RXUDF	RW	0x0	Receive underflow interrupt enable 0: Disable 1: Enable

Table 19-42: UART RXD Deglitch Control (UARTDGCTL) Layout

UARTDGCTL (UART RXD Deglitch Control) Offset: 0x54 Default: 0x00000000 Access: UARTx->UARTDGCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				FLTDIV			
7	6	5	4	3	2	1	0
FLTTH				FLTWIN			

Table 19-43: UART RXD Deglitch Control (UARTDGCTL) Description

Bits	Field Name	Type	Reset	Description
31:12	RESERVED_31_12	RO	0x0	Reserved.
11:8	FLTDIV	RW	0x0	Deglitch filter clock dividing ratio, which is (FLTDIV+1)
7:4	FLTTH	RW	0x0	Deglitch filter threshold is (FLTTH+1) 1's/0's for output to toggle
3:0	FLTWIN	RW	0x0	Deglitch filter window size is (FLTWIN+1) samples

Table 19-44: Uart Register Write-Allow Key Register (UARTREGKEY) Layout

UARTREGKEY (Uart Register Write-Allow Key Register) Offset: 0x58 Default: 0x1ACCE551 Access: UARTx->UARTREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 19-45: Uart Register Write-Allow Key Register (UARTREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected UART registers

20 SPI

20.1 SPI overview

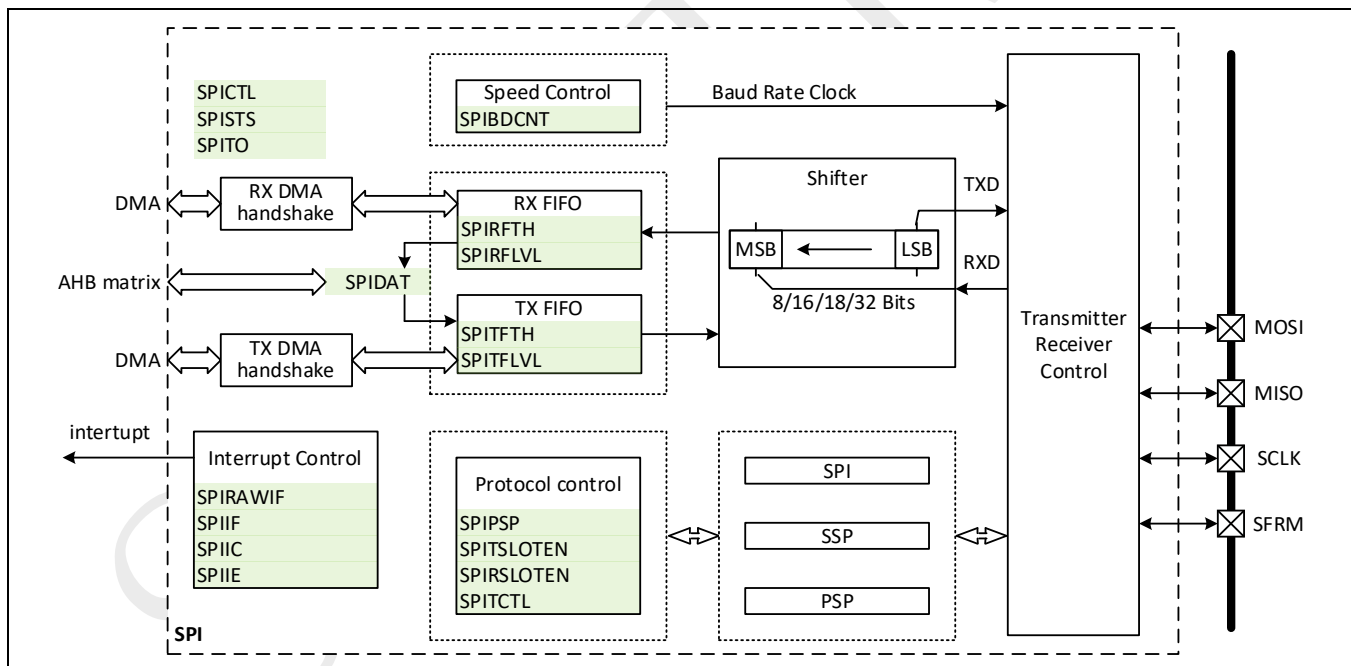
The SPC1169 SPI interface is a synchronous, full duplex, serial data transfer port that can be used as a connection to a variety of external interfaces, such as flash, ADC, CODEC, etc.

The SPI interface can be programmed to operate in Master Mode (the attached peripheral devices as slaves) or Slave Mode (the attached peripheral devices as masters). And also support SSP (TI Synchronous Serial Protocol), SPI (Motorola Serial Peripheral Interface), PSP (Programmable Serial Protocol).

The SPI port supports different serial bit rates through the configurable interface clocks. Serial data sample size can be configured to 8,16,18,32 bits in length. A FIFO is provided for Transmit data and a second independent FIFO is provided for Receive data. Both FIFOs are 16 sample in depth and 32 bit in width. The FIFOs can be loaded or emptied by CPU. The SPI can be driven by polling, interrupt or DMA.

SPI module is mainly composed of control, status and data registers, shifter logic, FIFO control logic, Transmitter and receiver control, interrupt control, as shown in Figure 20-1.

Figure 20-1: SPI block diagram



20.2 SPI features

The SPC1169 SPI has following features:

- Support SPI (TI Synchronous Serial Protocol)
- Support SPI (Motorola Serial Peripheral Interface)
- Support PSP (Programmable Serial Protocol)
- Master or Slave mode operation
- Data transfer rate up to 50 Mbps
- Receive only operation
- Programmable data frame size: 8/16/18/32bits
- Programmable polarity for SFRM
- Programmable tri-state output
- MSB-first data order
- Two independent Transmit and receive FIFO with 16 sample in depth and 32 bit in width, and all support packed mode
- DMA support

20.3 SPI signal description

Table 20-1 describes the SPC1169 SPI interface bus signals.

Table 20-1: SPI signal description

Signal Name	Type	Descriptions
MOSI	Input/ Output	This port is used to Transmit data out (TXD) of the SPI module when it is configured as a Master and receive data (RXD) when it is configured as Slave.
MISO	Input/ Output	This port is used to Transmit data out (TXD) of the SPI module when it is configured as a Slave and receive data (RXD) when it is configured as Master.
SCLK	Input/ Output	This port is used to output the clock with respect to which the SPI transfers data or receive clock in case of Slave.
SFRM	Input/ Output	This port is used to output the select signal from the SPI module to another peripheral with which a data transfer is to take place when its configured as a Master and its used as an input to receive the slave select signal when the SPI is configured as Slave.

20.4 SPI function description

This section describes the various modes of operation of the SPI, including baud rate generation, FIFO operation and interrupts.

20.4.1 Baud rate generation

The SPI contains a programmable baud-rate generator that can take a fixed-input clock and divide it down to generate the preferred baud rate. The baud rate is calculated by taking the SPI peripheral clock (CLK_SPI).

The baud-rate generator will generate baud rate clock used for transmit or receive data. SPIBDCNT store the divisor in a 16-bit binary format. Load these divisor latches during initialization to ensure that the baud-rate generator operates properly. The baud rate of the data shifted into or out of a SPI is given by the formula:

$$\text{BaudRate} = \frac{CLK_SSP}{SPIBDCNT + 1}$$

20.4.2 FIFO operation

The SPC1169 SPI has two independent FIFOs for transmitting and receiving. The CPU will load and empty those FIFOs. The CPU will transfer one FIFO entry per access and the accesses must always be 32-bits wide. The CPU writes to Transmit FIFO are 32-bits wide, but the serializing logic will ignore all bits beyond the programmed FIFO data size (SPICTL0.SIZE). The CPU reads from the RXFIFO are also 32-bits wide with zeros inserted in the MSBs down to the programmed data size.

The CPU only see one 32-bit location for both Transmit FIFO and Receive FIFO.

For data transmission, the SPI interface takes the data from Transmit FIFO, serializes the data and transmits the data through TXD interface signal to the external peripherals.

For data receiving, the serial data through RXD serial interface signal is converted to parallel data and written into the Receive FIFO.

Depends on whether the access is a READ or WRITE transfer from CPU, the FIFO Data Register automatically target the Receive FIFO or Transmit FIFO. From the memory-map perspective, the Transmit FIFO and Receive FIFO share the same address.

An interrupt/DMA service request is generated if a programmable FIFO trigger threshold exceeded which signals the CPU to empty the Receive FIFO or refill the Transmit FIFO.

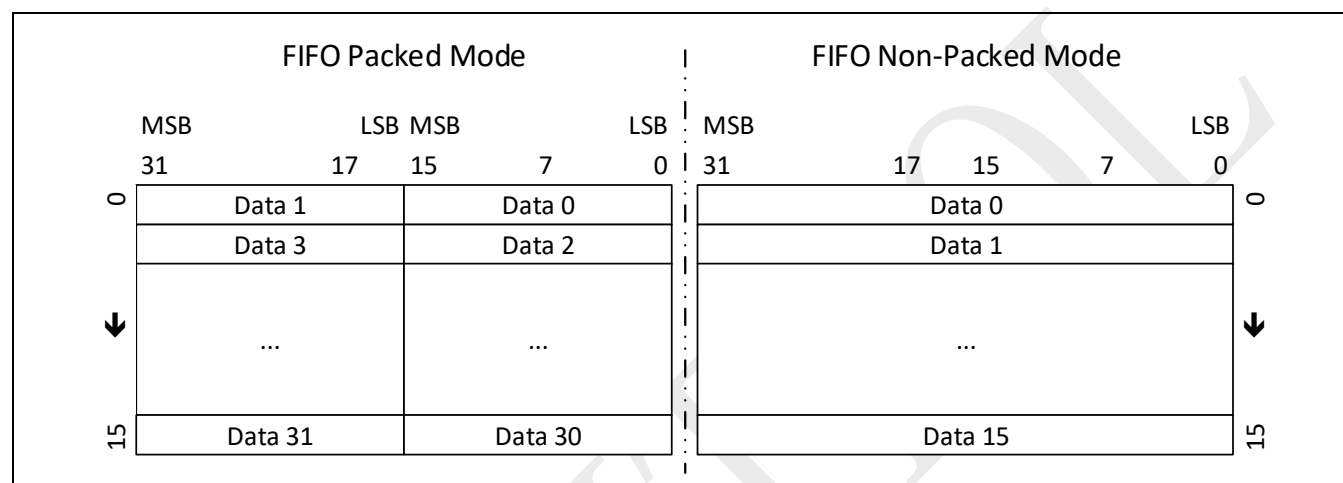
Note: If SPI was configured as slave and worked in receive only mode (SPICTL0.RXONLY=1), there is no need to write Transmit FIFO when receiving data (half-duplex). However, writing Transmit FIFO can still sending data.

20.4.2.1 Parallel data formats for FIFO storage

The sample data size can be 8, 16, 18 or 32 bits. The data in the FIFOs is either stored as 32-bits wide per data sample or as 16-bits wide per sample (packed mode). Within 32-bits or 16-bits sample mode, the stored data sample is right-justified, with LSB of the word in bit 0. The unused bits are packed as zero above the MSB for read mode and “don’t care” bit for write mode. The logic in SPI will automatically formats data in the Transmit FIFO so that the sample data will be properly Transmitted on TXD signal with the selected frame format.

When the FIFOs are operating in packed mode, as shown in Figure 20-2, each FIFO is 32 rows in depth by 16-bits in width for a total of 32 data samples. Each sample can be 8 or 16 bits in length. When the data is serialized and transmitted, bits15-0 are Transmitted first, followed by bits 31-16. The best practice for packed mode is to treat FIFO as a single entry of two samples. Thus, the CPU should write and read 32 bits of data at a time where each Write or Read transfers two samples. The entire FIFO width (32 bits) must be read or written in this mode. Also the FIFO trigger threshold need to be calculated in 32-bits read and write, not based on 16-bits.

Figure 20-2: FIFO packed and non-packed mode



Read SPISTS.RFEVEN and SPISTS.TFEVEN can show even sampled data in FIFO or not. These state bits will help to read data in packed FIFO correctly.

20.4.2.2 Trailing bytes in Receive FIFO

When the number of samples in the Receive FIFO is less than its trigger threshold level and no additional data is received, the remaining bytes are called Receive FIFO trailing bytes. Receive FIFO trailing bytes can be handled by the CPU. Receive FIFO trailing bytes are identified by means of a time-out mechanism and the existence of data within the Receive FIFO after timeout.

The trailing byte is triggered by timeout logic in implementation. A timeout happens when the Receive FIFO has been idled for a period time defined by SPITO. When a timeout occurs, the receiver timeout interrupt bit SPIIF.TO is set to 1 and if the receiver timeout interrupt enable bit SPIIE.TO is set, a timeout interrupt signals the CPU that a timeout condition has occurred. The timeout timer is reset after a new data sample is received into the Receive FIFO. Once the SPIIF.TO bit is set, it must be cleared by writing 0x1 to the SPIIC.TO bit.

The trailing bytes left in the Receive FIFO are handled by the CPU polling method by default. If a timeout occurs, the CPU is only interrupted by a timeout interrupt if it has been enabled by setting the Receiver Time-out Interrupt Enable field (SPIIE.TO). To read out the trailing bytes from the Receive FIFO, software should wait for the timeout interrupt and then read all trailing bytes as indicated by the Receive FIFO Even Sample Status (SPISTS.RFEVEN), Receive FIFO Level (SPIRFLVL), and Receive FIFO Not Empty (SPISTS.RXNE) fields in the SPI Status Register.

Note: If FIFO Packed mode is enabled (SPICTL0.FPACK=1), trailing bytes must be removed using programmed polling method. If the SPISTS.RFEVEN is clear, and SPISTS.RXNE is set, then the last FIFO line only contains one sample.

20.4.2.3 FIFO threshold

Each FIFO has a configurable trigger threshold that can be used to trigger an interrupt or DMA request. When the number of entries in Receive FIFO exceeds the Receive FIFO trigger threshold (SPIRFTH), an interrupt can be generated (if enabled) to signal the CPU to empty the Receive FIFO. When the number of entries in the Transmit FIFO is less than or equal to the Transmit FIFO trigger threshold (SPITFTH), an interrupt can be generated (if enabled) to signal the CPU to refill the Transmit FIFO.

The CPU can poll the SPI Status Register to determine how many samples are in a FIFO or whether the FIFO is full or empty. Software is responsible for ensuring that the proper Receive FIFO/Transmit FIFO trigger threshold values are chosen to prevent Receive FIFO overflow or Transmit FIFO underflow errors.

20.4.3 SPI interrupts

The interrupt flag registers (SPIRAWIF, SPIIF) contain bits that signal overrun errors as well as the Transmit FIFO and Receive FIFO service requests. Each of these hardware-detected events signals an interrupt request to the interrupt controller. SPISTS also contains flags that indicate if the SPI port is actively transmitting data, if the Transmit FIFO is not full, and if the Receive FIFO is not empty. A signal-interrupt signal is sent to the interrupt controller. These events can cause an interrupt request: receiver time out, Receive FIFO overrun, Receive FIFO service request, and Transmit FIFO service request. Receive FIFO service request and Transmit FIFO service request will also cause DMA requests.

Bits that cause an interrupt request remain set until they are cleared by writing a 1 to each bit in SPIIC, except TXREQ and RXREQ, which is controlled by FIFO hardware logic. Once a flag bit is cleared, the interrupt is cleared.

20.5 SPI (Motorola Serial Peripheral Interface)

20.5.1 Device connection

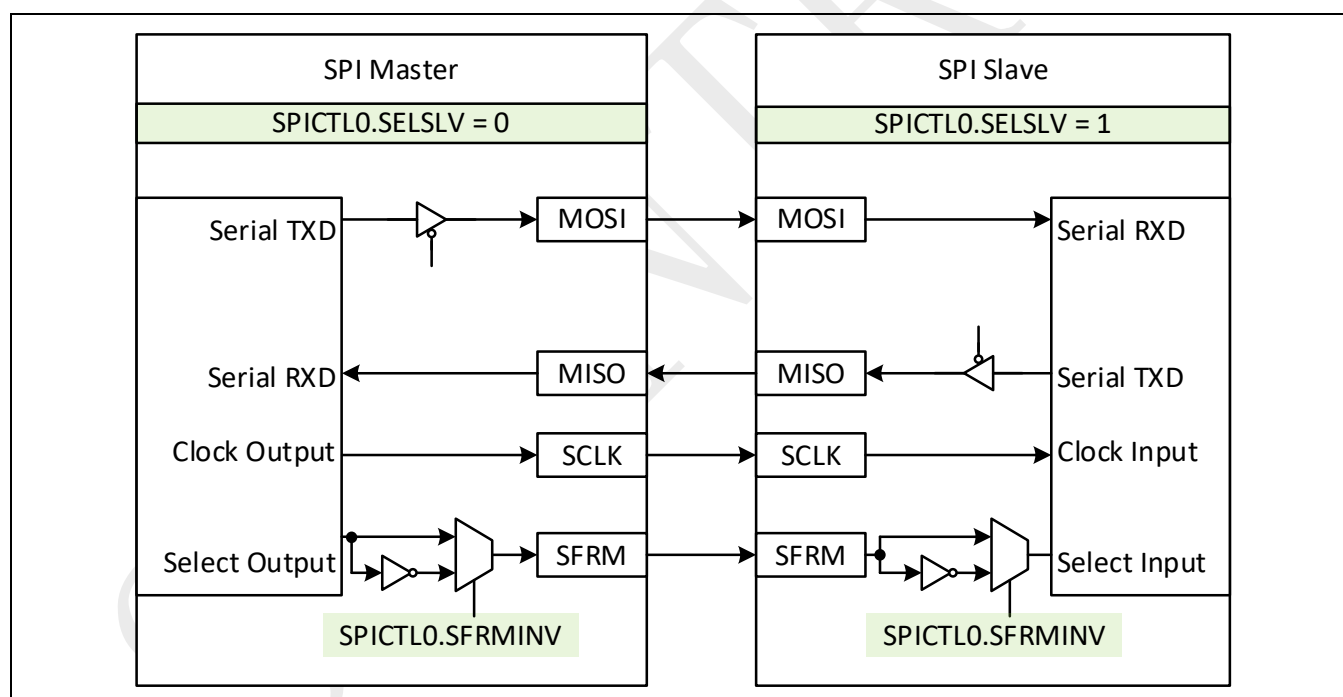
SFRM functions as a chip select to enable the external device (target the transfer) and is held active-low during the data transfer. Software can change active level. Master and Slave modes are supported.

SCLK only toggles during active transfers (does not run continuously). Software choose which edge of SCLK to use for switching Transmit data and for sampling Receive data. Master and Slave modes are supported.

SPI can be configured as master or slave, support full-duplex format. The master send clock SCLK and signal SFRM to slave, and slave begin transfer data based on SCLK and SFRM. The master data port (MOSI/MISO) connect with slave data port with the same name. Master send data to MOSI, receive data from MISO. Slave receive data from MOSI, and send data to MISO.

As shown in [Figure 20-3](#), a typical connections of the SPI for communications between two controllers: a master and a slave.

Figure 20-3: SPI master-slave connection



There are three possible methods for data transmission:

- Master sends data; slave sends dummy data.
- Master sends data; slave sends data.
- Master sends dummy data; slave sends data.

20.5.2 Protocol description

The SPI format has four sub-modes depending on the phase mode of SCLK (the edges used to transmit and receive data) and the polarity mode of SCLK (SCLK idle state), which controlled by register SPICTL0.SPIMODE:

- PHS is 0, POL is 0: SCLK is low when idle, first edge used for sample data
- PHS is 0, POL is 1: SCLK is high when idle, first edge used for sample data
- PHS is 1, POL is 0: SCLK is low when idle, second edge used for sample data
- PHS is 1, POL is 1: SCLK is high when idle, second edge used for sample data

The remanent edge used for sample data is based on first sample edge position, then following sample edge is same with the first sample edge polarity. And the opposite edge is used for Transmit data. As shown in [Figure 20-4](#) and

Figure 20-5.

When in SPI format, the SPI port can be either a master or a slave device, and the clock (SCLK) and frame (SFRM) direction are all controlled by master device (SPICTL0.SELSLV is 0).

When in SPI format, if the SPI port is the master and SPIPSP.ETDS is cleared, the end-of-transfer data state for TXD is low. If the SPI port is the master and SPIPSP.ETDS is set, the end-of-transfer data state for SPIx_TXD remains at the last bit Transmitted (LSB). If the SPI port is the slave, then the SPIPSP.ETDS is undefined.

RXD is undefined before the frame is active and after the LSB is received. RXD must not float.

When the SPI port is configured as a master and SPICTL0.TRIEN is set, SPIPSP.ETDS is ignored and TXD becomes high impedance between active transfers.

If the SPI port is disabled or reset, TXD is forced to zero (unless the TXD tri-state output enabled, in which case it goes into a high impedance state).

Figure 20-4 and

Figure 20-5 show the four possible configurations for the SPI frame protocol for a single Transmitted frame.

SPINTROL

Figure 20-4: SPI protocol (PHS = 0)

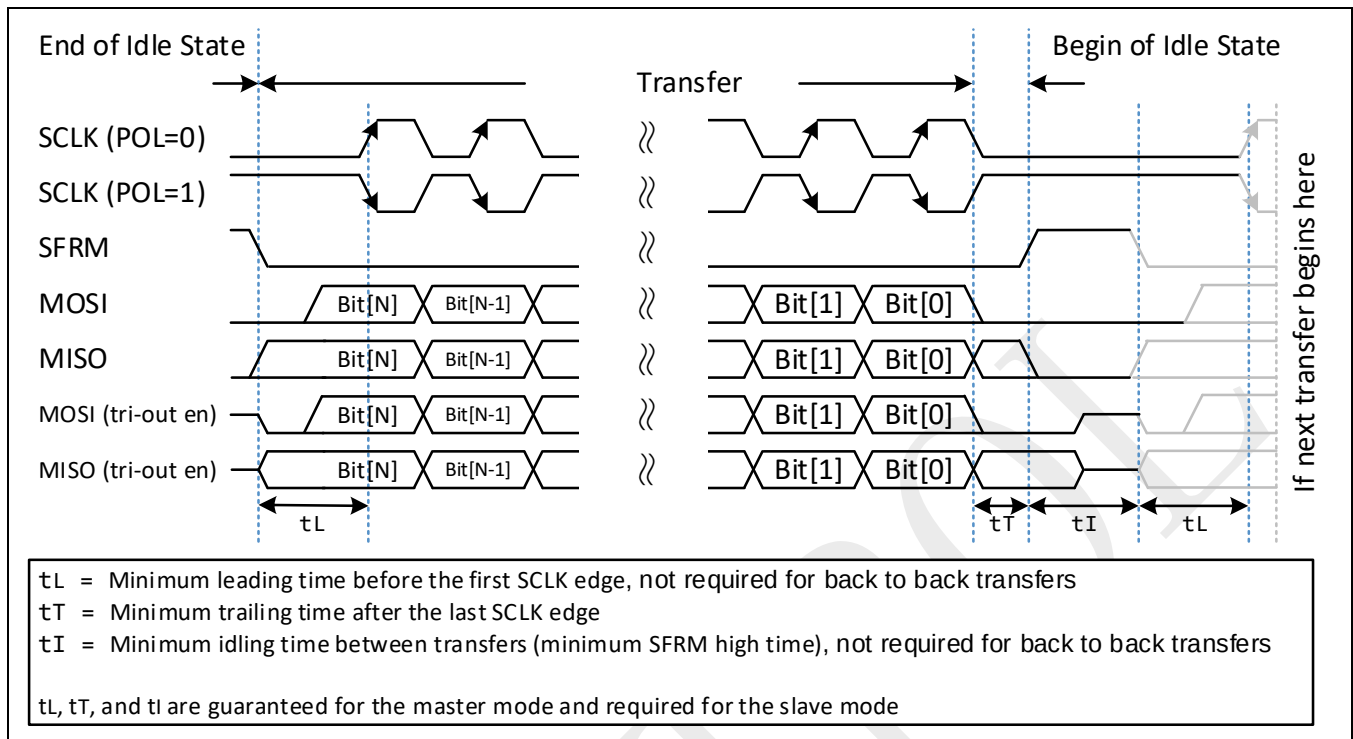


Figure 20-4 shows the SPI protocol when the PHS is 0. Explain this figure when SCLK POL is 0. When the SPI port is disabled or in idle mode, SCLK and TXD are low and SFRM is high. When Transmit data is ready to be sent, SFRM goes low (one clock period before the first rising edge of SCLK) and stays low for the remainder of the frame. The MSB (most significant bit) of the serial data is driven onto TXD after SCLK half-cycle. Halfway into the first bit period, SCLK asserts high and continues toggling for the remaining data bits. Data transitions on the falling edge of SCLK and is sampled on the rising edge of SCLK. 8, 16, 18, or 32 bits can be transferred per frame (Actually, 3-32 bits can be transferred per frame). With the assertion of SFRM, Receive data is driven simultaneously from the peripheral on RXD, MSB first. Data transitions on SCLK falling edges and is sampled by the controller on SCLK rising edges. At the end of the frame, SFRM is de-asserted high one clock period (half SCLK cycle after the last falling edge of SCLK) after the last bit has been latched at its destination and the completed incoming word is shifted into the Receive FIFO.

The peripheral can drive RXD to a high-impedance state after sending the last bit of the frame. TXD can be low level or retains the last value Transmitted when the controller goes into Idle mode.

Figure 20-5 shows the SPI protocol when the PHS is 1. The process is simila with Figure 20-4.

SPINTROL

Figure 20-5: SPI protocol (PHS = 1)

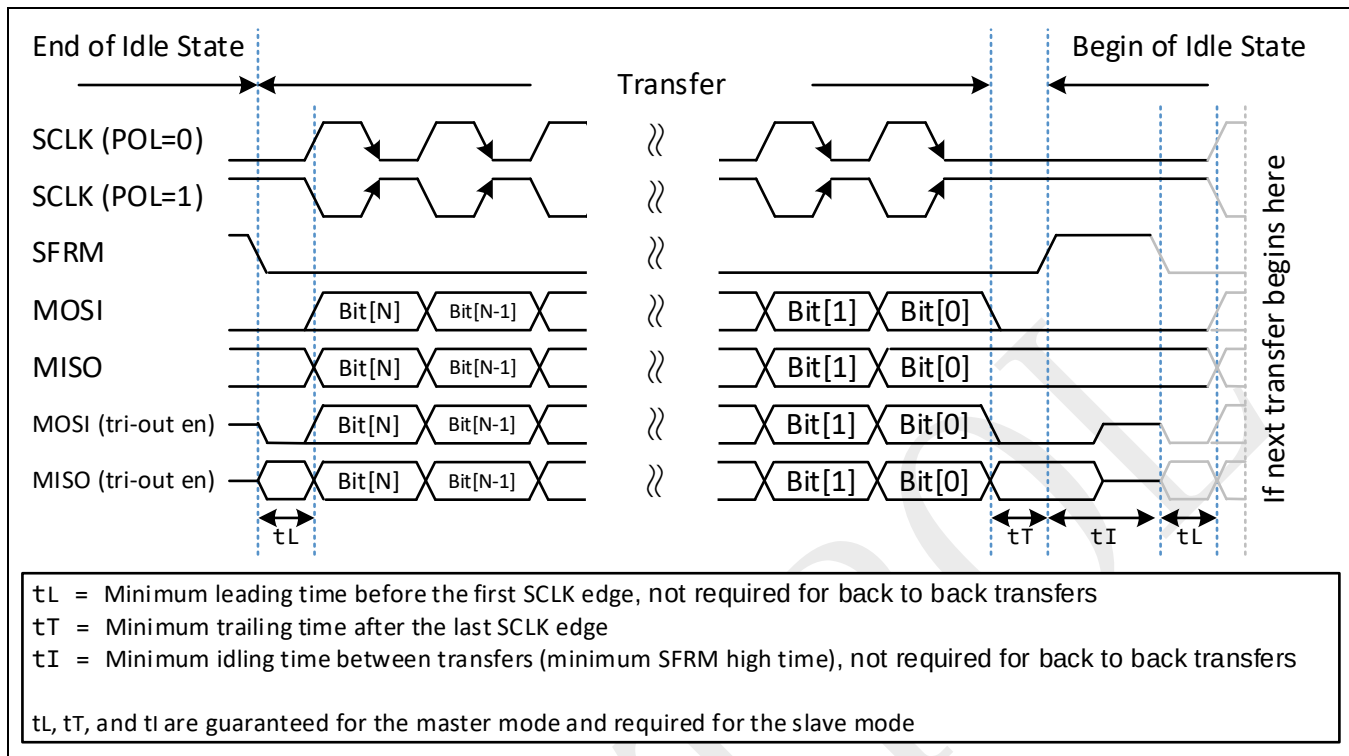


Figure 20-6: SPI protocol (back to back transfer)

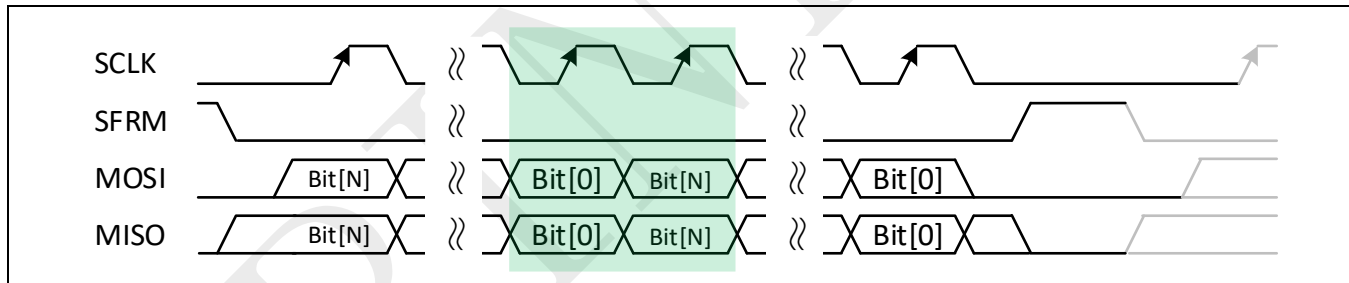


Figure 20-6 shows when back-to-back frames are Transmitted for the SPI frame protocol. For back-to-back transfers, start and completion are like those of a single transfer, but SFRM does not de-assert between words. The start of one frame immediately follows the completion of the previous. The MSB of one transfer immediately follows the LSB (least significant bit) of the preceding with no “dead” time between them. And the SFRM doesn’t de-assert during block transfer (a block of transfers is a group of back-to-back continuous transfers).

For enable back to back mode, be ensure that more than one data left in the Transmit FIFO during transfer data. If don’t want to use back to back mode, FIFO packed mode should be disabled, and be sure that Transmit FIFO should be empty before current data transfer finished.

- Note:
1. The input clock to SPI port must not be active when SFRM is de-asserted.
 2. When the SPI port is slave to clock and frame, SPICTL0.SLVCKRUN must be set.
 3. When the SPI tri-state output enabled, tri-state output phase (SPICTL0.TRIPHS) must be 0.

20.6 SSP (TI Synchronous Serial Protocol)

20.6.1 Device connection

SFRM pulsed active-high for one (serial) data period at the start of each frame. Software can change active level. Master and Slave modes are supported

SCLK only toggles during active transfers (does not run continuously). Transmit data on the rising edge of SCLK and sample Receive data on the falling edge. When the SCLK is driven by another device, it is allowed to be either continuous or driven only during transfers.

SSP can be configured as master or slave, support full-duplex format.

Device connection under this mode is same with [Figure 20-3](#).

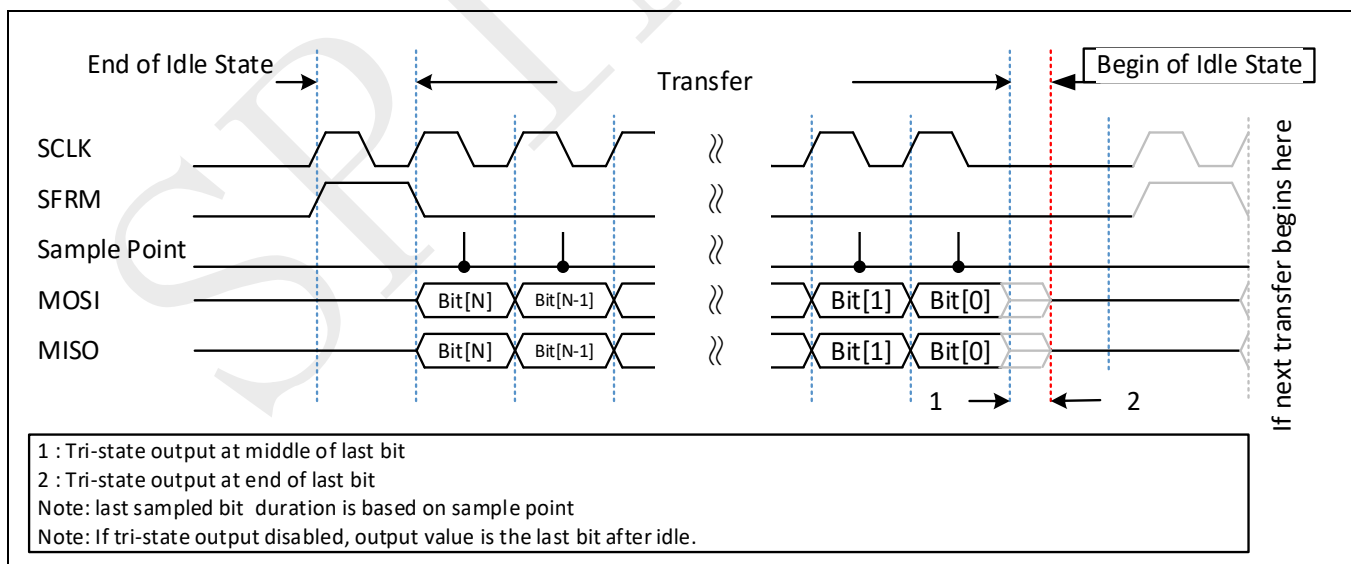
20.6.2 Protocol description

When outgoing data in the SSP controller is ready to Transmit, SFRM asserts for one clock period. On the following clock, data to be Transmitted is driven on TXD one bit at a time, with the MSB (most significant bit) first. For Receive data, the peripheral similarly drives data on the RXD pin. Word length can be 8, 16, 18, or 32 bits. All output transitions occur on the rising edge of SCLK while data sampling occurs on the falling edge.

The TXD signal either retains the value of the last bit sent (bit 0) or goes to a high impedance (SPICTLO.TRIEN, SPICTLO.TRIPHS) state at the end of the transfer as shown in [Figure 20-7](#) and [Figure 20-8](#).

If the SSP port is disabled or reset, TXD is forced to zero (unless the TXD tri-state output enabled, in which case it goes into a high impedance state).

Figure 20-7: SSP protocol (single transfer)



[Figure 20-7](#) shows the SSP for a single Transmitted frame. Once the Transmit FIFO contains data, SFRM is pulsed high for one SCLK period. And the value to be Transmitted is transferred from the Transmit FIFO to the serial shifter register. On the next rising edge of SCLK, the MSB of the data is shifted to the TXD

pin. Likewise, the MSB of the received data is shifted onto the RXD pin by the off-chip serial slave device. Both the SSP port and the off-chip serial slave device then latch each data bit into the serial shifter on the falling edge of each SCLK. The received data is transferred from the serial shifter register to the Receive FIFO on the last falling edge of SCLK after the last bit has been latched.

Figure 20-8: SSP protocol (back to back transfer)

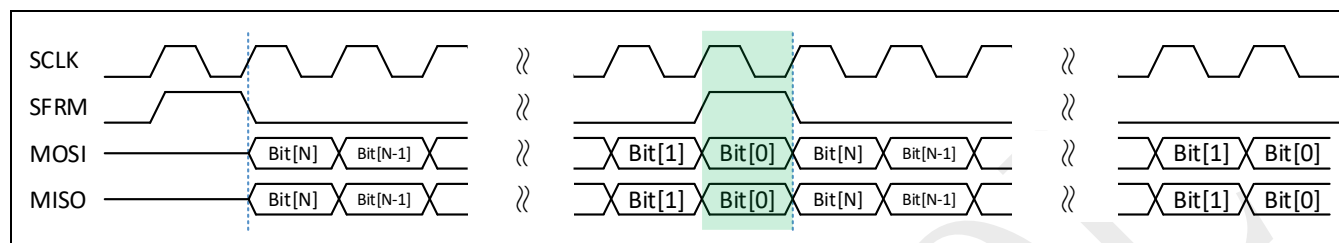


Figure 20-8 shows the SSP for when back-to-back frames are Transmitted. For back-to-back transfers, the start of one frame immediately follows the completion of the previous. The MSB of one transfer immediately follows the LSB (least significant bit) of the preceding with no “dead” time between them.

- Note:
1. If SPICTL0.EN is cleared, the TXD line goes low.
 2. SPIPSP.ETDS has no effect when in SSP mode.
 3. RXD is undefined before the MSB is sent and after the LSB is sent. RXD must not float.

For enable back to back mode, be ensure that more than one data left in the Transmit FIFO during transfer data. If don't want to use back to back mode, FIFO packed mode should be disabled, and be sure that Transmit FIFO should be empty before current data transfer finished.

- Note:
- When the SSP port is slave and SCLK not run continuously, SPICTL0.SLVCKRUN must be set. Else will lost first data for synchronize control signals from the CLK_AHB domain into the CLK_SPI domain after enable SPI module and start transfer.

20.7 PSP (Programmable Serial Protocol)

20.7.1 Device connection

SFRM is programmable in direction, delay, polarity, and width. Master and Slave modes are supported.

SCLK only toggles during active transfers. Software choose falling or rising edge of SCLK to transmit data and for sampling receive data. In addition, software can control the Idle state for SCLK and the number of active clocks that precede and follow the data transmission. Master and Slave modes are supported. When the SCLK is driven by another device, it is allowed to be either continuous or driven only during transfers, but certain restrictions on PSP parameters apply.

SPI can be configured as master or slave, and can be programmed to be either full- or half-duplex format.

Device connection under this mode is same with [Figure 20-3](#).

20.7.2 Protocol description

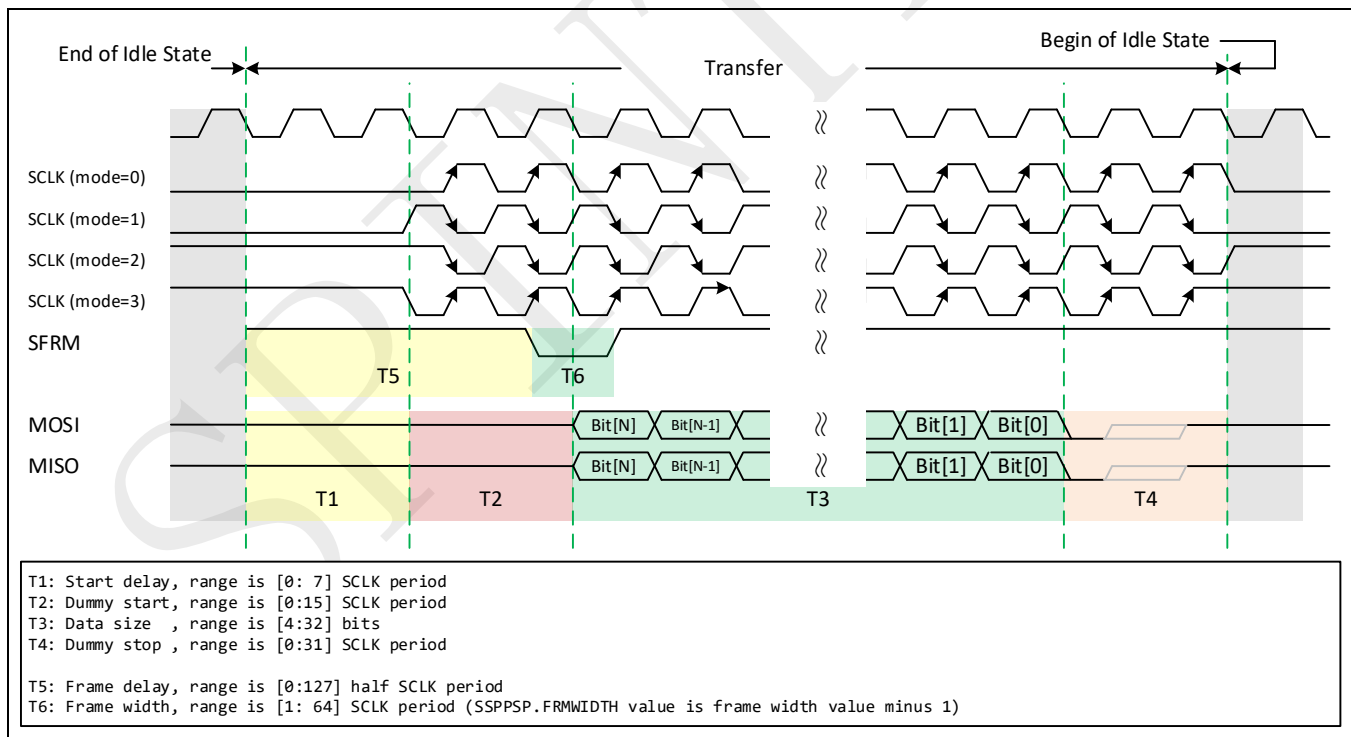
The PSP format defines programmable parameters that determine the transfer timings between data samples. Four serial clock modes are defined in SPIPSP.CLKMODE. These modes select the SCLK rising and falling edges for driving data and sampling received data.

As shown in Figure 20-9 and Figure 20-10, the Idle and Disable modes of the TXD, SCLK and SFRM interface signals are programmable using the following fields: SPIPSP.ETDS, SPIPSP.SFRMPOL and SPIPSP.CLKMODE. When Transmit data is ready, SCLK remains in its Idle state for the number of serial clock (SCLK) periods programmed into SPIPSP.STARTDLY.

SCLK then starts toggling. TXD remains in the idle state for the number of serial clock periods programmed into SPIPSP.DMYSTART. SFRM is asserted after the number of half serial clock periods programmed into the SPIPSP.FRMDLY. SFRM remains asserted for the number of serial clock periods programmed into the SPIPSP.FRWIDTH, then SFRM de-asserts.

Serial data of 8, 16, 18, or 32 bits can be transferred per frame by setting the SPICTL0.SIZE to the preferred data size select. Once the last bit (LSB) is transferred, SCLK continues toggling for the number of serial clock periods programmed into the SPIPSP.DMYSTOP. Depending on the value programmed into the SPIPSP.EDTS when the PSP port goes into Idle mode, TXD either retains the last bit-value transmitted or is forced to 0 when the PSP port is disabled or reset.

Figure 20-9: PSP protocol (single transfer)



With the assertion of SFRM, Receive data is driven simultaneously from the peripheral onto RXD, MSb first. Data transitions on the SCLK edge based on the serial-clock mode that is selected (SPIPSP.CLKMODE) and is sampled by the PSP port on the opposite clock edge.

Note: The PSP mode should not be configured as below:

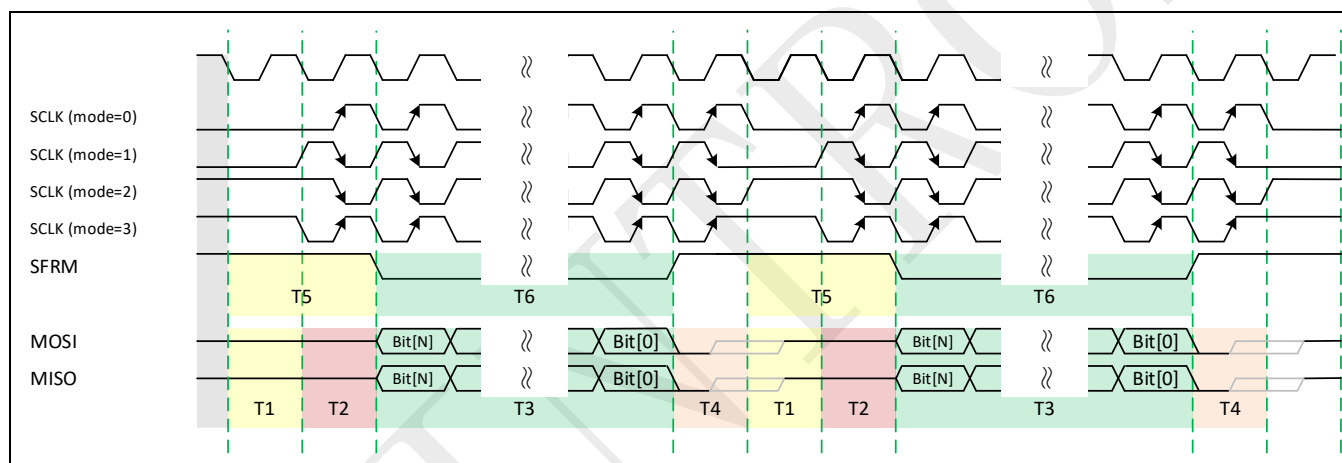
$(T5 / 2 > (T1 + T2))$, or $((T5 / 2 + T6) > (T1 + T2 + T3 + T4))$, or $(T5 / 2 < T1)$, or $((T2 == 0) \&\& (T4 == 0) \&\& (T6 \geq \text{SIZE}))$

This is an obscure case and the PSP mode will give an error (communication will be failed) if it is configured in this way.

The SFRM delay (T5) must not extend beyond the end of T4. The SFRM width (T6) must be asserted for at least one SCLK period and should be de-asserted before the end of T4.

Transmit data transitions from the end-of-transfer-data state (SPIPSP.EDTS) to the next MSB data value upon assertion of the internal version of SFRM.

Figure 20-10: PSP protocol (Consecutive Transfers)



20.8 Registers

20.8.1 SPI register map

Table 20-2: SPI Module Base Address

Peripheral Module	Base Address
SPI0	0x4000 1500
SPI1	0x4000 1700

Table 20-3: SPI Register Map

Register	Offset	Description	Reset Value
SPIDATA	0x0	SPI Data Register	0x00000000
SPICTL0 *	0x4	SPI Control Register 0	0x00000000
SPICTL1 *	0x8	SPI Control Register 1	0x00000000
SPISTS	0xC	SPI Status Register	0x00000003
SPIRFLVL	0x10	SPI Receive FIFO Level Register	0x00000000
SPITFLVL	0x14	SPI Transmit FIFO Level Register	0x00000000
SPIRFTH *	0x18	SPI Receive FIFO Threshold Register	0x00000000
SPITFTH *	0x1C	SPI Transmit FIFO Threshold Register	0x00000000
SPIRAWIF	0x20	SPI Interrupt Raw Flag Register	0x00000000
SPIIF	0x24	SPI Interrupt Flag Register	0x00000000
SPIIC	0x28	SPI Interrupt Clear Register	0x00000000
SPIIE *	0x2C	SPI Interrupt Enable Register	0x00000000
SPITO *	0x30	SPI Time Out Register	0x00000000
SPIPSP *	0x34	SPI Programmable Serial Protocol Register	0x00000000
SPITCTL *	0x40	SPI Test Control Register	0x00000000
SPIBDCNT *	0x44	SPI Baud Count Register	0x00000000
SPIREGKEY	0x48	SPI Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the SPIREGKEY=0x1ACCE551.

20.8.2 SPI registers

Table 20-4: SPI Data Register (SPIDATA) Layout

SPIDATA (SPI Data Register) Offset: 0x0 Default: 0x00000000 Access: SPIx->SPIDATA							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 20-5: SPI Data Register (SPIDATA) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Data to be written to the TXFIFO or read from the RXFIFO

Table 20-6: SPI Control Register 0 (SPICTL0) Layout

SPICTL0 (SPI Control Register 0) Offset: 0x4 Default: 0x00000000 Access: SPIx->SPICTL0							
31	30	29	28	27	26	25	24
RESERVED						RESERVED	SWAPPIN
23	22	21	20	19	18	17	16
DMATXREQ	DMARXREQ	MWIREDS	FRMSLOT			MODE	TRIPHS
15	14	13	12	11	10	9	8
TRIEN	FPACK	FRMFMT		SIZE			
7	6	5	4	3	2	1	0
SIZE	RESERVED	SFRMINV	SPIMODE		SELSLV	SLVCKRUN	EN

Table 20-7: SPI Control Register 0 (SPICTL0) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25	RESERVED_25	RW	0x0	Reserved.
24	SWAPPIN	RW	0x0	Swap MOSI and MISO pin 0: "MOSI is used for master data output and slave data input MISO is used for master data input and slave data output" 1:

Bits	Field Name	Type	Reset	Description
				"MOSI is used for master data input and slave data output MISO is used for master data output and slave data input"
23	DMATXREQ	RW	0x0	DMA Transmit Service Request Enable 0: Disable 1: Enable
22	DMARXREQ	RW	0x0	DMA Receive Service Request Enable 0: Disable 1: Enable
21	MWIREDS	RW	0x0	Microwire Transmit Command Data Size Only used for MWIRE. 0: 8-bit command words are transmitted. 1: 16-bit command words are transmitted.
20:18	FRMSLOT	RW	0x0	Frame Slot Only used for Network mode. Specifies the number of time slots per frame (the actual number of time slots is this field + 1, so 1 to 8 time slots can be specified).
17	MODE	RW	0x0	Mode Select 0: Normal SPI mode. 1: Network mode, worked with PSP format.
16	TRIPHS	RW	0x0	STXD Tri-state Enable On Last Phase 0: STXD is tri-stated 1/2 clock cycle after the beginning of the LSB. Note: Must be 0b0 for SPI/MWIRE frame format. 1: STXD is tri-stated on the clock edge that ends the LSB.
15	TRIEN	RW	0x0	STXD Tri-State Enable 0: STXD is not tri-stated 1: STXD is tri-stated when not transmitting data
14	FPACK	RW	0x0	FIFO Packing Enable 0: FIFO packing mode disabled 1: FIFO packing mode enabled
13:12	FRMFMT	RW	0x0	Frame Format 00: Motorola Serial Peripheral Interface (SPI) 01: Texas Instruments Synchronous Serial Protocol(SSP) 10: National Semiconductor Microwire 11: Programmable Serial Protocol (PSP)
11:7	SIZE	RW	0x0	Data Size Select(The min size is 3)

Bits	Field Name	Type	Reset	Description
6	RESERVED_6	RW	0x0	Reserved.
5	SFRMINV	RW	0x0	Invert Frame Signal 0: SFRM polarity is determined by the PSP polarity bit 1: SFRM is inverted from normal SFRM (as defined by the PSP polarity bit)
4:3	SPIMODE	RW	0x0	SPI Clock Mode Select (Only used for SPI frame format) 00: PHS = 0, POL = 0 01: PHS = 0, POL = 1 10: PHS = 1, POL = 0 11: PHS = 1, POL = 1
2	SESLV	RW	0x0	Select Slave Mode 0: Master mode, SPI port drives SFRM & SCLK 1: Slave mode, SPI port receives SFRM & SCLK
1	SLVCKRUN	RW	0x0	Slave Clock Free Running Only used for Slave mode. 0: Clock input to SCLK is only active during data transfers 1: Clock input to SCLK is continuously running. Must be enabled for SPI format.
0	EN	RW	0x0	Synchronous Serial Port Enable 0: Disable SPI 1: Enable SPI

Table 20-8: SPI Control Register 1 (SPICTL1) Layout

SPICTL1 (SPI Control Register 1) Offset: 0x8 Default: 0x00000000							
Access: SPIx->SPICTL1							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED			RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
15	14	13	12	11	10	9	8
RESERVED	RESERVED	RESERVED	RESERVED				
7	6	5	4	3	2	1	0
RESERVED							SFRMLCK

Table 20-9: SPI Control Register 1 (SPICTL1) Description

Bits	Field Name	Type	Reset	Description
31:21	RESERVED_31_21	RO	0x0	Reserved.
20	RESERVED_20	RW	0x0	Reserved.
19	RESERVED_19	RW	0x0	Reserved.
18	RESERVED_18	RW	0x0	Reserved.
17	RESERVED_17	RW	0x0	Reserved.
16	RESERVED_16	RW	0x0	Reserved.
15	RESERVED_15	RW	0x0	Reserved.
14	RESERVED_14	RW	0x0	Reserved.
13	RESERVED_13	RW	0x0	Reserved.
12:1	RESERVED_12_1	RW	0x0	Reserved.
0	SFRMLCK	RW	0x0	SFRM Lock Enable Only valid for SPI format, Master mode 0: Disable SFRM lock status 1: Enable SFRM lock function. SFRM will be locked at active level after SPI start send data, then need manually clear this bit to release locked SFRM to normal status.

Table 20-10: SPI Status Register (SPISTS) Layout

SPISTS (SPI Status Register) Offset: 0xC Default: 0x00000003 Access: SPIx->SPISTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			SFRMLCK	NETBUSY	ACTSLOT		
7	6	5	4	3	2	1	0
RFEVEN	TFEVEN	SLVCLK	BUSY	RXNE	RXFULL	TXEMPTY	TXNF

Table 20-11: SPI Status Register (SPISTS) Description

Bits	Field Name	Type	Reset	Description
31:13	RESERVED_31_13	RO	0x0	Reserved.
12	SFRMLCK	RO	0x0	SFRM Locked Only valid for SPI format 0: SFRM not locked 1: SFRM locked
11	NETBUSY	RO	0x0	Network mode busy 0: SPI port is in network mode and no frame is currently active

Bits	Field Name	Type	Reset	Description
				1: SPI port is in network mode and a frame is currently active
10:8	ACTSLOT	RO	0x0	Current active time slot. Because of synchronization between the SPI port's SCLK domain and an internal bus clock domain, the value in this field becomes stable between the beginning and end of the currently active time slot. 000: Time slot 0 is active 001: Time slot 1 is active 010: Time slot 2 is active 011: Time slot 3 is active 100: Time slot 4 is active 101: Time slot 5 is active 110: Time slot 6 is active 111: Time slot 7 is active
7	RFEVEN	RO	0x0	RXFIFO Even Sample Status This bit must be looked at only when FIFO Packing is enabled. Otherwise, this bit = 0. 0: RXFIFO entry has odd samples 1: RXFIFO entry has even sample
6	TFEVEN	RO	0x0	TXFIFO Even Sample Status This bit must be looked at only when FIFO Packing is enabled. Otherwise, this bit = 0. 0: TXFIFO entry has a odd number of samples 1: TXFIFO entry has an even number of samples
5	SLVCLK	RO	0x0	Clock Synchronization Status Software need only read this bit when the SPIx port is a slave or when the SPIx port is in the special FIFO test mode. Software should wait until this bit is cleared before allowing an external device to assert the external SFRM or SCLK signals. 0: The SPIx port is ready for slave clock operations. 1: The SPIx port is currently busy synchronizing slave mode signals.
4	BUSY	RO	0x0	SPI busy 0: SPIx port is idle or disabled. 1: SPIx port is currently transmitting or receiving framed data.
3	RXNE	RO	0x0	Receive FIFO Not Empty 0: RXFIFO is empty 1: RXFIFO is not empty

Bits	Field Name	Type	Reset	Description
2	RXFULL	RO	0x0	Receive FIFO full 0: Receive FIFO contains empty locations 1: Receive FIFO is completely full
1	TXEMPTY	RO	0x1	Transmit FIFO Not Full 0: Transmit FIFO contains one ore more data entries 1: Transmit FIFO is completely empty. But there may be still data in the transmit shift out register and the transmit is going on.
0	TXNF	RO	0x1	Transmit FIFO Not Full 0: TXFIFO is full 1: TXFIFO is not full

Table 20-12: SPI Receive FIFO Level Register (SPIRFLVL) Layout

SPIRFLVL (SPI Receive FIFO Level Register) Offset: 0x10 Default: 0x00000000 Access: SPIx->SPIRFLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

Table 20-13: SPI Receive FIFO Level Register (SPIRFLVL) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4:0	VAL	RO	0x0	Number of valid data entries in the receive FIFO

Table 20-14: SPI Transmit FIFO Level Register (SPITFLVL) Layout

SPITFLVL (SPI Transmit FIFO Level Register) Offset: 0x14 Default: 0x00000000 Access: SPIx->SPITFLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

Table 20-15: SPI Transmit FIFO Level Register (SPITFLVL) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4:0	VAL	RO	0x0	Number of valid data entries in the transmit FIFO

Table 20-16: SPI Receive FIFO Threshold Register (SPIRFTH) Layout

SPIRFTH (SPI Receive FIFO Threshold Register) Offset: 0x18 Default: 0x00000000 Access: SPIx->SPIRFTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

Table 20-17: SPI Receive FIFO Threshold Register (SPIRFTH) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3:0	VAL	RW	0x0	Trigger receive data available interrupt when SPIRFLVL>SPIRFTH

Table 20-18: SPI Transmit FIFO Threshold Register (SPITFTH) Layout

SPITFTH (SPI Transmit FIFO Threshold Register) Offset: 0x1C Default: 0x00000000 Access: SPIx->SPITFTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

Table 20-19: SPI Transmit FIFO Threshold Register (SPITFTH) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3:0	VAL	RW	0x0	Trigger transmit data request interrupt when SPITFLVL<=SPITFTH

Table 20-20: SPI Interrupt Raw Flag Register (SPIRAWIF) Layout

SPIRAWIF (SPI Interrupt Raw Flag Register) Offset: 0x20 Default: 0x00000000 Access: SPIx->SPIRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	TXREQ	TXOV	TXUD	TO	RXREQ	RXOV	RXUD

Table 20-21: SPI Interrupt Raw Flag Register (SPIRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	RESERVED_9	RO	0x0	Reserved.
8	RESERVED_8	RO	0x0	Reserved.
7	BITCNTERR	RO	0x0	Bit Count Error Interrupt Flag 0: Bit count error not occurred 1: Bit count error occurred
6	TXREQ	RO	0x0	Transmit FIFO Service Request Interrupt Flag 0: TXFIFO not trig threshold 1: TXFIFO trig threshold
5	TXOV	RO	0x0	Transmit FIFO overrun interrupt Flag 0: TXFIFO overrun not occurred 1: TXFIFO overrun occurred
4	TXUD	RO	0x0	Transmit FIFO Underrun Interrupt Flag 0: TXFIFO underrun not occurred 1: TXFIFO underrun occurred
3	TO	RO	0x0	Receiver Timeout Interrupt Flag 0: Timeout not occurred 1: Timeout occurred
2	RXREQ	RO	0x0	Receive FIFO Service Request Interrupt Flag 0: RXFIFO not trig threshold 1: RXFIFO trig threshold
1	RXOV	RO	0x0	Receive FIFO Overrun Interrupt Flag 0: RXFIFO overrun not occurred 1: RXFIFO overrun occurred
0	RXUD	RO	0x0	Receive FIFO underrun interrupt Flag 0: RXFIFO underrun not occurred 1: RXFIFO underrun occurred

Table 20-22: SPI Interrupt Flag Register (SPIIF) Layout

SPIIF (SPI Interrupt Flag Register) Offset: 0x24 Default: 0x00000000 Access: SPIx->SPIIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					INT	RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	TXREQ	TXOV	TXUD	TO	RXREQ	RXOV	RXUD

Table 20-23: SPI Interrupt Flag Register (SPIIF) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10	INT	RO	0x0	Global interrupt flag 0: No interrupt occurred 1: Some Interrupt occurred
9	RESERVED_9	RO	0x0	Reserved.
8	RESERVED_8	RO	0x0	Reserved.
7	BITCNTERR	RO	0x0	Bit Count Error Interrupt Flag 0: Bit count error not occurred 1: Bit count error occurred
6	TXREQ	RO	0x0	Transmit FIFO Service Request Interrupt Flag 0: TXFIFO not trig threshold 1: TXFIFO trig threshold
5	TXOV	RO	0x0	Transmit FIFO overrun interrupt Flag 0: TXFIFO overrun not occurred 1: TXFIFO overrun occurred
4	TXUD	RO	0x0	Transmit FIFO Underrun Interrupt Flag 0: TXFIFO underrun not occurred 1: TXFIFO underrun occurred
3	TO	RO	0x0	Receiver Timeout Interrupt Flag 0: Timeout not occurred 1: Timeout occurred
2	RXREQ	RO	0x0	Receive FIFO Service Request Interrupt Flag 0: RXFIFO not trig threshold 1: RXFIFO trig threshold
1	RXOV	RO	0x0	Receive FIFO Overrun Interrupt Flag 0: RXFIFO overrun not occurred 1: RXFIFO overrun occurred

Bits	Field Name	Type	Reset	Description
0	RXUD	RO	0x0	Receive FIFO underrun interrupt Flag 0: RXFIFO underrun not occurred 1: RXFIFO underrun occurred

Table 20-24: SPI Interrupt Clear Register (SPIIC) Layout

SPIIC (SPI Interrupt Clear Register) Offset: 0x28 Default: 0x00000000 Access: SPIx->SPIIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED					INT	RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	RESERVED	TXOV	TXUD	TO	RESERVED	RXOV	RXUD

Table 20-25: SPI Interrupt Clear Register (SPIIC) Description

Bits	Field Name	Type	Reset	Description
31:11	RESERVED_31_11	RO	0x0	Reserved.
10	INT	W1C	0x0	Global Interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
9	RESERVED_9	W1C	0x0	Reserved.
8	RESERVED_8	W1C	0x0	Reserved.
7	BITCNTERR	W1C	0x0	Bit Count Error Interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
6	RESERVED_6	RO	0x0	Reserved.
5	TXOV	W1C	0x0	Transmit FIFO overrun interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
4	TXUD	W1C	0x0	Transmit FIFO Underrun Interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
3	TO	W1C	0x0	Receiver Timeout Interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.
2	RESERVED_2	RO	0x0	Reserved.
1	RXOV	W1C	0x0	Receive FIFO Overrun Interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.

Bits	Field Name	Type	Reset	Description
0	RXUD	W1C	0x0	Receive FIFO underrun interrupt Clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self cleared.

Table 20-26: SPI Interrupt Enable Register (SPIIE) Layout

SPIIE (SPI Interrupt Enable Register) Offset: 0x2C Default: 0x00000000 Access: SPIx->SPIIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						RESERVED	RESERVED
7	6	5	4	3	2	1	0
BITCNTERR	TXREQ	TXOV	TXUD	TO	RXREQ	RXOV	RXUD

Table 20-27: SPI Interrupt Enable Register (SPIIE) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	RESERVED_9	RW	0x0	Reserved.
8	RESERVED_8	RW	0x0	Reserved.
7	BITCNTERR	RW	0x0	Bit Count Error Interrupt Enable 0: Disable 1: Enable
6	TXREQ	RW	0x0	Transmit FIFO Service Request Interrupt Enable 0: Disable 1: Enable
5	TXOV	RW	0x0	Transmit FIFO overrun interrupt enable 0: Disable 1: Enable
4	TXUD	RW	0x0	Transmit FIFO Underrun Interrupt Enable 0: Disable 1: Enable
3	TO	RW	0x0	Receiver Timeout Interrupt Enable 0: Disable 1: Enable
2	RXREQ	RW	0x0	Receive FIFO Service Request Interrupt Enable 0: Disable 1: Enable
1	RXOV	RW	0x0	Receive FIFO Overrun Interrupt Enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
0	RXUD	RW	0x0	Receive FIFO underrun interrupt enable 0: Disable 1: Enable

Table 20-28: SPI Time Out Register (SPITO) Layout

SPITO (SPI Time Out Register) Offset: 0x30 Default: 0x00000000 Access: SPIx->SPITO							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 20-29: SPI Time Out Register (SPITO) Description

Bits	Field Name	Type	Reset	Description
31:24	RESERVED_31_24	RO	0x0	Reserved.
23:0	VAL	RW	0x0	Timeout interval defined as $t_{CLK_SPI} \times (VAL+1)$

Table 20-30: SPI Programmable Serial Protocol Register (SPIPSP) Layout

SPIPSP (SPI Programmable Serial Protocol Register) Offset: 0x34 Default: 0x00000000 Access: SPIx->SPIPSP							
31	30	29	28	27	26	25	24
RESERVED	DMYSTOP					SYNC	RESERVED
23	22	21	20	19	18	17	16
FRMWIDTH						FRMDLY	
15	14	13	12	11	10	9	8
FRMDLY					DMYSTART		
7	6	5	4	3	2	1	0
DMYSTART	STARTDLY			ETDS	SFRMPOL	CLKMODE	

Table 20-31: SPI Programmable Serial Protocol Register (SPIPSP) Description

Bits	Field Name	Type	Reset	Description
31	RESERVED_31	RO	0x0	Reserved.
30:26	DMYSTOP	RW	0x0	Dummy stop Only used in PSP frame format. Do not use in PSP Network mode, and must be 0 when worked in Network mode.

Bits	Field Name	Type	Reset	Description
25	SYNC	RW	0x0	Frame Sync Relative Timing Bit Only used in PSP frame format. 0: Next frame is asserted after the end of the DMYSTOP timing. 1: Next frame is asserted with the LSB of the previous frame.
24	RESERVED_24	RO	0x0	Reserved.
23:18	FRMWIDTH	RW	0x0	Serial Frame Width Only used in PSP frame format. Only valid when SPI in Master mode.
17:11	FRMDLY	RW	0x0	Serial Frame Delay Only used in PSP frame format. Do not use in PSP Network mode, and must be 0 when worked in Network mode.
10:7	DMYSTART	RW	0x0	Dummy Start Only used in PSP frame format. Do not use in PSP Network mode, and must be 0 when worked in Network mode.
6:4	STARTDLY	RW	0x0	Start Delay Only used in PSP frame format. Do not use in PSP Network mode, and must be 0 when worked in Network mode.
3	ETDS	RW	0x0	End Of Transfer Data State Only used in PSP frame format. 0: Level low 1: Level of LSB value
2	SFRMPOL	RW	0x0	Serial Frame Polarity 0: SFRM is active low (0b0) 1: SFRM is active high (0b1)
1:0	CLKMODE	RW	0x0	Serial Bit-rate Clock Mode Only used in PSP frame format. 00: Data Driven (Falling), Data Sampled (Rising), Idle State (Low) 01: Data Driven (Rising), Data Sampled (Falling), Idle State (Low) 10: Data Driven (Rising), Data Sampled (Falling), Idle State (High) 11: Data Driven (Falling), Data Sampled (Rising), Idle State (High)

Table 20-32: SPI Test Control Register (SPITCTL) Layout

SPITCTL (SPI Test Control Register) Offset: 0x40 Default: 0x00000000 Access: SPIx->SPITCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	LOOP

Table 20-33: SPI Test Control Register (SPITCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	RESERVED_6	RW	0x0	Reserved.
5	RESERVED_5	RW	0x0	Reserved.
4	RESERVED_4	RW	0x0	Reserved.
3	RESERVED_3	RW	0x0	Reserved.
2	RESERVED_2	RW	0x0	Reserved.
1	RESERVED_1	RW	0x0	Reserved.
0	LOOP	RW	0x0	Loopback mode 0: Normal serial port operation is enabled. 1: Output of TX serial shifter is internally connected to input of RX serial shifter.

Table 20-34: SPI Baud Count Register (SPIBDCNT) Layout

SPIBDCNT (SPI Baud Count Register) Offset: 0x44 Default: 0x00000000 Access: SPIx->SPIBDCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 20-35: SPI Baud Count Register (SPIBDCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	SPI baud rate will be SPI clock / (SPIBDCNT+1)

Table 20-36: SPI Register Write-Allow Key Register (SPIREGKEY) Layout

SPIREGKEY (SPI Register Write-Allow Key Register) Offset: 0x48 Default: 0x1ACCE551 Access: SPIx->SPIREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 20-37: SPI Register Write-Allow Key Register (SPIREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected SPI registers

21 I2C

21.1 I2C overview

The SPC1169 implements one hardware I2C modules. Each I2C module provides an interface between a MCU and any I2C-bus-compatible device that connects via the I2C serial bus. The external components attached to the I2C bus can serially transmit/receive data to/from the MCU device through the two-wire I2C interface.

The I2C bus is multiple master bus and the I2C module supports the multi-master mode that allows more than one device capable of controlling the bus to be connected to it. Each I2C device is recognized by a unique address and can operate as either transmitter or receiver, according to the function of the device. In addition to being a transmitter or receiver, a device connected to the I2C bus can also be considered as master or slave when performing data transfers. Note that a master device is the device which initiates a data transfer on the bus and generates the clock signals to permit that transfer. During this transfer, any device addressed by this master is considered a slave.

21.2 I2C features

The SPC1169 I2C module has the following features:

- Compliant with I2C specification version 2.1
- Supports four speed modes:
 - Standard mode which maximum speed is 100 Kb/s
 - Fast mode which maximum speed is 400 Kb/s
 - Fast mode plus which maximum speed is 1 Mb/s
 - High-Speed mode which maximum speed is 3.4 Mb/s
- Clock synchronization
- 7- or 10-bit addressing
- 7- or 10-bit combined format transfers
- Two independent transmit and receive FIFOs with 16 sample in depth and 32 bit in width

21.3 I2C signal description

Table 21-1: I2C signal description

Signal Name	Type	Descriptions
I2C_SCL	Input/Output	I2C serial clock line (open drain)
I2C_SDA	Input/Output	I2C serial data line (open drain)

21.4 I2C operation

The I2C module can be controlled via software to be either:

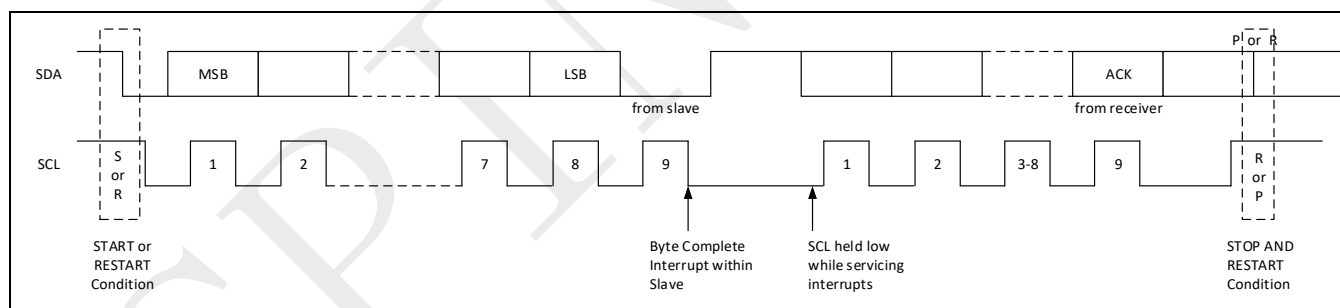
- An I2C master only, communicating with other I2C slaves; OR
- An I2C slave only, communicating with one more I2C masters.

The master is responsible for generating the clock and controlling the transfer of data. The slave is responsible for either transmitting or receiving data to/from the master. The acknowledgement of data is sent by the device that is receiving data, which can be either a master or a slave. As mentioned previously, the I2C protocol also allows multiple masters to reside on the I2C bus and uses an arbitration procedure to determine bus ownership.

Each slave has a unique address that is determined by the system designer. When a master wants to communicate with a slave, the master transmits a START/RESTART condition that is then followed by the slave's address and a control bit (R/W) to determine if the master wants to transmit data or receive data from the slave. The slave then sends an acknowledge (ACK) pulse after the address.

If the master (master-transmitter) is writing to the slave (slave-receiver), the receiver gets one byte of data. This transaction continues until the master terminates the transmission with a STOP condition. If the master is reading from a slave (master-receiver), the slave transmits (slave-transmitter) a byte of data to the master, and the master then acknowledges the transaction with the ACK pulse. This transaction continues until the master terminates the transmission by not acknowledging (NACK) the transaction after the last byte is received, and then the master issues a STOP condition or addresses another slave after issuing a RESTART condition. This behavior is illustrated in [Figure 21-1](#).

Figure 21-1: Data transfer on the I2C bus



The I2C is a synchronous serial interface. The SDA line is a bidirectional signal and changes only while the SCL line is low, except for STOP, START, and RESTART conditions. The output drivers are open-drain or open-collector to perform wire-AND functions on the bus. The maximum number of devices on the bus is limited by only the maximum capacitance specification of 400 pF. Data is transmitted in byte packages.

Note: Putting data into the FIFO generates a START, and emptying the FIFO generates a STOP. For more information, refer to [Section 21.4.1](#).

21.4.1 START and STOP generation

When operating as an I2C master, putting data into the Transmit FIFO causes the I2C to generate a START condition on the I2C bus. Allowing the Transmit FIFO to empty causes the I2C to generate a STOP condition on the I2C bus.

When operating as a slave, the I2C does not generate START and STOP conditions, as per the protocol. However, if a read request is made to the I2C, it holds the SCL line low until read data has been supplied to it. This action stalls the I2C bus until read data is provided by the slave I2C, or the I2C slave is disabled by writing a 0 to EN in register I2CCTL.

21.4.2 Combined formats

The SPC1169 I2C module supports mixed read and write combined format transactions in both 7-bit and 10-bit addressing modes.

The I2C does not support mixed address combined format transactions, that is, a 7-bit address transaction followed by a 10-bit address transaction or vice versa.

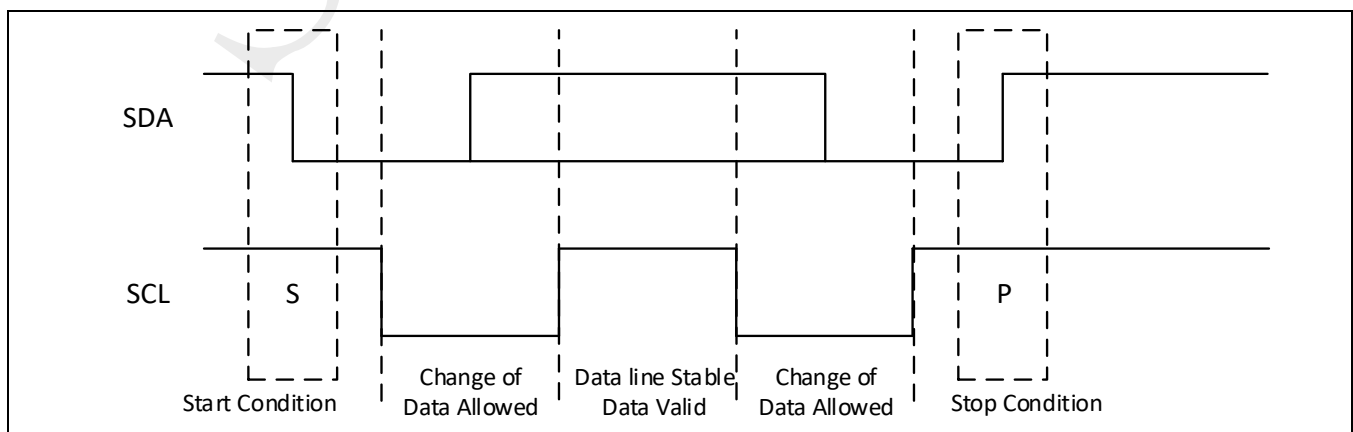
To initiate combined format transfers, set the register I2CCTL.RESTARTEN to 1. With this value set and operating as a master, when the I2C completes an I2C transfer, it checks the Transmit FIFO and executes the next transfer. If the direction of this transfer differs from the previous transfer, the combined format is used to issue the transfer. If the Transmit FIFO is empty when the current I2C transfer completes, a STOP is issued and the next transfer is issued following a START condition.

21.5 I2C protocols

21.5.1 START and STOP conditions

When the bus is idle, both the SCL and SDA signals are pulled high through external pull-up resistors on the bus. When the master wants to start a transmission on the bus, the master issues a START condition. This is defined to be a high-to-low transition of the SDA signal while SCL is 1. When the master must terminate the transmission, it issues a STOP condition, which is defined to be a low-to-high transition of the SDA line while SCL is 1. [Figure 21-2](#) shows the timing of the START and STOP conditions. When data is being transmitted on the bus, the SDA line must be stable when SCL is 1.

Figure 21-2: START and STOP condition



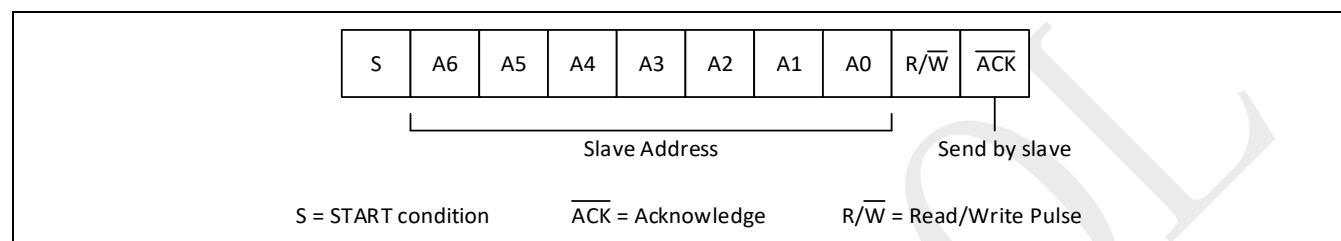
21.5.2 Address formats

The I2C module supports two address formats: the 7-bit and 10-bit address formats.

7-Bit Address Format

During the 7-bit address format, the first seven bits (bits 7:1) of the first byte set the slave address and the LSB bit (bit 0) is the R/W bit as shown in [Figure 21-3](#). When bit 0 (R/W) is set to 0, the master writes to the slave. When bit 0 (R/W) is set to 1, the master reads from the slave.

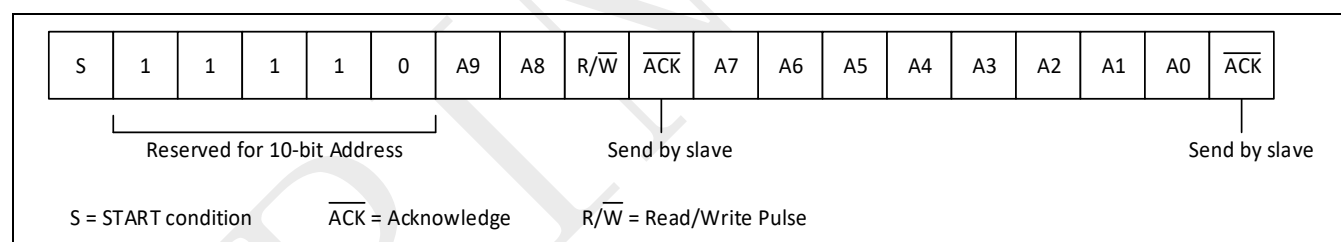
Figure 21-3: 7-bit address format



10-Bit Address Format

During 10-bit addressing, two bytes are transferred to set the 10-bit address. The transfer of the first byte contains the following bit definition. The first five bits (bits 7:3) notify the slaves that this is a 10-bit transfer followed by the next two bits (bits 2:1), which set the slaves address bits 9:8, and the LSB bit (bit 0) is the R/W bit. The second byte transferred sets bits 7:0 of the slave address. [Figure 21-4](#) shows the 10-bit address format.

Figure 21-4: 10-bit address format



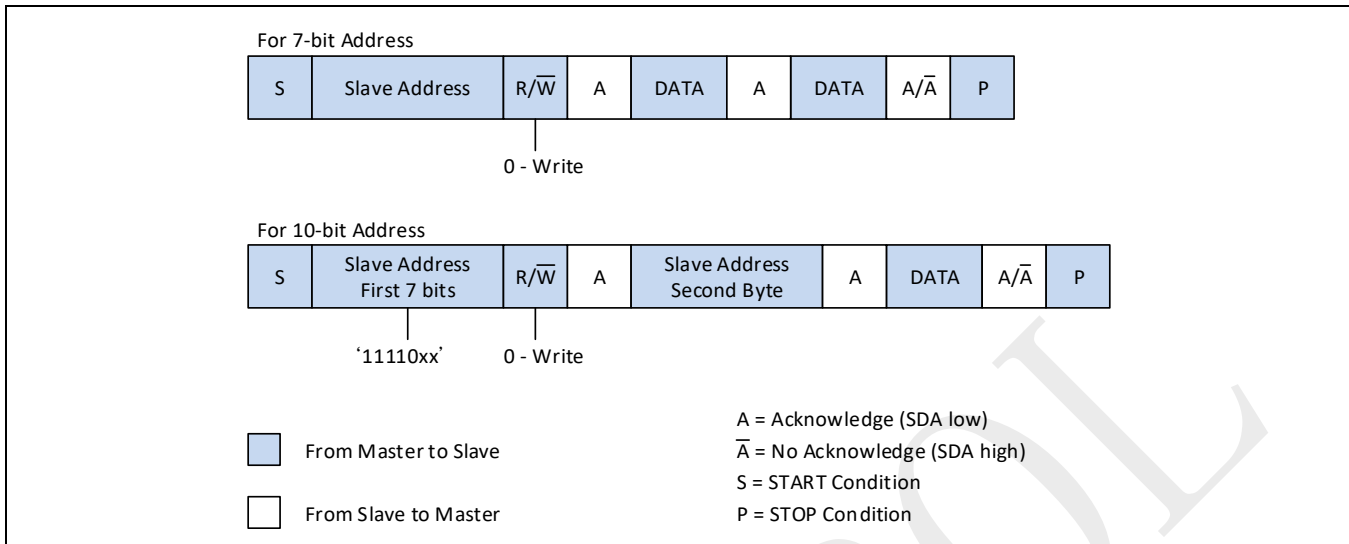
21.5.3 Transmitting and receiving protocol

The master can initiate data transmission and reception to/from the bus, acting as either a master-transmitter or master-receiver. A slave responds to requests from the master to either transmit data or receive data to/from the bus, acting as either a slave-transmitter or slave-receiver, respectively.

Master-Transmitter and Slave-Receiver

All data is transmitted in byte format, with no limit on the number of bytes transferred per data transaction. After the master sends the address and R/W bit or the master transmits a byte of data to the slave, the slave-receiver must respond with the acknowledge signal (ACK). When a slave-receiver does not respond with an ACK pulse, the master aborts the transfer by issuing a STOP condition. The slave must leave the SDA line high so that the master can abort the transfer.

Figure 21-5: Master-transmitter protocol

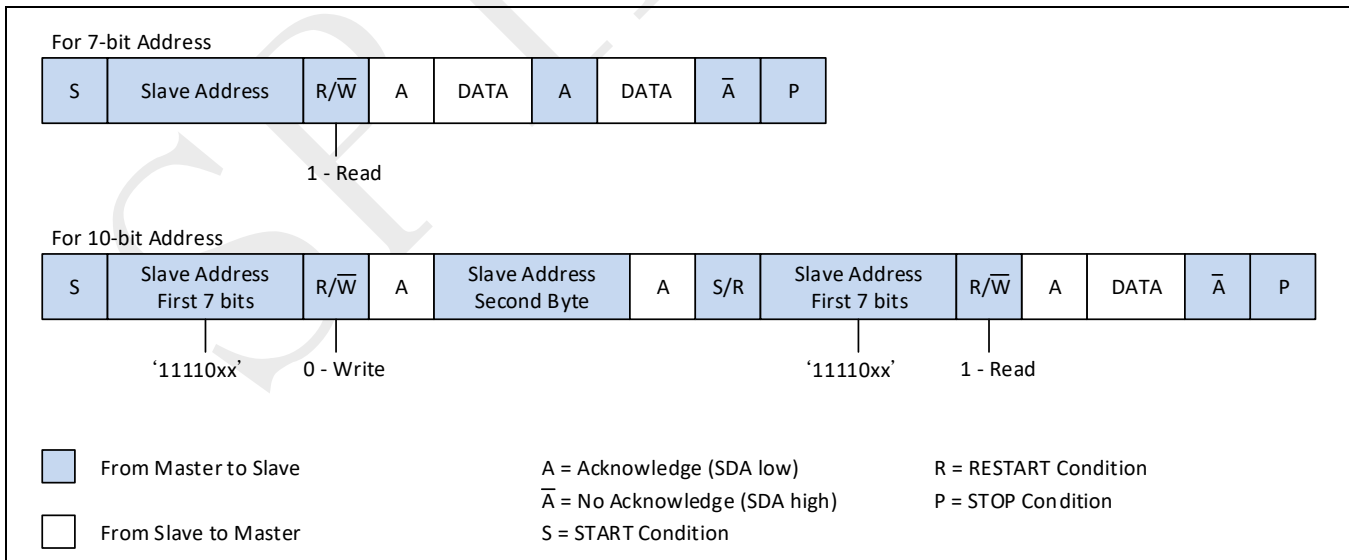


Master-Receiver and Slave-Transmitter

If the master is receiving data as shown in Figure 21-6, then the master responds to the slave-transmitter with an acknowledge pulse after a byte of data has been received, except for the last byte. This method is how the master-receiver notifies the slave-transmitter that this is the last byte. The slave-transmitter relinquishes the SDA line after detecting the No Acknowledge (NACK) so that the master can issue a STOP condition.

When a master refuses to relinquish the bus with a STOP condition, the master can issue a RESTART condition. This is identical to a START condition except it occurs after the ACK pulse. Operating in master mode, the I2C can then communicate with the same slave using a transfer of a different direction.

Figure 21-6: Master-receiver protocol



21.6 Registers

21.6.1 I2C register map

Table 21-2: I2C Module Base Address

Peripheral Module	Base Address
I2C0	0x4000 3500

Table 21-3: I2C Register Map

Register	Offset	Description	Reset Value
<u>I2CCTL</u> *	0x0	I2C Control Register	0x00000062
<u>I2CMADDR</u>	0x4	I2C Master Address Register	0x00090055
<u>I2CSADDR</u> *	0x8	I2C Slave Address Register	0x00000055
<u>I2CDATACMD</u>	0xC	I2C Data and Command Register	0x00000000
<u>I2CSSLCNT</u> *	0x10	Standard Speed I2C Clock SCL Low Count Register	0x0000024C
<u>I2CSSHCNT</u> *	0x14	Standard Speed I2C Clock SCL High Count Register	0x000001F4
<u>I2CFSLCNT</u> *	0x18	Fast Speed I2C Clock SCL Low Count Register	0x000000A3
<u>I2CFSHCNT</u> *	0x1C	Fast Speed I2C Clock SCL High Count Register	0x0000004B
<u>I2CHSLCNT</u> *	0x20	High Speed I2C Clock SCL Low Count Register	0x00000014
<u>I2CHSHCNT</u> *	0x24	High Speed I2C Clock SCL High Count Register	0x00000008
<u>I2CFSSPKLEN</u> *	0x28	Fast Speed I2C Spike Suppresion Limit Register	0x00000006
<u>I2CHSSPKLEN</u> *	0x2C	High Speed I2C Spike Suppresion Limit Register	0x00000002
<u>I2CSDASETUP</u> *	0x30	I2C SDA Setup Time Register	0x00000064
<u>I2CSDAHOLD</u> *	0x34	I2C SDA Hold Time Register	0x00000001
<u>I2CTXLVL</u>	0x38	I2C Transmit FIFO Level Register	0x00000000
<u>I2CRXLVL</u>	0x3C	I2C Receive FIFO Level Register	0x00000000
<u>I2CTXTH</u> *	0x40	I2C Transmit FIFO Threshold Register	0x00000000
<u>I2CRXTH</u> *	0x44	I2C Receive FIFO Threshold Register	0x00000000
<u>I2CSTS</u>	0x54	I2C Status Register	0x00000018
<u>I2CTXABRTSRC</u>	0x58	I2C Transmit Abort Source Register	0x00000000
<u>I2CRAWIF</u>	0x5C	I2C Interrupt Raw Flag Register	0x00000000
<u>I2CIF</u>	0x60	I2C Interrupt Flag Register	0x00000000
<u>I2CIE</u> *	0x64	I2C Interrupt Enable Register	0x000010FF
<u>I2CIC</u>	0x68	I2C Interrupt Enable Register	0x00000000
<u>I2CREGKEY</u>	0x6C	I2C Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the I2CREGKEY=0x1ACCE551.

21.6.2 I2C registers

Table 21-4: I2C Control Register (I2CCTL) Layout

I2CCTL (I2C Control Register) Offset: 0x0 Default: 0x00000062 Access: I2Cx->I2CCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CLKSTRETCH	ACKGC	RESTARTEN	ADDRMODE	SPEED		MODE	EN

Table 21-5: I2C Control Register (I2CCTL) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CLKSTRETCH	RW	0x0	Clock stretch on RX FIFO full 0: Disable 1: Enable
6	ACKGC	RW	0x1	Respond upon General Call. This bit is applicable only for slave mode 0: Responds General Call with an NACK 1: Responds General Call with an ACK
5	RESTARTEN	RW	0x1	Determines whether RESTART conditions may be sent when acting as a master. Some older slaves do not support handling RESTART conditions; however, RESTART conditions are used in several I2C operations. When RESTART is disabled, the master is prohibited from performing the following functions: - Change direction within a transfer (split) - Send a START BYTE - High-speed mode operation - Combined format transfers in 7-bit addressing modes - Read operation with a 10-bit address - Send multiple bytes per transfer By replacing RESTART condition followed by a STOP and a subsequent START condition, split operations are broken down into multiple I2C transfers. If the above operations are performed, it will result in setting bit 6 (TXABRT) of the I2CRAWIF register.

Bits	Field Name	Type	Reset	Description
				0: Disable 1: Enable
4	ADDRMODE	RW	0x0	Addressing mode 0: 7-bit addressing The I2C ignores transactions that involve 10-bit addressing; for 7-bit addressing, only the lower 7 bits of the I2CSLVADDR register are compared. 1: 10-bit addressing The I2C responds to only 10-bit addressing transfers that match the full 10 bits of the I2CSLVADDR register.
3:2	SPEED	RW	0x0	Speed in master mode 00: Standard mode (100 kbit/s) 01: Fast mode (400 kbit/s) 10: High speed mode (3.4 Mbit/s)
1	MODE	RW	0x1	I2C mode 0: Set I2C in slave mode 1: Set I2C in master mode
0	EN	RW	0x0	I2C enable (There are two I2C clock delay when enable or disable the I2C). When I2C is disabled, the following occurs: - The TX FIFO and RX FIFO get flushed. - Status bits in the I2CIF register are still active until I2C goes into IDLE state. - If the module is transmitting, it stops as well as deletes the contents of the transmit FIFO after the current transfer is complete. - If the module is receiving, the I2C stops the current transfer at the end of the current byte and does not acknowledge the transfer. 0: Disable I2C (Hold FIFOs in an erased state). Software can disable I2C while it is active. However, it is important that care be taken to ensure that i2c is disabled properly. 1: Enable I2C

Table 21-6: I2C Master Address Register (I2CMADDR) Layout

I2CMADDR (I2C Master Address Register) Offset: 0x4 Default: 0x00090055 Access: I2Cx->I2CMADDR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
HSMCODE							
15	14	13	12	11	10	9	8
RESERVED				SPECIALCMD		TARADDR	
7	6	5	4	3	2	1	0
TARADDR							

Table 21-7: I2C Master Address Register (I2CMADDR) Description

Bits	Field Name	Type	Reset	Description
31:24	RESERVED_31_24	RO	0x0	Reserved.
23:16	HSMCODE	RW	0x9	8-bit high-speed mode master code in format of (0x00001xxx) This register can be written only when the I2C interface is disabled (I2CCTL_EN=0) and the value to write is in valid format (00001xxx) Writes at other times have no effect.
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:10	SPECIALCMD	RW	0x0	Special command 00: Disable special command 10: General Call After issuing a General Call, only writes may be performed. Attempting to issue a read command results in setting bit 6 (TXABRT) of the I2CRAWIF register. The I2C remains in General Call mode until the SPECIAL bit is cleared 11: Start Byte
9:0	TARADDR	RW	0x55	Target address for any master transaction. These bits are ignored for issuing a General Call. To generate a START BYTE, the CPU needs to write only once into these bits. It should not be the same as I2CSLVADDR.

Table 21-8: I2C Slave Address Register (I2CSADDR) Layout

I2CSADDR (I2C Slave Address Register) Offset: 0x8 Default: 0x00000055 Access: I2Cx->I2CSADDR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						VAL	
7	6	5	4	3	2	1	0
VAL							

Table 21-9: I2C Slave Address Register (I2CSADDR) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9:0	VAL	RW	0x55	Slave address when I2C acts as a slave This register can be written only when I2CCTL_EN=0. Writes at other times have no effect. Note: The default values cannot be any of the reserved address locations: that is, 0x00 to 0x07, or 0x78 to 0x7f. The correct operation of the device is not guaranteed if you program the I2CSLVADDR or I2CMasterADDR to a reserved value.

Table 21-10: I2C Data and Command Register (I2CDATACMD) Layout

I2CDATACMD (I2C Data and Command Register) Offset: 0xC Default: 0x00000000 Access: I2Cx->I2CDATACMD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							CMD
7	6	5	4	3	2	1	0
DATA							

Table 21-11: I2C Data and Command Register (I2CDATACMD) Description

Bits	Field Name	Type	Reset	Description
31:9	RESERVED_31_9	RO	0x0	Reserved.
8	CMD	RW	0x0	Read or write control 0: Write 1: Read
7:0	DATA	RW	0x0	Data transmitted or received on the I2C bus. A write to the register sets the data to be transmitted. A read to the register gets the received data.

Table 21-12: Standard Speed I2C Clock SCL Low Count Register (I2CSSLCNT) Layout

I2CSSLCNT (Standard Speed I2C Clock SCL Low Count Register) Offset: 0x10 Default: 0x0000024C Access: I2Cx->I2CSSLCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-13: Standard Speed I2C Clock SCL Low Count Register (I2CSSLCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x24C	Duration for SCL low in standard speed mode. i.e. SCL is low for I2CSSLCNT clock cycles The minimum valid value is 4. Write a value below 4 will be reset as 4 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-14: Standard Speed I2C Clock SCL High Count Register (I2CSSHCNT) Layout

I2CSSHCNT (Standard Speed I2C Clock SCL High Count Register) Offset: 0x14 Default: 0x000001F4 Access: I2Cx->I2CSSHCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-15: Standard Speed I2C Clock SCL High Count Register (I2CSSHCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x1F4	Duration for SCL high in standard speed mode. i.e. SCL is high for I2CSSHCNT clock cycles The minimum valid value is 5. Write a value below 5 will be reset as 5 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-16: Fast Speed I2C Clock SCL Low Count Register (I2CFSLCNT) Layout

I2CFSLCNT (Fast Speed I2C Clock SCL Low Count Register) Offset: 0x18 Default: 0x000000A3 Access: I2Cx->I2CFSLCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-17: Fast Speed I2C Clock SCL Low Count Register (I2CFSLCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0xA3	Duration for SCL low in fast speed mode. i.e. SCL is low for I2CFSLCNT clock cycles The minimum valid value is 4. Write a value below 4 will be reset as 4 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-18: Fast Speed I2C Clock SCL High Count Register (I2CFSHCNT) Layout

I2CFSHCNT (Fast Speed I2C Clock SCL High Count Register) Offset: 0x1C Default: 0x0000004B Access: I2Cx->I2CFSHCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-19: Fast Speed I2C Clock SCL High Count Register (I2CFSHCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x4B	Duration for SCL high in fast speed mode. i.e. SCL is high for I2CFSHCNT clock cycles The minimum valid value is 5. Write a value below 5 will be reset as 5 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-20: High Speed I2C Clock SCL Low Count Register (I2CHSLCNT) Layout

I2CHSLCNT (High Speed I2C Clock SCL Low Count Register) Offset: 0x20 Default: 0x00000014 Access: I2Cx->I2CHSLCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-21: High Speed I2C Clock SCL Low Count Register (I2CHSLCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x14	Duration for SCL low in high speed mode. i.e. SCL is low for I2CHSLCNT clock cycles The minimum valid value is 4. Write a value below 4 will be reset as 4 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-22: High Speed I2C Clock SCL High Count Register (I2CHSHCNT) Layout

I2CHSHCNT (High Speed I2C Clock SCL High Count Register) Offset: 0x24 Default: 0x00000008 Access: I2Cx->I2CHSHCNT							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-23: High Speed I2C Clock SCL High Count Register (I2CHSHCNT) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x8	Duration for SCL high in high speed mode. i.e. SCL is high for I2CHSHCNT clock cycles The minimum valid value is 1. Write a 0 will be reset as 1 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-24: Fast Speed I2C Spike Suppresion Limit Register (I2CFSSPKLEN) Layout

I2CFSSPKLEN (Fast Speed I2C Spike Suppresion Limit Register)					Offset: 0x28	Default: 0x00000006	
Access: I2Cx->I2CFSSPKLEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

Table 21-25: Fast Speed I2C Spike Suppresion Limit Register (I2CFSSPKLEN) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	VAL	RW	0x6	Duration in fast speed mode of the longest spike in the SCL or SDA lines that are filtered out by the spike suppression logic. The minimum value is 1 and write a 0 will be reset as 1 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-26: High Speed I2C Spike Suppresion Limit Register (I2CHSSPKLEN) Layout

I2CHSSPKLEN (High Speed I2C Spike Suppresion Limit Register) Offset: 0x2C Default: 0x00000002 Access: I2Cx->I2CHSSPKLEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

Table 21-27: High Speed I2C Spike Suppresion Limit Register (I2CHSSPKLEN) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	VAL	RW	0x2	Duration in high speed mode of the longest spike in the SCL or SDA lines that are filtered out by the spike suppression logic. The minimum value is 1 and write a 0 will be reset as 1 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-28: I2C SDA Setup Time Register (I2CSDASETUP) Layout

I2CSDASETUP (I2C SDA Setup Time Register) Offset: 0x30 Default: 0x00000064 Access: I2Cx->I2CSDASETUP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
VAL							

Table 21-29: I2C SDA Setup Time Register (I2CSDASETUP) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	VAL	RW	0x64	Configure SDA setup time as VAL CLK_I2C cycles i.e. If the required setup delay is 1000ns, then for an CLK_I2C frequency of 10 MHz, this register is recommended to be programmed as 11. The minimum value is 2 and write a value below 2 will be reset as 2 by the hardware. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-30: I2C SDA Hold Time Register (I2CSDAHOLD) Layout

I2CSDAHOLD (I2C SDA Hold Time Register) Offset: 0x34 Default: 0x00000001 Access: I2Cx->I2CSDAHOLD							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 21-31: I2C SDA Hold Time Register (I2CSDAHOLD) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x1	Configure SDA hold time as VAL CLK_I2C cycles i.e. If the required hold delay is 1000ns, then for an CLK_I2C frequency of 10 MHz, this register is recommended to be programmed as 11. This register can be written only when I2CCTL_EN=0. Writes at other times have no effect.

Table 21-32: I2C Transmit FIFO Level Register (I2CTXLVL) Layout

I2CTXLVL (I2C Transmit FIFO Level Register) Offset: 0x38 Default: 0x00000000 Access: I2Cx->I2CTXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

Table 21-33: I2C Transmit FIFO Level Register (I2CTXLVL) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4:0	VAL	RO	0x0	Number of valid data entries in the transmit FIFO.

Table 21-34: I2C Receive FIFO Level Register (I2CRXLVL) Layout

I2CRXLVL (I2C Receive FIFO Level Register) Offset: 0x3C Default: 0x00000000 Access: I2Cx->I2CRXLVL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			VAL				

Table 21-35: I2C Receive FIFO Level Register (I2CRXLVL) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4:0	VAL	RO	0x0	Number of valid data entries in the receive FIFO.

Table 21-36: I2C Transmit FIFO Threshold Register (I2CTXTH) Layout

I2CTXTH (I2C Transmit FIFO Threshold Register) Offset: 0x40 Default: 0x00000000 Access: I2Cx->I2CTXTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

Table 21-37: I2C Transmit FIFO Threshold Register (I2CTXTH) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3:0	VAL	RW	0x0	Transmit FIFO threshold level to trigger the TFEMPTY interrupt

Table 21-38: I2C Receive FIFO Threshold Register (I2CRXTH) Layout

I2CRXTH (I2C Receive FIFO Threshold Register) Offset: 0x44 Default: 0x00000000 Access: I2Cx->I2CRXTH							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				VAL			

Table 21-39: I2C Receive FIFO Threshold Register (I2CRXTH) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3:0	VAL	RW	0x0	Receive FIFO threshold level to trigger the RFFULL interrupt

Table 21-40: I2C Status Register (I2CSTS) Layout

I2CSTS (I2C Status Register) Offset: 0x54 Default: 0x00000018 Access: I2Cx->I2CSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						SRXDLOST	SRXABRT
7	6	5	4	3	2	1	0
EN	RXFULL	RXNE	TXNF	TXEMPTY	SACT	MACT	ACT

Table 21-41: I2C Status Register (I2CSTS) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	SRXDLOST	RO	0x0	Slave received data lost upon disable I2C 0: Slave receiver operation is done before I2CENABLE is changed from 1 to 0 1: Slave receiver operation is aborted with at least one data byte received from an I2C transfer when I2CENABLE is changed from 1 to 0
8	SRXABRT	RO	0x0	Slave disabled while it is busy 0: I2CENABLE is stable during slave operation 1: I2CENABLE is changed from 1 to 0 when I2C slave is receiving from a remote master on (a) The address byte of the slave transmitter (b) The address and data bytes of the slave receiver
7	EN	RO	0x0	I2C enabled status 0: I2C is deemed completely inactive SLVRDLOST and SLVDISONBUSY bits can be read safely 1: I2C is deemed to be in an enabled state
6	RXFULL	RO	0x0	Receive FIFO completely full 0: Receive FIFO contains empty location 1: Receive FIFO is completely full
5	RXNE	RO	0x0	Receive FIFO not empty (so can read data from it) 0: Receive FIFO is empty 1: Receive FIFO contains one or more entries
4	TXNF	RO	0x1	Transmit FIFO not full (so can write data to it) 0: Transmit FIFO is full 1: Transmit FIFO contains empty locations

Bits	Field Name	Type	Reset	Description
3	TXEMPTY	RO	0x1	Transmit FIFO completely empty 0: Transmit FIFO contains one or more entries 1: Transmit FIFO is empty
2	SACT	RO	0x0	Slave FSM (Finite State Machine) activity status 0: Slave FSM is in IDLE state (Inactive) 1: Slave FSM is not in IDLE state (Active)
1	MACT	RO	0x0	Master FSM activity status 0: Master FSM is in IDLE state (Inactive) 1: Master FSM is not in IDLE state (Active)
0	ACT	RO	0x0	I2C activity status (Logic OR of SACT and MACT) 0: Both master and slave FSM are in idle state 1: Either master or slave FSM is active

Table 21-42: I2C Transmit Abort Source Register (I2CTXABRTSRC) Layout

I2CTXABRTSRC (I2C Transmit Abort Source Register) Offset: 0x58 Default: 0x00000000 Access: I2Cx->I2CTXABRTSRC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 SCMDERR	14 SARBLOST	13 STFIFOCLR	12 MARBLOST	11 MDISABLED	10 RD10BNORESTART	9 STARTNORESTART	8 HSNORESTART
7 STARTACK	6 HSMCAACK	5 GCREAD	4 GCNACK	3 TXDNACK	2 ADDR10B2NACK	1 ADDR10B1NACK	0 ADDR7BNACK

Table 21-43: I2C Transmit Abort Source Register (I2CTXABRTSRC) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	SCMDERR	RO	0x0	Role of I2C: Slave transmitter This bit is set when the processor side responds to a slave mode request for data to be transmitted to a remote master and user writes a 1 in CMD (bit 8) of I2CDATAACMD register. 0: Event did not occur 1: Event occurred
14	SARBLOST	RO	0x0	Role of I2C: Slave transmitter This bit is set when slave lost the bus while transmitting data to a remote master. I2CTXABRTSRC.MARBLOST is set at the same time. Note: Even though the slave never 'owns' the

Bits	Field Name	Type	Reset	Description
				bus, something could go wrong on the bus. This is a failsafe check. For instance, during a data transmission at the low-to-high transition of SCL, if what is on the data bus is not what is supposed to be transmitted, then I2C no longer own the bus. 0: Event did not occur 1: Event occurred
13	STFIFOCLR	RO	0x0	Role of I2C: Slave transmitter This bis is set when the slave has received an NACK from the external master and the TX FIFO is not empty, so the slave issues a TXABRT interrupt to clear old data in TX FIFO. 0: Event did not occur 1: Event occurred
12	MARBLOST	RO	0x0	Role of I2C: Master/slave transmitter This bit is set when master has lost arbitration, or if I2CTXABRTSRC.SARBLOST is also set, then the slave transmitter has lost arbitration. 0: Event did not occur 1: Event occurred
11	MDISABLED	RO	0x0	Role of I2C: Slave transmitter/receiver This bit is set upon the writing to TX FIFO without the request from a master. i.e. The device itself is a slave instead of a master. And it has not received a read request from the external master 0: Event did not occur 1: Event occurred
10	RD10BNORESTART	RO	0x0	Role of I2C: Master receiver This bit is set when the restart is disabled (I2CCTL.RESTARTEN=0) and the master sends a read command in 10-bit addressing mode. 0: Event did not occur 1: Event occurred
9	STARTNORESTART	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when the restart is disabled (I2CCTL.RESTARTEN=0) and the user is trying to send a START Byte. Follow configurations below to avoid this error (1) I2CCTL.RESTARTEN=1 (2) I2CMASTERADDR.SPECIAL=0 or I2CMASTERADDR.GCORSTART=0 Otherwise, this bit is set again at next clock after it is cleared by writing a 1 to I2CIC_TXABRT

Bits	Field Name	Type	Reset	Description
				0: Event did not occur 1: Event occurred
8	HSNORESTART	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when the restart is disabled (I2CCTL.RESTARTEN=0) and the user is trying to use the master to transfer data in high speed mode. 0: Event did not occur 1: Event occurred
7	STARTACK	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when master has sent a START Byte and the START Byte was acknowledged (wrong behavior). 0: Event did not occur 1: Event occurred
6	HSMCAACK	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when master is in high speed mode and the high speed master code was acknowledged (wrong behavior). 0: Event did not occur 1: Event occurred
5	GCREAD	RO	0x0	Role of I2C: Master transmitter This bit is set when master sent a General Call but the user programmed the byte following the General Call to be a read from the bus (I2CDATAACMD_CMD=1). 0: Event did not occur 1: Event occurred
4	GCNACK	RO	0x0	Role of I2C: Master transmitter This bit is set when master sent a General Call and no slave on the bus acknowledged the General Call. 0: Event did not occur 1: Event occurred
3	TXDNACK	RO	0x0	Role of I2C: Master transmitter This bit is set when master has received an acknowledgement for the address, but when it sent data byte(s) following the address, it did not receive an acknowledge from the remote slave(s). 0: Event did not occur 1: Event occurred
2	ADDR10B2NACK	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when master is in 10-bit address mode and the second address byte of the 10-bit

Bits	Field Name	Type	Reset	Description
				address was not acknowledged by any slave. 0: Event did not occur 1: Event occurred
1	ADDR10B1NACK	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when master is in 10-bit address mode and the first 10-bit address byte was not acknowledged by any slave. 0: Event did not occur 1: Event occurred
0	ADDR7BNACK	RO	0x0	Role of I2C: Master transmitter/receiver This bit is set when master is in 7-bit addressing mode and the address sent was not acknowledged by any slave. 0: Event did not occur 1: Event occurred

Table 21-44: I2C Interrupt Raw Flag Register (I2CRAWIF) Layout

I2CRAWIF (I2C Interrupt Raw Flag Register) Offset: 0x5C Default: 0x00000000 Access: I2Cx->I2CRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

Table 21-45: I2C Interrupt Raw Flag Register (I2CRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:13	RESERVED_31_13	RO	0x0	Reserved.
12	GENCALL	RO	0x0	General Call address received and acknowledged flag 0: Not occur 1: Occurred
11	STARTDET	RO	0x0	START or RESTART condition flag 0: Condition is not detected 1: Condition is detected
10	STOPDET	RO	0x0	STOP condition flag 0: Condition is not detected 1: Condition is detected
9	ACKDET	RO	0x0	Latched flag for detected ACK from external receiver

Bits	Field Name	Type	Reset	Description
				0: ACK is not detected 1: ACK is detected
8	ACT	RO	0x0	Latched I2C activity status flag 0: There is no activity detected on I2C interface 1: Activity has been detected on I2C interface
7	RXDONE	RO	0x0	Flag to indicate external master receive is done when I2C acts as a slave transmitter 0: Slave transmission is acknowledged by the other master, which continues data receiving. 1: Slave transmission is not acknowledge by the other master, which finishes the data receiving
6	TXABRT	RO	0x0	Latched transmission abort flag 0: No transmission abortion occurred 1: Transimission aborted and TX FIFO remains in flushed state until write a 1 to I2CIC_TXABRT
5	RDREQ	RO	0x0	Latched external master read request flag when the I2C is acting as a slave 0: No read request is received from another master 1: Read request was received from another master. The processor must respond to this interrupt and then write the requested data to the I2CDATAACMD register.
4	TXDREQ	RO	0x0	Flag to indicate there is not enough data in the transmit FIFO and request for more data 0: I2CTXLVL>I2CTXTH 1: I2CTXLVL<=I2CTXTH This bit is self-cleared when I2CTXLVL>I2CTXTH
3	TXOVF	RO	0x0	Flag to indicate an attempt to write I2CDATAACMD register when transmit FIFO is full 0: Transmit FIFO does not overflow 1: Transmit FIFO overflows
2	RXDAV	RO	0x0	Flag to indicate there is enough data in the receive FIFO and available for read 0: I2CRXLVL<= I2CRXTH 1: I2CRXLVL>I2CRXTH This bit is self-cleared when I2CRXLVL<=I2CRXTH
1	RXOVF	RO	0x0	Flag to indicate the receive FIFO is completely filled to 16 and an additional byte is received from an external I2C device. The I2C acknowledges this, but any data bytes received after the FIFO is full are lost. 0: Receive FIFO does not overflow 1: Receive FIFO overflows

Bits	Field Name	Type	Reset	Description
0	RXUDF	RO	0x0	Flag to indicate the processor attempts to read the receive FIFO via I2CDATA CMD register when it is empty 0: Receive FIFO does not underflow 1: Receive FIFO underflows

Table 21-46: I2C Interrupt Flag Register (I2CIF) Layout

I2CIF (I2C Interrupt Flag Register) Offset: 0x60 Default: 0x00000000 Access: I2Cx->I2CIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		INT	GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

Table 21-47: I2C Interrupt Flag Register (I2CIF) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13	INT	RO	0x0	Global interrupt flag 0: Not occur 1: Occurred
12	GENCALL	RO	0x0	General Call address received and acknowledged flag 0: Not occur 1: Occurred
11	STARTDET	RO	0x0	START or RESTART condition flag 0: Condition is not detected 1: Condition is detected
10	STOPDET	RO	0x0	STOP condition flag 0: Condition is not detected 1: Condition is detected
9	ACKDET	RO	0x0	Latched flag for detected ACK from external receiver 0: ACK is not detected 1: ACK is detected
8	ACT	RO	0x0	Latched I2C activity status flag 0: There is no activity detected on I2C interface 1: Activity has been detected on I2C interface
7	RXDONE	RO	0x0	Flag to indicate external master receive is done when I2C acts as a slave transmitter

Bits	Field Name	Type	Reset	Description
				0: Slave transmission is acknowledged by the other master, which continues data receiving. 1: Slave transmission is not acknowledge by the other master, which finishes the data receiving
6	TXABRT	RO	0x0	Latched transmission abort flag 0: No transmission abortion occurred 1: Transimission aborted and TX FIFO remains in flushed state until write a 1 to I2CIC_TXABRT
5	RDREQ	RO	0x0	Latched external master read request flag when the I2C is acting as a slave 0: No read request is received from another master 1: Read request was received from another master. The processor must respond to this interrupt and then write the requested data to the I2CDATACMD register.
4	TXDREQ	RO	0x0	Flag to indicate there is not enough data in the transmit FIFO and request for more data 0: I2CTXLVL>I2CTXTH 1: I2CTXLVL<=I2CTXTH This bit is self-cleared when I2CTXLVL>I2CTXTH
3	TXOVF	RO	0x0	Flag to indicate an attempt to write I2CDATACMD register when transmit FIFO is full 0: Transmit FIFO does not overflow 1: Transmit FIFO overflows
2	RXDAV	RO	0x0	Flag to indicate there is enough data in the receive FIFO and available for read 0: I2CRXLVL<= I2CRXTH 1: I2CRXLVL>I2CRXTH This bit is self-cleared when I2CRXLVL<=I2CRXTH
1	RXOVF	RO	0x0	Flag to indicate the receive FIFO is completely filled to 16 and an additional byte is received from an external I2C device. The I2C acknowledges this, but any data bytes received after the FIFO is full are lost. 0: Receive FIFO does not overflow 1: Receive FIFO overflows
0	RXUDF	RO	0x0	Flag to indicate the processor attempts to read the receive FIFO via I2CDATACMD register when it is empty 0: Receive FIFO does not underflow 1: Receive FIFO underflows

Table 21-48: I2C Interrupt Enable Register (I2CIE) Layout

I2CIE (I2C Interrupt Enable Register) Offset: 0x64 Default: 0x000010FF Access: I2Cx->I2CIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

Table 21-49: I2C Interrupt Enable Register (I2CIE) Description

Bits	Field Name	Type	Reset	Description
31:13	RESERVED_31_13	RO	0x0	Reserved.
12	GENCALL	RW	0x1	Enable/disable GENCALL interrupt 0: Disable GENCALL interrupt 1: Enable GENCALL interrupt
11	STARTDET	RW	0x0	Enable/disable STARTDET interrupt 0: Disable STARTDET interrupt 1: Enable STARTDET interrupt
10	STOPDET	RW	0x0	Enable/disable STOPDET interrupt 0: Disable STOPDET interrupt 1: Enable STOPDET interrupt
9	ACKDET	RW	0x0	Enable/disable ACKDET interrupt 0: Disable ACKDET interrupt 1: Enable ACKDET interrupt
8	ACT	RW	0x0	Enable/disable ACTIVITY interrupt 0: Disable ACTIVITY interrupt 1: Enable ACTIVITY interrupt
7	RXDONE	RW	0x1	Enable/disable RXDONE interrupt 0: Disable RXDONE interrupt 1: Enable RXDONE interrupt
6	TXABRT	RW	0x1	Enable/disable TXABRT interrupt 0: Disable TXABRT interrupt 1: Enable TXABRT interrupt
5	RDREQ	RW	0x1	Enable/disable RDREQ interrupt 0: Disable RDREQ interrupt 1: Enable RDREQ interrupt
4	TXDREQ	RW	0x1	Enable/disable TXDREQ interrupt 0: Disable TXDREQ interrupt 1: Enable TXDREQ interrupt

Bits	Field Name	Type	Reset	Description
3	TXOVF	RW	0x1	Enable/disable TXOVF interrupt 0: Disable TXOVF interrupt 1: Enable TXOVF interrupt
2	RXDAV	RW	0x1	Enable/disable RXDAV interrupt 0: Disable RXDAV interrupt 1: Enable RXDAV interrupt
1	RXOVF	RW	0x1	Enable/disable RXOVF interrupt 0: Disable RXOVF interrupt 1: Enable RXOVF interrupt
0	RXUDF	RW	0x1	Enable/disable RXUDF interrupt 0: Disable RXUDF interrupt 1: Enable RXUDF interrupt

Table 21-50: I2C Interrupt Enable Register (I2CIC) Layout

I2CIC (I2C Interrupt Enable Register) Offset: 0x68 Default: 0x00000000 Access: I2Cx->I2CIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		INT	GENCALL	STARTDET	STOPDET	ACKDET	ACT
7	6	5	4	3	2	1	0
RXDONE	TXABRT	RDREQ	TXDREQ	TXOVF	RXDAV	RXOVF	RXUDF

Table 21-51: I2C Interrupt Enable Register (I2CIC) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13	INT	W1C	0x0	Clear global combined interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
12	GENCALL	W1C	0x0	Clear GENCALL interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
11	STARTDET	W1C	0x0	Clear STARTDET interrupt 0: Write a 0 has no effect and always reads back 0

Bits	Field Name	Type	Reset	Description
				1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
10	STOPDET	W1C	0x0	Clear STOPDET interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
9	ACKDET	W1C	0x0	Clear ACKDET interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
8	ACT	W1C	0x0	Clear ACTIVITY interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
7	RXDONE	W1C	0x0	Clear RXDONE interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
6	TXABRT	W1C	0x0	Clear TXABRT interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
5	RDREQ	W1C	0x0	Clear RDREQ interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
4	TXDREQ	W1C	0x0	Clear TXDREQ interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
3	TXOVF	W1C	0x0	Clear TXOVF interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0

Bits	Field Name	Type	Reset	Description
2	RXDAV	W1C	0x0	Clear RXDAV interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
1	RXOVF	W1C	0x0	Clear RXOVF interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0
0	RXUDF	W1C	0x0	Clear RXUDF interrupt 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears the interrupt flag. This bit is self-cleared to 0

Table 21-52: I2C Register Write-Allow Key Register (I2CREGKEY) Layout

I2CREGKEY (I2C Register Write-Allow Key Register)				Offset: 0x6C	Default: 0x1ACCE551		
Access: I2Cx->I2CREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 21-53: I2C Register Write-Allow Key Register (I2CREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected I2C registers. The 28 MSBs of this register cannot be 0x55555555

22 CAN

22.1 Features

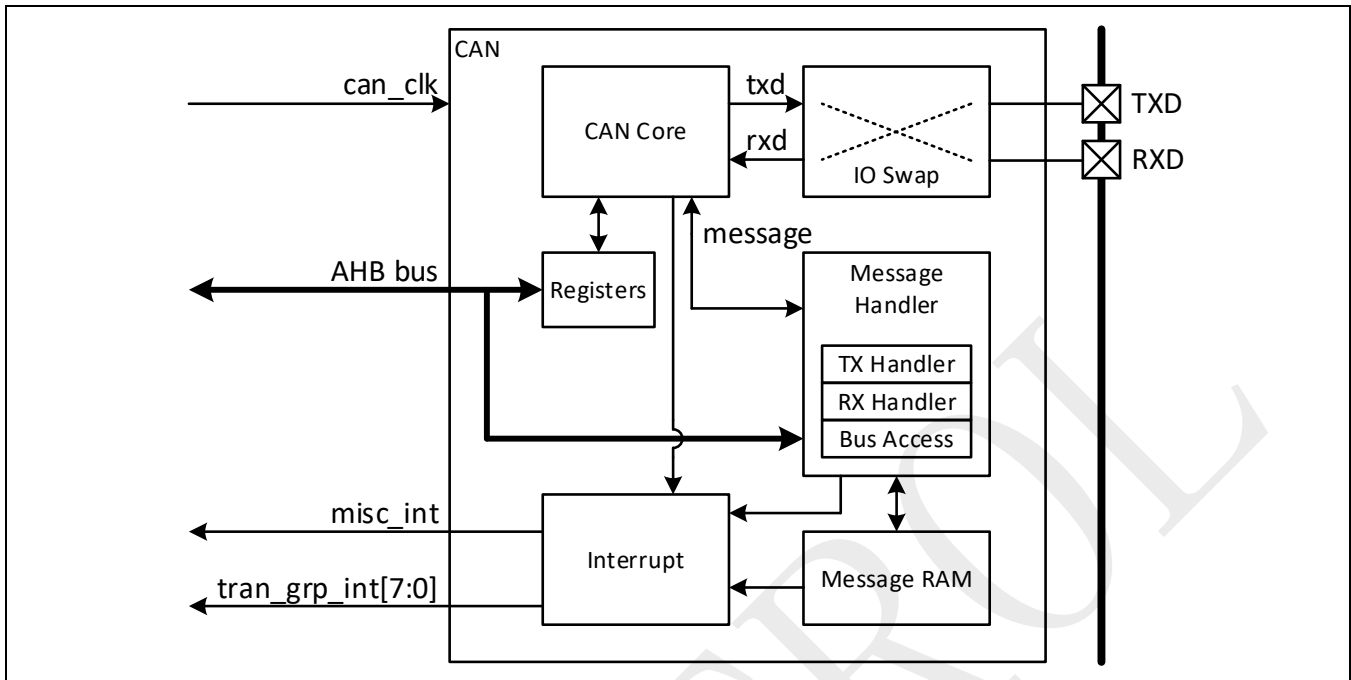
The SPC1169 has implemented one CAN module. It is used for communication based on CAN protocol. An external CAN transceiver is required for the connection to the physical layer (CAN bus). The CAN module features:

- Compatible with ISO-11898-1:2015 specification, whose frame format is called classical CAN and ISO-FDCAN.
- Compatible with Bosch CAN2.0B specification.
- Compatible with Bosch FDCAN V1.0 specification, whose frame format is called NONISO-FDCAN.
- Support Bus-Off recovery.
- Support automatic message re-transmission after transmission failure.
- Support restricted mode.
- Support monitor mode.
- Support test mode.
- Support 64 mailboxes with independent message identifier mask for each mailbox.
- Support automatic remote frame handling.
- Support CAN bus error recording.
- Support 32-bit timestamp.

Note: The CAN module of SPC1169 does not support wake-up function, users need to select a CAN PHY that supports wake-up function and connect the CAN PHY wake-up signal to the MON pin or GPIO pin of SPC1169 to realize the CAN wake-up function.

The frame formats for classical CAN and Bosch CAN2.0B specification are compatible, while the ISO-FDCAN is incompatible with NONISO-FDCAN due to the difference in the frame format and CRC calculation method.

Figure 22-1: CAN Block Diagram



As shown in Figure 22-1, the CAN module mainly consists of register control logic, IO swap logic, CAN core, message handler, message RAM and interrupt handler.

The CAN core sub-module controls the CAN frame transfer and CAN bus status detection.

To communicate on a CAN network, the message objects can be configured individually. Message RAM is used as mailbox to store the message object including its message identifier mask. It supports 32-bit write access and 8-bit, 16-bit, 32-bit read access. ECC protection is provided for data integrity.

The message handler controls message RAM access and arbitrates priority automatically for access via TX handler, RX handler, CAN core and AHB.

After the CAN module is enabled, the TX handler starts from the 1st mailbox to check all the mailboxes one by one. If there is a mailbox requesting message transmission, the TX handler makes the CAN core to read the content and transmit the message. It will start to check the next mailbox after current message transfer is finished successfully or cancelled by user.

The RX handler always searches mailbox from the 1st one when a message identifier is received. Once a valid mailbox is found to be able to store the data, CAN core will save all the received message data into this mailbox.

AHB can write any mailbox at any time except that the mailbox to be written to is currently used for message transfer. In this case, AHB write access to this mailbox is blocked. To configure the mailbox, user should disable the mailbox first and read back the status to check whether it is disabled successfully. If it is still enabled, user should not configure the mailbox, but wait until current message transfer is finished.

The IO swap logic supports pin function swap between TXD and RXD.

Every 8 consecutive mailboxes share one message transfer done interrupt. Therefore, the CAN module has 8 message transfer done interrupts. Other events, such as RAM ECC error or CAN bus error, share one miscellaneous interrupt.

Note: DMA access to the message RAM is allowed.

22.2 CAN I/O signal

Table 22-1 describes the I/O signals connected to a CAN module. They are connected to the I/O pin in CMOS-level and shared with other functional channels. Refer to the pin function description for the details.

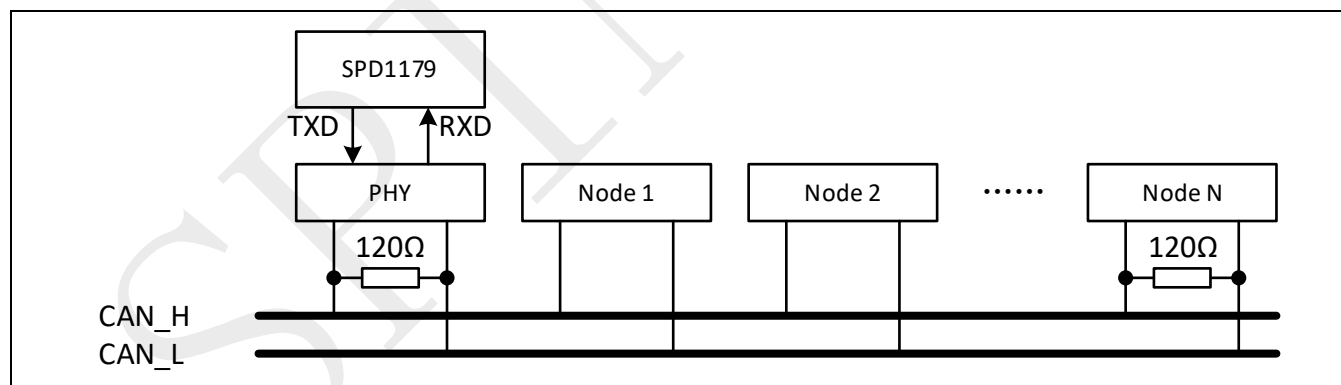
Table 22-1: CAN I/O signal

Name	Type	Description
RXD	Input	Serial Data Input Serial data input to the Receive Shift register, it is connected to CAN PHY RXD pin.
TXD	Output	Serial Data Output Serial data output, it is connected to CAN PHY TXD pin.

22.2.1 CAN device connection

Figure 22-2 shows how the CAN module in this device is connected to other CAN devices on the network. The CAN_H and CAN_L is a twisted pair.

Figure 22-2: CAN device connection

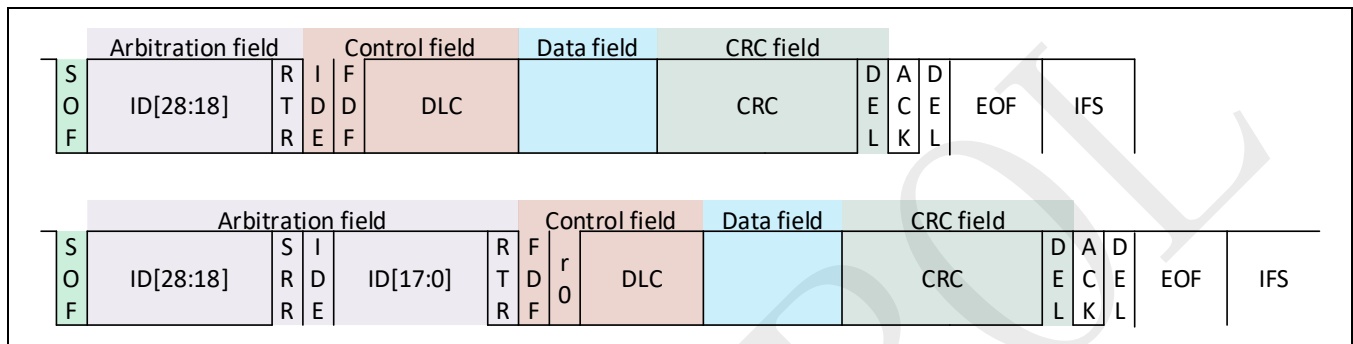


22.3 CAN frame format

22.3.1 ISO classical CAN

The ISO classical CAN supports two frame formats as show in [Figure 22-3](#). The identifier is 11-bit for standard frame and 29-bit for identifier extended frame. The CRC field is always 15-bit. For details, please refer to the ISO-11898-1:2015 specification.

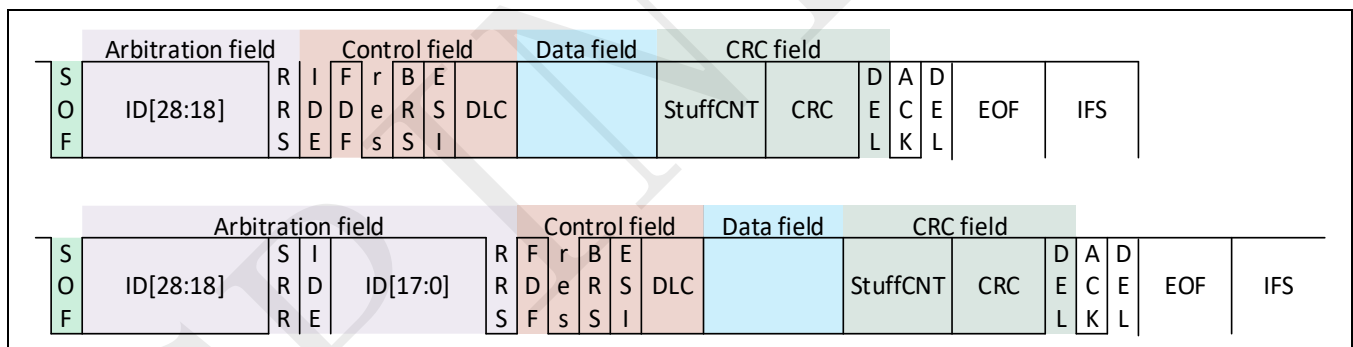
Figure 22-3: ISO classical CAN frame format



22.3.2 ISO FDCAN

The ISO FDCAN supports two frame formats as show in [Figure 22-4](#). It only support data frame transfer. For details, please refer to the ISO-11898-1:2015 specification.

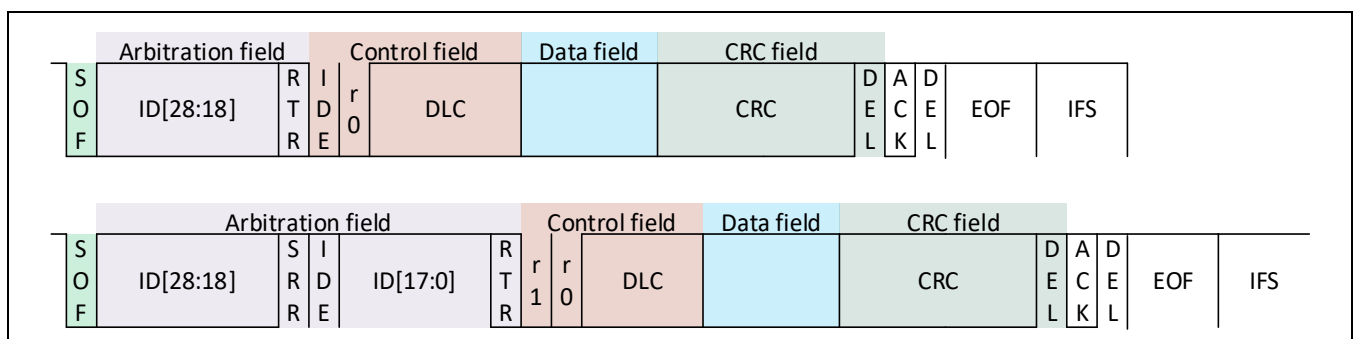
Figure 22-4: ISO FDCAN frame format



22.3.3 CAN frame based on CAN2.0B

The CAN2.0B CAN supports two frame formats as show in [Figure 22-5](#). They are the same as the frame format for ISO classical CAN. For details, please refer to the BOSCH CAN2.0B specification.

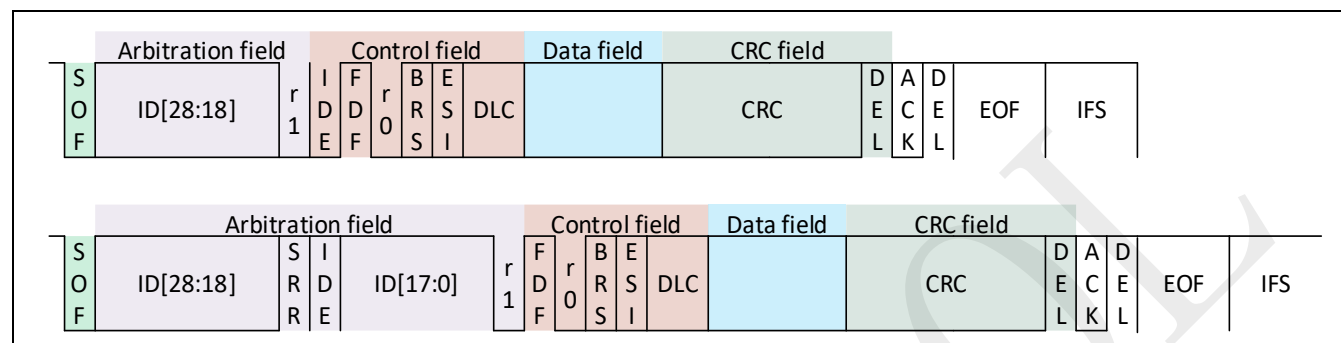
Figure 22-5: CAN2.0B CAN frame format



22.3.4 NONISO FDCAN

The NONISO FDCAN supports two frame formats as show in [Figure 22-6](#). It only supports data frame transfer. Compared to ISO FDCAN, NONISO FDCAN has no stuff count in the CRC field, and uses different CRC algorithm. For details, please refer to the BOSCH FDCAN V1.0 specification.

Figure 22-6: NONISO FDCAN frame format

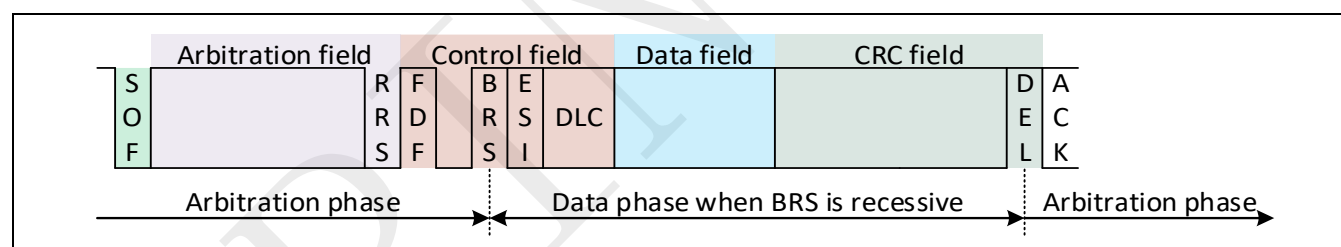


22.3.5 Arbitration phase and Data phase

For bit rate switch enabled ISO-FDCAN or NONISO-FDCAN frame transfer, there are two phases with independent bit rate configurations. As shown in [Figure 22-7](#), the bit transfer after the BRS bit sample point and before the CRC delimiter sample point is called data phase. All other bit transfer is called arbitration phase. The bit rate during data phase is usually faster than that in the arbitration phase.

For bit rate switch disabled FDCAN frame or classical CAN frame, only arbitration phase is available.

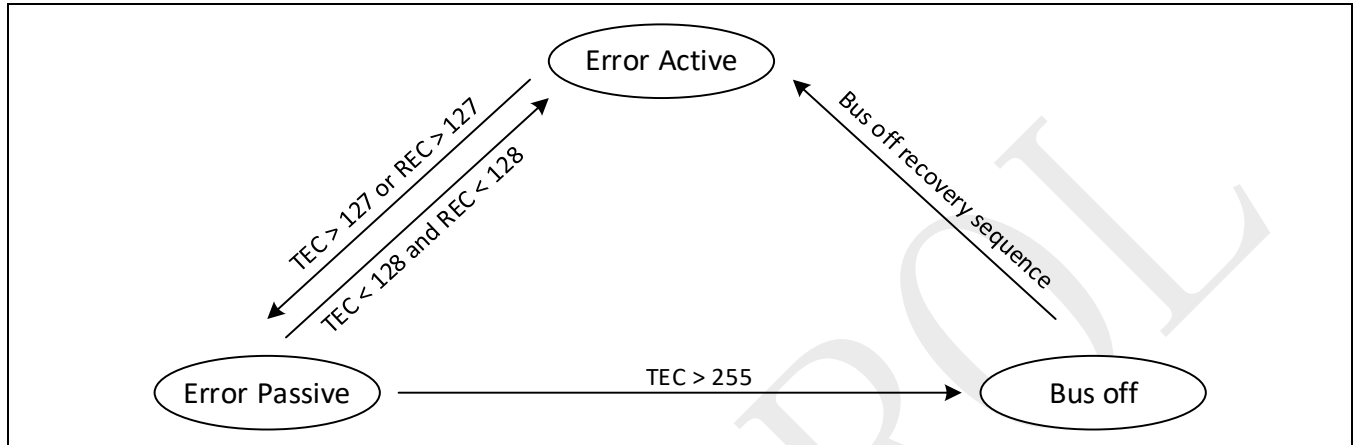
Figure 22-7: Arbitration phase and data phase



22.4 CAN error state

The error state of a CAN node is shown in Figure 22-8. TEC is the transmitter error count and REC is the receiver error count. The rules to update TEC and REC for various CAN modes are defined in the corresponding specification.

Figure 22-8: CAN node error state conversion



22.4.1 Error record

The CAN error includes bit error, stuff error, CRC error, form error and ACK error. The error occurred during arbitration phase is recorded in CANLEC.LEC, and error occurred during data phase is recorded in CANLEC.DLEC.

22.4.2 Error active

When both TEC and REC are below 128, the CAN node is in error active state and sends active error flag upon the detected error event. Once the TEC or REC increases across the threshold as defined in CANECCTL.ERRWARN, an error warning interrupt will be generated.

22.4.3 Error passive

When TEC or REC is above 127, the CAN node is in error passive state and sends passive error flag upon the detected error event. An error passive interrupt is generated when the node enters error passive state. When both TEC and REC decrease below 128, the node enters error active state again.

If the CAN node enters error passive state since REC increases above 127 while the TEC keeps below 128, REC will be loaded as CANECCTL.RECLD after a frame is successfully received.

22.4.4 Bus-Off recovery

When TEC increases above 255, the CAN node enters Bus-Off state and generates a Bus-Off interrupt. If auto Bus-Off recovery is enabled, the Bus-Off recovery sequence will be started after the specified delay. User can also start Bus-Off recovery sequence immediately by setting CANCTL.RECOVERN.

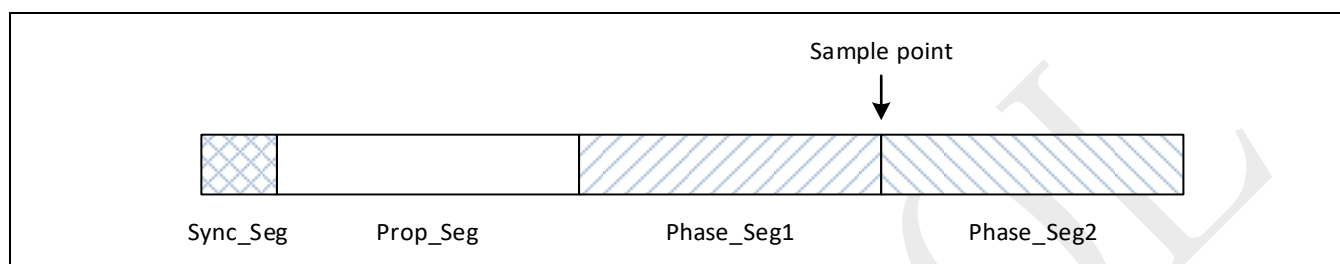
During Bus-Off recovery sequence, REC will decrease by 1 each time 11 consecutive recessive bits are detected. Meanwhile, a bit_0_error interrupt is generated. When REC reaches 0, the Bus-Off recovery sequence finishes, and the CAN node is ready to transfer message normally again.

22.5 CAN operation

22.5.1 Bit rate configuration

Each bit in CAN frame can be divided into 4 segments as show in Figure 22-9. Each duration is in terms of certain time-quantum generated by module clock prescaler.

Figure 22-9: Bit timing



Sync_Seg is called the synchronization segment. It shall be used to synchronize the various CAN nodes on the bus. An edge is expected to be detected within this segment.

Prop_Seg is called the propagation time segment. It shall be used to compensate for physical delay within the network. The delay consists of the signal propagation time on the bus and the internal delay of the CAN nodes.

Phase_Seg1 and Phase_Seg2 are called phase buffer segments 1 and 2. They are used to compensate for edge phase errors. These segments may be lengthened or shortened by resynchronization. Sample point is right at the boundary between Phase_Seg1 and Phase_Seg2.

The bit rate for arbitration phase is called nominal bit rate, which is configured via CANNBTQ register. It can be calculated as: $f_{CLK_CAN} / (NBRP+1) / (1+NSEG1+1+NSEG2+1)$.

The bit rate for data phase is called data bit rate, which is configured via CANDBTQ register. It can be calculated as: $f_{CLK_CAN} / (DBRP+1) / (1+DSEG1+1+DSEG2+1)$.

The Prop_Seg and Phase_Seg1 are merged together and the overall duration is configured via SEG1 field in CANNBTQ or CANDBTQ register.

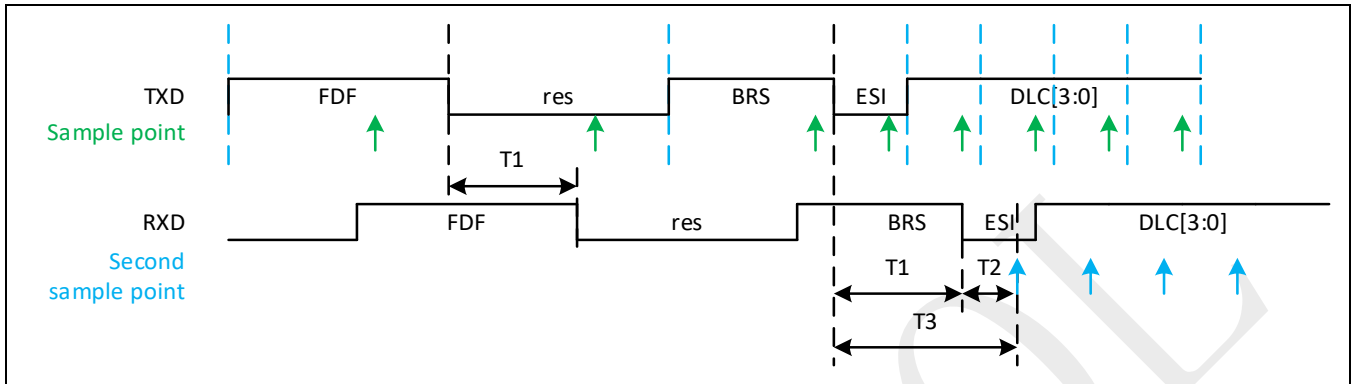
Note: CANNBTQ.NBRP and CANDBTQ.DBRP must be the same. Otherwise, it may lead to transfer exception.

During FDCAN frame transmission with BRS enabled, transmitter delay compensation is supported to generate the second sample point. So it can check whether the sampled data matches the transmitted data. As show in Figure 22-10:

- T1 is the measured delay from TXD sends LOW to RXD detects LOW. It must be longer than the delay window specified via CANTDC.TDCW. Otherwise, the specified delay window will be used as the measured delay.
- T2 is the transmitter delay offset set by CANTDC.TDCO.
- T3 is the overall transmitter delay summed by T1 and T2. It can be read via CANTDC.TDCV.

After the sample point of BRS bit, the CAN node changes from nominal bit rate to data bit rate. The second sample point is generated after the overall transmitter delay for each TXD bit.

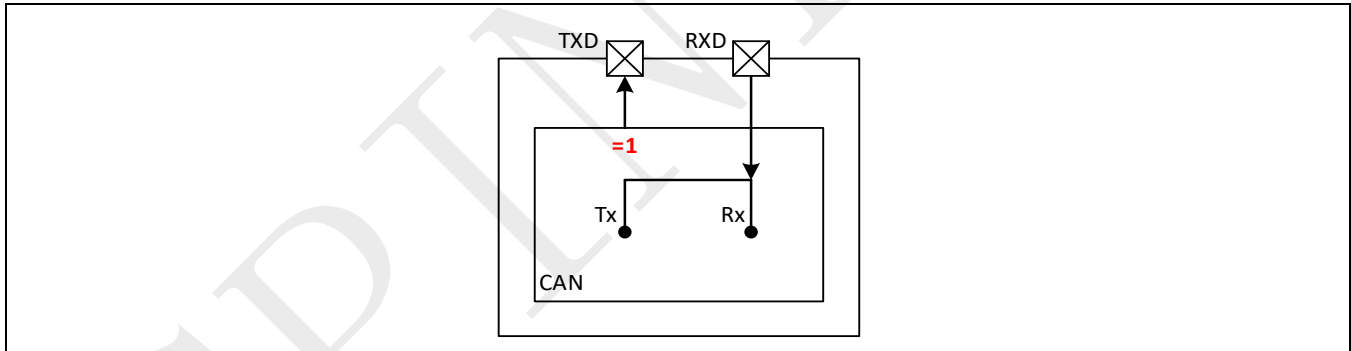
Figure 22-10: Transmitter delay compensation



22.5.2 Monitor mode

In monitor mode (CANCTL.MONMODE bit is set), the CAN node cannot transmit frame but only receives valid data frames or remote frames. The TXD pin keeps recessive. The internal Tx bit is directly looped back to generate the dominant bit necessary for cases such as ACK bit, overload flag, and active error flag.

Figure 22-11: Monitor mode



22.5.3 Restricted mode

In restricted mode (CANCTL.RESMODE bit is set), the CAN node is able to receive data frames, remote frames and acknowledge valid frames. However, it can't send data frames, remote frame, active error frames and overload frames. Upon error event or overload event, the node enters integration state and recovers when idle condition (11 consecutive recessive sample bits) is detected. The TEC and REC are not affected in restricted mode, while the error event detection is still active.

The restricted mode can't be used together with monitor mode or any loopback mode.

22.5.4 Test mode

In test mode (CANCTL.TESTMODE bit is set), user can enable internal loopback mode (CANTEST.LOOP) or external loopback mode (CANTEST.ELOOP). User can also set TXD output to always low, always high

or sample point mode by CANTEST.TXMODE register bits. When CANCTL.TESTMODE bit is cleared, the LOOP, ELOOP and TXMODE in register CANTEST will be reset to default value.

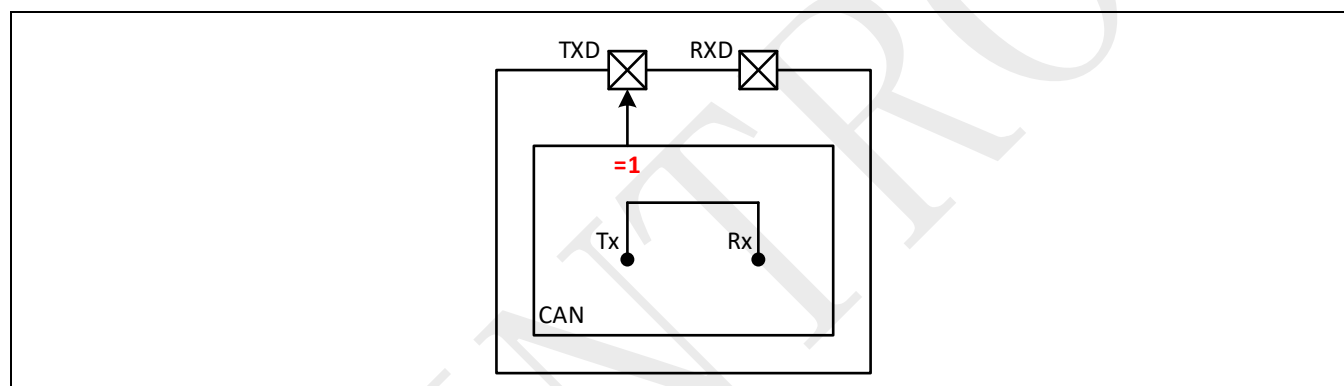
The CAN node can be used as normal in test mode when the LOOP, ELOOP and TXMODE bits in register CANTEST keep default value.

Loopback mode can't be used together with restricted mode or monitor mode. And internal loopback mode and external loopback mode cannot be both enabled.

22.5.4.1 Internal loopback

The connection for internal loopback mode is shown in Figure 22-12. The TXD is always recessive. The internal Rx is disconnected from RXD pin. During frame transmission when ACK bit from other node is required, the internal logic will generate an ACK bit to RX. So that no ACK error will be generated for loopback test.

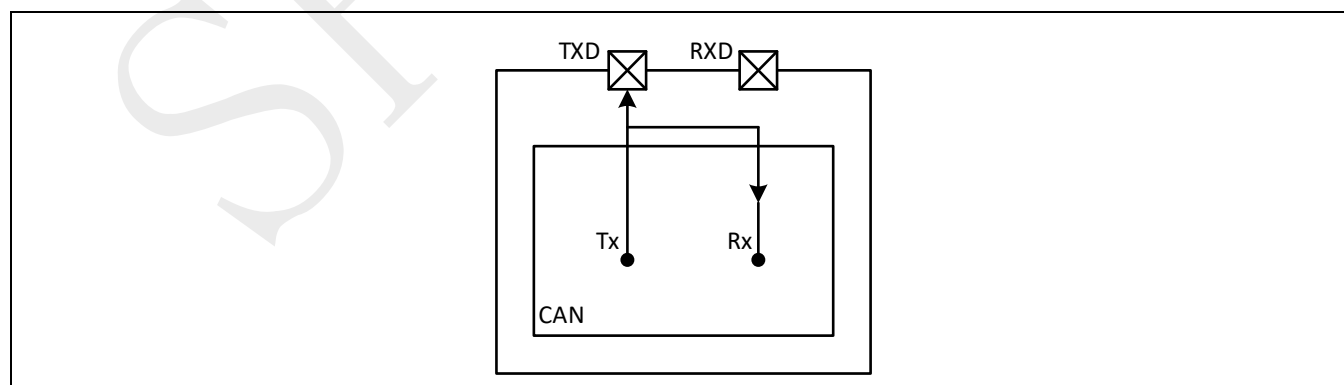
Figure 22-12: Loopback mode



22.5.4.2 External loopback

In external loopback mode, the CAN node behaves similarly as in internal loopback mode. But TXD pin is now the actual transmitted data, and connected to Rx, which is shown in Figure 22-13.

Figure 22-13: External loopback mode



22.5.4.3 Test TXD output

User can set TXD output to always low, always high or sample point mode by register CANTEST.TXMODE. For sample point mode, TXD will output dominant at the sample point and recessive at the drive point.

22.6 Mailbox control

The message RAM must be initialized with value 0 before the module is enabled, so as to avoid malfunction due to the unknown power-on value stored in the RAM.

22.6.1 Mailbox information

Table 22-2: Mailbox control bits

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
0	0	0	0	0	0	0	0	0	0	0

The mailbox control bits are show in [Table 22-2](#):

- EN: Enable current mailbox.
- DIR: Control message data transfer direction.
 - 0: Mailbox is currently used to receive message data.
 - 1: Mailbox is currently used to transmit message data.
- RMTEN: Enable auto response to remote frame or remote frame transmission to request data frame.
 - When DIR is 0, it transmits remote frame to other nodes first, then waits and receives data frame from other node that responses to this remote frame.
 - When DIR is 1, it transmits data frame to response to the received remote frame from other node.
- NEW: New message indicator.
 - When DIR is 0, this bit should be cleared before the mailbox is enabled. It will be set by hardware again after message is successfully received.
Note that if EOB and OVWR are enabled, and the NEW bit is already set, it will be cleared if a new message with the matched identifier is detected. In this case, the LOST bit will be set at the same time by hardware. The NEW bit will be set again by hardware after the whole new message is received.
 - When DIR is 1, this bit should be set before mailbox is enabled:
If auto retransmission is enabled, the NEW bit will be cleared by hardware when message transmission in this mailbox starts.
If auto retransmission is disabled, the NEW bit will be cleared by hardware when message transmission in this mailbox finishes.
- LOST: Message data lost indicator.
 - When DIR is 0 and NEW is already set, it is set by hardware if a new message with matched identifier is detected.
 - When DIR is 1, this bit is useless.
- TXREQ: Frame transmission request.
 - When DIR is 0:
If RMTEN is 0, TXREQ should be cleared before mailbox is enabled.
If RMTEN is 1 and auto retransmission is disabled, TXREQ should be set before mailbox is enabled. It is cleared by hardware when remote frame request transmission from this mailbox starts.
If RMTEN is 1 and auto retransmission is enabled, TXREQ should be set before mailbox is enabled. It is cleared by hardware when remote frame request transmission from this mailbox finishes.

➤ When DIR is 1:

If RMTEN is 0 and auto retransmission is disabled, TXREQ should be set before mailbox is enabled. It is cleared by hardware when frame transmission from this mailbox starts.

If RMTEN is 0 and auto retransmission is enabled, TXREQ should be set before mailbox is enabled. It is cleared by hardware when frame transmission from this mailbox finishes.

If RMTEN is 1 and auto retransmission is disabled, TXREQ should be cleared before mailbox is enabled. It will be set by hardware when matched remote frame request is received, and cleared by hardware when data frame transmission from this mailbox starts.

If RMTEN is 1 and auto retransmission is enabled, TXREQ should be cleared before mailbox is enabled. It will be set by hardware after matched remote frame request is received, and cleared by hardware when data frame transmission from this mailbox finishes.

- IF: Message transfer done interrupt flag.
- IE: Enable message transfer done interrupt.
- EOB: Consecutive mailboxes can be grouped as a block. This bit set current mailbox to be the last mailbox of a block. All mailboxes in one block can receive message frame and may have similar identifier mask filter condition. For message transmission, this bit is useless.
- OVWR: Enable to overwrite the mailbox with the latest received message. It is only used by EOB enabled mailbox.
- LOCK: This bit is set by hardware when message in the mailbox is transferring, and the mailbox is write protected for AHB access. When the transfer is finished, this bit is cleared by hardware and AHB can write the mailbox again.

Note: Message is always saved to the first matched and unused mailbox. User should handle all messages stored in one block first, and then enable message reception for the next block.

22.6.2 Message transmission without remote frame auto response

Table 22-3: Mailbox control bits setting for message transmission

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	1	0	1	0	1	0	0/1	0	0	0

User should follow steps below to transmit message with polling mode.

1. Wait until both NEW and TXREQ are cleared by hardware, which means the mailbox is ready to be used for transmission.
2. Configure the message information via FID, FDF, BRS, ESICTL, ESI, DLC, FDW0 to FDW15 registers. It is recommended to fill the data field beyond the DLC expression range with 0xCC.
3. Configure the mailbox control bits as show in [Table 22-3](#).
4. Wait until both NEW and TXREQ are cleared by hardware, which means message is transmitted successfully.

- It is capable to cancel message transfer before the mailbox truly starts working. User can disable mailbox by clear MCTL.EN bit and must read back this bit to check whether it is cleared successfully. Then this mailbox can be reconfigured for other usage. If MCTL.EN cannot be cleared, message transfer in this mailbox is ongoing and user has to wait until it finishes.

22.6.3 Message transmission with remote frame auto response

Table 22-4: Mailbox control bits setting for message transmission with remote frame response

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	1	1	1	0	0	0	0/1	0	0	0

User should follow steps below to transmit message with polling mode.

- Wait until both NEW and TXREQ are cleared by hardware, which means the mailbox is ready to be used for transmission.
- Configure the remote frame identifier mask filter with FID and FMSK registers. RTR bit mask is ignored by RX handler and received RTR must be 1. The FID is used as reference identifier during received identifier comparison. When the received identifier meets the criteria that $((\text{reference_ID} \& \text{FMSK}) == (\text{received_ID} \& \text{FMSK}))$, the frame can be saved to current mailbox.
- Configure the response message data with ESICTL, FDW0 to FDW15 register. It is recommended to fill the data field beyond the DLC expression range with 0xCC.
- Configure the mailbox control bits as show in [Table 22-4](#).
- When matched remote frame is received, TXREQ will be set by hardware. Value in FID (RTR bit will be not changed), FDF, BRS, ESI, DLC will be changed by the received remote frame. Then this mailbox will transmit data frame to response to the remote node request.
- Wait until both NEW and TXREQ are cleared by hardware, which means message is transmitted successfully.
- It is capable to cancel message transfer before the mailbox truly starts working. Users can disable mailbox by clear MCTL.EN bit and must read back this bit to check whether it is cleared successfully. Then this mailbox can be reconfigured for other usage. If MCTL.EN cannot be cleared, message transfer in this mailbox is ongoing and user has to wait until it finishes.

22.6.4 Message reception without request data from remote node

Table 22-5: Mailbox control bits setting to receive message

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	0	0	0	0	0	0	0/1	0/1	0/1	0

User should follow steps below to receive message with polling mode.

- Configure frame identifier mask filter with FID and FMSK registers. The FID is used as reference identifier during received identifier comparison. When the received identifier meets the criteria that $((\text{reference_ID} \& \text{FMSK}) == (\text{received_ID} \& \text{FMSK}))$, this frame can be saved to current mailbox.

2. Configure the mailbox control bits as show in [Table 22-5](#).
3. Wait until the NEW bit is set by hardware after matched message is received.
4. It is capable to cancel message transfer before the mailbox truly starts working. Users can disable mailbox by clear MCTL.EN bit and must read back this bit to check whether it is cleared successfully. Then this mailbox can be reconfigured for other usage. If MCTL.EN cannot be cleared, message transfer in this mailbox is ongoing and user has to wait until it finishes.

22.6.5 Message reception with request data from remote node

Table 22-6: Mailbox control bits setting to receive message with remote frame response

EN	DIR	RMTEN	NEW	LOST	TXREQ	IF	IE	EOB	OVWR	LOCK
1	0	1	0	0	1	0	0/1	0/1	0/1	0

User should follow steps below to receive message with polling mode.

1. Configure the message information with FDF, BRS, ESICTL, ESI, DLC registers.
2. Configure the frame identifier mask filter with FID and FMSK registers. RTR bit should be cleared and it is ignored by TX handler which always sends RTR as 1. The FID is used as reference identifier during received identifier comparison. When the received identifier meets the criteria that $((\text{reference_ID} \& \text{FMSK}) == (\text{received_ID} \& \text{FMSK}))$, this frame can be saved to current mailbox. FID is also used as remote frame identifier to transmit request to other nodes.
3. Configure the mailbox control bits as show in [Table 22-6](#).
4. TXREQ will be cleared by hardware after remote frame is transmitted. Then this mailbox will wait for frame from remote node.
5. Wait until NEW is set by hardware after matched message from remote node is received.
6. It is capable to cancel message transfer before the mailbox truly starts working. Users can disable mailbox by clear MCTL.EN bit and must read back this bit to check whether it is cleared successfully. Then this mailbox can be reconfigured for other usage. If MCTL.EN cannot be cleared, message transfer in this mailbox is ongoing and user has to wait until it finishes.

22.7 Registers

22.7.1 CAN register map

Table 22-7: CAN Module Base Address

Peripheral Module	Base Address
CAN	0x40005400

Table 22-8: CAN Register Map

Register	Offset	Description	Reset Value
<u>CANCTL</u> *	0x00	CAN Control Register	0x00000060
<u>CANNBTQ</u> *	0x04	CAN Nominal Bit Timing Quantum Register	0x00090403
<u>CANDBTQ</u> *	0x08	CAN Data Bit Timing Quantum Register	0x00090403
<u>CANTDC</u> *	0x0C	CAN Transmitter Delay Compensation Register	0x00000000
<u>CANBRDLY</u> *	0x10	CAN Bus-Off Recovery Delay Register	0x00000000
<u>CANSTS</u>	0x14	CAN Status Register	0x00000000
<u>CANLEC</u>	0x18	CAN Last Error Code Register	0x00000000
<u>CANMIE</u> *	0x1C	CAN Miscellaneous Interrupt Enable Register	0x00000000
<u>CANMRAWIF</u>	0x20	CAN Miscellaneous Interrupt Raw Flag Register	0x00000000
<u>CANMIF</u>	0x24	CAN Miscellaneous Interrupt Flag Register	0x00000000
<u>CANMIC</u>	0x28	CAN Miscellaneous Interrupt Flag Clear Register	0x00000000
<u>CANTGIF</u>	0x2C	CAN Mailbox Group Transfer Done Interrupt Flag Register	0x00000000
<u>CANTGIC</u>	0x30	CAN Mailbox Group Transfer Done Interrupt Clear Register	0x00000000
<u>CANMEMERRADDR</u>	0x34	CAN Memory Error Address Register	0x00000000
<u>CANTEST</u> *	0x38	CAN Test Control Register	0x00000030
<u>CANECCTL</u> *	0x3C	CAN Error Counter Control Register	0x0000307F
<u>CANTXREQ</u>	0x40	CAN Transmission Request Register	0x00000000
<u>CANMSGNEW</u>	0x50	CAN New Message Register	0x00000000
<u>CANMSGIF</u>	0x60	CAN Mailbox Transfer Done Interrupt Flag Register	0x00000000
<u>CANMBEN</u>	0x70	CAN Mailbox Enable Register	0x00000000
<u>CANTSCTL</u> *	0x80	CAN Timestamp Control Register	0x00000000
<u>CANTSCNT</u> *	0x84	CAN Timestamp Counter Register	0x00000000
<u>CANREGKEY</u>	0x8C	CAN Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the CANREGKEY=0x1ACCE551.

22.7.2 CAN registers

Table 22-9: CAN Control Register (CANCTL) Layout

CANCTL (CAN Control Register) Offset: 0x00 Default: 0x00000060 Access: CAN->CANCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							ECCEN
15	14	13	12	11	10	9	8
NONISO	FDEN	TSTMODE	MONMODE	RESMODE	EDGEFLT	TXPAUSE	PXEN
7	6	5	4	3	2	1	0
IOSWAP	RETX	AUTOON	RECOVEN	RST	RESERVED		EN

Table 22-10: CAN Control Register (CANCTL) Description

Bits	Field Name	Type	Reset	Description
31:17	RESERVED_31_17	RO	0x0	Reserved.
16	ECCEN	RW	0x0	ECC enable. 0: Disable 1: Enable
15	NONISO	RW	0x0	NON-ISO CAN protocol mode. This bit is writable only when CANSTS.OP is STOP. 0: Use ISO-FDCAN format defined by ISO11898-1:2015. 1: Use NONISO-FDCAN format defined by Bosch CAN2.0B and FDCAN_V1.0
14	FDEN	RW	0x0	FDCAN enable. This bit is writable only when CANSTS.OP is STOP. 0: Disable 1: Enable
13	TSTMODE	RW	0x0	Test mode enable This bit is writable only when CANSTS.OP is STOP. 0: Disable 1: Enable
12	MONMODE	RW	0x0	Monitor mode enable This bit can be set only when CANSTS.OP is STOP. This bit can be cleared at any time. 0: Disable 1: Enable
11	RESMODE	RW	0x0	Restricted mode enable This bit can be set only when CANSTS.OP is STOP. This bit can be cleared at any time. 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
10	EDGEFLT	RW	0x0	Edge filter enable in integration state. When it is enabled, the falling edge should be 2 consecutive dominant bits after the recessive bit. 0: Disable 1: Enable
9	TXPAUSE	RW	0x0	Enable 2 nominal bit insertion between consecutive frame transmission 0: Disable 1: Enable
8	PXEN	RW	0x0	Protocol exception enable. This bit is writable only when CANSTS.OP is STOP. 0: Disable 1: Enable
7	IOSWAP	RW	0x0	Enable swap function for TXD/RXD Pin. This bit is writable only when CANSTS.OP is STOP. 0: Disable 1: Enable
6	RETX	RW	0x1	Enable automatic retransmission of "not successful" messages. This bit is writable only when CANSTS.OP is STOP. 0: Disable 1: Enable
5	AUTOON	RW	0x1	Enable auto bus on, so Bus-Off recovery sequence can be started automatically after the delay specified by CANBRDLY. This feature is not supported in debug mode. 0: Disable 1: Enable
4	RECOVEN	W1C	0x0	Enable Bus-Off recovery. This bit is NOT written protected by CANREGKEY. 0: Write 0 has no effect and always reads back 0 1: Write 1 starts Bus-Off recovery sequence immediately. This bit is self-cleared.
3	RST	W1C	0x0	Software reset. User should stop the CAN first by clearing CANCTL.RUN before issuing the software reset. Actual reset would not take place until the CAN is in idle state. 0: Write 0 has no effect. 1: Write a 1 forces the module to reset. This bit is self-cleared when the reset is finished.
2:1	RESERVED_2_1	RO	0x0	Reserved.
0	EN	RW	0x0	CAN module enable 0: Disable 1: Enable

Table 22-11: CAN Nominal Bit Timing Quantum Register (CANNBTQ) Layout

CANNBTQ (CAN Nominal Bit Timing Quantum Register) Offset: 0x04 Default: 0x00090403 Access: CAN->CANNBTQ							
31 RESERVED	30	29	28	27 NSJW	26	25	24
23 RESERVED	22	21	20	19 NSEG2	18	17	16
15	14	13	12	11 NSEG1	10	9	8
7	6	5	4	3 NBRP	2	1	0

Table 22-12: CAN Nominal Bit Timing Quantum Register (CANNBTQ) Description

Bits	Field Name	Type	Reset	Description
31	RESERVED_31	RO	0x0	Reserved.
30:24	NSJW	RW	0x0	Synchronization jumps width for nominal bit rate, which would be (NSJW+1) NTq. NSJW should be within [0, 127] This field is writable only when CANSTS.OP is STOP.
23	RESERVED_23	RO	0x0	Reserved.
22:16	NSEG2	RW	0x9	Time segment after the sample point for nominal bit rate, which would be (NSEG2+1) NTq. NSEG2 should be within [1, 127] This field is writable only when CANSTS.OP is STOP.
15:8	NSEG1	RW	0x4	Time segment before the sample point for nominal bit rate, which would be (NSEG1+1) NTq. NSEG1 should be within [1, 255] This field is writable only when CANSTS.OP is STOP.
7:0	NBRP	RW	0x3	Nominal bit rate prescaler value. Arbitration phase timing quantum NTq would be (NBRP+1) CAN module clocks. This field is writable only when CANSTS.OP is STOP.

Table 22-13: CAN Data Bit Timing Quantum Register (CANDBTQ) Layout

CANDBTQ (CAN Data Bit Timing Quantum Register) Offset: 0x08 Default: 0x00090403 Access: CAN->CANDBTQ							
31 TDC	30	29	28	27	26	25	24
	RESERVED			DSJW			
23	22	21	20	19	18	17	16
	RESERVED			DSEG2			
15	14	13	12	11	10	9	8
	RESERVED		DSEG1				
7	6	5	4	3	2	1	0
DBRP							

Table 22-14: CAN Data Bit Timing Quantum Register (CANDBTQ) Description

Bits	Field Name	Type	Reset	Description
31	TDC	RW	0x0	Transmitter delay compensation enable. This field is writable only when CANSTS.OP is STOP. 0: Disable 1: Enable
30:28	RESERVED_30_28	RO	0x0	Reserved.
27:24	DSJW	RW	0x0	Synchronization jumps width for data bit rate, which would be (DSJW+1) DTq. DSJW should be within [0, 15]. This field is writable only when CANSTS.OP is STOP.
23:20	RESERVED_23_20	RO	0x0	Reserved.
19:16	DSEG2	RW	0x9	Time segment after the sample point for data bit rate, which would be (DSEG2+1) DTq. DSEG2 should be within [1, 15]. This field is writable only when CANSTS.OP is STOP.
15:13	RESERVED_15_13	RO	0x0	Reserved.
12:8	DSEG1	RW	0x4	Time segment before the sample point for data bit rate, which would be (DSEG1+1) DTq. DSEG1 should be within [0, 31]. This field is writable only when CANSTS.OP is STOP.
7:0	DBRP	RW	0x3	Data bit rate prescaler value. Data phase timing quantum DTq would be (DBRP+1) CAN module clocks. This field is writable only when CANSTS.OP is STOP. NOTE: It should be the same as NBRP.

Table 22-15: CAN Transmitter Delay Compensation Register (CANTDC) Layout

CANTDC (CAN Transmitter Delay Compensation Register) Offset: 0x0C Default: 0x00000000 Access: CAN->CANTDC							
31	30	29	28	27	26	25	24
RESERVED		TDCV					
23	22	21	20	19	18	17	16
TDCV				TDCW			
15	14	13	12	11	10	9	8
TDCW						TDCO	
7	6	5	4	3	2	1	0
TDCO							

Table 22-16: CAN Transmitter Delay Compensation Register (CANTDC) Description

Bits	Field Name	Type	Reset	Description
31:30	RESERVED_31_30	RO	0x0	Reserved.
29:20	TDCV	RO	0x0	Transmitter delay compensation value in DTq. It is the sum of TDCO and measured delay time.
19:10	TDCW	RW	0x0	Transmitter delay compensation filter window size, which is TDCW DTq.
9:0	TDCO	RW	0x0	Transmitter delay compensation second sample point offset, which is TDCO DTq.

Table 22-17: CAN Bus-Off Recovery Delay Register (CANBRDLY) Layout

CANBRDLY (CAN Bus-Off Recovery Delay Register) Offset: 0x10 Default: 0x00000000 Access: CAN->CANBRDLY							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
DLY							
7	6	5	4	3	2	1	0
DLY							

Table 22-18: CAN Bus-Off Recovery Delay Register (CANBRDLY) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	DLY	RW	0x0	Bus-Off recovery delay. When CANCTL.AUTOON=1, recovery sequence would be started automatically after (DLY+1) nominal bit time since Bus-Off.

Table 22-19: CAN Status Register (CANSTS) Layout

CANSTS (CAN Status Register) Offset: 0x14 Default: 0x00000000 Access: CAN->CANSTS							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				OPMODE			ERRWARN
15	14	13	12	11	10	9	8
ERRPASSIVE	REC						
7	6	5	4	3	2	1	0
TEC							

Table 22-20: CAN Status Register (CANSTS) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19:17	OPMODE	RO	0x0	Operation mode. 000: Stop state 001: Integration state 010: Idle state 011: Busy transmitting message 100: Busy receiving message 110: Bus-Off
16	ERRWARN	RO	0x0	Error warning. 0: Both TEC and REC are below CANECCTL.ERRWARN 1: Either TEC or REC reaches CANECCTL.ERRWARN
15	ERRPASSIVE	RO	0x0	Error passive state. 0: CAN would send active error frames on bus error. 1: CAN would send passive error frames on bus error.
14:8	REC	RO	0x0	Receiver error count. It is within [0, 127] NOTE: It would decrease by 1 each time 11 consecutive recessive bits are detected during Bus-Off recovery.
7:0	TEC	RO	0x0	Transmitter error count. It is within [0, 255]

Table 22-21: CAN Last Error Code Register (CANLEC) Layout

CANLEC (CAN Last Error Code Register) Offset: 0x18 Default: 0x00000000 Access: CAN->CANLEC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							WRLCK
15	14	13	12	11	10	9	8
RESERVED					DLEC		
7	6	5	4	3	2	1	0
RESERVED					LEC		

Table 22-22: CAN Last Error Code Register (CANLEC) Description

Bits	Field Name	Type	Reset	Description
31:17	RESERVED_31_17	RO	0x0	Reserved.
16	WRLCK	RC	0x0	Write on locked mailbox (busy transferring message) occurred. The write access would be ignored. It is recommended to read this field in byte. 0: Not occurred. 1: Occurred.
15:11	RESERVED_15_11	RO	0x0	Reserved.
10:8	DLEC	RS	0x0	Last error code during data phase. It is recommended to read this field in byte. 000: No error. 001: Stuff error. 010: Form error. 100: Transmit 1 but readback 0 101: Transmit 0 but readback 1 111: No error after last read.
7:3	RESERVED_7_3	RO	0x0	Reserved.
2:0	LEC	RS	0x0	Last error code during arbitration phase. It is recommended to read this field in byte. 000: This field will be cleared to '0' when a message has been transferred (reception or transmission) without error. 001: More than five equal bits in a row have been detected in a part of a received message where this is not allowed. 010: A fixed format part of a received frame has the wrong format. 011: The message transmitted was not acknowledged by another node. 100: Transmit 1 but readback 0

Bits	Field Name	Type	Reset	Description
				101: Transmit 0 but readback 1 Note: During Bus-Off recovery, this status is set each time 11 recessive bits are detected. This enables the CPU to monitor the proceeding of the Bus-Off recovery sequence (indicating the bus is not stuck at dominant or continuously disturbed). 110: In a received message, the CRC field does not match the calculated value based on the received data. 111: This field will be reset to '7' whenever the CPU reads the register.

Table 22-23: CAN Miscellaneous Interrupt Enable Register (CANMIE) Layout

CANMIE (CAN Miscellaneous Interrupt Enable Register) Offset: 0x1C Default: 0x00000000 Access: CAN->CANMIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						MSGLOST	TSOV
7	6	5	4	3	2	1	0
MEMMBE	MEMSBE	PXEVT	FLTOK	BUSOFF	ERRPASSIVE	ERRWARN	TRANERR

Table 22-24: CAN Miscellaneous Interrupt Enable Register (CANMIE) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	MSGLOST	RW	0x0	Message lost interrupt enable 0: Disable 1: Enable
8	TSOV	RW	0x0	Timestamp counter overflow interrupt enable 0: Disable 1: Enable
7	MEMMBE	RW	0x0	Memory multi bit error interrupt enable 0: Disable 1: Enable
6	MEMSBE	RW	0x0	Memory single bit error interrupt enable 0: Disable 1: Enable
5	PXEVT	RW	0x0	Protocol exception event interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
4	FLTOK	RW	0x0	Filtered received message identifier match interrupt enable 0: Disable 1: Enable
3	BUSOFF	RW	0x0	Bus-Off state entry interrupt enable 0: Disable 1: Enable
2	ERRPASSIVE	RW	0x0	Passive state entry interrupt enable 0: Disable 1: Enable
1	ERRWARN	RW	0x0	Error warning interrupt enable 0: Disable 1: Enable
0	TRANERR	RW	0x0	Message transfer error interrupt enable 0: Disable 1: Enable

Table 22-25: CAN Miscellaneous Interrupt Raw Flag Register (CANMRAWIF) Layout

CANMRAWIF (CAN Miscellaneous Interrupt Raw Flag Register) Offset: 0x20 Default: 0x00000000 Access: CAN->CANMRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						MSGLOST	TSOV
7	6	5	4	3	2	1	0
MEMMBE	MEMSB	PXEVT	FLTOK	BUSOFF	ERRPASSIVE	ERRWARN	TRANERR

Table 22-26: CAN Miscellaneous Interrupt Raw Flag Register (CANMRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	MSGLOST	RO	0x0	Latched message lost event flag 0: Event does not occur 1: Event occurred
8	TSOV	RO	0x0	Latched timestamp counter overflow event flag 0: Event does not occur 1: Event occurred
7	MEMMBE	RO	0x0	Latched memory multi bit error event flag 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
6	MEMSBE	RO	0x0	Latched memory single bit error event flag 0: Event does not occur 1: Event occurred
5	PXEVT	RO	0x0	Latched protocol exception event flag 0: Event does not occur 1: Event occurred
4	FLTOK	RO	0x0	Latched filtered received message identifier match event flag 0: Event does not occur 1: Event occurred
3	BUSOFF	RO	0x0	Latched Bus-Off state entry event flag 0: Event does not occur 1: Event occurred
2	ERRPASSIVE	RO	0x0	Latched Passive state entry event flag 0: Event does not occur 1: Event occurred
1	ERRWARN	RO	0x0	Latched error warning event flag 0: Event does not occur 1: Event occurred
0	TRANERR	RO	0x0	Latched message transfer error event flag 0: Event does not occur 1: Event occurred

Table 22-27: CAN Miscellaneous Interrupt Flag Register (CANMIF) Layout

CANMIF (CAN Miscellaneous Interrupt Flag Register) Offset: 0x24 Default: 0x00000000 Access: CAN->CANMIF							
31 INT	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9 MSGLOST	8 TSOV
RESERVED							
7 MEMMBE	6 MEMSBE	5 PXEVT	4 FLTOK	3 BUSOFF	2 ERRPASSIVE	1 ERRWARN	0 TRANERR

Table 22-28: CAN Miscellaneous Interrupt Flag Register (CANMIF) Description

Bits	Field Name	Type	Reset	Description
31	INT	RO	0x0	Global interrupt 0: Miscellaneous interrupt does not occur 1: Miscellaneous interrupt occurred and issued to the CPU. Further interrupt will not be generated until this bit is cleared.
30:10	RESERVED_30_10	RO	0x0	Reserved.
9	MSGLOST	RO	0x0	Latched message lost interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
8	TSOV	RO	0x0	Latched timestamp counter overflow interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
7	MEMMBE	RO	0x0	Latched memory multi bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
6	MEMSBE	RO	0x0	Latched memory single bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
5	PXEVT	RO	0x0	Latched protocol exception interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
4	FLTOK	RO	0x0	Latched filtered received message identifier match interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
3	BUSOFF	RO	0x0	Latched Bus-Off state entry interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
2	ERRPASSIVE	RO	0x0	Latched Passive state entry interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
1	ERRWARN	RO	0x0	Latched error warning interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Bits	Field Name	Type	Reset	Description
0	TRANERR	RO	0x0	Latched message transfer error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Table 22-29: CAN Miscellaneous Interrupt Flag Clear Register (CANMIC) Layout

CANMIC (CAN Miscellaneous Interrupt Flag Clear Register) Offset: 0x28 Default: 0x00000000 Access: CAN->CANMIC							
31 INT	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9 MSGLOST	8 TSOV
RESERVED							
7 MEMMBE	6 MEMSBE	5 PXEVT	4 FLTOK	3 BUSOFF	2 ERRPASSIVE	1 ERRWARN	0 TRANERR

Table 22-30: CAN Miscellaneous Interrupt Flag Clear Register (CANMIC) Description

Bits	Field Name	Type	Reset	Description
31	INT	W1C	0x0	Global interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
30:10	RESERVED_30_10	RO	0x0	Reserved.
9	MSGLOST	W1C	0x0	Message lost interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
8	TSOV	W1C	0x0	Timestamp counter overflow interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
7	MEMMBE	W1C	0x0	Memory multi bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	MEMSBE	W1C	0x0	Memory single bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	PXEVT	W1C	0x0	Protocol exception event interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
4	FLTOK	W1C	0x0	Filtered received message identifier match interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
3	BUSOFF	W1C	0x0	Bus-Off state entry interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	ERRPASSIVE	W1C	0x0	Passive state entry interrupt enable 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	ERRWARN	W1C	0x0	Error warning interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	TRANERR	W1C	0x0	Message transfer error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 22-31: CAN Mailbox Group Transfer Done Interrupt Flag Register (CANTGIF) Layout

CANTGIF (CAN Mailbox Group Transfer Done Interrupt Flag Register)					Offset: 0x2C	Default: 0x00000000									
Access: CAN->CANTGIF															
31		30		29		28		27		26		25		24	
RESERVED															
23		22		21		20		19		18		17		16	
RESERVED															
15		14		13		12		11		10		9		8	
RESERVED															
7		6		5		4		3		2		1		0	
GRP7		GRP6		GRP5		GRP4		GRP3		GRP2		GRP1		GRP0	

Table 22-32: CAN Mailbox Group Transfer Done Interrupt Flag Register (CANTGIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	GRP7	RO	0x0	Mailbox group 7 transfer done interrupt flag. It is set when any one in mailbox 56~63 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
6	GRP6	RO	0x0	Mailbox group 6 transfer done interrupt flag. It is set when any one in mailbox 48~55 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Bits	Field Name	Type	Reset	Description
5	GRP5	RO	0x0	Mailbox group 5 transfer done interrupt flag. It is set when any one in mailbox 40~47 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
4	GRP4	RO	0x0	Mailbox group 4 transfer done interrupt flag. It is set when any one in mailbox 32~39 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
3	GRP3	RO	0x0	Mailbox group 3 transfer done interrupt flag. It is set when any one in mailbox 24~31 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
2	GRP2	RO	0x0	Mailbox group 2 transfer done interrupt flag. It is set when any one in mailbox 16~23 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
1	GRP1	RO	0x0	Mailbox group 1 transfer done interrupt flag. It is set when any one in mailbox 8~15 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
0	GRP0	RO	0x0	Mailbox group 0 transfer done interrupt flag. It is set when any one in mailbox 0~7 finishes transfer. 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Table 22-33: CAN Mailbox Group Transfer Done Interrupt Clear Register (CANTGIC) Layout

CANTGIC (CAN Mailbox Group Transfer Done Interrupt Clear Register)					Offset: 0x30	Default: 0x00000000	
Access: CAN->CANTGIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
GRP7	GRP6	GRP5	GRP4	GRP3	GRP2	GRP1	GRP0

Table 22-34: CAN Mailbox Group Transfer Done Interrupt Clear Register (CANTGIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	GRP7	W1C	0x0	Mailbox group 7 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	GRP6	W1C	0x0	Mailbox group 6 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	GRP5	W1C	0x0	Mailbox group 5 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
4	GRP4	W1C	0x0	Mailbox group 4 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	GRP3	W1C	0x0	Mailbox group 3 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	GRP2	W1C	0x0	Mailbox group 2 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	GRP1	W1C	0x0	Mailbox group 1 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	GRP0	W1C	0x0	Mailbox group 0 transfer done interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 22-35: CAN Memory Error Address Register (CANMEMERRADDR) Layout

CANMEMERRADDR (CAN Memory Error Address Register) Offset: 0x34 Default: 0x00000000 Access: CAN->CANMEMERRADDR							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

Table 22-36: CAN Memory Error Address Register (CANMEMERRADDR) Description

Bits	Field Name	Type	Reset	Description
31:0	ADDR	RO	0x0	Memory error address

Table 22-37: CAN Test Control Register (CANTEST) Layout

CANTEST (CAN Test Control Register) Offset: 0x38 Default: 0x00000030 Access: CAN->CANTEST							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	RESERVED	RXD	TXD	TXMODE		ELOOP	LOOP

Table 22-38: CAN Test Control Register (CANTEST) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6	RESERVED_6	RW	0x0	Reserved.
5	RXD	RO	0x1	RXD pin monitored level 0: Low level 1: High level
4	TXD	RO	0x1	TXD pin monitored level 0: Low level 1: High level
3:2	TXMODE	RW	0x0	Control of TXD Pin. Disable CANCTL.TSTMODE will reset this field. 00: Drive TXD pin by CAN protocol engine 01: Drive TXD pin dominant at sample point and

Bits	Field Name	Type	Reset	Description
				recessive at drive point. Sample point can be monitored on TXD pin in this way. 10: Drive TXD pin as dominant 11: Drive TXD pin as recessive
1	ELOOP	RW	0x0	Enable external loop back mode. Disable CANCTL.TSTMODE will reset this field. 0: Disable 1: Enable
0	LOOP	RW	0x0	Enable internal loop back mode. Disable CANCTL.TSTMODE will reset this field. 0: Disable 1: Enable

Table 22-39: CAN Error Counter Control Register (CANECCTL) Layout

CANECCTL (CAN Error Counter Control Register) Offset: 0x3C Default: 0x0000307F Access: CAN->CANECCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		ERRWARN					
7	6	5	4	3	2	1	0
ERRWARN	RECLD						

Table 22-40: CAN Error Counter Control Register (CANECCTL) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13:7	ERRWARN	RW	0x60	Error warning threshold. It should be within [80, 127]
6:0	RECLD	RW	0x7F	REC load value after successful frame reception when REC is greater than 127. It should be within [119, 127]

Table 22-41: CAN Transmission Request Register (CANTXREQ) Layout

CANTXREQ (CAN Transmission Request Register) Offset: 0x40 Default: 0x00000000 Access: CAN->CANTXREQ							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 22-42: CAN Transmission Request Register (CANTXREQ) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Transmission request for the corresponding mailbox (0 ~ 63) 0: No transmission request for this mailbox 1: Transmission request is pending or processing for this mailbox.

Table 22-43: CAN New Message Register (CANMSGNEW) Layout

CANMSGNEW (CAN New Message Register) Offset: 0x50 Default: 0x00000000 Access: CAN->CANMSGNEW							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 22-44: CAN New Message Register (CANMSGNEW) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	New message indicator for the corresponding mailbox (0 ~ 63) 0: No new message is stored into this mailbox 1: There is new message stored in this mailbox

Table 22-45: CAN Mailbox Transfer Done Interrupt Flag Register (CANMSGIF) Layout

CANMSGIF (CAN Mailbox Transfer Done Interrupt Flag Register) Offset: 0x60 Default: 0x00000000 Access: CAN->CANMSGIF							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 22-46: CAN Mailbox Transfer Done Interrupt Flag Register (CANMSGIF) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Transfer done flag for the corresponding mailbox (0 ~ 63) 0: No transfer done event occurred for this mailbox 1: Transfer done event occurred for this mailbox

Table 22-47: CAN Mailbox Enable Register (CANMBEN) Layout

CANMBEN (CAN Mailbox Enable Register) Offset: 0x70 Default: 0x00000000 Access: CAN->CANMBEN							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 22-48: CAN Mailbox Enable Register (CANMBEN) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	Enable control for the corresponding mailbox (0 ~ 63) 0: Mailbox is not used for message transfer. 1: Mailbox is enabled for message transfer.

Table 22-49: CAN Timestamp Control Register (CANTSCTL) Layout

CANTSCTL (CAN Timestamp Control Register) Offset: 0x80 Default: 0x00000000 Access: CAN->CANTSCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							TSEN

Table 22-50: CAN Timestamp Control Register (CANTSCTL) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	TSEN	RW	0x0	Timestamp enable 0: Disable 1: Enable

Table 22-51: CAN Timestamp Counter Register (CANTSCNT) Layout

CANTSCNT (CAN Timestamp Counter Register) Offset: 0x84 Default: 0x00000000 Access: CAN->CANTSCNT							
31	30	29	28	27	26	25	24
TSCNT							
23	22	21	20	19	18	17	16
TSCNT							
15	14	13	12	11	10	9	8
TSCNT							
7	6	5	4	3	2	1	0
TSCNT							

Table 22-52: CAN Timestamp Counter Register (CANTSCNT) Description

Bits	Field Name	Type	Reset	Description
31:0	TSCNT	RW	0x0	Timestamp value of sample point based up-counter. The counter is set to the given value upon write and continues up-counting on sample point.

Table 22-53: CAN Register Write-Allow Key Register (CANREGKEY) Layout

CANREGKEY (CAN Register Write-Allow Key Register) Offset: 0x8C Default: 0x1ACCE551 Access: CAN->CANREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 22-54: CAN Register Write-Allow Key Register (CANREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected registers.

22.7.3 CANMBOX register map

Table 22-55: CANMBOX Module Base Address

Peripheral Module	Base Address
CAN->CANCANMBOX	0x40004000

Table 22-56: CANMBOX Register Map

Register	Offset	Description	Reset Value
CANMBOXFDW	0x00	CAN Mailbox Data Word n ([0:15]) Register	0x00000000
CANMBOXFMSK	0x40	CAN Mailbox Frame Mask Register	0x00000000
CANMBOXFID	0x44	CAN Mailbox Frame Identifier Register	0x00000000
CANMBOXMTS	0x48	CAN Mailbox Message Timestamp Register	0x00000000
CANMBOXMCTL	0x4C	CAN Mailbox Message Control Register	0x00000000

22.7.4 CANMBOX registers

Table 22-57: CAN Mailbox Data Word Register n (CANMBOXFDW) Layout

CANMBOXFDW (CAN Mailbox Data Word Register n) Offset: 0x00 Default: 0x00000000 Access: CAN->CANMBOX[m].CANMBOXFDW[n] (m is from 0 to 63, n is from 0 to 15)							
31	30	29	28	27	26	25	24
BYTE3							
23	22	21	20	19	18	17	16
BYTE2							
15	14	13	12	11	10	9	8
BYTE1							
7	6	5	4	3	2	1	0
BYTE0							

Table 22-58: CAN Mailbox Data Word Register n (CANMBOXFDW) Description

Bits	Field Name	Type	Reset	Description
31:24	BYTE3	RW	0x0	Byte (4 * N + 3) of CAN frame, N is CANMBOXFDW index. It is recommended to padding with 0xCC for unused byte.
23:16	BYTE2	RW	0x0	Byte (4 * N + 2) of CAN frame, N is CANMBOXFDW index. It is recommended to padding with 0xCC for unused byte.
15:8	BYTE1	RW	0x0	Byte (4 * N + 1) of CAN frame, N is CANMBOXFDW index. It is recommended to padding with 0xCC for unused byte.
7:0	BYTE0	RW	0x0	Byte (4 * N + 0) of CAN frame, N is CANMBOXFDW index. It is recommended to padding with 0xCC for unused byte.

Table 22-59: CAN Mailbox Frame Mask Register (CANMBOXFMSK) Layout

CANMBOXFMSK (CAN Mailbox Frame Mask Register) Offset: 0x40 Default: 0x00000000 Access: CAN->CANMBOX[m].CANMBOXFMSK (m is from 0 to 63)							
31 RESERVED	30 RTR	29 IDE	28	27	26 ID	25	24
23	22	21	20	19	18	17	16
ID							
15	14	13	12	11	10	9	8
ID							
7	6	5	4	3	2	1	0
ID							

Table 22-60: CAN Mailbox Frame Mask Register (CANMBOXFMSK) Description

Bits	Field Name	Type	Reset	Description
31	RESERVED_31	RO	0x0	Reserved.
30	RTR	RW	0x0	Remote transmission request bit mask. 0: This bit in the message identifier is not compared for identifier filtering 1: This bit in the message identifier is compared for identifier filtering
29	IDE	RW	0x0	Identifier extension bit mask. 0: This bit in the message identifier is not compared for identifier filtering 1: This bit in the message identifier is compared for identifier filtering
28:0	ID	RW	0x0	Identifier mask, which is ID[28:18] if received IDE is 0, or ID[28:0] if received IDE is 1. 0: This bit in the message identifier is not compared for identifier filtering 1: This bit in the message identifier is compared for identifier filtering

Table 22-61: CAN Mailbox Frame Identifier Register (CANMBOXFID) Layout

CANMBOXFID (CAN Mailbox Frame Identifier Register) Offset: 0x44 Default: 0x00000000 Access: CAN->CANMBOX[m].CANMBOXFID (m is from 0 to 63)							
31 RESERVED	30 RTR	29 IDE	28	27	26 ID	25	24
23	22	21	20	19	18	17	16
ID							
15	14	13	12	11	10	9	8
ID							
7	6	5	4	3	2	1	0
ID							

Table 22-62: CAN Mailbox Frame Identifier Register (CANMBOXFID) Description

Bits	Field Name	Type	Reset	Description
31	RESERVED_31	RO	0x0	Reserved.
30	RTR	RW	0x0	Remote transmission request bit. 0: Message is data frame. 1: Message is remote frame.
29	IDE	RW	0x0	Identifier extension bit. 0: Message is standard CAN frame. 1: Message is extended CAN frame.
28:0	ID	RW	0x0	Identifier, which is ID[28:18] if IDE is 0, or ID[28:0] if IDE is 1.

Table 22-63: CAN Mailbox Message Timestamp Register (CANMBOXMTS) Layout

CANMBOXMTS (CAN Mailbox Message Timestamp Register) Offset: 0x48 Default: 0x00000000 Access: CAN->CANMBOX[m].CANMBOXMTS (m is from 0 to 63)							
31	30	29	28	27	26	25	24
TS							
23	22	21	20	19	18	17	16
TS							
15	14	13	12	11	10	9	8
TS							
7	6	5	4	3	2	1	0
TS							

Table 22-64: CAN Mailbox Message Timestamp Register (CANMBOXMTS) Description

Bits	Field Name	Type	Reset	Description
31:0	TS	RW	0x0	Timestamp User should write this register only for initialization, and read back the valid value updated by hardware after transfer is finished.

Table 22-65: CAN Mailbox Message Control Register (CANMBOXMCTL) Layout

CANMBOXMCTL (CAN Mailbox Message Control Register) Offset: 0x4C Default: 0x00000000 Access: CAN->CANMBOX[m].CANMBOXMCTL (m is from 0 to 63)							
31 EN	30 DIR	29 RMTEN	28 NEW	27 LOST	26 TXREQ	25 IF	24 IE
23 EOB	22 OVWR	21 LOCK	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 ESICTL	6 FDF	5 BRS	4 ESI	3	2	1	0
DLC							

Table 22-66: CAN Mailbox Message Control Register (CANMBOXMCTL) Description

Bits	Field Name	Type	Reset	Description
31	EN	RW	0x0	Mailbox enable 0: Disable 1: Enable
30	DIR	RW	0x0	Message data transfer direction on bus. 0: Receive 1: Transmit
29	RMTEN	RW	0x0	Remote frame auto response enable 0: Disable 1: Enable
28	NEW	RW	0x0	New message indicator 0: Old message 1: New message
27	LOST	RW	0x0	Received message lost. User should NOT write 1 to this bit 0: Not occurred. 1: Occurred.
26	TXREQ	RW	0x0	Transmit request. 0: Disable 1: Enable
25	IF	RW	0x0	Interrupt pending flag. User should NOT write 1 to this bit 0: Not occurred. 1: Occurred.
24	IE	RW	0x0	Mailbox transfer done interrupt enable 0: Disable 1: Enable
23	EOB	RW	0x0	End of block. 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
22	OVWR	RW	0x0	Enable overwrite mailbox which is end of block. 0: Disable 1: Enable
21	LOCK	RW	0x0	Mailbox locked flag. User should NOT write 1 to this bit 0: Not occurred. 1: Occurred.
20:8	RESERVED_20_8	RW	0x0	Reserved.
7	ESICTL	RW	0x0	Error state indicator software overwrite enable 0: ESI is driven by CAN protocol engine 1: ESI is driven by CANMBOXMCTL.ESI
6	FDF	RW	0x0	FD format indicator. 0: Disable 1: Enable
5	BRS	RW	0x0	Bit rate switch. 0: Disable 1: Enable
4	ESI	RW	0x0	Error state indicator Write to this bit is meaningless if ESICTL=0 0: Error active 1: Error passive
3:0	DLC	RW	0x0	Data length code. For Non-FDCAN format, if this field is written to a value above 8, 8-byte data length is actually used for the data transfer. 0000: 0-byte 0001: 1-byte 0010: 2-byte 0011: 3-byte 0100: 4-byte 0101: 5-byte 0110: 6-byte 0111: 7-byte 1000: 8-byte 1001: 12-byte 1010: 16-byte 1011: 20-byte 1100: 24-byte 1101: 32-byte 1110: 48-byte 1111: 64-byte

23 CRC

23.1 CRC overview

A Cyclic Redundancy Check (CRC) or polynomial code checksum is a hash function designed to detect accidental changes to raw computer data, and is used to verify data transmission or storage. It generates up to 32-bit CRC code for error detection.

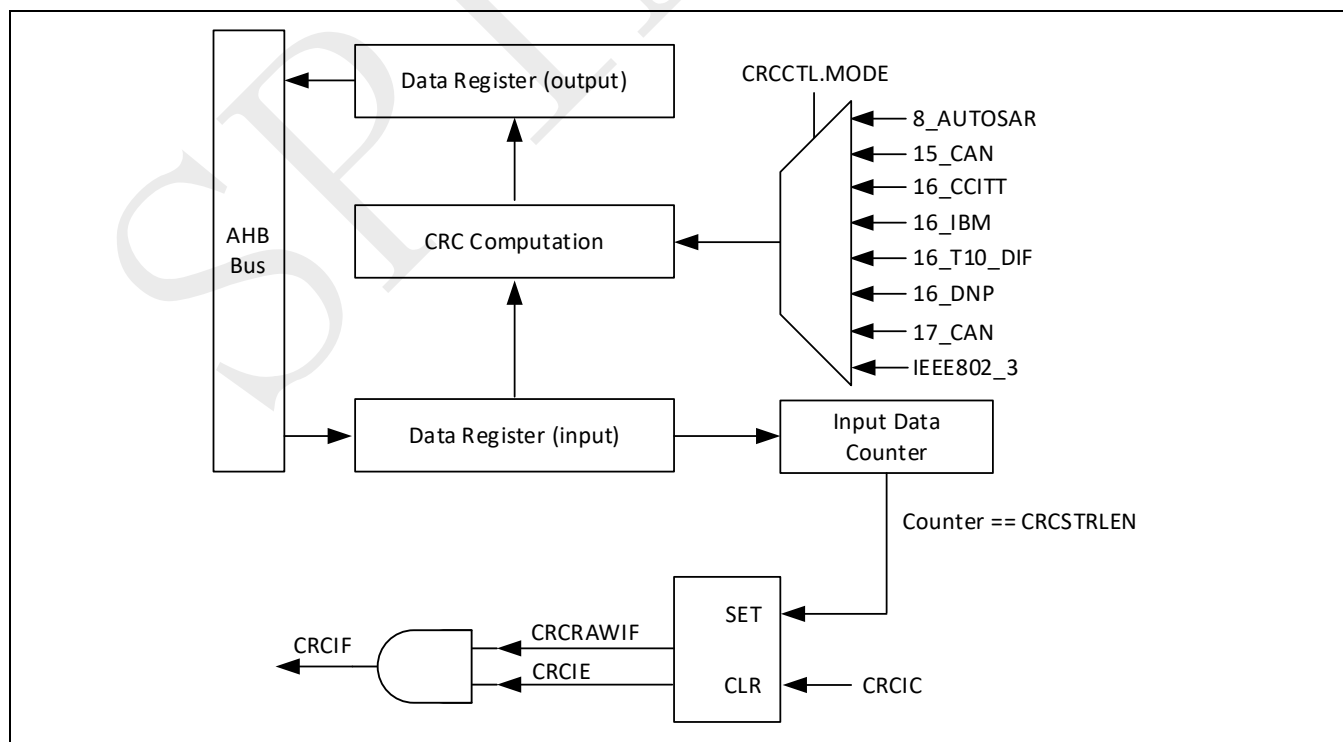
23.2 Main features

A standard AHB slave interface is used to configure the module, input the bit stream, and output the CRC result. The CRC module supports the following features:

- Supports 32-bit parallel bit stream input, and supports up to 32-bit CRC output
- Supports up to 2^{32} (4294967296) byte input data stream for CRC calculation
- Supports the 8 CRC standards with polynomials below.
 - CRC-8-AUTOSAR: $x^8+x^5+x^3+x^2+x+1$
 - CRC-15-CAN: $x^{15}+x^{14}+x^{10}+x^8+x^7+x^4+x^3+1$
 - CRC-16-CCITT: $x^{16}+x^{12}+x^5+1$
 - CRC-16-IBM: $x^{16}+x^{15}+x^2+1$
 - CRC-16-T10-DIF: $x^{16}+x^{15}+x^{11}+x^9+x^8+x^7+x^5+x^4+x^2+1$
 - CRC-16-DNP: $x^{16}+x^{13}+x^{12}+x^{11}+x^{10}+x^8+x^6+x^5+x^2+1$
 - CRC-17-CAN: $x^{17}+x^{16}+x^{14}+x^{13}+x^{11}+x^6+x^4+x^3+x+1$
 - CRC-32-IEEE 802.3: $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

The block diagram is shown in [Figure 23-1](#).

Figure 23-1: CRC block diagram



23.3 CRC operation flow

User can use the CRC module according to the following steps:

- Step 1: Disable CRC (Set CRCCTL.EN to 0).
- Step 2: Disable the interrupt (Set CRCIE.DONE to 0).
- Step 3: Clear all the interrupts (Set CRCIC.DONE to 1).
- Step 4: Configure the stream length (Set register CRCSTRLEN).
- Step 5: Configure CRC mode (Set bit CRCCTL.MODE).
- Step 6: Enable the interrupt (Set CRCIE.DONE to 1).
- Step 7: Enable CRC (Set CRCCTL.EN to 1).
- Step 8: Write stream in and waiting for interrupt to occur (If interrupt occurred, go to Step 9).
- Step 9: Get CRC calculation result (Read register CRCRESULT).
- Step 10: CRC operation complete.

Note: The CRC input stream registers can accept a word (32 bit) at a time. If the input data is not 4 bytes aligned, user should pad zeros at the start of the data stream. For example, if the data stream consists of 5 bytes starting from the lower address: 0xA1, 0xA2, 0xA3, 0xA4, 0xA5, the following two words should be written to the stream input register:

- 0xA1000000
- 0xA5A4A3A2

The CRC result bit order is:

- 16 bit CRC: x0~x15 [MSB->LSB]
 - 32 bit CRC: x0~x31 [MSB->LSB]
-

23.4 Registers

23.4.1 CRC register map

Table 23-1: CRC Module Base Address

Peripheral Module	Base Address
CRC	0x4000 3600

Table 23-2: CRC Register Map

Register	Offset	Description	Reset Value
<u>CRCCTL</u> *	0x00	CRC Control Register	0x00000000
<u>CRCSTRLEN</u> *	0x04	CRC Stream Length Register	0x00000000
<u>CRCSEED</u> *	0x08	CRC Seed Register	0x00000000
<u>CRCSTRIN</u>	0x0C	CRC Stream Input Register	0x00000000
<u>CRCRESULT</u>	0x10	CRC Result Register	0x00000000
<u>CRCRAWIF</u>	0x14	CRC Interrupt Raw Flag Register	0x00000000
<u>CRCIF</u>	0x18	CRC Interrupt Flag Register	0x00000000
<u>CRCIC</u>	0x1C	CRC Interrupt Clear Register	0x00000000
<u>CRCIE</u> *	0x20	CRC Interrupt Enable Register	0x00000000
<u>CRCREGKEY</u>	0x24	CRC Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the CRCREGKEY = 0x1ACCE551.

23.4.2 CRC registers

Table 23-3: CRC Control Register (CRCCTL) Layout

CRCCTL (CRC Control Register)							
Access: CRCx->CRCCTL		Offset: 0x00		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED						PERBYTE	INVOUT
7	6	5	4	3	2	1	0
INVIN	ENDIANOUT	ENDIANIN	CONTINUOUS	MODE			EN

Table 23-4: CRC Control Register (CRCCTL) Description

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	PERBYTE	RW	0x0	Set the data format for each write to the CRCSTRIN register 0: CRCSTRIN is written per word (4 bytes) 1: CRCSTRIN is written per byte
8	INVOUT	RW	0x0	Bitwise inversion of the output result 0: Output result is the original value in LFSR 1: Output result is the bitwise inversed value in LFSR
7	INVIN	RW	0x0	Bitwise inversion of the input data 0: Put the original data into LFSR 1: Put the bitwise inversion of the data into LFSR
6	ENDIANOUT	RW	0x0	Output result endian 0: LSB is last stage in LFSR 1: LSB is first stage in LFSR
5	ENDIANIN	RW	0x0	Input data endian (per byte) 0: LSB is firstly sent into the LFSR per byte 1: MSB is firstly sent into the LFSR per byte
4	CONTINUOUS	RW	0x0	Continuous frame calculation 0: Internal register will be cleared when CRCCTL.EN=0 so next frame calculation is a fresh new start 1: Internal register hold the value when CRCCTL.EN=0 so next frame calculation is continous to current one

Bits	Field Name	Type	Reset	Description
3:1	MODE	RW	0x0	CRC mode select 000: $x^8+x^5+x^3+x^2+x+1$ (CRC-8-AUTOSAR) 001: $x^{15}+x^{14}+x^{10}+x^8+x^7+x^4+x^3+1$ (CRC-15-CAN) 010: $x^{16}+x^{12}+x^5+1$ (CRC-16-CCITT) 011: $x^{16}+x^{15}+x^2+1$ (CRC-16-IBM) 100: $x^{16}+x^{15}+x^{11}+x^9+x^8+x^7+x^5+x^4+x^2+1$ (CRC-16-T10-DIF) 101: $x^{16}+x^{13}+x^{12}+x^{11}+x^{10}+x^8+x^6+x^5+x^2+1$ (CRC-16-DNP) 110: $x^{17}+x^{16}+x^{14}+x^{13}+x^{11}+x^6+x^4+x^3+x+1$ (CRC-17-CAN) 111: $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$ (CRC-32-IEEE 802.3)
0	EN	RW	0x0	CRC calculate enable 0: Disable CRC calculation 1: Enable CRC calculation

Table 23-5: CRC Stream Length Register (CRCSTRLEN) Layout

CRCSTRLEN (CRC Stream Length Register)							
Access: CRCx->CRCSTRLEN		Offset: 0x04		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 23-6: CRC Stream Length Register (CRCSTRLEN) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Input stream length, which is (CRCSTRLEN+1) bytes

Table 23-7: CRC Seed Register (CRCSEED) Layout

CRCSEED (CRC Seed Register)							
Access: CRCx->CRCSEED		Offset: 0x08		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 23-8: CRC Seed Register (CRCSEED) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Set the LFSR values to the seed

Table 23-9: CRC Stream Input Register (CRCSTRIN) Layout

CRCSTRIN (CRC Stream Input Register)							
Access: CRCx->CRCSTRIN		Offset: 0x0C		Default: 0x00000000			
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 23-10: CRC Stream Input Register (CRCSTRIN) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Stream Input data register The 24 MSBs will be always read as 0 if CRCCTLEN.PERBYTE=1

Table 23-11: CRC Result Register (CRCRESULT) Layout

CRCRESULT (CRC Result Register) Offset: 0x10 Default: 0x00000000 Access: CRCx->CRCRESULT							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 23-12: CRC Result Register (CRCRESULT) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	CRC calculation result

Table 23-13: CRC Interrupt Raw Flag Register (CRCRAWIF) Layout

CRCRAWIF (CRC Interrupt Raw Flag Register) Access: CRCx->CRCRAWIF Offset: 0x14 Default: 0x00000000							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						STROVF	DONE

Table 23-14: CRC Interrupt Raw Flag Register (CRCRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	STROVF	RO	0x0	CRC stream overflow event flag 0: Input stream does not overflow 1: Input stream exceeds the specified length
0	DONE	RO	0x0	CRC calculation done event flag 0: CRC calculation is not done 1: CRC calculation is done

Table 23-15: CRC Interrupt Flag Register (CRCIF) Layout

CRCIF (CRC Interrupt Flag Register)							
Access: CRCx->CRCIF		Offset: 0x18		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					INT	STROVF	DONE

Table 23-16: CRC Interrupt Flag Register (CRCIF) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	INT	RO	0x0	Global CRC interrupt flag issued to CPU 0: No interrupt was issued to CPU 1: Interrupt was issued to CPU. Further interrupt will not be generated until this bit is cleared
1	STROVF	RO	0x0	Masked CRC stream overflow interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not generate interrupt until this bit is cleared.
0	DONE	RO	0x0	Masked CRC calculation done interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not generate interrupt until this bit is cleared.

Table 23-17: CRC Interrupt Clear Register (CRCIC) Layout

CRCIC (CRC Interrupt Clear Register)							
Access: CRCx->CRCIC		Offset: 0x1C		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					INT	STROVF	DONE

Table 23-18: CRC Interrupt Clear Register (CRCIC) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2	INT	W1C	0x0	Global CRC interrupt clear 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears CRCIF.INT This bit is self-cleared to 0
1	STROVF	W1C	0x0	CRC stream overflow interrupt clear 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears STROVF bit in both CRCRAWIF and CRCIF This bit is self-cleared to 0
0	DONE	W1C	0x0	CRC calculation done interrupt clear 0: Write a 0 has no effect and always reads back 0 1: Write a 1 clears DONE bit in both CRCRAWIF and CRCIF This bit is self-cleared to 0

Table 23-19: CRC Interrupt Enable Register (CRCIE) Layout

CRCIE (CRC Interrupt Enable Register)							
Access: CRCx->CRCIE		Offset: 0x20		Default: 0x00000000			
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED						STROVF	DONE

Table 23-20: CRC Interrupt Enable Register (CRCIE) Description

Bits	Field Name	Type	Reset	Description
31:2	RESERVED_31_2	RO	0x0	Reserved.
1	STROVF	RW	0x0	CRC stream overflow interrupt enable 0: Disable 1: Enable
0	DONE	RW	0x0	CRC calculation done interrupt enable 0: Disable 1: Enable

Table 23-21: CRC Register Write-Allow Key Register (CRCREGKEY) Layout

CRCREGKEY (CRC Register Write-Allow Key Register)							
Access: CRCx->CRCREGKEY		Offset: 0x24		Default: 0x1ACCE551			
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 23-22: CRC Register Write-Allow Key Register (CRCREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected CRC registers

24 Flash memory

24.1 Features

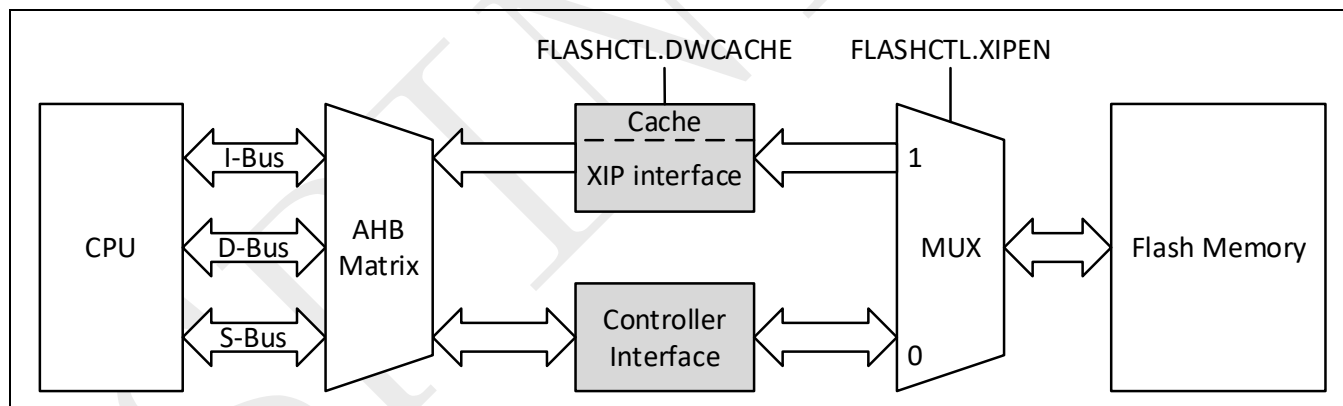
The SPC1169 Flash memory module has the following key features:

- Up to 128KB main on-chip flash memory
- Memory organization:
 - Main memory array: with total 32 sectors; 512 x 64 bits per sector
 - NVR memory array: with total 3 sectors; 512 x 64 bits per sector
- Program/Sector erase cycles: 100,000 cycles @ $T_J=85^\circ\text{C}$
- Data retention period: over 20 years @ $T_J=85^\circ\text{C}$
- ECC (Error Checking and Correction) protection
- Qualification conforms to AEC-Q100 grade 1

As shown in [Figure 24-1](#), flash memory interface features:

- XIP (Execute-In-Place) interface to support direct AHB read access with 64-bit cache.
- Controller interface for read/program/erase operation based on state machine
- Read-out/Write protection

Figure 24-1: Flash memory access interfaces



24.2 Flash module organization

The flash memory is based on 64-bit (DWord, double word) memory cells. It is organized as a main memory array containing 32 sectors of 512 DWord and an NVR array containing 3 sectors of 512 DWord as shown in [Figure 24-2](#) and [Table 24-1](#).

The NVR memory array can be used as:

- Data memory: The 3 NVR sectors are just used as normal flash memory.

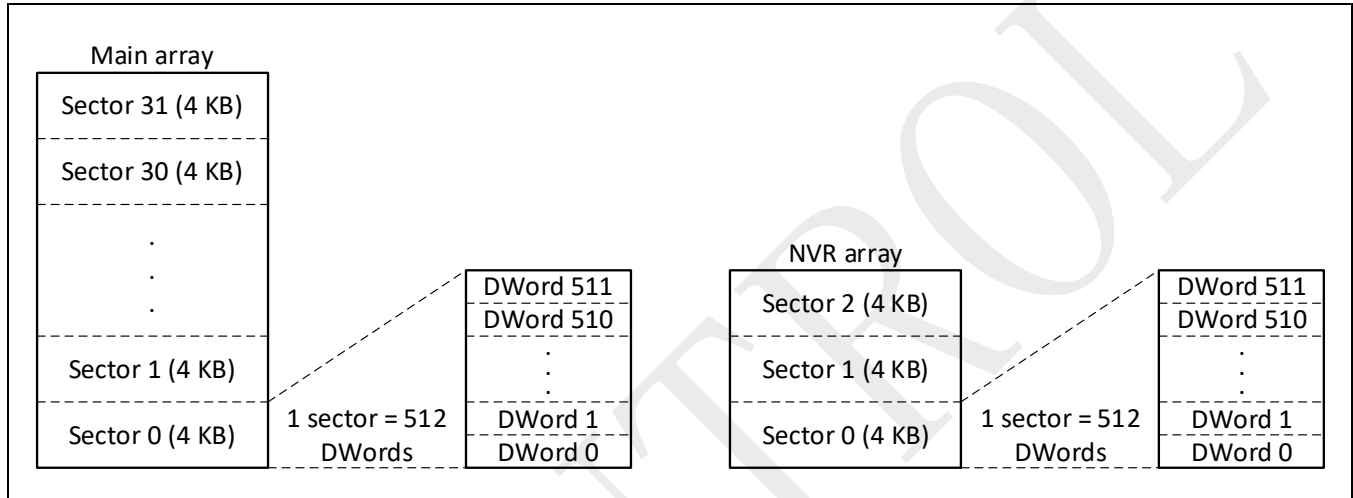
Write operation to the main memory block and the NVR memory block are managed by the controller interface as shown in [Figure 24-1](#). The high voltage needed for program/erase operations is internally generated.

To avoid different types of unwanted access, the main flash memory can be protected as below:

- Sector write protection
- Read-out protection

Note: For operations on the flash memory through the controller interface, the XIP interface should be disabled first (FLASHCTL.XIPEN = 0).

Figure 24-2: Logical structure of the Flash module



[1] DWord = double word, 1 DWord = 64-bit.

Table 24-1: Flash module organization

Block	Name	Base address	Size
Main Memory	Sector 0	0x1000 0000	4096 bytes
	Sector 1	0x1000 1000	4096 bytes
	Sector 2	0x1000 2000	4096 bytes
	Sector 3	0x1000 3000	4096 bytes
			
	Sector 31	0x1001 F000	4096 bytes
NVR Memory	Sector 0	0x1100 2000	4096 bytes
	Sector 1	0x1100 3000	4096 bytes
	Sector 2	0x1100 4000	4096 bytes

24.3 Read operation

The CPU can read data from the embedded flash module through the XIP interface or the controller interface. The read granularity is exactly the same as the size of memory cell, namely 1 DWord.

24.3.1 Read through XIP interface

With the XIP interface, the flash memory can be addressed as a common memory space. That means the CPU can access the flash memory in bytes, half-words (16 bits) or words (32 bits) by AHB interface directly. The main task of the XIP interface is mapping the AHB read access signals to control signals for reading from the flash memory. Therefore, the CPU can use it to fetch the instruction over the I-bus and the literal pool (constant/data) over the D-bus.

Read accesses may be performed with latency. It is the clock cycles needed for a read operation on-the-fly, namely the ratio of flash memory read time to the system clock period. Please refer to [Table 24-2](#) for the details.

The read granularity is 64-bit while the CPU access is up to 32-bit each time. Therefore, a 64-bit cache is provided to speed up the execution. The next word to be fetched may be already in the cache when it is enabled.

Table 24-2: Flash memory read access latency

System Clock Frequency	Latency
$0 < \text{SYSCLK} \leq 25 \text{ MHz}$	0
$25 \text{ MHz} < \text{SYSCLK} \leq 50 \text{ MHz}$	1
$50 \text{ MHz} < \text{SYSCLK} \leq 75 \text{ MHz}$	2
$75 \text{ MHz} < \text{SYSCLK} \leq 100 \text{ MHz}$	3

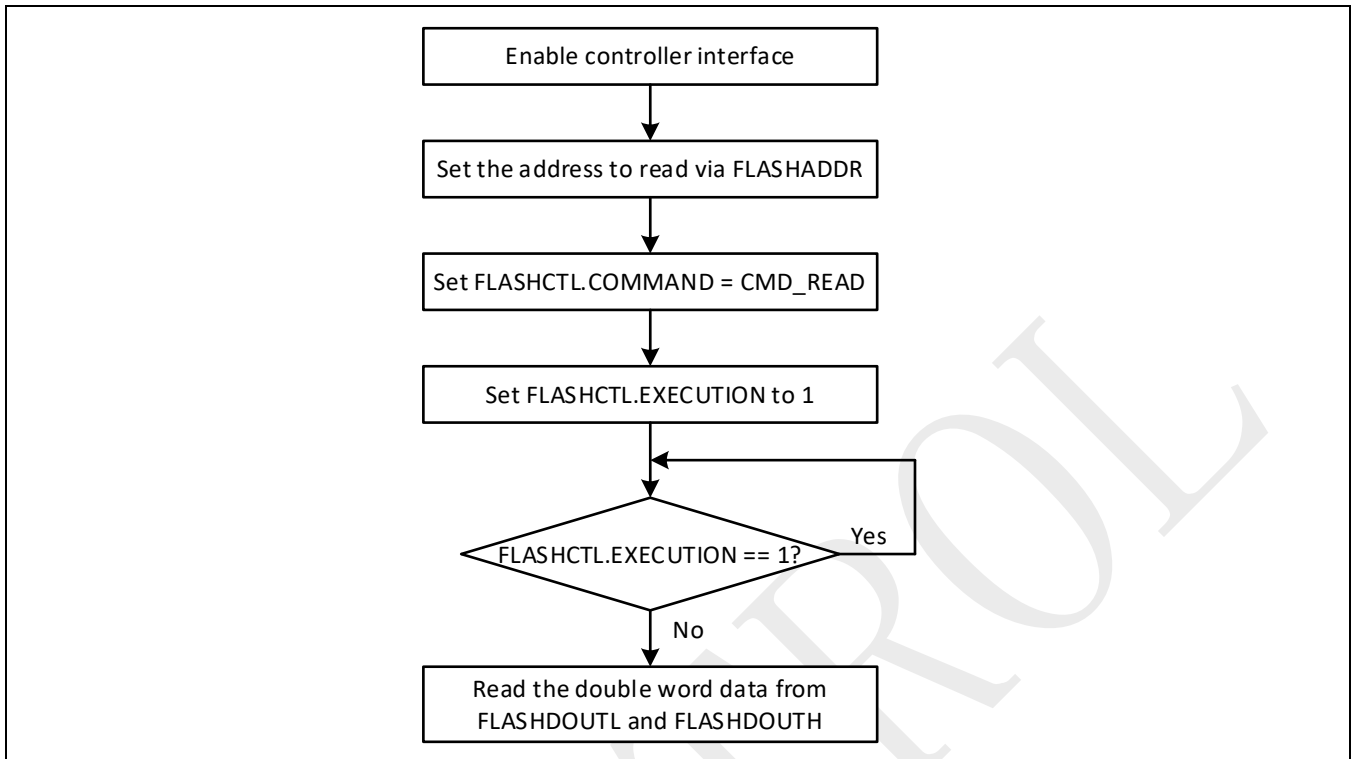
24.3.2 Read through controller interface

The CPU can also use the controller interface to read a DWord data at a time from the flash memory. The reading sequence is as below:

- Write FLASHCREGKEY as 0x1ACCE551 to enable write access to other registers
- Write FLASHCTL.XIPEN as 0 to disable the XIP interface and activate the controller interface
- Set the FLASHADDR register to select a memory cell to read
- Set the COMMAND bits in the FLASHCTL register with CMD_READ
- Set the EXECUTION bit in the FLASHCTL register
- Wait for the EXECUTION bit to be cleared by hardware
- Read the double word data from the FLASHDOUTL and FLASHDOUTH registers
- Write FLASHCTL.XIPEN as 1 to switch back to the XIP interface
- Write FLASHCREGKEY as 0x0 (any value other than 0x1ACCE551) to enable register write protection

Users can also call `pHWLIB->FLASHC_Read()` function to read through the controller interface. Refer to the descriptions in the SDK for the details.

Figure 24-3: Flash memory reading procedure



24.4 Program and erase operation

The flash controller interface handles the program and erase operations of the flash memory. It consists of eight 32-bit registers related to the program and erase.

- Flash control register (FLASHCTL)
- Flash address register (FLASHADDR)
- Flash input data low word register (FLASHDINL)
- Flash input data high word register (FLASHDINH)
- Flash output data low word register (FLASHDOUTL)
- Flash output data high word register (FLASHDOUTH)
- Flash write protection register (FLASHWP)
- Flash register write-allow key register (FLASHCREGKEY)

24.4.1 Enable flash controller interface

After reset, the flash controller interface is disabled. It can be enabled by setting FLASHCTL.XIPEN to 0. However, the FLASHCTL register is write-protected. An unlock key value (0x1ACCE551) should be written to the FLASHCREGKEY register to enable the write access to the protected registers.

24.4.2 Flash memory programming

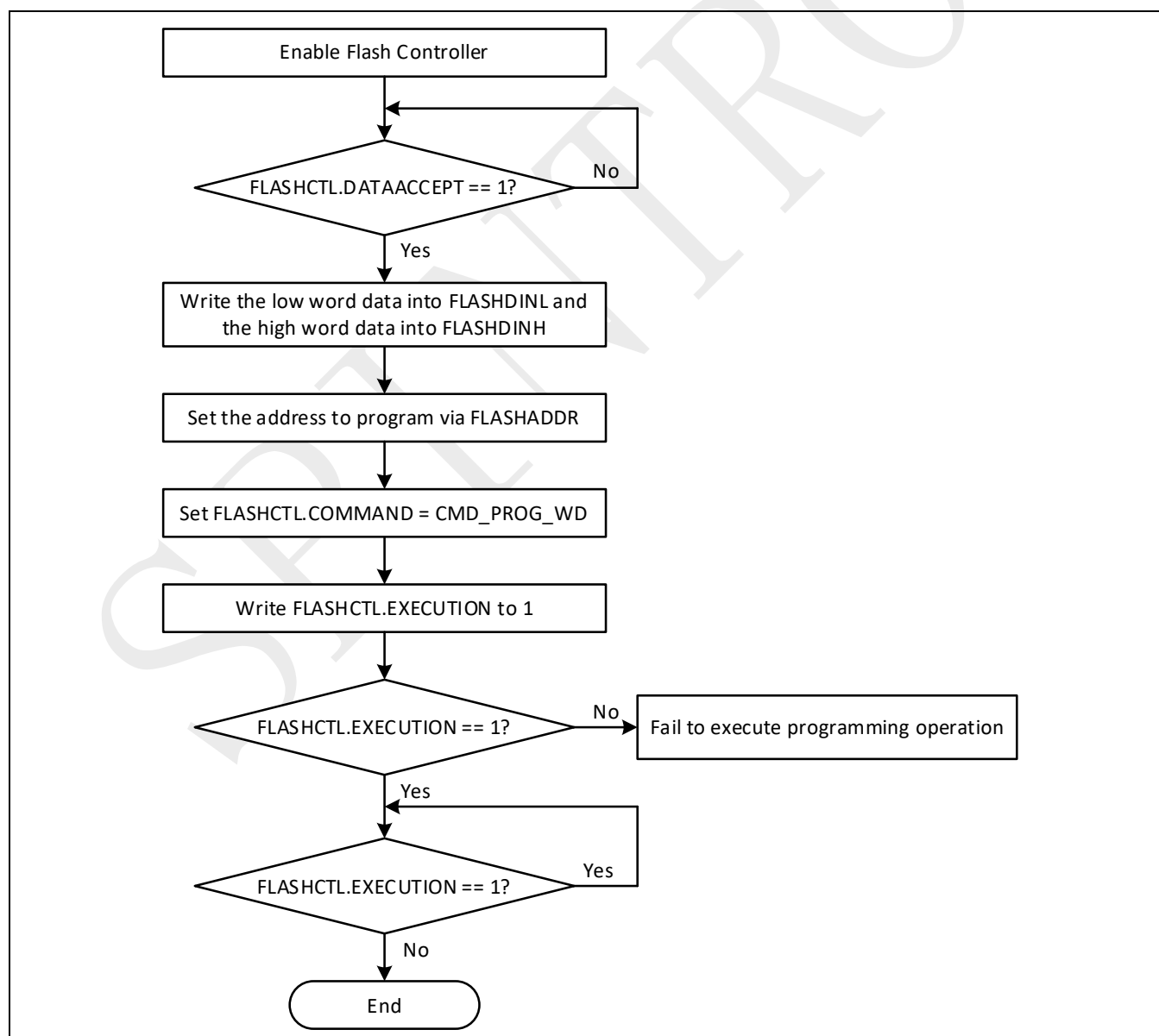
The program granularity is also the same as the size of memory cell, namely 1 DWord. The flash memory programming sequence is as below:

- Wait for the DATAACCEPT bit in the FLASHCTL register to be set
- Write the low word data to the FLASHDINL register and the high word data to the FLASHDINH register
- Set the FLASHADDR register to select a memory cell to program
- Set the COMMAND bits in the FLASHCTL register with CMD_PROG_DWORD
- Set the EXECUTION bit in the FLASHCTL register and then check whether the bit value was set
- Wait for the EXECUTION bit to be cleared

If the addressed main flash memory location is write-protected by the FLASHWP register, the EXECUTION bit could not be set and the program operation is skipped.

Users can also call pHWLIB->FLASHC_ProgramDWord() or pHWLIB->FLASHC_Program() function to program through the controller interface. Refer to the descriptions in the SDK for the details.

Figure 24-4: Flash memory programming procedure



24.4.3 Flash memory erase

The main flash memory can be erased by sector or by whole chip. And the NVR memory can only be erased by sector.

Sector Erase

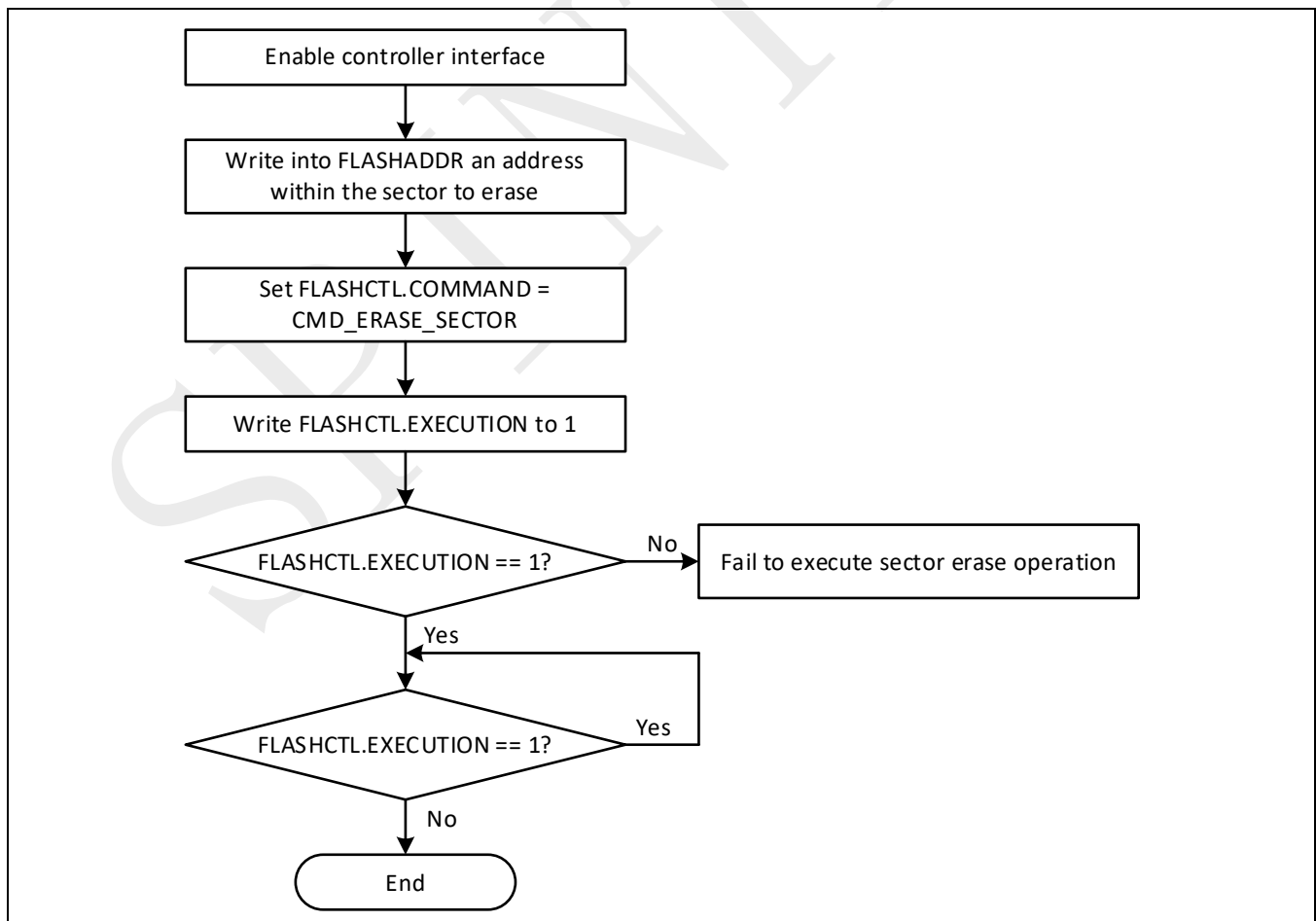
A sector of the flash memory can be erased using the sector erase by the flash controller interface. To erase a sector, the procedure below should be followed:

- Set the FLASHADDR register to select a sector to erase
- Set the COMMAND bits in the FLASHCTL register with CMD_ERASE_SECTOR
- Set the EXECUTION bit in the FLASHCTL register and then check whether the bit value was set
- Wait for the EXECUTION bit to be cleared

If the addressed main flash memory location is write-protected by the FLASHWP register, the EXECUTION bit could not be set and the sector erase operation is skipped.

Users can also call pHWLIB->FLASHC_EraseSector() function to erase the sector through the controller interface, and pHWLIB->FLASHC_VerifyErase() function to check whether the content is all 1's after erase. Refer to the descriptions in the SDK for the details.

Figure 24-5: Flash memory Sector Erase procedure



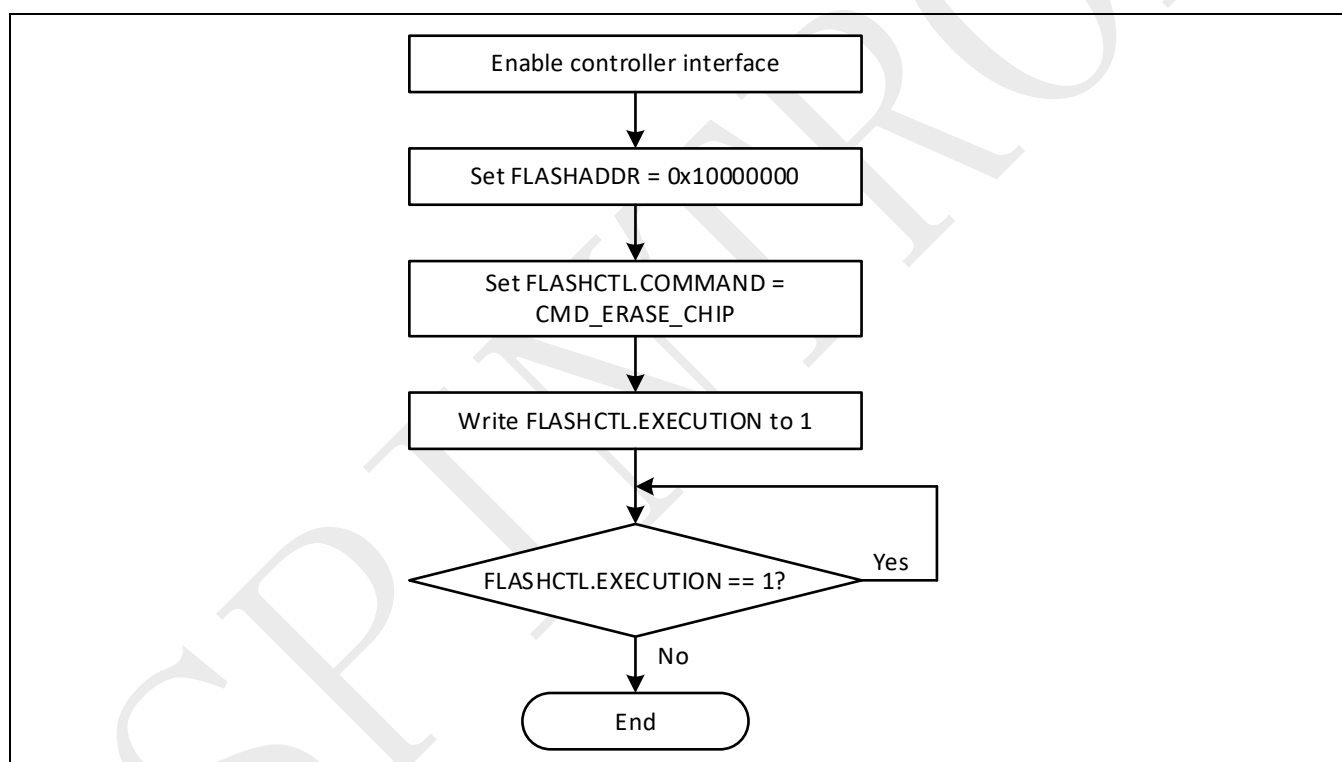
Chip Erase

The Chip Erase command can be used to completely erase all the sectors of the main flash memory. The NVR memory block is not affected by this procedure. The following sequence is recommended:

- Set the FLASHADDR register with the main flash start address
- Set the COMMAND bits in the FLASHCTL register with CMD_ERASE_CHIP
- Set the EXECUTION bit in the FLASHCTL register
- Wait for the EXECUTION bit to be reset

Users can also call pHWLIB->FLASHC_EraseChip() function to erase the whole chip through the Flash controller, and pHWLIB->FLASHC_VerifyErase() function to check whether the content is all 1's after erase. Refer to the descriptions in the SDK for the details.

Figure 24-6: Flash memory Chip Erase procedure



24.5 Protections

24.5.1 Read-out protection

The SPC1169 provides a read-out protection of the flash memory. It prevents reading and modifying the flash memory via the debug interface. This protection is active in all device operation modes.

The read-out protection is activated by setting the CHIP_SECURITY field in the configuration words area and then, by applying a system reset to reload the new configuration.

To disable the read-out protection:

- Erase the whole chip. The read-out protections are still enabled by this moment.
- Reset the device to reload the configuration and the chip would disable the read-out protection.

24.5.2 Write protection

The sectors of the main flash memory can be protected against unwanted write due to loss of program counter contexts. The write protection is implemented with a granularity of one sector at a time. If a program or an erase operation is performed on a protected sector, the operation will be skipped.

The write protection is activated by set the FLASHWP register bits and reset the register bits will disable the write protection.

24.6 Setup Flash timing

There are physically timing requirements for the flash access to avoid malfunction. It is a list of absolute values for the timing relationship of a series of events. An finite state machine (FSM) is designed in both the XIP interface and the flash controller interface, which guarantee the timing based on counting clock cycles with a list of timeout counts. For the ease of use, users can call pHWLIB->FLASHC_SetTiming() function. It takes the frequency of CPU clock as input parameter and automatically sets all the timeout counts.

Whenever user plans to change the CPU clock frequency, FLASHC_RelaxXIPTiming() should be called before the frequency change so as to relax the timing requirement on flash access. After the frequency change, pHWLIB->FLASHC_SetTiming() should be called to set the optimized timing parameters.

24.7 Configuration words description

There are two configuration words for system settings. They are programmed by the end user depending on the application requirements. The configuration words are located at the end (the last DWord) of the main flash memory, which is 0x1001FFF8.

The description of these configuration words are shown in [Table 24-3](#). The new programmed configuration words are loaded and activated after power-on reset.

Table 24-3: Description of the configuration words

Address	Name	Description
0x1001FFF8	WDT_ENABLE	Watchdog enable word 0xFFFFFFFF: Disable watchdog when chip start-up Others: Enable watchdog when chip start-up
0x1001FFFC	CHIP_SECURITY	Chip debug interface lock word 0xFFFFFFFF: the chip debug interface would not be locked Others: the chip debug interface would be locked

24.8 Registers

24.8.1 Flash register map

Table 24-4: Flash Module Base Address

Peripheral Module	Base Address
FLASHC	0x4000 1B00

Table 24-5: FLASHC Register Map

Register	Offset	Description	Reset Value
<u>FLASHCTL*</u>	0x00	Flash Control Register	0x00000010
<u>FLASHADDR</u>	0x08	Flash Address Register	0x00000000
<u>FLASHDINL</u>	0x10	Flash Data Input Low Word Register	0x00000000
<u>FLASHDINH</u>	0x14	Flash Data Input High Word Register	0x00000000
<u>FLASHDINE</u>	0x18	Flash Data Input ECC Byte Register	0x00000000
<u>FLASHDINP</u>	0x1C	Flash Data Input Pattern Register	0x00000000
<u>FLASHDOUTL</u>	0x20	Flash Data Output Low Word Register	0x00000000
<u>FLASHDOUTH</u>	0x24	Flash Data Output High Word Register	0x00000000
<u>FLASHDOUTE</u>	0x28	Flash Data Output ECC Byte Register	0x00000000
<u>FLASHDOUTP</u>	0x2C	Flash Data Output Pattern Register	0x00000000
<u>FLASHWP*</u>	0x58	Flash Write Protect Register	0x00000000
<u>FLASHCREGKEY</u>	0x60	Flash Controller Register Write-Allow Key Register	0x00000000

Note: Registers marked with * are write-allowed only when the FLASHREGKEY=0x1ACCE551.

24.8.2 Flash registers

Table 24-6: Flash Control Register (FLASHCTL) Layout

FLASHCTL (Flash Control Register) Offset: 0x0 Default: 0x00000010 Access: FLASHC->FLASHCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							DWCACHEWE
7	6	5	4	3	2	1	0
DWCACHE	XIPEN	EXECUTION	DATAACCEPT	COMMAND			

Table 24-7: Flash Control Register (FLASHCTL) Description

Bits	Field Name	Type	Reset	Description
31:9	RESERVED_31_9	RO	0x0	Reserved.
8	DWCACHEWE	W1S	0x0	DWCACHE bit write enable. Need to write as 1 at the same time when writing DWCACHE bit 0: Write 0 has no effect and always reads back 0 1: Write 1 enables concurrent write to DWCACHE. This bit is self-cleared to 0
7	DWCACHE	RW	0x0	Enable simple cache if the new address are in the same double-word as the last address 0: Disable double-word cache 1: Enable double-word cache
6	XIPEN	RW	0x0	XIP module enable 0: Disable XIP module 1: Enable XIP module
5	EXECUTION	W1C	0x0	Command is in execution 0: Write a 0 has no effect Read a 0 indicates no command execution 1: Write a 1 starts a new command execution cycle. Read a 1 indicates command is still executing. It will change to 0 after execution is accomplished.
4	DATAACCEPT	RO	0x1	FLASHDIN cannot be changed when this bit is 0
3:0	COMMAND	RW	0x0	Flash command 0000: Power up 0001: Read 0010: Program word 0011: Program page

Bits	Field Name	Type	Reset	Description
				0100: Erase sector 0101: Erase chip 0110: Recall read(used only when reading NVR sectors) 0111: Set configuration 1110: Verify read 1111: Powe down

Table 24-8: Flash Address Register (FLASHADDR) Layout

FLASHADDR (Flash Address Register) Offset: 0x8 Default: 0x00000000 Access: FLASHC->FLASHADDR							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

Table 24-9: Flash Address Register (FLASHADDR) Description

Bits	Field Name	Type	Reset	Description
31:0	ADDR	RW	0x0	Flash address

Table 24-10: Flash Data Input Low Word Register (FLASHDINL) Layout

FLASHDINL (Flash Data Input Low Word Register) Offset: 0x10 Default: 0x00000000 Access: FLASHC->FLASHDINL							
31	30	29	28	27	26	25	24
DINL							
23	22	21	20	19	18	17	16
DINL							
15	14	13	12	11	10	9	8
DINL							
7	6	5	4	3	2	1	0
DINL							

Table 24-11: Flash Data Input Low Word Register (FLASHDINL) Description

Bits	Field Name	Type	Reset	Description
31:0	DINL	RW	0x0	Flash input low word data

Table 24-12: Flash Data Input High Word Register (FLASHDINH) Layout

FLASHDINH (Flash Data Input High Word Register) Offset: 0x14 Default: 0x00000000 Access: FLASHC->FLASHDINH							
31	30	29	28	27	26	25	24
DINH							
23	22	21	20	19	18	17	16
DINH							
15	14	13	12	11	10	9	8
DINH							
7	6	5	4	3	2	1	0
DINH							

Table 24-13: Flash Data Input High Word Register (FLASHDINH) Description

Bits	Field Name	Type	Reset	Description
31:0	DINH	RW	0x0	Flash input high word data

Table 24-14: Flash Data Input ECC Byte Register (FLASHDINE) Layout

FLASHDINE (Flash Data Input ECC Byte Register) Offset: 0x18 Default: 0x00000000 Access: FLASHC->FLASHDINE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DINE							

Table 24-15: Flash Data Input ECC Byte Register (FLASHDINE) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	DINE	RW	0x0	Flash input ECC byte data It is only effective when hardware ECC is disabled

Table 24-16: Flash Data Input Pattern Register (FLASHDINP) Layout

FLASHDINP (Flash Data Input Pattern Register) Offset: 0x1C Default: 0x00000000 Access: FLASHC->FLASHDINP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					PATTERN		

Table 24-17: Flash Data Input Pattern Register (FLASHDINP) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2:0	PATTERN	RW	0x0	Flash raw input pattern without ECC correction 000: Input is set by FLASHDIN{E H L} 100: All 0's 101: All 0x55 checkboard bytes 110: All 0xAA checkboard bytes 111: All 1's

Table 24-18: Flash Data Output Low Word Register (FLASHDOUTL) Layout

FLASHDOUTL (Flash Data Output Low Word Register) Offset: 0x20 Default: 0x00000000 Access: FLASHC->FLASHDOUTL							
31	30	29	28	27	26	25	24
DOUTL							
23	22	21	20	19	18	17	16
DOUTL							
15	14	13	12	11	10	9	8
DOUTL							
7	6	5	4	3	2	1	0
DOUTL							

Table 24-19: Flash Data Output Low Word Register (FLASHDOUTL) Description

Bits	Field Name	Type	Reset	Description
31:0	DOUTL	RO	0x0	Flash output low word data

Table 24-20: Flash Data Output High Word Register (FLASHDOUTH) Layout

FLASHDOUTH (Flash Data Output High Word Register) Offset: 0x24 Default: 0x00000000 Access: FLASHC->FLASHDOUTH							
31	30	29	28	27	26	25	24
DOUTH							
23	22	21	20	19	18	17	16
DOUTH							
15	14	13	12	11	10	9	8
DOUTH							
7	6	5	4	3	2	1	0
DOUTH							

Table 24-21: Flash Data Output High Word Register (FLASHDOUTH) Description

Bits	Field Name	Type	Reset	Description
31:0	DOUTH	RO	0x0	Flash output high word data

Table 24-22: Flash Data Output ECC Byte Register (FLASHDOUTE) Layout

FLASHDOUTE (Flash Data Output ECC Byte Register) Offset: 0x28 Default: 0x00000000 Access: FLASHC->FLASHDOUTE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
DOUTE							

Table 24-23: Flash Data Output ECC Byte Register (FLASHDOUTE) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7:0	DOUTE	RO	0x0	Flash output ECC byte data

Table 24-24: Flash Data Output Pattern Register (FLASHDOUTP) Layout

FLASHDOUTP (Flash Data Output Pattern Register) Offset: 0x2C Default: 0x00000000 Access: FLASHC->FLASHDOUTP							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED					PATTERN		

Table 24-25: Flash Data Output Pattern Register (FLASHDOUTP) Description

Bits	Field Name	Type	Reset	Description
31:3	RESERVED_31_3	RO	0x0	Reserved.
2:0	PATTERN	RO	0x0	Flash raw output pattern without ECC correction 000: No special pattern recognised 100: All 0's 101: All 0x55 checkboard bytes 110: All 0xAA checkboard bytes 111: All 1's

Table 24-26: Flash Write Protect Register (FLASHWP) Layout

FLASHWP (Flash Write Protect Register) Offset: 0x58 Default: 0x00000000 Access: FLASHC->FLASHWP							
31	30	29	28	27	26	25	24
SECTOR31	SECTOR30	SECTOR29	SECTOR28	SECTOR27	SECTOR26	SECTOR25	SECTOR24
23	22	21	20	19	18	17	16
SECTOR23	SECTOR22	SECTOR21	SECTOR20	SECTOR19	SECTOR18	SECTOR17	SECTOR16
15	14	13	12	11	10	9	8
SECTOR15	SECTOR14	SECTOR13	SECTOR12	SECTOR11	SECTOR10	SECTOR9	SECTOR8
7	6	5	4	3	2	1	0
SECTOR7	SECTOR6	SECTOR5	SECTOR4	SECTOR3	SECTOR2	SECTOR1	SECTOR0

Table 24-27: Flash Write Protect Register (FLASHWP) Description

Bits	Field Name	Type	Reset	Description
31	SECTOR31	RW	0x0	Write protection for flash sector 31 0: Can be programed or erased 1: Cannot be programed or sector erased
30	SECTOR30	RW	0x0	Write protection for flash sector 30 0: Can be programed or erased 1: Cannot be programed or sector erased

Bits	Field Name	Type	Reset	Description
29	SECTOR29	RW	0x0	Write protection for flash sector 29 0: Can be programed or erased 1: Cannot be programed or sector erased
28	SECTOR28	RW	0x0	Write protection for flash sector 28 0: Can be programed or erased 1: Cannot be programed or sector erased
27	SECTOR27	RW	0x0	Write protection for flash sector 27 0: Can be programed or erased 1: Cannot be programed or sector erased
26	SECTOR26	RW	0x0	Write protection for flash sector 26 0: Can be programed or erased 1: Cannot be programed or sector erased
25	SECTOR25	RW	0x0	Write protection for flash sector 25 0: Can be programed or erased 1: Cannot be programed or sector erased
24	SECTOR24	RW	0x0	Write protection for flash sector 24 0: Can be programed or erased 1: Cannot be programed or sector erased
23	SECTOR23	RW	0x0	Write protection for flash sector 23 0: Can be programed or erased 1: Cannot be programed or sector erased
22	SECTOR22	RW	0x0	Write protection for flash sector 22 0: Can be programed or erased 1: Cannot be programed or sector erased
21	SECTOR21	RW	0x0	Write protection for flash sector 21 0: Can be programed or erased 1: Cannot be programed or sector erased
20	SECTOR20	RW	0x0	Write protection for flash sector 20 0: Can be programed or erased 1: Cannot be programed or sector erased
19	SECTOR19	RW	0x0	Write protection for flash sector 19 0: Can be programed or erased 1: Cannot be programed or sector erased
18	SECTOR18	RW	0x0	Write protection for flash sector 18 0: Can be programed or erased 1: Cannot be programed or sector erased
17	SECTOR17	RW	0x0	Write protection for flash sector 17 0: Can be programed or erased 1: Cannot be programed or sector erased
16	SECTOR16	RW	0x0	Write protection for flash sector 16 0: Can be programed or erased 1: Cannot be programed or sector erased
15	SECTOR15	RW	0x0	Write protection for flash sector 15 0: Can be programed or erased 1: Cannot be programed or sector erased

Bits	Field Name	Type	Reset	Description
14	SECTOR14	RW	0x0	Write protection for flash sector 14 0: Can be programed or erased 1: Cannot be programed or sector erased
13	SECTOR13	RW	0x0	Write protection for flash sector 13 0: Can be programed or erased 1: Cannot be programed or sector erased
12	SECTOR12	RW	0x0	Write protection for flash sector 12 0: Can be programed or erased 1: Cannot be programed or sector erased
11	SECTOR11	RW	0x0	Write protection for flash sector 11 0: Can be programed or erased 1: Cannot be programed or sector erased
10	SECTOR10	RW	0x0	Write protection for flash sector 10 0: Can be programed or erased 1: Cannot be programed or sector erased
9	SECTOR9	RW	0x0	Write protection for flash sector 9 0: Can be programed or erased 1: Cannot be programed or sector erased
8	SECTOR8	RW	0x0	Write protection for flash sector 8 0: Can be programed or erased 1: Cannot be programed or sector erased
7	SECTOR7	RW	0x0	Write protection for flash sector 7 0: Can be programed or erased 1: Cannot be programed or sector erased
6	SECTOR6	RW	0x0	Write protection for flash sector 6 0: Can be programed or erased 1: Cannot be programed or sector erased
5	SECTOR5	RW	0x0	Write protection for flash sector 5 0: Can be programed or erased 1: Cannot be programed or sector erased
4	SECTOR4	RW	0x0	Write protection for flash sector 4 0: Can be programed or erased 1: Cannot be programed or sector erased
3	SECTOR3	RW	0x0	Write protection for flash sector 3 0: Can be programed or erased 1: Cannot be programed or sector erased
2	SECTOR2	RW	0x0	Write protection for flash sector 2 0: Can be programed or erased 1: Cannot be programed or sector erased
1	SECTOR1	RW	0x0	Write protection for flash sector 1 0: Can be programed or erased 1: Cannot be programed or sector erased
0	SECTOR0	RW	0x0	Write protection for flash sector 0 0: Can be programed or erased 1: Cannot be programed or sector erased

Table 24-28: Flash Controller Register Write-Allow Key Register (FLASHCREGKEY) Layout

FLASHCREGKEY (Flash Controller Register Write-Allow Key Register) Offset: 0x60 Default: 0x00000000 Access: FLASHC->FLASHCREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 24-29: Flash Controller Register Write-Allow Key Register (FLASHCREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x0	Write 0x1ACCE551 to unlock protected flash registers

25 System control

25.1 Overview

The system control module provides:

- Unique device ID register (64 bits)
- System monitor control registers
- GPIO wake-up control register
- Memory ECC error address registers
- Non-maskable Interrupt (NMI) event control registers
- Reset event control registers
- System information register
- Boot information register

25.2 Unique device ID register (64 bits)

The 64-bit unique device identifier is the manufacture time of the chip. It also provides a reference number which is unique for any device and in any context. These bits can never be altered by the user. It is ideally to:

- be used as serial numbers
- be used as security keys in order to increase the security of code in flash memory.
 - It is achieved by using and combining this unique ID with software cryptographic primitives and protocols before programming the internal flash memory
- to activate secure boot processes, etc.

25.3 System monitor control registers

These registers are used to enable system voltage or temperature monitoring.

25.4 GPIO wake-up control register

The register enables the selected GPIO pin to wake up MCU from sleep or stop mode.

25.5 Memory ECC error address registers

These registers store the memory ECC error addresses.

25.6 NMI event control registers

These registers are used to enable, indicate and clear NMI events.

25.7 Reset event control registers

These registers are used to enable, indicate and clear reset events.

25.8 System information register

The register provides information about flash size, RAM size, ADC effective bits and clock frequency option.

25.9 Boot information register

The register provides information of RAM built-in self-test result status, flash initialization status, critical register default value check status, and the count of chip boot start-up after power-on. The RAM built-in self-test is only executed after Power-on-Reset (POR).

25.10 Registers

25.10.1 System register map

Table 25-1: SYSTEM Module Base Address

Peripheral Module	Base Address
SYSTEM	0x4000 0000

Table 25-2: SYSTEM Register Map

Register	Offset	Description	Reset Value
CID0	0x00	Chip ID Register 0	0x00000000
CID1	0x04	Chip ID Register 1	0x00000000
UID0	0x08	Unique ID Register 0	0x00000000
UID1	0x0C	Unique ID Register 1	0x00000000
RND0	0x10	Random Number Register 0	0x00000000
RND1	0x14	Random Number Register 1	0x00000000
REV0	0x18	Revision Information Register 0	0x63681010
REV1	0x1C	Revision Information Register 1	0x20200200
SYSMONCTL *	0x20	System Monitor Control Register	0xF800FFFF
TMCUTH *	0x40	Controller Temperature Monitor Threshold Register	0x0FFF0000
VDD33TH *	0x44	VDD33 Monitor Threshold Register	0x0FFF0000
VDD12TH *	0x48	VDD12 Monitor Threshold Register	0x0FFF0000
VADCREFTH *	0x4C	ADC Reference Voltage Monitor Threshold Register	0x0FFF0000
TMCUCODE	0x6C	Controller Temperature Monitor Code Register	0x00000000
VDD33CODE	0x70	VDD33 Monitor Code Register	0x00000000
VDD12CODE	0x74	VDD12 Monitor Threshold Register	0x00000000
VADCREFCODE	0x78	13-bit ADC Reference Voltage Monitor Code Register	0x00000000
GPIOWKUPCTL *	0x7C	GPIO Wake-Up Control Register	0x00010001
GPIONMICTL *	0x80	GPIO Non-Maskable Interrupt Control Register	0x00000001
NMRAWIF	0x84	Non-Maskable Interrupt Raw Flag Register	0x00000000
NMIF	0x88	Non-Maskable Interrupt Flag Register	0x00000000
NMIC	0x8C	Non-Maskable Interrupt Clear Register	0x00000000
NMIE *	0x90	Non-Maskable Interrupt Enable Register	0x00000400
MEMRAWIF	0x94	Memory Interrupt Raw Flag Register	0x00000000
MEMIF	0x98	Memory Interrupt Flag Register	0x00000000
MEMIC	0x9C	Memory Interrupt Clear Register	0x00000000
MEMIE *	0xA0	Memory Interrupt Enable Register	0x00000000
RSTEVTF LG	0xA4	Reset Event Flag Register	0x00000000
RSTEVTC LR	0xA8	Reset Event Flag Clear Register	0x00000000
RSTEVTEN *	0xAC	Reset Event Enable Register	0x00000C00
SYSINFO	0xB0	System Information Register	0x000001FF

Register	Offset	Description	Reset Value
ROMERRADDR	0xB4	ROM ECC Error Address	0xFFFFFFFF
FLASHERRADDR	0xB8	FLASH ECC Error Address	0xFFFFFFFF
RAM0ERRADDR	0xBC	RAM0 ECC Error Address	0xFFFFFFFF
RAM1ERRADDR	0xC0	RAM1 ECC Error Address	0xFFFFFFFF
DBGIFCTL	0xC8	Debug Interface Control Register	0x00000000
BOOTINFO	0xCC	Boot Information Register	0x00000000
AHBMBEFLG	0xD0	AHB Multi-Bit ECC Error Event Flag Register	0x00000000
AHBMBECLR	0xD4	AHB Multi-bit ECC Error Event Flag Register	0x00000000
AHBSBEFLG	0xD8	AHB Single-Bit ECC Error Event Flag Register	0x00000000
AHBSBECLR	0xDC	AHB Single-bit ECC Error Event Flag Register	0x00000000
PERIRSTFLG	0xE0	Peripheral Reset Event Flag Register	0x0000FFFF
PERIRSTCLR	0xE4	Peripheral Reset Event Clear Register	0x00000000
NVREG0	0xF0	Non-Volatile Register 0	0x00000000
NVREG1	0xF4	Non-Volatile Register 1	0x00000000
NVREG2	0xF8	Non-Volatile Register 2	0x00000000
SYSREGKEY	0xFC	System Register Write-Allow Key Register	0x1ACCE551

Note: Registers marked with * are write-allowed only when the SYSREGKEY=0x1ACCE551.

25.10.2 System registers

Table 25-3: Chip ID Register 0 (CID0) Layout

CID0 (Chip ID Register 0) Offset: 0x0 Default: 0x00000000 Access: SYSTEM->CID0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-4: Chip ID Register 0 (CID0) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	32 LSBs of the 64-bit chip ID

Table 25-5: Chip ID Register 1 (CID1) Layout

CID1 (Chip ID Register 1) Offset: 0x4 Default: 0x00000000 Access: SYSTEM->CID1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-6: Chip ID Register 1 (CID1) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	32 MSBs of the 64-bit chip ID

Table 25-7: Unique ID Register 0 (UID0) Layout

UID0 (Unique ID Register 0) Offset: 0x8 Default: 0x00000000 Access: SYSTEM->UID0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-8: Unique ID Register 0 (UID0) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	32 LSBs of the 64-bit unique ID It provides the manufacture time information in format of “HHmmSSsn”, where HH is the hour, mm is the minute, SS is the second, sn is the site number

Table 25-9: Unique ID Register 1 (UID1) Layout

UID1 (Unique ID Register 1) Offset: 0xC Default: 0x00000000 Access: SYSTEM->UID1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-10: Unique ID Register 1 (UID1) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	32 MSBs of the 64-bit unique ID. It provides the manufacture time information in format of “YYYYMMDD”, where YYYY is the year, MM is the month, DD is the day

Table 25-11: Random Number Register 0 (RND0) Layout

RND0 (Random Number Register 0) Offset: 0x10 Default: 0x00000000 Access: SYSTEM->RND0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-12: Random Number Register 0 (RND0) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	32 LSBs of the 64-bit random number

Table 25-13: Random Number Register 1 (RND1) Layout

RND1 (Random Number Register 1) Offset: 0x14 Default: 0x00000000 Access: SYSTEM->RND1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-14: Random Number Register 1 (RND1) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x0	32 MSBs of the 64-bit random number

Table 25-15: Revision Information Register 0 (REV0) Layout

REV0 (Revision Information Register 0) Offset: 0x18 Default: 0x63681010 Access: SYSTEM->REV0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-16: Revision Information Register 0 (REV0) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x63681010	32 LSBs of the 64-bit revision number

Table 25-17: Revision Information Register 1 (REV1) Layout

REV1 (Revision Information Register 1) Offset: 0x1C Default: 0x20200200 Access: SYSTEM->REV1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-18: Revision Information Register 1 (REV1) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0x20200200	32 MSBs of the 64-bit revision number

Table 25-19: System Monitor Control Register (SYSMONCTL) Layout

SYSMONCTL (System Monitor Control Register) Offset: 0x20 Default: 0xF800FFFF Access: SYSTEM->SYSMONCTL							
31	30	29	28	27	26	25	24
SCWIN					VBATEN	VCPEN	VMONEN
23	22	21	20	19	18	17	16
VDD5EN	VBGEN	TPMUEN	TLINEN	TMCUEN	VDD33EN	VDD12EN	VADCREFEN
15	14	13	12	11	10	9	8
MONPRD							
7	6	5	4	3	2	1	0
MONPRD							

Table 25-20: System Monitor Control Register (SYSMONCTL) Description

Bits	Field Name	Type	Reset	Description
31:27	SCWIN	RW	0x1F	Both sample and convert time is set to (SCWIN + 1)*4 HCLK cycles
26	VBATEN	RW	0x0	VBAT monitor enable 0: Disable 1: Enable
25	VCPEN	RW	0x0	VCP monitor enable 0: Disable 1: Enable
24	VMONEN	RW	0x0	VMON monitor enable 0: Disable 1: Enable
23	VDD5EN	RW	0x0	VDD5 monitor enable 0: Disable 1: Enable
22	VBGEN	RW	0x0	Power bandgap monitor enable 0: Disable 1: Enable
21	TPMUEN	RW	0x0	Power temperature monitor enable 0: Disable 1: Enable
20	TLINEN	RW	0x0	LIN PHY temperature monitor enable 0: Disable 1: Enable
19	TMCUEN	RW	0x0	Controller temperature monitor enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
18	VDD33EN	RW	0x0	VDD33 monitor enable 0: Disable 1: Enable
17	VDD12EN	RW	0x0	VDD12 monitor enable 0: Disable 1: Enable
16	VADCREFEN	RW	0x0	13-bit ADC reference voltage monitor enable 0: Disable 1: Enable
15:0	MONPRD	RW	0xFFFF	System monitor interval. i.e. Wait for (MONPRD*8+1) HCLK cycles to measure the next item

Table 25-21: VBAT Monitor Threshold Register (VBATTH) Layout

VBATTH (VBAT Monitor Threshold Register) Offset: 0x24 Default: 0x0FFF0000 Access: SYSTEM->VBATTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
				MAX			
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-22: VBAT Monitor Threshold Register (VBATTH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFF	Maximum allowed VBAT monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed VBAT monitor code

Table 25-23: VCP Monitor Threshold Register (VCP TH) Layout

VCP TH (VCP Monitor Threshold Register) Offset: 0x28 Default: 0x0FFF0000 Access: SYSTEM->VCP TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-24: VCP Monitor Threshold Register (VCP TH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFFF	Maximum allowed VCP monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed VCP monitor code

Table 25-25: VMON Monitor Threshold Register (VMON TH) Layout

VMON TH (VMON Monitor Threshold Register) Offset: 0x2C Default: 0x0FFF0000 Access: SYSTEM->VMON TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-26: VMON Monitor Threshold Register (VMON TH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFFF	Maximum allowed VMON monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed VMON monitor code

Table 25-27: VDD5 Monitor Threshold Register (VDD5TH) Layout

VDD5TH (VDD5 Monitor Threshold Register) Offset: 0x30 Default: 0x0FFF0000 Access: SYSTEM->VDD5TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-28: VDD5 Monitor Threshold Register (VDD5TH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFFF	Maximum allowed VDD5 monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed VDD5 monitor code

Table 25-29: Power Bandgap Monitor Threshold Register (VBGTH) Layout

VBGTH (Power Bandgap Monitor Threshold Register) Offset: 0x34 Default: 0x0FFF0000 Access: SYSTEM->VBGTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-30: Power Bandgap Monitor Threshold Register (VBGTH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFFF	Maximum allowed power bandgap monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed power bandgap monitor code

Table 25-31: Power Module Temperature Monitor Threshold Register (TPMUTH) Layout

TPMUTH (Power Module Temperature Monitor Threshold Register) Offset: 0x38 Default: 0x0FFF0000 Access: SYSTEM->TPMUTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-32: Power Module Temperature Monitor Threshold Register (TPMUTH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFF	Maximum allowed power module temperature code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed power module temperature code

Table 25-33: LIN PHY Temperature Monitor Threshold Register (TLINTH) Layout

TLINTH (LIN PHY Temperature Monitor Threshold Register) Offset: 0x3C Default: 0x0FFF0000 Access: SYSTEM->TLINTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-34: LIN PHY Temperature Monitor Threshold Register (TLINTH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFF	Maximum allowed LIN PHY temperature code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed LIN PHY temperature code

Table 25-35: Controller Temperature Monitor Threshold Register (TMCUTH) Layout

TMCUTH (Controller Temperature Monitor Threshold Register) Offset: 0x40 Default: 0x0FFF0000 Access: SYSTEM->TMCUTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-36: Controller Temperature Monitor Threshold Register (TMCUTH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFF	Maximum allowed controller temperature code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed controller temperature code

Table 25-37: VDD33 Monitor Threshold Register (VDD33TH) Layout

VDD33TH (VDD33 Monitor Threshold Register) Offset: 0x44 Default: 0x0FFF0000 Access: SYSTEM->VDD33TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-38: VDD33 Monitor Threshold Register (VDD33TH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFF	Maximum allowed VDD33 monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed VDD33 monitor code

Table 25-39: VDD12 Monitor Threshold Register (VDD12TH) Layout

VDD12TH (VDD12 Monitor Threshold Register) Offset: 0x48 Default: 0x0FFF0000 Access: SYSTEM->VDD12TH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-40: VDD12 Monitor Threshold Register (VDD12TH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFFF	Maximum allowed VDD12 monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed VDD12 monitor code

Table 25-41: 13-bit ADC Reference Voltage Monitor Threshold Register (VADCREFTH) Layout

VADCREFTH (13-bit ADC Reference Voltage Monitor Threshold Register) Offset: 0x4C Default: 0x0FFF0000 Access: SYSTEM->VADCREFTH							
31	30	29	28	27	26	25	24
RESERVED				MAX			
23	22	21	20	19	18	17	16
MAX							
15	14	13	12	11	10	9	8
RESERVED				MIN			
7	6	5	4	3	2	1	0
MIN							

Table 25-42: 13-bit ADC Reference Voltage Monitor Threshold Register (VADCREFTH) Description

Bits	Field Name	Type	Reset	Description
31:28	RESERVED_31_28	RO	0x0	Reserved.
27:16	MAX	RW	0xFFFF	Maximum allowed 13-bit ADC reference monitor code
15:12	RESERVED_15_12	RO	0x0	Reserved.
11:0	MIN	RW	0x0	Minimum allowed 13-bit ADC reference monitor code

Table 25-43: VBAT Monitor Code Register (VBATCODE) Layout

VBATCODE (VBAT Monitor Code Register) Offset: 0x50 Default: 0x00000000 Access: SYSTEM->VBATCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-44: VBAT Monitor Code Register (VBATCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured VBAT code

Table 25-45: VCP Monitor Code Register (VCPCODE) Layout

VCPCODE (VCP Monitor Code Register) Offset: 0x54 Default: 0x00000000 Access: SYSTEM->VCPCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-46: VCP Monitor Code Register (VCPCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured VCP code

Table 25-47: VMON Monitor Code Register (VMONCODE) Layout

VMONCODE (VMON Monitor Code Register) Offset: 0x58 Default: 0x00000000 Access: SYSTEM->VMONCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-48: VMON Monitor Code Register (VMONCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured VMON code

Table 25-49: VDD5 Monitor Code Register (VDD5CODE) Layout

VDD5CODE (VDD5 Monitor Code Register) Offset: 0x5C Default: 0x00000000 Access: SYSTEM->VDD5CODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-50: VDD5 Monitor Code Register (VDD5CODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured VDD5 code

Table 25-51: Power Bandgap Monitor Code Register (VBGCODE) Layout

VBGCODE (Power Bandgap Monitor Code Register) Offset: 0x60 Default: 0x00000000 Access: SYSTEM->VBGCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-52: Power Bandgap Monitor Code Register (VBGCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured power bandgap code

Table 25-53: Power Module Temperature Monitor Code Register (TPMUCODE) Layout

TPMUCODE (Power Module Temperature Monitor Code Register) Offset: 0x64 Default: 0x00000000 Access: SYSTEM->TPMUCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-54: Power Module Temperature Monitor Code Register (TPMUCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured power module temperature code

Table 25-55: LIN PHY Temperature Monitor Code Register (TLINCODE) Layout

TLINCODE (LIN PHY Temperature Monitor Code Register) Offset: 0x68 Default: 0x00000000 Access: SYSTEM->TLINCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-56: LIN PHY Temperature Monitor Code Register (TLINCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured LIN PHY temperature code

Table 25-57: Controller Temperature Monitor Code Register (TMCUCODE) Layout

TMCUCODE (Controller Temperature Monitor Code Register) Offset: 0x6C Default: 0x00000000 Access: SYSTEM->TMCUCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-58: Controller Temperature Monitor Code Register (TMCUCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured controller temperature code

Table 25-59: VDD33 Monitor Code Register (VDD33CODE) Layout

VDD33CODE (VDD33 Monitor Code Register) Offset: 0x70 Default: 0x00000000 Access: SYSTEM->VDD33CODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-60: VDD33 Monitor Code Register (VDD33CODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured VDD33 code

Table 25-61: VDD12 Monitor Threshold Register (VDD12CODE) Layout

VDD12CODE (VDD12 Monitor Threshold Register) Offset: 0x74 Default: 0x00000000 Access: SYSTEM->VDD12CODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-62: VDD12 Monitor Threshold Register (VDD12CODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured VDD12 code

Table 25-63: 13-bit ADC Reference Voltage Monitor Code Register (VADCREFCODE) Layout

VADCREFCODE (13-bit ADC Reference Voltage Monitor Code Register) Offset: 0x78 Default: 0x00000000 Access: SYSTEM->VADCREFCODE							
31	30	29	28	27	26	25	24
CODE							
23	22	21	20	19	18	17	16
CODE							
15	14	13	12	11	10	9	8
CODE							
7	6	5	4	3	2	1	0
CODE							

Table 25-64: 13-bit ADC Reference Voltage Monitor Code Register (VADCREFCODE) Description

Bits	Field Name	Type	Reset	Description
31:0	CODE	RO	0x0	Measured 13-bit ADC reference code

Table 25-65: GPIO Wake-Up Control Register (GPIOWKUPCTL) Layout

GPIOWKUPCTL (GPIO Wake-Up Control Register) Offset: 0x7C Default: 0x00010001 Access: SYSTEM->GPIOWKUPCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED					SLEEPWKUPWE	SLEEPWKUIPIO	SLEEPWKUPPOL
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
STOPWKUPWE	STOPWKUIPIO						STOPWKUPPOL

Table 25-66: GPIO Wake-Up Control Register (GPIOWKUPCTL) Description

Bits	Field Name	Type	Reset	Description
31:19	RESERVED_31_19	RO	0x0	Reserved.
18	SLEEPWKUPWE	W1S	0x0	GPIO wake up from sleep state write enable This bit should be write 1 at the same time when write SLEEPWKUIPIO and SLEEPWKUPPOL bits 0: Disable 1: Enable
17	SLEEPWKUIPIO	RW	0x0	GPIO number to wake up from sleep state 0: Use GPIO28 to wake up the MCU from sleep 1: Use GPIO4 to wake up the MCU from sleep
16	SLEEPWKUPPOL	RW	0x1	GPIO level to wake up from sleep state 0: Wake up from sleep when low 1: Wake up from sleep when high

Bits	Field Name	Type	Reset	Description
15:8	RESERVED_15_8	RO	0x0	Reserved.
7	STOPWKUPWE	W1S	0x0	GPIO wake up from stop state write enable This bit should be write 1 at the same time when write STOPWKUPIO and STOPWKUPPOL bits 0: Disable 1: Enable
6:1	STOPWKUPIO	RW	0x0	GPIO number to wake up from stop state
0	STOPWKUPPOL	RW	0x1	GPIO level to wake up from stop state 0: Wake up from stop when low 1: Wake up from stop when high

Table 25-67: GPIO Non-Maskable Interrupt Control Register (GPIONMCTL) Layout

GPIONMCTL (GPIO Non-Maskable Interrupt Control Register) Offset: 0x80 Default: 0x00000001 Access: SYSTEM->GPIONMCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED	IOSEL						POL

Table 25-68: GPIO Non-Maskable Interrupt Control Register (GPIONMCTL) Description

Bits	Field Name	Type	Reset	Description
31:7	RESERVED_31_7	RO	0x0	Reserved.
6:1	IOSEL	RW	0x0	GPIO number to trigger the NMI
0	POL	RW	0x1	GPIO edge to trigger the NMI 0: Falling edge 1: Rising edge

Table 25-69: Non-Maskable Interrupt Raw Flag Register (NMRAWIF) Layout

NMRAWIF (Non-Maskable Interrupt Raw Flag Register) Offset: 0x84 Default: 0x00000000 Access: SYSTEM->NMRAWIF							
31	30	29	28	27	26	25	24
RESERVED						PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFerr	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-70: Non-Maskable Interrupt Raw Flag Register (NMRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25	PTM	RO	0x0	Latched production test mode entry flag 0: Event does not occur 1: Event occurred
24	DEBUG	RO	0x0	Latched debug mode entry flag 0: Event does not occur 1: Event occurred
23	AHBMBE	RO	0x0	Latched AHB address or data ECC multi-bit error flag 0: Event does not occur 1: Event occurred
22	VBATERR	RO	0x0	Latched VBAT monitor error flag 0: Event does not occur 1: Event occurred
21	VCPERR	RO	0x0	Latched VCP monitor error flag 0: Event does not occur 1: Event occurred
20	VMONERR	RO	0x0	Latched VMON monitor error flag 0: Event does not occur 1: Event occurred
19	VDD5ERR	RO	0x0	Latched VDD5 monitor error flag 0: Event does not occur 1: Event occurred
18	VBGERR	RO	0x0	Latched power bandgap monitor error flag 0: Event does not occur 1: Event occurred
17	TPMUERR	RO	0x0	Latched power module temperature monitor error flag 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
16	TLINERR	RO	0x0	Latched LIN PHY temperature monitor error flag 0: Event does not occur 1: Event occurred
15	TMCUERR	RO	0x0	Latched controller temperature monitor error flag 0: Event does not occur 1: Event occurred
14	VDD33ERR	RO	0x0	Latched VDD33 monitor error flag 0: Event does not occur 1: Event occurred
13	VDD12ERR	RO	0x0	Latched VDD12 monitor error flag 0: Event does not occur 1: Event occurred
12	VADCREFERR	RO	0x0	Latched 13-bit ADC reference monitor error flag 0: Event does not occur 1: Event occurred
11	GPIONMI	RO	0x0	Latched GPIO non-maskable interrupt event flag 0: Event does not occur 1: Event occurred
10	WDT0	RO	0x0	Latched WDT0 timeout flag 0: Event does not occur 1: Event occurred
9	RAM1MBE	RO	0x0	Latched RAM1 ECC multi-bit error flag 0: Event does not occur 1: Event occurred
8	RAM0MBE	RO	0x0	Latched RAM0 ECC multi-bit error flag 0: Event does not occur 1: Event occurred
7	FLASHMBE	RO	0x0	Latched FLASH ECC multi-bit error flag 0: Event does not occur 1: Event occurred
6	ROMMBE	RO	0x0	Latched ROM ECC multi-bit error flag 0: Event does not occur 1: Event occurred
5	CLKDETERR	RO	0x0	Latched CLKDET error event flag It means CLKDETCNT is out of the range defined by CLKDETCTH 0: Event does not occur 1: Event occurred
4	PLLUNLOCK	RO	0x0	Latched PLL unlock event flag Detailed reason can be referred to CLKSTS[PLLFAST] and CLKSTS[PLLSLOW] 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
3	VDD33OV	RO	0x0	Latched VDD33 over-voltage event flag 0: Event does not occur 1: Event occurred
2	VDD33UV	RO	0x0	Latched VDD33 under-voltage event flag 0: Event does not occur 1: Event occurred
1	VDD12OV	RO	0x0	Latched VDD12 over-voltage event flag 0: Event does not occur 1: Event occurred
0	VDD12UV	RO	0x0	Latched VDD12 under-voltage event flag 0: Event does not occur 1: Event occurred

Table 25-71: Non-Maskable Interrupt Flag Register (NMIF) Layout

NMIF (Non-Maskable Interrupt Flag Register) Offset: 0x88 Default: 0x00000000 Access: SYSTEM->NMIF							
31	30	29	28	27	26	25	24
RESERVED					INT	PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFerr	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-72: Non-Maskable Interrupt Flag Register (NMIF) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	INT	RO	0x0	Global non-maskable interrupt flag 0: Non-maskable interrupt does not occur 1: Non-maskable interrupt occurred and issued to the CPU. Further interrupt will not be generated until this bit is cleared.
25	PTM	RO	0x0	Latched production test mode entry interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
24	DEBUG	RO	0x0	Latched debug mode entry interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Bits	Field Name	Type	Reset	Description
23	AHBMBE	RO	0x0	Latched AHB address or data ECC multi-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
22	VBATERR	RO	0x0	Latched VBAT monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
21	VCPERR	RO	0x0	Latched VCP monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
20	VMONERR	RO	0x0	Latched VMON monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
19	VDD5ERR	RO	0x0	Latched VDD5 monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
18	VBGERR	RO	0x0	Latched power bandgap monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
17	TPMUERR	RO	0x0	Latched power module temperature monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
16	TLINERR	RO	0x0	Latched LIN PHY temperature monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
15	TMCUERR	RO	0x0	Latched controller temperature monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Bits	Field Name	Type	Reset	Description
14	VDD33ERR	RO	0x0	Latched VDD33 monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
13	VDD12ERR	RO	0x0	Latched VDD12 monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
12	VADCREFERR	RO	0x0	Latched 13-bit ADC reference monitor error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
11	GPIONMI	RO	0x0	Latched GPIO non-maskable interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
10	WDT0	RO	0x0	Latched WDT0 interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
9	RAM1MBE	RO	0x0	Latched RAM1 ECC multi-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
8	RAM0MBE	RO	0x0	Latched RAM0 ECC multi-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
7	FLASHMBE	RO	0x0	Latched FLASH ECC multi-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
6	ROMMBE	RO	0x0	Latched ROM ECC multi-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
5	CLKDETERR	RO	0x0	Latched CLKDET error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Bits	Field Name	Type	Reset	Description
4	PLLUNLOCK	RO	0x0	Latched PLL unlock interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
3	VDD33OV	RO	0x0	Latched VDD33 over-voltage interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
2	VDD33UV	RO	0x0	Latched VDD33 under-voltage interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
1	VDD12OV	RO	0x0	Latched VDD12 over-voltage interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
0	VDD12UV	RO	0x0	Latched VDD12 under-voltage interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Table 25-73: Non-Maskable Interrupt Clear Register (NMIC) Layout

NMIC (Non-Maskable Interrupt Clear Register) Offset: 0x8C Default: 0x00000000 Access: SYSTEM->NMIC							
31	30	29	28	27	26	25	24
RESERVED					INT	PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFFERR	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-74: Non-Maskable Interrupt Clear Register (NMIC) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	INT	W1C	0x0	Global non-maskable interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
25	PTM	W1C	0x0	Latched production test mode entry interrupt clear

Bits	Field Name	Type	Reset	Description
				0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
24	DEBUG	RW	0x0	Latched debug mode entry interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
23	AHBMBE	W1C	0x0	AHB address or data ECC multi-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
22	VBATERR	W1C	0x0	VBAT monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
21	VCPERR	W1C	0x0	VCP monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
20	VMONERR	W1C	0x0	VMON monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
19	VDD5ERR	W1C	0x0	VDD5 monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
18	VBGERR	W1C	0x0	Power bandgap monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
17	TPMUERR	W1C	0x0	Power module temperature monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
16	TLINERR	W1C	0x0	LIN PHY temperature monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
15	TMCUERR	W1C	0x0	Controller temperature monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
14	VDD33ERR	W1C	0x0	VDD33 monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
13	VDD12ERR	W1C	0x0	VDD12 monitor error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
12	VADCREFerr	W1C	0x0	13-bit ADC reference monitor error interrupt clear

Bits	Field Name	Type	Reset	Description
				0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
11	GPIONMI	W1C	0x0	GPIO non-maskable interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
10	WDT0	W1C	0x0	WDT0 interrupt clear This bit is equivalent to WDTIC register bit of WDT0 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
9	RAM1MBE	W1C	0x0	RAM1 ECC multi-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
8	RAM0MBE	W1C	0x0	RAM0 ECC multi-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
7	FLASHMBE	W1C	0x0	FLASH ECC multi-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	ROMMBE	W1C	0x0	ROM ECC multi-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	CLKDETERR	W1C	0x0	CLKDET error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
4	PLLUNLOCK	W1C	0x0	PLL unlock interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	VDD33OV	W1C	0x0	VDD33 over-voltage interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	VDD33UV	W1C	0x0	VDD33 under-voltage interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	VDD12OV	W1C	0x0	VDD12 over-voltage interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	VDD12UV	W1C	0x0	VDD12 under-voltage interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 25-75: Non-Maskable Interrupt Enable Register (NMIE) Layout

NMIE (Non-Maskable Interrupt Enable Register) Offset: 0x90 Default: 0x00000400 Access: SYSTEM->NMIE							
31	30	29	28	27	26	25	24
RESERVED						PTM	DEBUG
23	22	21	20	19	18	17	16
AHBMBE	VBATERR	VCPERR	VMONERR	VDD5ERR	VBGERR	TPMUERR	TLINERR
15	14	13	12	11	10	9	8
TMCUERR	VDD33ERR	VDD12ERR	VADCREFerr	GPIONMI	WDT0	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-76: Non-Maskable Interrupt Enable Register (NMIE) Description

Bits	Field Name	Type	Reset	Description
31:26	RESERVED_31_26	RO	0x0	Reserved.
25	PTM	RW	0x0	Production test mode entry interrupt enable 0: Disable 1: Enable
24	DEBUG	RW	0x0	Debug mode entry interrupt enable 0: Disable 1: Enable
23	AHBMBE	RW	0x0	AHB address or data ECC multi-bit error interrupt enable 0: Disable 1: Enable
22	VBATERR	RW	0x0	VBAT monitor error interrupt enable 0: Disable 1: Enable
21	VCPERR	RW	0x0	VCP monitor error interrupt enable 0: Disable 1: Enable
20	VMONERR	RW	0x0	VMON monitor error interrupt enable 0: Disable 1: Enable
19	VDD5ERR	RW	0x0	VDD5 monitor error interrupt enable 0: Disable 1: Enable
18	VBGERR	RW	0x0	Power bandgap monitor error interrupt enable 0: Disable 1: Enable
17	TPMUERR	RW	0x0	Power module temperature monitor error interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
16	TLINERR	RW	0x0	LIN PHY temperature monitor error interrupt enable 0: Disable 1: Enable
15	TMCUERR	RW	0x0	Controller temperature monitor error interrupt enable 0: Disable 1: Enable
14	VDD33ERR	RW	0x0	VDD33 monitor error interrupt enable 0: Disable 1: Enable
13	VDD12ERR	RW	0x0	VDD12 monitor error interrupt enable 0: Disable 1: Enable
12	VADCREFERR	RW	0x0	13-bit ADC reference monitor error interrupt enable 0: Disable 1: Enable
11	GPIONMI	RW	0x0	GPIO non-maskable interrupt enable 0: Disable 1: Enable
10	WDT0	RW	0x1	WDT0 interrupt enable This bit is equivalent to WDTIE register bit of WDT0 0: Disable 1: Enable
9	RAM1MBE	RW	0x0	RAM1 ECC multi-bit error interrupt enable 0: Disable 1: Enable
8	RAM0MBE	RW	0x0	RAM0 ECC multi-bit error interrupt enable 0: Disable 1: Enable
7	FLASHMBE	RW	0x0	FLASH ECC multi-bit error interrupt enable 0: Disable 1: Enable
6	ROMMBE	RW	0x0	ROM ECC multi-bit error interrupt enable 0: Disable 1: Enable
5	CLKDETERR	RW	0x0	CLKDET error interrupt enable 0: Disable 1: Enable
4	PLLUNLOCK	RW	0x0	PLL unlock interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
3	VDD33OV	RW	0x0	VDD33 over-voltage interrupt enable 0: Disable 1: Enable
2	VDD33UV	RW	0x0	VDD33 under-voltage interrupt enable 0: Disable 1: Enable
1	VDD12OV	RW	0x0	VDD12 over-voltage interrupt enable 0: Disable 1: Enable
0	VDD12UV	RW	0x0	VDD12 under-voltage interrupt enable 0: Disable 1: Enable

Table 25-77: Memory Interrupt Raw Flag Register (MEMRAWIF) Layout

MEMRAWIF (Memory Interrupt Raw Flag Register) Offset: 0x94 Default: 0x00000000 Access: SYSTEM->MEMRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				RAM1SBE	RAM0SBE	FLASHSBE	ROMSBE

Table 25-78: Memory Interrupt Raw Flag Register (MEMRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	RAM1SBE	RO	0x0	Latched RAM1 single-bit error event flag 0: Event does not occur 1: Event occurred
2	RAM0SBE	RO	0x0	Latched RAM0 single-bit error event flag 0: Event does not occur 1: Event occurred
1	FLASHSBE	RO	0x0	Latched FLASH single-bit error event flag 0: Event does not occur 1: Event occurred
0	ROMSBE	RO	0x0	Latched ROM single-bit error event flag 0: Event does not occur 1: Event occurred

Table 25-79: Memory Interrupt Flag Register (MEMIF) Layout

MEMIF (Memory Interrupt Flag Register) Offset: 0x98 Default: 0x00000000 Access: SYSTEM->MEMIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			INT	RAM1SBE	RAM0SBE	FLASHSBE	ROMSBE

Table 25-80: Memory Interrupt Flag Register (MEMIF) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4	INT	RO	0x0	Global memory error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further interrupt will not be generated until this bit is cleared
3	RAM1SBE	RO	0x0	Latched ECC RAM1 single-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
2	RAM0SBE	RO	0x0	Latched ECC RAM0 single-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
1	FLASHSBE	RO	0x0	Latched ECC FLASH single-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared
0	ROMSBE	RO	0x0	Latched ECC ROM single-bit error interrupt flag 0: Interrupt does not occur 1: Interrupt occurred. Further event will not trigger interrupt until this bit is cleared

Table 25-81: Memory Interrupt Clear Register (MEMIC) Layout

MEMIC (Memory Interrupt Clear Register) Offset: 0x9C Default: 0x00000000 Access: SYSTEM->MEMIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			INT	RAM1SBE	RAMOSBE	FLASHSBE	ROMSBE

Table 25-82: Memory Interrupt Clear Register (MEMIC) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4	INT	W1C	0x0	Global memory error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	RAM1SBE	W1C	0x0	RAM1 ECC single-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	RAMOSBE	W1C	0x0	RAM0 ECC single-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	FLASHSBE	W1C	0x0	FLASH ECC single-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	ROMSBE	W1C	0x0	ROM ECC single-bit error interrupt clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 25-83: Memory Interrupt Enable Register (MEMIE) Layout

MEMIE (Memory Interrupt Enable Register) Offset: 0xA0 Default: 0x00000000 Access: SYSTEM->MEMIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED				RAM1SBE	RAMOSBE	FLASHSBE	ROMSBE

Table 25-84: Memory Interrupt Enable Register (MEMIE) Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	RAM1SBE	RW	0x0	RAM1 ECC single-bit error interrupt enable 0: Disable 1: Enable
2	RAM0SBE	RW	0x0	RAM0 ECC single-bit error interrupt enable 0: Disable 1: Enable
1	FLASHSBE	RW	0x0	FLASH ECC single-bit error interrupt enable 0: Disable 1: Enable
0	ROMSBE	RW	0x0	ROM ECC single-bit error interrupt enable 0: Disable 1: Enable

Table 25-85: Reset Event Flag Register (RSTEVTF LG) Layout

RSTEVTF LG (Reset Event Flag Register) Offset: 0xA4 Default: 0x00000000 Access: SYSTEM->RSTEVTF LG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		PINRST	SYSRST	WDT1RST	WDTORST	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETER	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-86: Reset Event Flag Register (RSTEVTF LG) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13	PINRST	RO	0x0	Latched reset flag upon XRSTn pin reset event 0: Specified reset not happen 1: Specified reset happened
12	SYSRST	RO	0x0	Latched reset flag upon system reset event 0: Specified reset not happen 1: Specified reset happened
11	WDT1RST	RO	0x0	Latched reset flag upon WDT1 reset event 0: Specified reset not happen 1: Specified reset happened
10	WDTORST	RO	0x0	Latched reset flag upon WDT0 reset event 0: Specified reset not happen 1: Specified reset happened

Bits	Field Name	Type	Reset	Description
9	RAM1MBE	RO	0x0	Latched reset flag upon RAM1 multi-bit error event 0: Specified reset does not happen 1: Specified reset happened
8	RAM0MBE	RO	0x0	Latched reset flag upon RAM0 multi-bit error event 0: Specified reset does not happen 1: Specified reset happened
7	FLASHMBE	RO	0x0	Latched reset flag upon FLASH multi-bit error event 0: Specified reset does not happen 1: Specified reset happened
6	ROMMBE	RO	0x0	Latched reset flag upon ROM multi-bit error event 0: Specified reset does not happen 1: Specified reset happened
5	CLKDETERR	RO	0x0	Latched reset flag upon CLKDETERR event 0: Specified reset not happen 1: Specified reset happened
4	PLLUNLOCK	RO	0x0	Latched reset flag upon PLL unlock event 0: Specified reset not happen 1: Specified reset happened
3	VDD33OV	RO	0x0	Latched reset flag upon VDD33OV event 0: Specified reset does not happen 1: Specified reset happened
2	VDD33UV	RO	0x0	Latched reset flag upon VDD33UV event 0: Specified reset does not happen 1: Specified reset happened
1	VDD12OV	RO	0x0	Latched reset flag upon VDD12OV event 0: Specified reset does not happen 1: Specified reset happened
0	VDD12UV	RO	0x0	Latched reset flag upon VDD12UV event 0: Specified reset does not happen 1: Specified reset happened

Table 25-87: Reset Event Flag Clear Register (RSTEVTCLR) Layout

RSTEVTCLR (Reset Event Flag Clear Register) Offset: 0xA8 Default: 0x00000000 Access: SYSTEM->RSTEVTCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED		PINRST	SYSRST	WDT1RST	WDTORST	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-88: Reset Event Flag Clear Register (RSTEVTCLR) Description

Bits	Field Name	Type	Reset	Description
31:14	RESERVED_31_14	RO	0x0	Reserved.
13	PINRST	W1C	0x0	Clear reset flag upon XRSTn pin reset event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
12	SYSRST	W1C	0x0	Clear reset flag upon system reset event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
11	WDT1RST	W1C	0x0	Clear reset flag upon WDT1 reset event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
10	WDTORST	W1C	0x0	Clear reset flag upon WDT0 reset event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
9	RAM1MBE	W1C	0x0	Clear reset flag upon RAM1 multi-bit error event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
8	RAM0MBE	W1C	0x0	Clear reset flag upon RAM0 multi-bit error event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
7	FLASHMBE	W1C	0x0	Clear reset flag upon FLASH multi-bit error event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	ROMMBE	W1C	0x0	Clear reset flag upon ROM multi-bit error event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	CLKDETERR	W1C	0x0	Clear reset flag upon CLKDETERR event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
4	PLLUNLOCK	W1C	0x0	Clear reset flag upon PLL unlock event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	VDD33OV	W1C	0x0	Clear reset flag upon VDD33OV event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	VDD33UV	W1C	0x0	Clear reset flag upon VDD33UV event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	VDD12OV	W1C	0x0	Clear reset flag upon VDD12OV event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	VDD12UV	W1C	0x0	Clear reset flag upon VDD12UV event 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 25-89: Reset Event Enable Register (RSTEV TEN) Layout

RSTEV TEN (Reset Event Enable Register) Offset: 0xAC Default: 0x000000C00 Access: SYSTEM->RSTEV TEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED				WDT1RST	WDTORST	RAM1MBE	RAM0MBE
7	6	5	4	3	2	1	0
FLASHMBE	ROMMBE	CLKDETERR	PLLUNLOCK	VDD33OV	VDD33UV	VDD12OV	VDD12UV

Table 25-90: Reset Event Enable Register (RSTEV TEN) Description

Bits	Field Name	Type	Reset	Description
31:12	RESERVED_31_12	RO	0x0	Reserved.
11	WDT1RST	RW	0x1	Enable reset upon WDT1 reset request 0: Disable 1: Enable
10	WDTORST	RW	0x1	Enable reset upon WDT0 reset request 0: Disable 1: Enable
9	RAM1MBE	RW	0x0	Enable reset upon RAM1 multi-bit error event 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
8	RAM0MBE	RW	0x0	Enable reset upon RAM0 multi-bit error event 0: Disable 1: Enable
7	FLASHMBE	RW	0x0	Enable reset upon FLASH multi-bit error event 0: Disable 1: Enable
6	ROMMBE	RW	0x0	Enable reset upon ROM multi-bit error event 0: Disable 1: Enable
5	CLKDETERR	RW	0x0	Enable reset upon CLKDETERR event 0: Disable 1: Enable
4	PLLUNLOCK	RW	0x0	Enable reset upon PLL unlock event 0: Disable 1: Enable
3	VDD33OV	RW	0x0	Enable reset upon VDD33OV event 0: Disable 1: Enable
2	VDD33UV	RW	0x0	Enable reset upon VDD33UV event 0: Disable 1: Enable
1	VDD12OV	RW	0x0	Enable reset upon VDD12OV event 0: Disable 1: Enable
0	VDD12UV	RW	0x0	Enable reset upon VDD12UV event 0: Disable 1: Enable

Table 25-91: System Information Register (SYSINFO) Layout

SYSINFO (System Information Register) Offset: 0xB0 Default: 0x000001FF Access: SYSTEM->SYSINFO							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							ADCBIT
7	6	5	4	3	2	1	0
ADCBIT	CLKOPT		RAMOSIZE		RESERVED		ROMSIZE

Table 25-92: System Information Register (SYSINFO) Description

Bits	Field Name	Type	Reset	Description
31:9	RESERVED_31_9	RO	0x0	Reserved.
8:7	ADCBIT	RO	0x3	ADC bit resolution 00: 10 bit 01: 11 bit 10: 12 bit 11: 13 bit
6:5	CLKOPT	RO	0x3	Clock option 00: Minimum PLL output dividing ratio is 12 and no XO 01: Minimum PLL output dividing ratio is 6 and no XO 10: Minimum PLL output dividing ratio is 12 11: Minimum PLL output dividing ratio is 6
4:3	RAMOSIZE	RO	0x3	RAM0 size 00: 4KB from 0x1FFFB000 to 0x1FFFBFFF or from 0x20003000 to 0x20003FFF 01: 8KB from 0x1FFFA000 to 0x1FFFBFFF or from 0x20002000 to 0x20003FFF 10: 12KB from 0x1FFF9000 to 0x1FFFBFFF or from 0x20001000 to 0x20003FFF 11: 16KB from 0x1FFF8000 to 0x1FFFBFFF or from 0x20000000 to 0x20003FFF
2:1	RESERVED_2_1	RO	0x3	Reserved.
0	ROMSIZE	RO	0x1	ROM size 0: Basic support 1: Full support with motor hardware libraries

Table 25-93: ROM ECC Error Address (ROMERRADDR) Layout

ROMERRADDR (ROM ECC Error Address) Offset: 0xB4 Default: 0xFFFFFFFF							
Access: SYSTEM->ROMERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-94: ROM ECC Error Address (ROMERRADDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0xFFFFFFFF	Address where ECC error happened for ROM access

Table 25-95: FLASH ECC Error Address (FLASHERRADDR) Layout

FLASHERRADDR (FLASH ECC Error Address) Offset: 0xB8 Default: 0xFFFFFFFF Access: SYSTEM->FLASHERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-96: FLASH ECC Error Address (FLASHERRADDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0xFFFFFFFF	Address where ECC error happened for FLASH access

Table 25-97: RAM0 ECC Error Address (RAM0ERRADDR) Layout

RAM0ERRADDR (RAM0 ECC Error Address) Offset: 0xBC Default: 0xFFFFFFFF Access: SYSTEM->RAM0ERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-98: RAM0 ECC Error Address (RAM0ERRADDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0xFFFFFFFF	Address where ECC error happened for RAM0 access

Table 25-99: RAM1 ECC Error Address (RAM1ERRADDR) Layout

RAM1ERRADDR (RAM1 ECC Error Address) Offset: 0xC0 Default: 0xFFFFFFFF Access: SYSTEM->RAM1ERRADDR							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-100: RAM1 ECC Error Address (RAM1ERRADDR) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RO	0xFFFFFFFF	Address where ECC error happened for RAM1 access

Table 25-101: Debug Interface Control Register (DBGIFCTL) Layout

DBGIFCTL (Debug Interface Control Register) Offset: 0xC8 Default: 0x00000000 Access: SYSTEM->DBGIFCTL							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							JTAGMODE

Table 25-102: Debug Interface Control Register (DBGIFCTL) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	JTAGMODE	RW	0x0	JTAG or SWD debug interface select 0: SWD mode 1: JTAG mode

Table 25-103: Boot Information Register (BOOTINFO) Layout

BOOTINFO (Boot Information Register) Offset: 0xCC Default: 0x00000000 Access: SYSTEM->BOOTINFO							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED				RESERVED	RESERVED	FLASHINITERR	RAMBISTERR
15	14	13	12	11	10	9	8
BOOTCNT							
7	6	5	4	3	2	1	0
BOOTCNT							

Table 25-104: Boot Information Register (BOOTINFO) Description

Bits	Field Name	Type	Reset	Description
31:20	RESERVED_31_20	RO	0x0	Reserved.
19	RESERVED	RW	0x0	Reserved.
18	RESERVED	RO	0x0	Reserved.
17	FLASHINITERR	RW	0x0	FLASH initialization error 0: FLASH initialization passed 1: FLASH initialization failed
16	RAMBISTERR	RW	0x0	RAM built-in self test error 0: RAM test passed 1: RAM test failed
15:0	BOOTCNT	RW	0x0	Boot counter, increased by 1 when each time boot code is executed

Table 25-105: AHB Multi-Bit ECC Error Event Flag Register (AHBMBEFLG) Layout

AHBMBEFLG (AHB Multi-Bit ECC Error Event Flag Register) Offset: 0xD0 Default: 0x00000000 Access: SYSTEM->AHBMBEFLG							
31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPIO	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

Table 25-106: AHB Multi-Bit ECC Error Event Flag Register (AHBMBEFLG) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	CRC	RO	0x0	CRC AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
25	ECAP	RO	0x0	ECAP AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
24	AFE	RO	0x0	AFE AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
23	PWM	RO	0x0	PWM AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
22	DMAC	RO	0x0	DMAC AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
21	FLASHC	RO	0x0	FLASHC AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
20	TIMER2	RO	0x0	TIMER2 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
19	TIMER1	RO	0x0	TIMER1 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
18	TIMER0	RO	0x0	TIMER0 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
17	CAN	RO	0x0	CAN AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
16	I2C	RO	0x0	I2C AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
15	SPI1	RO	0x0	SPI1 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
14	SPI0	RO	0x0	SPI0 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
13	UART1	RO	0x0	UART1 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
12	UART0	RO	0x0	UART0 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
11	GPIO	RO	0x0	GPIO AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
10	WDT1	RO	0x0	WDT1 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
9	WDT0	RO	0x0	WDT0 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
8	GLBREG	RO	0x0	Global control AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
7	RAM1	RO	0x0	RAM1 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
6	RAM0	RO	0x0	RAM0 AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
5	XIP	RO	0x0	XIP AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
4	ROM	RO	0x0	ROM AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
3	DMA	RO	0x0	DMA AHB port multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
2	SBUS	RO	0x0	S-bus multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
1	IBUS	RO	0x0	I-bus multi-bit ECC error event flag 0: Event does not occur 1: Event occurred
0	DBUS	RO	0x0	D-bus multi-bit ECC error event flag 0: Event does not occur 1: Event occurred

Table 25-107: AHB Multi-bit ECC Error Event Flag Register (AHBMBECLR) Layout

AHBMBECLR (AHB Multi-bit ECC Error Event Flag Register) Offset: 0xD4 Default: 0x00000000 Access: SYSTEM->AHBMBECLR							
31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

Table 25-108: AHB Multi-bit ECC Error Event Flag Register (AHBMBECLR) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	CRC	W1C	0x0	CRC AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
25	ECAP	W1C	0x0	ECAP AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
24	AFE	W1C	0x0	AFE AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
23	PWM	W1C	0x0	PWM AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
22	DMAC	W1C	0x0	DMAC AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
21	FLASHC	W1C	0x0	FLASHC AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
20	TIMER2	W1C	0x0	TIMER2 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
19	TIMER1	W1C	0x0	TIMER1 AHB port uncorrectable ECC error event flag clear

Bits	Field Name	Type	Reset	Description
				0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
18	TIMER0	W1C	0x0	TIMER0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
17	CAN	W1C	0x0	CAN AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
16	I2C	W1C	0x0	I2C AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
15	SPI1	W1C	0x0	SPI1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
14	SPI0	W1C	0x0	SPI0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
13	UART1	W1C	0x0	UART1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
12	UART0	W1C	0x0	UART0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
11	GPIO	W1C	0x0	GPIO AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
10	WDT1	W1C	0x0	WDT1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
9	WDT0	W1C	0x0	WDT0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
8	GLBREG	W1C	0x0	Global control AHB port uncorrectable ECC error event flag clear

Bits	Field Name	Type	Reset	Description
				0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
7	RAM1	W1C	0x0	RAM1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	RAM0	W1C	0x0	RAM0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	XIP	W1C	0x0	XIP AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
4	ROM	W1C	0x0	ROM AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	DMA	W1C	0x0	DMA AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	SBUS	W1C	0x0	S-bus uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	IBUS	W1C	0x0	I-bus uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	DBUS	W1C	0x0	D-bus uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 25-109: AHB Single-Bit ECC Error Event Flag Register (AHBSBEFLG) Layout

AHBSBEFLG (AHB Single-Bit ECC Error Event Flag Register) Offset: 0xD8 Default: 0x00000000 Access: SYSTEM->AHBSBEFLG							
31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

Table 25-110: AHB Single-Bit ECC Error Event Flag Register (AHBSBEFLG) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	CRC	RO	0x0	CRC AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
25	ECAP	RO	0x0	ECAP AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
24	AFE	RO	0x0	AFE AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
23	PWM	RO	0x0	PWM AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
22	DMAC	RO	0x0	DMAC AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
21	FLASHC	RO	0x0	FLASHC AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
20	TIMER2	RO	0x0	TIMER2 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
19	TIMER1	RO	0x0	TIMER1 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
18	TIMER0	RO	0x0	TIMER0 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
17	CAN	RO	0x0	CAN AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
16	I2C	RO	0x0	I2C AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
15	SPI1	RO	0x0	SPI1 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
14	SPI0	RO	0x0	SPI0 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
13	UART1	RO	0x0	UART1 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
12	UART0	RO	0x0	UART0 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
11	GPIO	RO	0x0	GPIO AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
10	WDT1	RO	0x0	WDT1 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
9	WDT0	RO	0x0	WDT0 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
8	GLBREG	RO	0x0	Global control AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
7	RAM1	RO	0x0	RAM1 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
6	RAM0	RO	0x0	RAM0 AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
5	XIP	RO	0x0	XIP AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
4	ROM	RO	0x0	ROM AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
3	DMA	RO	0x0	DMA AHB port Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
2	SBUS	RO	0x0	S-bus Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
1	IBUS	RO	0x0	I-bus Single-bit ECC error event flag 0: Event does not occur 1: Event occurred
0	DBUS	RO	0x0	D-bus Single-bit ECC error event flag 0: Event does not occur 1: Event occurred

Table 25-111: AHB Single-bit ECC Error Event Flag Register (AHBSBECLR) Layout

AHBSBECLR (AHB Single-bit ECC Error Event Flag Register) Offset: 0xDC Default: 0x00000000 Access: SYSTEM->AHBSBECLR							
31	30	29	28	27	26	25	24
RESERVED					CRC	ECAP	AFE
23	22	21	20	19	18	17	16
PWM	DMAC	FLASHC	TIMER2	TIMER1	TIMER0	CAN	I2C
15	14	13	12	11	10	9	8
SPI1	SPI0	UART1	UART0	GPIO	WDT1	WDT0	GLBREG
7	6	5	4	3	2	1	0
RAM1	RAM0	XIP	ROM	DMA	SBUS	IBUS	DBUS

Table 25-112: AHB Single-bit ECC Error Event Flag Register (AHBSBECLR) Description

Bits	Field Name	Type	Reset	Description
31:27	RESERVED_31_27	RO	0x0	Reserved.
26	CRC	W1C	0x0	CRC AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
25	ECAP	W1C	0x0	ECAP AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
24	AFE	W1C	0x0	AFE AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
23	PWM	W1C	0x0	PWM AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
22	DMAC	W1C	0x0	DMAC AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
21	FLASHC	W1C	0x0	FLASHC AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
20	TIMER2	W1C	0x0	TIMER2 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
19	TIMER1	W1C	0x0	TIMER1 AHB port uncorrectable ECC error event flag clear

Bits	Field Name	Type	Reset	Description
				0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
18	TIMER0	W1C	0x0	TIMER0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
17	CAN	W1C	0x0	CAN AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
16	I2C	W1C	0x0	I2C AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
15	SPI1	W1C	0x0	SPI1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
14	SPI0	W1C	0x0	SPI0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
13	UART1	W1C	0x0	UART1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
12	UART0	W1C	0x0	UART0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
11	GPIO	W1C	0x0	GPIO AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
10	WDT1	W1C	0x0	WDT1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
9	WDT0	W1C	0x0	WDT0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
8	GLBREG	W1C	0x0	Global control AHB port uncorrectable ECC error event flag clear

Bits	Field Name	Type	Reset	Description
				0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
7	RAM1	W1C	0x0	RAM1 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	RAM0	W1C	0x0	RAM0 AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	XIP	W1C	0x0	XIP AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
4	ROM	W1C	0x0	ROM AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	DMA	W1C	0x0	DMA AHB port uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	SBUS	W1C	0x0	S-bus uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	IBUS	W1C	0x0	I-bus uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	DBUS	W1C	0x0	D-bus uncorrectable ECC error event flag clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 25-113: Peripheral Reset Event Flag Register (PERIRSTFLG) Layout

PERIRSTFLG (Peripheral Reset Event Flag Register) Offset: 0xE0 Default: 0x0000FFFF Access: SYSTEM->PERIRSTFLG							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
WDT1	WDT0	DG	AHB	CAN	I2C	SPI1	SPI0
7	6	5	4	3	2	1	0
UART1	UART0	TIMER2	TIMER1	TIMER0	ECAP	PWM	ADC

Table 25-114: Peripheral Reset Event Flag Register (PERIRSTFLG) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	WDT1	RO	0x1	Watchdog 1 latched reset event flag 0: Event does not occur 1: Event occurred
14	WDT0	RO	0x1	Watchdog 0 latched reset event flag 0: Event does not occur 1: Event occurred
13	DG	RO	0x1	Deglitch module latched reset event flag 0: Event does not occur 1: Event occurred
12	AHB	RO	0x1	Bus and CPU latched reset event flag 0: Event does not occur 1: Event occurred
11	CAN	RO	0x1	CAN module latched reset event flag 0: Event does not occur 1: Event occurred
10	I2C	RO	0x1	I2C module latched reset event flag 0: Event does not occur 1: Event occurred
9	SPI1	RO	0x1	SPI1 module latched reset event flag 0: Event does not occur 1: Event occurred
8	SPI0	RO	0x1	SPI0 module latched reset event flag 0: Event does not occur 1: Event occurred
7	UART1	RO	0x1	UART1 module latched reset event flag 0: Event does not occur 1: Event occurred
6	UART0	RO	0x1	UART0 module latched reset event flag 0: Event does not occur 1: Event occurred
5	TIMER2	RO	0x1	TIMER2 module latched reset event flag 0: Event does not occur 1: Event occurred
4	TIMER1	RO	0x1	TIMER1 module latched reset event flag 0: Event does not occur 1: Event occurred
3	TIMER0	RO	0x1	TIMER0 module latched reset event flag 0: Event does not occur 1: Event occurred

Bits	Field Name	Type	Reset	Description
2	ECAP	RO	0x1	ECAP module latched reset event flag 0: Event does not occur 1: Event occurred
1	PWM	RO	0x1	PWM module latched reset event flag 0: Event does not occur 1: Event occurred
0	ADC	RO	0x1	ADC module latched reset event flag 0: Event does not occur 1: Event occurred

Table 25-115: Peripheral Reset Event Clear Register (PERIRSTCLR) Layout

PERIRSTCLR (Peripheral Reset Event Clear Register) Offset: 0xE4 Default: 0x00000000 Access: SYSTEM->PERIRSTCLR							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 WDT1	14 WDT0	13 DG	12 AHB	11 CAN	10 I2C	9 SPI1	8 SPI0
7 UART1	6 UART0	5 TIMER2	4 TIMER1	3 TIMER0	2 ECAP	1 PWM	0 ADC

Table 25-116: Peripheral Reset Event Clear Register (PERIRSTCLR) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	WDT1	W1C	0x0	Watchdog 1 reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
14	WDT0	W1C	0x0	Watchdog 0 reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
13	DG	W1C	0x0	Deglintch module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
12	AHB	W1C	0x0	Bus and CPU reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
11	CAN	W1C	0x0	CAN module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
10	I2C	W1C	0x0	I2C module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Bits	Field Name	Type	Reset	Description
9	SPI1	W1C	0x0	SPI1 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
8	SPI0	W1C	0x0	SPI0 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
7	UART1	W1C	0x0	UART1 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
6	UART0	W1C	0x0	UART0 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
5	TIMER2	W1C	0x0	TIMER2 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
4	TIMER1	W1C	0x0	TIMER1 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
3	TIMER0	W1C	0x0	TIMER0 module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
2	ECAP	W1C	0x0	ECAP module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
1	PWM	W1C	0x0	PWM module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.
0	ADC	W1C	0x0	ADC module reset event clear 0: Write 0 has no effect and always reads back 0 1: Write 1 clears the flag. This bit is self-cleared.

Table 25-117: Non-Volatile Register 0 (NVREG0) Layout

NVREG0 (Non-Volatile Register 0) Offset: 0xF0 Default: 0x00000000 Access: SYSTEM->NVREG0							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-118: Non-Volatile Register 0 (NVREG0) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Value in this register can only be reset via power-cycling

Table 25-119: Non-Volatile Register 1 (NVREG1) Layout

NVREG1 (Non-Volatile Register 1) Offset: 0xF4 Default: 0x00000000 Access: SYSTEM->NVREG1							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-120: Non-Volatile Register 1 (NVREG1) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Value in this register can only be reset via power-cycling

Table 25-121: Non-Volatile Register 2 (NVREG2) Layout

NVREG2 (Non-Volatile Register 2) Offset: 0xF8 Default: 0x00000000 Access: SYSTEM->NVREG2							
31	30	29	28	27	26	25	24
VAL							
23	22	21	20	19	18	17	16
VAL							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

Table 25-122: Non-Volatile Register 2 (NVREG2) Description

Bits	Field Name	Type	Reset	Description
31:0	VAL	RW	0x0	Value in this register can only be reset via power-cycling

Table 25-123: System Register Write-Allow Key Register (SYSREGKEY) Layout

SYSREGKEY (System Register Write-Allow Key Register) Offset: 0xFC Default: 0x1ACCE551 Access: SYSTEM->SYSREGKEY							
31	30	29	28	27	26	25	24
KEY							
23	22	21	20	19	18	17	16
KEY							
15	14	13	12	11	10	9	8
KEY							
7	6	5	4	3	2	1	0
KEY							

Table 25-124: System Register Write-Allow Key Register (SYSREGKEY) Description

Bits	Field Name	Type	Reset	Description
31:0	KEY	RW	0x1ACCE551	Write 0x1ACCE551 to unlock protected system registers

26 Direct memory access controller (DMA)

26.1 DMA Introduction

Direct memory access (DMA) is used in order to provide high-speed data transfer between peripherals and memory as well as memory to memory. Data can be quickly moved by DMA without CPU involvement. This keeps CPU resources free for other operations.

The DMA controller in SPC1169 has 8 channels in total to manage memory access requests from one or more peripherals. It has an arbiter for handling the priority between DMA requests. The DMA controller is interconnected with RAM, ADC, UART and SPI through the bus matrix.

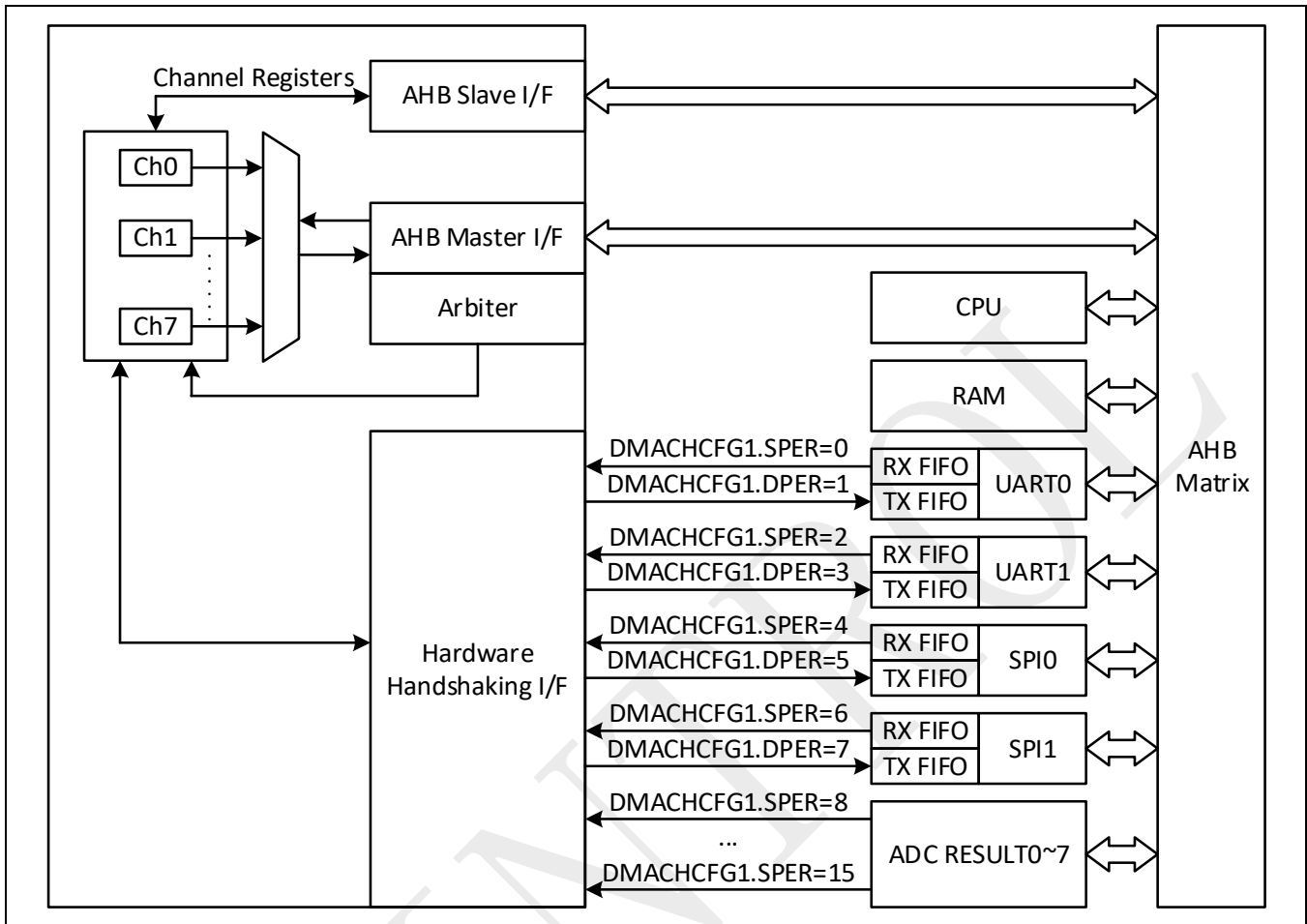
26.2 DMA features

The SPC1169 DMA controller has the following features:

- 8 independent channels and each one has a FIFO with 16 words (32-bit)
- AHB peripheral devices can request a DMA transfer through hardware handshaking interface
 - The peripheral devices include the ADC, UART and SPI.
- Programmable channel priority
- Memory-to-memory transfer
- Peripheral-to-memory and memory-to-peripheral transfers
- Peripheral-to-peripheral transfer
- Support for single block transfer with configurable block size up to 4095 bytes
- Programmable burst transaction size for source and destination
- Programmable source and destination addresses stepping level
 - Increase to the next word-aligned address
 - Decrease to the next word-aligned address
 - Freeze the address
- Two interrupt sources with flags:
 - Transfer complete interrupt
 - Error interrupt

The block diagram is shown in [Figure 26-1](#).

Figure 26-1: DMA block diagram



26.3 DMA operation

26.3.1 Basic definitions

The following terms are definitions of the DMA concepts used throughout this chapter.

- **Source** – Device from which DMA reads data. The DMA reads data from source device and then stores the data in the channel FIFO.
- **Destination** – Device to which the DMA writes the stored data from the FIFO (previously read from the source device).
- **Channel** – Read/write data path between a source device and a destination device.
- **Block** – Block of DMA data, the amount of which is the block size. For transfers between the DMA and memory, a block is directly broken into a sequence of bursts and single transfers. For transfers between the DMA and a peripheral, a block is broken into a sequence of DMA transactions (single and bursts).
- **Transaction** – Basic unit of DMA transfer. A transaction is relevant only for transfers between the DMA and a source or destination device if the device is a non-memory device. There are two types of transactions:
 - Single transaction: Length of a single transaction is always 1 and is converted to a single AHB transfer.
 - Burst transaction: Length of a burst transaction is programmed into the DMA. The burst transaction is converted into a sequence of bursts and AHB single transfers.

26.3.2 Hardware handshaking interface

DMA hardware handshaking interfaces are used at the transaction level to control the flow of single or burst transactions. When the DMA channel and the peripheral are enabled simultaneously, the hardware handshake interface can automatically complete the data transfer.

In SPC1169, the DMA hardware handshaking interfaces are allocated for UART0, UART1, SPI0, SPI1 and the ADC module as shown in [Table 26-1](#).

Table 26-1: DMA hardware handshaking interface for UART, SPI and ADC

DMACHx->DMACHCFG1.DPER/SPER	Hardware handshaking interface
0	UART0_RX
1	UART0_TX
2	UART1_RX
3	UART1_TX
4	SPI0_RX
5	SPI0_TX
6	SPI1_RX
7	SPI1_TX
8	ADC_RESULT0
9	ADC_RESULT1
10	ADC_RESULT2

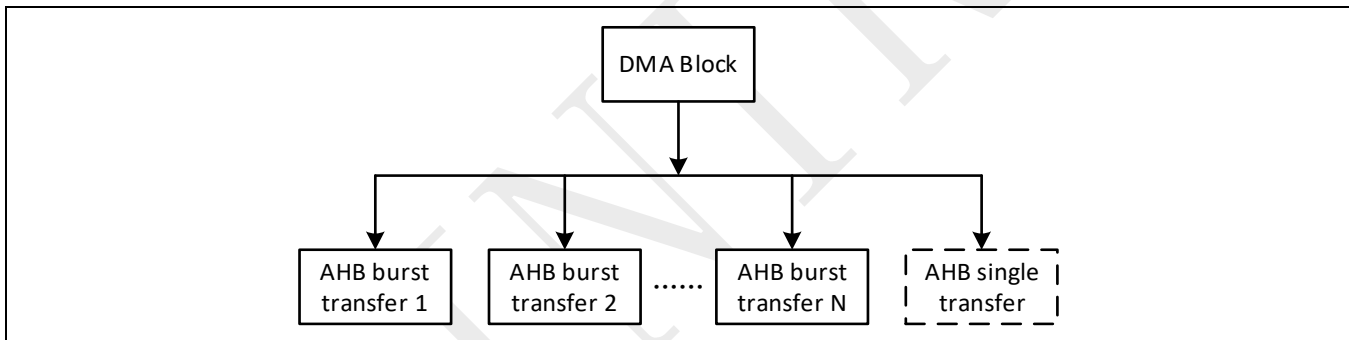
DMACHx->DMACHCFG1.DPER/SPER	Hardware handshaking interface
11	ADC_RESULT3
12	ADC_RESULT4
13	ADC_RESULT5
14	ADC_RESULT6
15	ADC_RESULT7

26.3.3 Memory-to-memory transfer operation

DMA supports only single block transfer, and the number of the block can be configured by programming DMACHCTL1.BKTS. For transfers between the DMA and memory, a block data transfer is directly broken into a sequence of burst and single transfers, as shown in Figure 26-2.

When the block size programmed into the DMA channel is not a multiple of the burst transfer length, as shown in Figure 26-2, a series of burst transfers followed by single transfers are needed to complete the block transfer.

Figure 26-2: Breakdown of DMA transfer for memory



Example: DMA transfer between two different RAM regions

Refer to the steps below to use DMA channel 0 for 12-byte data transfer from 0x1FFFE000 to 0x1FFF0000.

- Enable the DMA module by setting DMAEN as 1.
- Configure channel 0
 - Set source address DMACHSA as 0x1FFFE000.
 - Set destination address DMACHDA as 0x1FFF0000.
 - Set source address to be incremental by setting DMACHCTL0.SINC as 0.
 - Set destination address to be incremental by setting DMACHCTL0.DINC as 0.
 - Set the transfer type to be from memory to memory by setting DMACHCTL0.TT as 0.
 - Enable the channel interrupt by setting DMACHCTL0.IE as 1.
 - Set the transfer to be in byte by setting DMACHCTL0.SWIDTH and DMACHCTL0.DWIDTH as 0.
 - Set the overall block transfer size by setting DMACHCTL1.BKTS as 12.
- Enable channel 0 transfer complete interrupt by writing 0x101 to DMACHIE.
- Enable the DMA module by setting DMAEN as 1.
- Start the transfer by writing 0x101 to DMACHEN.

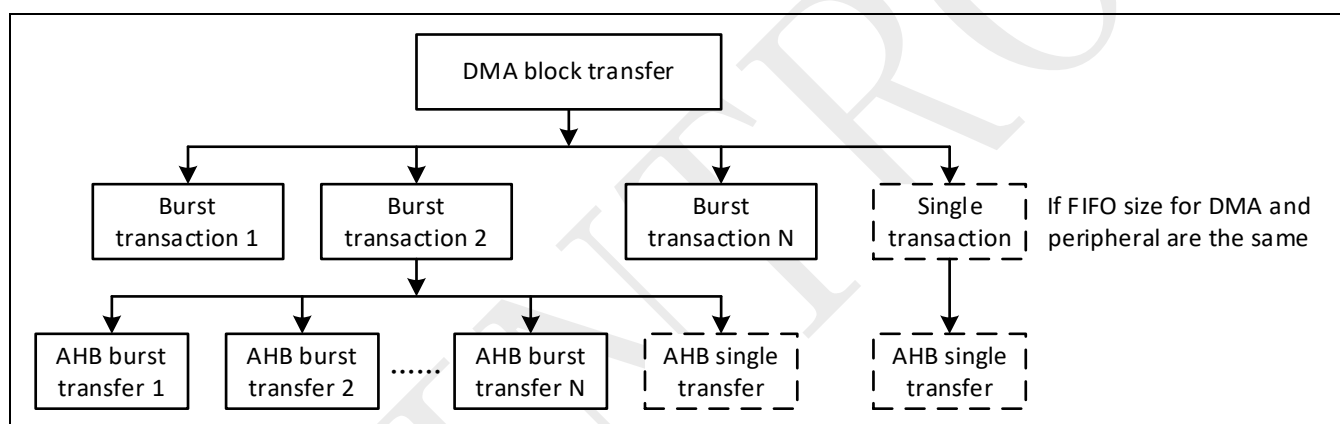
26.3.4 Memory-to-peripheral and Peripheral-to-memory transfer operation

The memory-to-peripheral and peripheral-to-memory are non-memory transfers. For the transfer between DMA and peripheral, the block is broken into a number of transactions (single or burst), each initiated by a request from the peripheral. Then, each transaction is converted to AHB transfers (single or burst) to the non-memory peripheral.

The burst transaction length of the DMA controller is defined by DMACHCTL0.SSIZE/DSIZE for source and destination, respectively. It is the number of data items (such as FIFO entries) to be transferred for each DMA request. The FIFO size of the DMA and the peripherals should be considered in determining the burst transaction length to avoid the FIFO overflow or underflow.

In SPC1169, DMA transfer is completed based on memory-to-peripheral operation for UART transmission and SPI transmission. And it is completed based on peripheral-to-memory operation for UART reception, SPI reception and ADC result readout.

Figure 26-3: Breakdown of DMA transfer for non-memory peripherals



Example: DMA transfer between memory and UART0/SPI0

Refer to the steps below to use DMA for 16-word data transfer between the memory and UART0/SPI0.

- Enable the DMA module by setting DMAEN as 1.
- Configure UART0 to support the DMA transfer
 - Enable transfer, DMA interface, and 32-bit bus via UARTCTL register.
 - Set RX FIFO threshold to be 8 words by setting UARTRXTH as 8.
 - Set TX FIFO threshold to be 0 by setting UARTRXTH as 0.
- Configure channel 0 for UART0 data reception
 - Set source address DMACHSA to be the address of UART0 FIFO interface (i.e. UARTDAT register).
 - Set destination address DMACHDA to be the start address of memory region 0.
 - Set source address to be frozen by setting DMACHCTL0.SINC as 2.
 - Set destination address to be incremental by setting DMACHCTL0.DINC as 0.
 - Set the transfer type to be from peripheral to memory by setting DMACHCTL0.TT as 2.
 - Enable the channel interrupt by setting DMACHCTL0.IE as 1.
 - Set the transfer to be in word by setting DMACHCTL0.SWIDTH and DMACHCTL0.DWIDTH as 2.
 - Set the overall block transfer size by setting DMACHCTL1.BKTS as 16.
 - Enable hardware handshaking by setting DMACHCFG0.SSWHS as 0.
 - Set hardware handshaking polarity to be active high by setting DMACHCFG0.SHSPOL as 1.

- Set hardware handshaking interface to be UART0 RX by setting DMACHCFG1.SPER as 0.
- Configure channel 1 for UART0 data transmission
 - Set source address DMACHSA to the start address of memory region 1.
 - Set destination address DMACHDA to be the address of UART0 FIFO interface (UARTDAT register).
 - Set source address to be incremental by setting DMACHCTL0.SINC as 0.
 - Set destination address to be frozen by setting DMACHCTL0.DINC as 2.
 - Set the transfer type to be from memory to peripheral by setting DMACHCTL0.TT as 1.
 - Enable the channel interrupt by setting DMACHCTL0.IE as 1.
 - Set the transfer to be in word by setting DMACHCTL0.SWIDTH and DMACHCTL0.DWIDTH as 2.
 - Set the overall block transfer size by setting DMACHCTL1.BLKTS as 16.
 - Enable hardware handshaking by setting DMACHCFG0.DSWHS as 0.
 - Set hardware handshaking polarity to be active high by setting DMACHCFG0.DHSPOL as 1.
 - Set hardware handshaking interface to be UART0 TX by setting DMACHCFG1.DPER as 0.
- Configure SPI0 to support the DMA transfer
 - Enable DMA interface by setting SPICTL0.DMATXREQ and SPICTL0.DMARXREQ as 1.
 - Set RX FIFO threshold to be 8 words by setting SPIRFTH as 8.
 - Set TX FIFO threshold to be 8 words by setting SPITFTH as 8.
- Configure channel 2 for SPI0 data reception
 - Set source address DMACHSA to be the address of SPI0 FIFO interface (i.e. SPIDAT register).
 - Set destination address DMACHDA to be the start address of memory region 2.
 - Set source address to be frozen by setting DMACHCTL0.SINC as 2.
 - Set destination address to be incremental by setting DMACHCTL0.DINC as 0.
 - Set the transfer type to be from peripheral to memory by setting DMACHCTL0.TT as 2.
 - Enable the channel interrupt by setting DMACHCTL0.IE as 1.
 - Set the transfer to be in word by setting DMACHCTL0.SWIDTH and DMACHCTL0.DWIDTH as 2.
 - Set the overall block transfer size by setting DMACHCTL1.BLKTS as 16.
 - Enable hardware handshaking by setting DMACHCFG0.SSWHS as 0.
 - Set hardware handshaking polarity to be active high by setting DMACHCFG0.SHSPOL as 1.
 - Set hardware handshaking interface to be SPI0 RX by setting DMACHCFG1.SPER as 4.
- Configure channel 3 for SPI0 data transmission
 - Set source address DMACHSA to the start address of memory region 3.
 - Set destination address DMACHDA to be the address of SPI0 FIFO interface (i.e. SPIDAT register).
 - Set source address to be incremental by setting DMACHCTL0.SINC as 0.
 - Set destination address to be frozen by setting DMACHCTL0.DINC as 2.
 - Set the transfer type to be from memory to peripheral by setting DMACHCTL0.TT as 1.
 - Enable the channel interrupt by setting DMACHCTL0.IE as 1.
 - Set the transfer to be in word by setting DMACHCTL0.SWIDTH and DMACHCTL0.DWIDTH as 2.
 - Set the overall block transfer size by setting DMACHCTL1.BLKTS as 16.
 - Enable hardware handshaking by setting DMACHCFG0.DSWHS as 0.
 - Set hardware handshaking polarity to be active high by setting DMACHCFG0.DHSPOL as 1.
 - Set hardware handshaking interface to be SPI0 TX by setting DMACHCFG1.DPER as 5.
- Enable channel 0-3 transfer complete interrupt by writing 0xF0F to DMACHIE.
- Enable the DMA module by setting DMAEN as 1.

- Start the transfer by writing 0xF0F to DMACHEN.

26.3.5 Peripheral-to-peripheral transfer operation

The peripheral-to-peripheral transfer in SPC1169 provides a physical communication channel between UART0/1 and SPI0/1, such as from UART0_RX to UART1_TX, UART0_RX to SPI0_TX, from SPI0_RX to UART1_TX, etc.

Example: DMA transfer between SPI0 reception and UART0 transmission

Refer to the steps below to use DMA for 16-word data transfer from SPI0 reception to UART0 transmission.

- Enable the DMA module by setting DMAEN as 1.
- Configure UART0 to support the DMA transfer for transmission
 - Enable transfer, DMA interface, and 32-bit bus via UARTCTL register.
 - Set TX FIFO threshold to be 0 by setting UARTEXTH as 0.
- Configure SPI0 to support the DMA transfer for reception
 - Enable DMA interface by setting SPICTL0.DMARXREQ as 1.
 - Set RX FIFO threshold to be 8 words by setting SPIRFTH as 8.
- Configure channel 0 for transferring the received data from SPI0 to be transmitted via UART0
 - Set source address DMACHSA to be the address of SPI0 FIFO interface (i.e. SPIDAT register).
 - Set destination address DMACHDA to be the address of UART0 FIFO interface (UARTDAT register).
 - Set source address to be frozen by setting DMACHCTL0.SINC as 2.
 - Set destination address to be frozen by setting DMACHCTL0.DINC as 2.
 - Set the transfer type to be from peripheral to peripheral by setting DMACHCTL0.TT as 3.
 - Enable the channel interrupt by setting DMACHCTL0.IE as 1.
 - Set the transfer to be in word by setting DMACHCTL0.SWIDTH and DMACHCTL0.DWIDTH as 2.
 - Set the overall block transfer size by setting DMACHCTL1.BLKTS as 16.
 - Enable hardware handshaking by setting DMACHCFG0.SSWHS and DMACHCFG0.DSWHS as 0.
 - Set hardware handshaking polarity to be active high by setting DMACHCFG0.SHSPOL and DMACHCFG0.DHSPOL as 1.
 - Set source hardware handshaking interface to be SPI0 RX by setting DMACHCFG1.SPER as 4.
 - Set destination hardware handshaking interface to be UART0 TX by setting DMACHCFG1.DPER as 1.
- Enable channel 0 transfer complete interrupt by writing 0x101 to DMACHIE.
- Enable the DMA module by setting DMAEN as 1.
- Start the transfer by writing 0x101 to DMACHEN.

26.4 AHB Transfer Error Handling

Upon occurrence of an eorr in an AHB transfer, the following occurs:

- DMA transfer in progress stops immediately
- Relevant channel is disabled.
- If enabled, an interrupt is issued.

If multiple channels are enabled, only the one where the AHB error was detected is disabled. The contents of the FIFO becomes inaccessible and are overwritten once the channel is re-enabled to start a new sequence.

There is no support for automatically resuming the transfer from the point where the error occurred, and the full block transfer has to be re-initiated in order to be successfully completed.

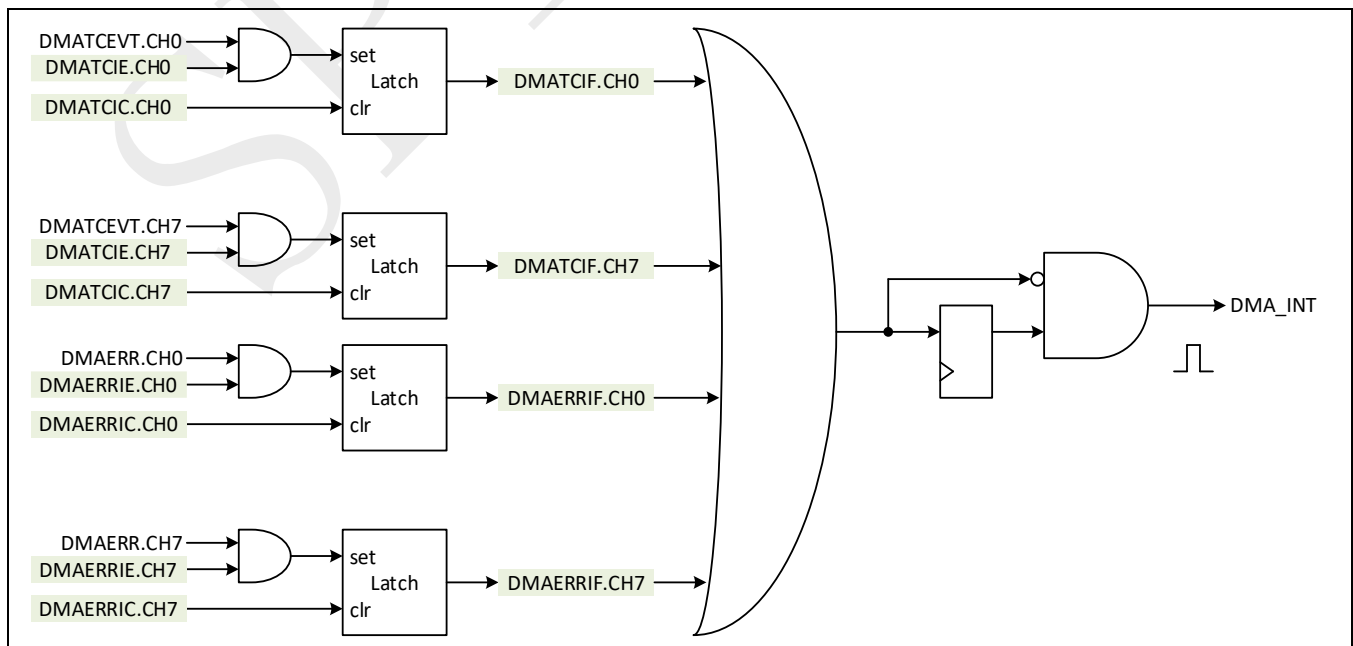
The DMA does not use the hardware handshaking interface to signal the error occurrence, nor does it signal the end of the transfer. Therefore, if the request from a peripheral is active when the error occurs, it would not be automatically de-activated by DMA acknowledge. The hardware handshake interface on the peripheral side has to be reset or re-initiated by software upon the error interrupt. The DMA request signal needs to be brought low before the channel is re-enabled.

26.5 Interrupt handling

When the DMA transfer completes for channel x, the event is latched to DMATCRAWIF.CHx. If it is enabled by setting DMATCIE.CHx as 1, an interrupt request can be generated and latched to DMATCIF.CHx. Writing 1 to DMATCIC clears both the DMATCRAWIF.CHx and DMATCIF.CHx.

When an ERROR response is received from an AHB slave during a DMA transfer for channel x, DMA transfer is cancelled and the channel is disabled. The event is latched to DMAERRRAWIF.CHx. If it is enabled by setting DMAERRIE.CHx as 1, an interrupt request can be generated and latched to DMAERRIF.CHx. Writing 1 to DMAERRIC clears both the DMAERRRAWIF.CHx and DMAERRIF.CHx.

Figure 26-4: DMA interrupt generation



As shown in Figure 26-4, the interrupt flag bits are combined together to generate the final single interrupt to the CPU. It can be seen that the interrupt flag bits would block the interrupt issuing to the CPU until they are all cleaned.

Based on the structure, it is recommended to implement the ISR (interrupt service routine) as the pseudo-code shown in Figure 26-5.

Figure 26-5: Pseduo-code for DMA ISR

```

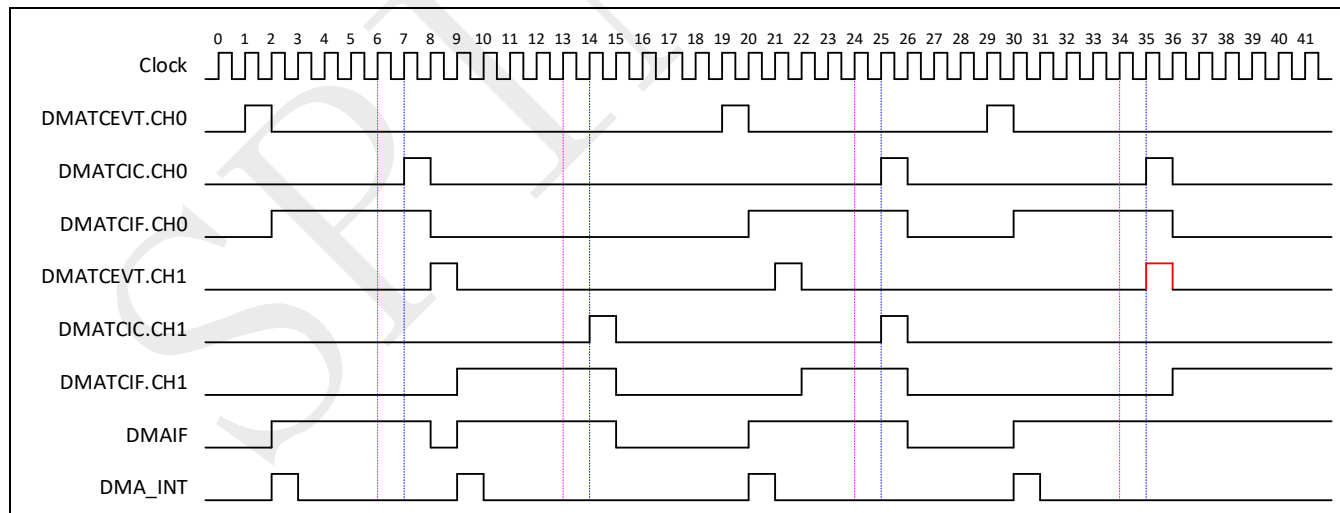
u32TCFlag = DMAC->DMATCIF;
DMAC->DMATCIC = u32TCFlag; // Make sure it will not be interrupt here by ISR with higher priority
u32ErrFlag = DMAC->DMAERRIF;
DMAC->DMAERRIC = u32ErrFlag; // Make sure it will not be interrupt here by ISR with higher priority
if ((u32TCFlag & 1U) !=0) { // Code to handle channel 0 transfer completion }
if ((u32TCFlag & 2U) !=0) { // Code to handle channel 1 transfer completion }
...
if ((u32ErrFlag & 1U) !=0) { // Code to handle channel 0 transfer error }
if ((u32ErrFlag & 2U) !=0) { // Code to handle channel 1 transfer error }

```

26.5.1 Risk mitigation for interrupt loss and blocking

Figure 26-6 illustrates various scenario when DMA interrupts are enabled for multiple channels.

Figure 26-6: Scenario DMA interrupt events for multiple channels



- ISR entry is triggered independently by request from each channel. Therefore, there is no interrupt loss or double serve.
 - Transfer of channel 0 completes at t_1 , and triggers the interrupt request. DMATCIF.CH0 is set accordingly.
 - The software reads the DMATCIF in the ISR at t_6 , and finds that only DMATCIF.CH0 is set. Then it clears the flag at t_7 accordingly to enable the new interrupt coming later. After that, it serves the request from channel 0.

- Transfer of channel 1 completes at t_8 , and triggers the interrupt request. DMATCIF.CH1 is set accordingly.
- Since there is no overlap between the DMATCIF.CH0 and DMATCIF.CH1, a rising edge can be recognized in the combined DMAIF signal, and a new interrupt pulse is issued to the CPU. Therefore, when the software exits the ISR, it will enter the ISR again.
- The software reads the DMATCIF in the ISR at t_{13} , and finds that only DMATCIF.CH1 is set. Then it clears the flag at t_{14} accordingly to enable the new interrupt coming later. After the request from channel 1 is served, it exits the ISR.
- ISR entry is triggered by request from one channel but multiple requests are served within the ISR. Therefore, there is no interrupt loss or double serve.
 - Transfer of channel 0 completes at t_{19} , and triggers the interrupt request. DMATCIF.CH0 is set accordingly.
 - Before the software reads the DMATCIF in the ISR at t_{24} , transfer of channel 1 completes at t_{21} . Therefore, the software will find that both DMATCIF.CH0 and DMATCIF.CH1 are set at t_{24} , and clear them at t_{25} accordingly to enable the new interrupt coming later. After the requests from both channel 0 and channel 1 are served, it exits the ISR.
- New interrupt request comes after the interrupt flag readout in the ISR but before the interrupt clear, leading to the interrupt loss and blocking for future interrupts.
 - Transfer of channel 0 completes at t_{29} , and triggers the DMA interrupt. DMATCIF.CH0 is set accordingly.
 - The software reads the DMATCIF in the ISR at t_{34} and find that only DMATCIF.CH0 is set. Then it clears the flag at t_{35} accordingly to enable the new interrupt coming later, and starts to serve for channel 0
 - Transfer of channel 1 completes at t_{35} , and triggers the interrupt request. DMATCIF.CH1 is set accordingly.
 - Since DMATCIF.CH0 is cleared at the same time when DMATCIF.CH1 is set, there will be no rising edge in the combined DMAIF signal. So no new interrupt pulse will be issued to the CPU.
 - There is no chance for the clearance of DMATCIF.CH1 since CPU is unaware of this situation. **The combined DMAIF is always 1 and DMA interrupt pulse will not be generated any more.**

Based on the analysis above, it causes problem when the new interrupt request comes after the flag readout and before the flag clear. To minimize the risk, this time window should be as small as possible. The coding style shown in [Figure 26-5](#) reduces the window to only 1 clock cycle, but there is still potential risk as shown in [Figure 26-6](#). To fully solve the problem, user need to check the DMA interrupt flag periodically (i.e. in the ISR of a timer) and take actions.

26.6 Registers

26.6.1 DMACH register map

Table 26-2: DMACH Module Base Address

Peripheral Module	Base Address
DMACH0	0x4000 1C00
DMACH1	0x4000 1C58
DMACH2	0x4000 1CB0
DMACH3	0x4000 1D08
DMACH4	0x4000 1D60
DMACH5	0x4000 1DB8
DMACH6	0x4000 1E10
DMACH7	0x4000 1E68

Table 26-3: DMACH Register Map

Register	Offset	Description	Reset Value
DMACHSA	0x0	DMA Channel Source Address Register	0x00000000
DMACHDA	0x8	DMA Channel Destination Address Register	0x00000000
DMACHCTL0	0x18	DMA Channel Control Register 0	0x00304801
DMACHCTL1	0x1C	DMA Channel Control Register 1	0x00000002
DMACHCFG0	0x40	DMA Channel Configure Register 0	0x00000E00
DMACHCFG1	0x44	DMA Channel Configure Register 1	0x00000004

26.6.2 DMACH registers

Table 26-4: DMA Channel Source Address Register (DMACHSA) Layout

DMACHSA (DMA Channel Source Address Register) Offset: 0x0 Default: 0x00000000 Access: DMACHx->DMACHSA							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

Table 26-5: DMA Channel Source Address Register (DMACHSA) Description

Bits	Field Name	Type	Reset	Description
31:0	ADDR	RW	0x0	Current source address of DMA transfer. This register must be written with a 32-bit aligned address. It is updated after each source transfer according to DMACHCTL0.SINC

Table 26-6: DMA Channel Destination Address Register (DMACHDA) Layout

DMACHDA (DMA Channel Destination Address Register) Offset: 0x8 Default: 0x00000000 Access: DMACHx->DMACHDA							
31	30	29	28	27	26	25	24
ADDR							
23	22	21	20	19	18	17	16
ADDR							
15	14	13	12	11	10	9	8
ADDR							
7	6	5	4	3	2	1	0
ADDR							

Table 26-7: DMA Channel Destination Address Register (DMACHDA) Description

Bits	Field Name	Type	Reset	Description
31:0	ADDR	RW	0x0	Current Destination address of DMA transfer. This register must be written with a 32-bit aligned address. It is updated after each source transfer according to DMACHCTL0.DINC

Table 26-8: DMA Channel Control Register 0 (DMACHCTL0) Layout

DMACHCTL0 (DMA Channel Control Register 0) Offset: 0x18 Default: 0x00304801 Access: DMACHx->DMACHCTL0									
31		30		29	28	27	26	25	24
RESERVED			RESERVED		RESERVED	RESERVED			RESERVED
23	22		21		20	19	18	17	16
RESERVED	TT			RESERVED		RESERVED	RESERVED	RESERVED	SSIZE
15		14	13		12	11	10	9	8
SSIZE			DSIZE			SINC			DINC
7	6		5		4	3	2	1	0
DINC	SWIDTH				DWIDTH				IE

Table 26-9: DMA Channel Control Register 0 (DMACHCTL0) Description

Bits	Field Name	Type	Reset	Description
31:29	RESERVED_31_29	RO	0x0	Reserved.
28	RESERVED_28	RW	0x0	Reserved.
27	RESERVED_27	RW	0x0	Reserved.
26:25	RESERVED_26_25	RW	0x0	Reserved.
24:23	RESERVED_24_23	RW	0x0	Reserved.
22:20	TT	RW	0x3	Transfer type. For multi-block transfers using linked list operation, TT must be constant for all blocks of this multi-block transfer. 000: Memory to memory, DMAC is flow controller 001: Memory to peripheral, DMAC is flow controller 010: Peripheral to memory, DMAC is flow controller 011: Peripheral to peripheral, DMAC is flow controller 100: Peripheral to memory, source peripheral is flow controller 101: Peripheral to peripheral, source peripheral is flow controller 110: Memory to peripheral, destination peripheral is flow controller 111: Peripheral to peripheral, destination peripheral is flow controller
19	RESERVED_19	RO	0x0	Reserved.
18	RESERVED_18	RW	0x0	Reserved.
17	RESERVED_17	RW	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
16:14	SSIZE	RW	0x1	Source burst transaction length. Number of data items, each of width SWIDTH, to be read from the source every time a source burst transaction request is made the corresponding handshaking interface. Note: This value is not related to the AHB master HBURST signal. 000: 1 word (32-bit) 001: 4 words 010: 8 words 011: 16 words 100: 32 words 101: 64 words 110: 128 words 111: 256 words
13:11	DSIZE	RW	0x1	Destination burst transaction length. Number of data items, each of width DWIDTH, to be written to the destination every time a destination burst transaction request is made from the corresponding handshaking interface. Note: This value is not related to the AHB master HBURST signal. 000: 1 word (32-bit) 001: 4 words 010: 8 words 011: 16 words 100: 32 words 101: 64 words 110: 128 words 111: 256 words
10:9	SINC	RW	0x0	Source address increment after each transfer. Incrementing or decrementing is done for alignment to the next SWIDTH boundary. 00: Increase to next word aligned address 01: Decrease to next word aligned address 10: Freeze the address
8:7	DINC	RW	0x0	Destination address increment after each transfer. Incrementing or decrementing is done for alignment to the next DWIDTH boundary. 00: Increase to next word aligned address 01: Decrease to next word aligned address 10: Freeze the address
6:4	SWIDTH	RW	0x0	Source transfer width. It should not exceed AHB bus HDATA width (32-

Bits	Field Name	Type	Reset	Description
				bits). For a non-memory peripheral, typically the peripheral (source) FIFO width. 000: Byte (8-bits) 001: Half-word (16-bits) 010: Word (32-bits) 011: Double-word (64-bits) 100: Four-word (128-bits) 101: Eight-word (256-bits)
3:1	DWIDTH	RW	0x0	Destination transfer width. It should not exceed AHB bus HDATA width (32-bits). For a non-memory peripheral, typically the peripheral (destination) FIFO width. 000: Byte (8-bits) 001: Half-word (16-bits) 010: Word (32-bits) 011: Double-word (64-bits) 100: Four-word (128-bits) 101: Eight-word (256-bits)
0	IE	RW	0x1	Interrupt enable This bit does not affect the Interrupt Raw flag generation 0: Disable interrupt 1: Enable interrupt

Table 26-10: DMA Channel Control Register 1 (DMACHCTL1) Layout

DMACHCTL1 (DMA Channel Control Register 1) Offset: 0x1C Default: 0x00000002 Access: DMACHx->DMACHCTL1							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED			RESERVED	BLKTS			
7	6	5	4	3	2	1	0
BLKTS							

Table 26-11: DMA Channel Control Register 1 (DMACHCTL1) Description

Bits	Field Name	Type	Reset	Description
31:13	RESERVED_31_13	RO	0x0	Reserved.
12	RESERVED_12	RW	0x0	Reserved.
11:0	BLKTS	RW	0x2	Block transfer size. Which is the total number of single transactions to perform for every block transfer; a single transaction is mapped to a single AMBA beat. The width of the single transaction is determined by DMACHCTL0.SWIDTH. This field must be written before the channel is enabled in order to indicate the block size.

Table 26-12: DMA Channel Configure Register 0 (DMACHCFG0) Layout

DMACHCFG0 (DMA Channel Configure Register 0) Offset: 0x40 Default: 0x00000E00 Access: DMACHx->DMACHCFG0							
31 DRLD	30 SRLD	29	28	27	26	25	24
RESERVED							
23	22	21	20	19 SHSPOL	18 DHSPOL	17 RESERVED	16 RESERVED
RESERVED		RESERVED		11 SSWHS	10 DSWHS	9 FIFOEMPTY	8 SUSPEND
15 RESERVED	14 RESERVED	13 RESERVED	12 RESERVED	7	6	5	4
PRIORITY				3	2	1	0
				RESERVED			

Table 26-13: DMA Channel Configure Register 0 (DMACHCFG0) Description

Bits	Field Name	Type	Reset	Description
31	DRLD	RW	0x0	Automatic destination reload. The DMACHDA register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated. 0: 1:
30	SRLD	RW	0x0	Automatic source reload. The DMACHSA register can be automatically reloaded from its initial value at the end of every block for multi-block transfers. A new block transfer is then initiated. 0: 1:
29:20	RESERVED_29_20	RW	0x0	Reserved.

Bits	Field Name	Type	Reset	Description
19	SHSPOL	RW	0x0	Source handshaking interface polarity. 0: Active low 1: Active high
18	DHSPOL	RW	0x0	Destination handshaking interface polarity. 0: Active low 1: Active high
17	RESERVED_17	RW	0x0	Reserved.
16	RESERVED_16	RW	0x0	Reserved.
15:14	RESERVED_15_14	RW	0x0	Reserved.
13:12	RESERVED_13_12	RW	0x0	Reserved.
11	SSWHS	RW	0x1	Source software handshaking select. This bit is ignored if the source is memory 0: Hardware handshaking 1: Software handshaking
10	DSWHS	RW	0x1	Destination software handshaking select. This bit is ignored if the destination is memory 0: Hardware handshaking 1: Software handshaking
9	FIFOEMPTY	RO	0x1	Indicates if there is data left in the channel FIFO. Can be used in conjunction with SUSPEND bit to cleanly disable a channel. 0: There is data left in the FIFO 1: FIFO is fully empty
8	SUSPEND	RW	0x0	Channel suspend. Suspends all DMA data transfers from the source until this bit is cleared. There is no guarantee that the current transaction will complete. Can also be used in conjunction with FIFODAV bit to cleanly disable a channel without losing any data. 0: Not suspended. 1: Suspend DMA transfer from the source.
7:5	PRIORITY	RW	0x0	Channel priority. This field must be written as a value within 0 to 7 while 0 is the lowest priority 000: 8th priority (Lowest) 001: 7th priority 010: 6th priority 011: 5th priority 100: 4th priority 101: 3rd priority 110: 2nd priority 111: 1st priority (Highest)

Bits	Field Name	Type	Reset	Description
4:0	RESERVED_4_0	RO	0x0	Reserved.

Table 26-14: DMA Channel Configure Register 1 (DMACHCFG1) Layout

DMACHCFG1 (DMA Channel Configure Register 1) Offset: 0x44 Default: 0x00000004 Access: DMACHx->DMACHCFG1							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 RESERVED	14	13	12	11	10	9	8
DPER				SPER			
7 SPER	6 RESERVED	5 RESERVED	4	3	2	1 FIFOTH	0 RESERVED
RESERVED				RESERVED			

Table 26-15: DMA Channel Configure Register 1 (DMACHCFG1) Description

Bits	Field Name	Type	Reset	Description
31:15	RESERVED_31_15	RO	0x0	Reserved.
14:11	DPER	RW	0x0	Destination peripheral that communicated through hardware handshaking interface. This field is ignored if DMACHCFG0.DSWHS=1. 0000: 0001: 0010: 0011: 0100: 0101: 0110: 0111: 1000: 1001: 1010: 1011: 1100: 1101: 1110: 1111:
10:7	SPER	RW	0x0	Source peripheral that communicated through hardware handshaking interface. This field is ignored if DMACHCFG0.SSWHS=1. 0000: 0001: 0010: 0011:

Bits	Field Name	Type	Reset	Description
				0100: 0101: 0110: 0111: 1000: 1001: 1010: 1011: 1100: 1101: 1110: 1111:
6	RESERVED_6	RW	0x0	Reserved.
5	RESERVED_5	RW	0x0	Reserved.
4:2	RESERVED_4_2	RW	0x1	Reserved.
1	FIFOTH	RW	0x0	FIFO threshold to determine how much space or data needs to be available in the FIFO before a burst transaction request is serviced. 0: Transfer whenever data or space is available in the FIFO 1: Destination transfer will not start until data available is greater than or equal to half the FIFO depth. Source transfer will not start until data available is greater than half the FIFO depth. The exceptions are at the end of a burst transaction request or at the end of a block transfer.
0	RESERVED_0	RW	0x0	Reserved.

26.6.3 DMAC register map

Table 26-16: DMAC Module Base Address

Peripheral Module	Base Address
DMAC	0x4000 1C00

Table 26-17: DMAC Register Map

Register	Offset	Description	Reset Value
DMATCRAWIF	0x2C0	DMA Transfer Complete Interrupt Raw Flag Register	0x00000000
DMABTCRAWIF	0x2C8	DMA Block Transfer Complete Interrupt Raw Flag Register	0x00000000
DMASTCRAWIF	0x2D0	DMA Source Transfer Complete Interrupt Raw Flag Register	0x00000000
DMADTCRAWIF	0x2D8	DMA Destination Transfer Complete Interrupt Raw Flag Register	0x00000000
DMAERRRAWIF	0x2E0	DMA Error Interrupt Raw Flag Register	0x00000000
DMATCIF	0x2E8	DMA Transfer Complete Interrupt Flag Register	0x00000000
DMABTCIF	0x2F0	DMA Block Transfer Complete Interrupt Flag Register	0x00000000
DMASTCIF	0x2F8	DMA Source Transfer Complete Interrupt Flag Register	0x00000000
DMADTCIF	0x300	DMA Destination Transfer Complete Interrupt Flag Register	0x00000000
DMAERRIF	0x308	DMA Error Interrupt Flag Register	0x00000000
DMATCIE	0x310	DMA Transfer Complete Interrupt Enable Register	0x00000000
DMABTCIE	0x318	DMA Block Transfer Complete Interrupt Enable Register	0x00000000
DMASTCIE	0x320	DMA Source Transfer Complete Interrupt Enable Register	0x00000000
DMADTCIE	0x328	DMA Destination Transfer Complete Interrupt Enable Register	0x00000000
DMAERRIE	0x330	DMA Error Interrupt Enable Register	0x00000000
DMATCIC	0x338	DMA Transfer Complete Interrupt Clear Register	0x00000000
DMABTCIC	0x340	DMA Block Transfer Complete Interrupt Clear Register	0x00000000
DMASTCIC	0x348	DMA Source Transfer Complete Interrupt Clear Register	0x00000000
DMADTCIC	0x350	DMA Destination Transfer Complete Interrupt Clear Register	0x00000000
DMAERRIC	0x358	DMA Error Interrupt Clear Register	0x00000000
DMAIF	0x360	DMA Combined Interrupt Flag Register	0x00000000
DMASSREQ	0x368	DMA Source Software Transaction Request Register	0x00000000

Register	Offset	Description	Reset Value
<u>DMADSREQ</u>	0x370	DMA Destination Software Transaction Request Register	0x00000000
<u>DMASSSREQ</u>	0x378	DMA Source Software Transaction Single Request Register	0x00000000
<u>DMADSSREQ</u>	0x380	DMA Destination Software Transaction Single Request Register	0x00000000
<u>DMASSLREQ</u>	0x388	DMA Source Software Transaction Last Request Register	0x00000000
<u>DMADSLREQ</u>	0x390	DMA Destination Software Transaction Last Request Register	0x00000000
<u>DMAEN</u>	0x398	DMA Enable Register	0x00000000
<u>DMACHEN</u>	0x3A0	DMA Channel Enable Register	0x00000000

26.6.4 DMAC registers

Table 26-18: DMA Transfer Complete Interrupt Raw Flag Register (DMATCRAWIF) Layout

DMATCRAWIF (DMA Transfer Complete Interrupt Raw Flag Register) Offset: 0x2C0 Default: 0x00000000 Access: DMACx->DMATCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-19: DMA Transfer Complete Interrupt Raw Flag Register (DMATCRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
6	CH6	RO	0x0	Channel 6 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
5	CH5	RO	0x0	Channel 5 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
4	CH4	RO	0x0	Channel 4 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
3	CH3	RO	0x0	Channel 3 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
2	CH2	RO	0x0	Channel 2 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not

Bits	Field Name	Type	Reset	Description
				recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
1	CH1	RO	0x0	Channel 1 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
0	CH0	RO	0x0	Channel 0 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed

Table 26-20: DMA Block Transfer Complete Interrupt Raw Flag Register (DMABTCRAWIF) Layout

DMABTCRAWIF (DMA Block Transfer Complete Interrupt Raw Flag Register) Offset: 0x2C8 Default: 0x00000000 Access: DMACx->DMABTCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-21: DMA Block Transfer Complete Interrupt Raw Flag Register (DMABTCRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
6	CH6	RO	0x0	Channel 6 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
5	CH5	RO	0x0	Channel 5 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation

Bits	Field Name	Type	Reset	Description
				0: Transfer not completed 1: Transfer completed
4	CH4	RO	0x0	Channel 4 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
3	CH3	RO	0x0	Channel 3 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
2	CH2	RO	0x0	Channel 2 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
1	CH1	RO	0x0	Channel 1 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
0	CH0	RO	0x0	Channel 0 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed

Table 26-22: DMA Source Transfer Complete Interrupt Raw Flag Register (DMASTCRAWIF) Layout

DMASTCRAWIF (DMA Source Transfer Complete Interrupt Raw Flag Register) Offset: 0x2D0 Default: 0x00000000 Access: DMACx->DMASTCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-23: DMA Source Transfer Complete Interrupt Raw Flag Register (DMASTCRAWIF)
Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
6	CH6	RO	0x0	Channel 6 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
5	CH5	RO	0x0	Channel 5 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
4	CH4	RO	0x0	Channel 4 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
3	CH3	RO	0x0	Channel 3 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
2	CH2	RO	0x0	Channel 2 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
1	CH1	RO	0x0	Channel 1 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
0	CH0	RO	0x0	Channel 0 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed

Table 26-24: DMA Destination Transfer Complete Interrupt Raw Flag Register (DMADTCRAWIF) Layout

DMADTCRAWIF (DMA Destination Transfer Complete Interrupt Raw Flag Register) Offset: 0x2D8 Default: 0x00000000 Access: DMACx->DMADTCRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-25: DMA Destination Transfer Complete Interrupt Raw Flag Register (DMADTCRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
6	CH6	RO	0x0	Channel 6 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
5	CH5	RO	0x0	Channel 5 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
4	CH4	RO	0x0	Channel 4 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
3	CH3	RO	0x0	Channel 3 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed

Bits	Field Name	Type	Reset	Description
2	CH2	RO	0x0	Channel 2 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
1	CH1	RO	0x0	Channel 1 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed
0	CH0	RO	0x0	Channel 0 transfer complete Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Transfer not completed 1: Transfer completed

Table 26-26: DMA Error Interrupt Raw Flag Register (DMAERRRAWIF) Layout

DMAERRRAWIF (DMA Error Interrupt Raw Flag Register) Offset: 0x2E0 Default: 0x00000000 Access: DMACx->DMAERRRAWIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-27: DMA Error Interrupt Raw Flag Register (DMAERRRAWIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred
6	CH6	RO	0x0	Channel 6 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred

Bits	Field Name	Type	Reset	Description
5	CH5	RO	0x0	Channel 5 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred
4	CH4	RO	0x0	Channel 4 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred
3	CH3	RO	0x0	Channel 3 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred
2	CH2	RO	0x0	Channel 2 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred
1	CH1	RO	0x0	Channel 1 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred
0	CH0	RO	0x0	Channel 0 error Interrupt Raw flag This bit can be written for test purpose. It is not recommended to write for normal operation 0: Error not occurred 1: Error occurred

Table 26-28: DMA Transfer Complete Interrupt Flag Register (DMATCIF) Layout

DMATCIF (DMA Transfer Complete Interrupt Flag Register) Offset: 0x2E8 Default: 0x00000000 Access: DMACx->DMATCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-29: DMA Transfer Complete Interrupt Flag Register (DMATCIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
6	CH6	RO	0x0	Channel 6 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
5	CH5	RO	0x0	Channel 5 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
4	CH4	RO	0x0	Channel 4 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
3	CH3	RO	0x0	Channel 3 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
2	CH2	RO	0x0	Channel 2 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
1	CH1	RO	0x0	Channel 1 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
0	CH0	RO	0x0	Channel 0 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred

Table 26-30: DMA Block Transfer Complete Interrupt Flag Register (DMABTCIF) Layout

DMABTCIF (DMA Block Transfer Complete Interrupt Flag Register) Offset: 0x2F0 Default: 0x00000000 Access: DMACx->DMABTCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-31: DMA Block Transfer Complete Interrupt Flag Register (DMABTCIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
6	CH6	RO	0x0	Channel 6 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
5	CH5	RO	0x0	Channel 5 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
4	CH4	RO	0x0	Channel 4 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
3	CH3	RO	0x0	Channel 3 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
2	CH2	RO	0x0	Channel 2 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
1	CH1	RO	0x0	Channel 1 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
0	CH0	RO	0x0	Channel 0 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred

Table 26-32: DMA Source Transfer Complete Interrupt Flag Register (DMASTCIF) Layout

DMASTCIF (DMA Source Transfer Complete Interrupt Flag Register) Offset: 0x2F8 Default: 0x00000000 Access: DMACx->DMASTCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-33: DMA Source Transfer Complete Interrupt Flag Register (DMASTCIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
6	CH6	RO	0x0	Channel 6 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
5	CH5	RO	0x0	Channel 5 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
4	CH4	RO	0x0	Channel 4 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
3	CH3	RO	0x0	Channel 3 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
2	CH2	RO	0x0	Channel 2 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
1	CH1	RO	0x0	Channel 1 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
0	CH0	RO	0x0	Channel 0 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred

Table 26-34: DMA Destination Transfer Complete Interrupt Flag Register (DMADTCIF) Layout

DMADTCIF (DMA Destination Transfer Complete Interrupt Flag Register) Offset: 0x300 Default: 0x00000000 Access: DMACx->DMADTCIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-35: DMA Destination Transfer Complete Interrupt Flag Register (DMADTCIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
6	CH6	RO	0x0	Channel 6 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
5	CH5	RO	0x0	Channel 5 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
4	CH4	RO	0x0	Channel 4 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
3	CH3	RO	0x0	Channel 3 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
2	CH2	RO	0x0	Channel 2 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
1	CH1	RO	0x0	Channel 1 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred
0	CH0	RO	0x0	Channel 0 transfer complete interrupt flag 0: Transfer complete interrupt not occurred 1: Transfer complete interrupt occurred

Table 26-36: DMA Error Interrupt Flag Register (DMAERRIF) Layout

DMAERRIF (DMA Error Interrupt Flag Register) Offset: 0x308 Default: 0x00000000 Access: DMACx->DMAERRIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-37: DMA Error Interrupt Flag Register (DMAERRIF) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	RO	0x0	Channel 7 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
6	CH6	RO	0x0	Channel 6 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
5	CH5	RO	0x0	Channel 5 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
4	CH4	RO	0x0	Channel 4 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
3	CH3	RO	0x0	Channel 3 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
2	CH2	RO	0x0	Channel 2 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
1	CH1	RO	0x0	Channel 1 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred
0	CH0	RO	0x0	Channel 0 error interrupt flag 0: Error interrupt not occurred 1: Error interrupt occurred

Table 26-38: DMA Transfer Complete Interrupt Enable Register (DMATCIE) Layout

DMATCIE (DMA Transfer Complete Interrupt Enable Register) Offset: 0x310 Default: 0x00000000 Access: DMACx->DMATCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-39: DMA Transfer Complete Interrupt Enable Register (DMATCIE) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 transfer complete interrupt enable 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 transfer complete interrupt enable 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 transfer complete interrupt enable 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 transfer complete interrupt enable 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 transfer complete interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
2	CH2	RW	0x0	Channel 2 transfer complete interrupt enable 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 transfer complete interrupt enable 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 transfer complete interrupt enable 0: Disable 1: Enable

Table 26-40: DMA Block Transfer Complete Interrupt Enable Register (DMABTCIE) Layout

DMABTCIE (DMA Block Transfer Complete Interrupt Enable Register) Offset: 0x318 Default: 0x00000000 Access: DMACx->DMABTCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-41: DMA Block Transfer Complete Interrupt Enable Register (DMABTCIE) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
10	CH2WE	W1S	0x0	CH2 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 transfer complete interrupt enable 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 transfer complete interrupt enable 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 transfer complete interrupt enable 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 transfer complete interrupt enable 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 transfer complete interrupt enable 0: Disable 1: Enable
2	CH2	RW	0x0	Channel 2 transfer complete interrupt enable 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 transfer complete interrupt enable 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 transfer complete interrupt enable 0: Disable 1: Enable

Table 26-42: DMA Source Transfer Complete Interrupt Enable Register (DMASTCIE) Layout

DMASTCIE (DMA Source Transfer Complete Interrupt Enable Register) Offset: 0x320 Default: 0x00000000 Access: DMACx->DMASTCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-43: DMA Source Transfer Complete Interrupt Enable Register (DMASTCIE) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 transfer complete interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
6	CH6	RW	0x0	Channel 6 transfer complete interrupt enable 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 transfer complete interrupt enable 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 transfer complete interrupt enable 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 transfer complete interrupt enable 0: Disable 1: Enable
2	CH2	RW	0x0	Channel 2 transfer complete interrupt enable 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 transfer complete interrupt enable 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 transfer complete interrupt enable 0: Disable 1: Enable

Table 26-44: DMA Destination Transfer Complete Interrupt Enable Register (DMADTCIE) Layout

DMADTCIE (DMA Destination Transfer Complete Interrupt Enable Register) Offset: 0x328 Default: 0x00000000 Access: DMACx->DMADTCIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-45: DMA Destination Transfer Complete Interrupt Enable Register (DMADTCIE) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 transfer complete interrupt enable 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 transfer complete interrupt enable 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 transfer complete interrupt enable 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 transfer complete interrupt enable 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 transfer complete interrupt enable 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
2	CH2	RW	0x0	Channel 2 transfer complete interrupt enable 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 transfer complete interrupt enable 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 transfer complete interrupt enable 0: Disable 1: Enable

Table 26-46: DMA Error Interrupt Enable Register (DMAERRIE) Layout

DMAERRIE (DMA Error Interrupt Enable Register) Offset: 0x330 Default: 0x00000000 Access: DMACx->DMAERRIE							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-47: DMA Error Interrupt Enable Register (DMAERRIE) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
10	CH2WE	W1S	0x0	CH2 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 error interrupt enable 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 error interrupt enable 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 error interrupt enable 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 error interrupt enable 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 error interrupt enable 0: Disable 1: Enable
2	CH2	RW	0x0	Channel 2 error interrupt enable 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 error interrupt enable 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 error interrupt enable 0: Disable 1: Enable

Table 26-48: DMA Transfer Complete Interrupt Clear Register (DMATCIC) Layout

DMATCIC (DMA Transfer Complete Interrupt Clear Register) Offset: 0x338 Default: 0x00000000 Access: DMACx->DMATCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-49: DMA Transfer Complete Interrupt Clear Register (DMATCIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	W1C	0x0	Channel 7 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
6	CH6	W1C	0x0	Channel 6 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
5	CH5	W1C	0x0	Channel 5 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
4	CH4	W1C	0x0	Channel 4 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
3	CH3	W1C	0x0	Channel 3 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
2	CH2	W1C	0x0	Channel 2 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0.

Bits	Field Name	Type	Reset	Description
				1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
1	CH1	W1C	0x0	Channel 1 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
0	CH0	W1C	0x0	Channel 0 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Table 26-50: DMA Block Transfer Complete Interrupt Clear Register (DMABTCIC) Layout

DMABTCIC (DMA Block Transfer Complete Interrupt Clear Register) Offset: 0x340 Default: 0x00000000 Access: DMACx->DMABTCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-51: DMA Block Transfer Complete Interrupt Clear Register (DMABTCIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	W1C	0x0	Channel 7 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
6	CH6	W1C	0x0	Channel 6 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
5	CH5	W1C	0x0	Channel 5 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
4	CH4	W1C	0x0	Channel 4 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
3	CH3	W1C	0x0	Channel 3 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
2	CH2	W1C	0x0	Channel 2 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
1	CH1	W1C	0x0	Channel 1 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
0	CH0	W1C	0x0	Channel 0 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Table 26-52: DMA Source Transfer Complete Interrupt Clear Register (DMASTCIC) Layout

DMASTCIC (DMA Source Transfer Complete Interrupt Clear Register) Offset: 0x348 Default: 0x00000000 Access: DMACx->DMASTCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-53: DMA Source Transfer Complete Interrupt Clear Register (DMASTCIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	W1C	0x0	Channel 7 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
6	CH6	W1C	0x0	Channel 6 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
5	CH5	W1C	0x0	Channel 5 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
4	CH4	W1C	0x0	Channel 4 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
3	CH3	W1C	0x0	Channel 3 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
2	CH2	W1C	0x0	Channel 2 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
1	CH1	W1C	0x0	Channel 1 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
0	CH0	W1C	0x0	Channel 0 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Table 26-54: DMA Destination Transfer Complete Interrupt Clear Register (DMADTCIC) Layout

DMADTCIC (DMA Destination Transfer Complete Interrupt Clear Register) Offset: 0x350 Default: 0x00000000 Access: DMACx->DMADTCIC							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-55: DMA Destination Transfer Complete Interrupt Clear Register (DMADTCIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	W1C	0x0	Channel 7 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
6	CH6	W1C	0x0	Channel 6 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
5	CH5	W1C	0x0	Channel 5 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
4	CH4	W1C	0x0	Channel 4 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
3	CH3	W1C	0x0	Channel 3 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
2	CH2	W1C	0x0	Channel 2 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
1	CH1	W1C	0x0	Channel 1 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
0	CH0	W1C	0x0	Channel 0 transfer complete interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Table 26-56: DMA Error Interrupt Clear Register (DMAERRIC) Layout

DMAERRIC (DMA Error Interrupt Clear Register)				Offset: 0x358	Default: 0x00000000										
Access: DMACx->DMAERRIC															
31		30		29		28		27		26		25		24	
RESERVED															
23		22		21		20		19		18		17		16	
RESERVED															
15		14		13		12		11		10		9		8	
RESERVED															
7		6		5		4		3		2		1		0	
CH7		CH6		CH5		CH4		CH3		CH2		CH1		CH0	

Table 26-57: DMA Error Interrupt Clear Register (DMAERRIC) Description

Bits	Field Name	Type	Reset	Description
31:8	RESERVED_31_8	RO	0x0	Reserved.
7	CH7	W1S	0x0	Channel 7 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
6	CH6	W1S	0x0	Channel 6 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
5	CH5	W1S	0x0	Channel 5 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
4	CH4	W1S	0x0	Channel 4 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
3	CH3	W1S	0x0	Channel 3 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
2	CH2	W1S	0x0	Channel 2 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
1	CH1	W1S	0x0	Channel 1 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.
0	CH0	W1S	0x0	Channel 0 error interrupt clear 0: Write a 0 has no effect and always reads back a 0. 1: Write a 1 clears the interrupt raw flag and flag. This bit is self-cleared to 0.

Table 26-58: DMA Combined Interrupt Flag Register (DMAIF) Layout

DMAIF (DMA Combined Interrupt Flag Register) Offset: 0x360 Default: 0x00000000							
Access: DMACx->DMAIF							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED			ERROR	DC	SC	BC	TC

Table 26-59: DMA Combined Interrupt Flag Register (DMAIF) Description

Bits	Field Name	Type	Reset	Description
31:5	RESERVED_31_5	RO	0x0	Reserved.
4	ERROR	RO	0x0	Global error interrupt flag. This bit is the logic OR of DMAERRIF bits 0: No error interrupt was generated 1: Error interrupt was generated
3	DC	RO	0x0	Global destination transfer complete interrupt flag This bit is the logic OR of DMATCIF bits 0: No transfer complete interrupt was generated 1: Transfer complete interrupt was generated
2	SC	RO	0x0	Global source transfer complete interrupt flag This bit is the logic OR of DMATCIF bits 0: No transfer complete interrupt was generated 1: Transfer complete interrupt was generated
1	BC	RO	0x0	Global block transfer complete interrupt flag This bit is the logic OR of DMATCIF bits 0: No transfer complete interrupt was generated 1: Transfer complete interrupt was generated
0	TC	RO	0x0	Global transfer complete interrupt flag This bit is the logic OR of DMATCIF bits 0: No transfer complete interrupt was generated 1: Transfer complete interrupt was generated

Table 26-60: DMA Source Software Transaction Request Register (DMASSREQ) Layout

DMASSREQ (DMA Source Software Transaction Request Register) Offset: 0x368 Default: 0x00000000 Access: DMACx->DMASSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-61: DMA Source Software Transaction Request Register (DMASSREQ) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 enable. 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 enable. 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 enable. 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 enable. 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 enable. 0: Disable 1: Enable

Bits	Field Name	Type	Reset	Description
2	CH2	RW	0x0	Channel 2 enable. 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 enable. 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 enable. 0: Disable 1: Enable

Table 26-62: DMA Destination Software Transaction Request Register (DMADSREQ) Layout

DMADSREQ (DMA Destination Software Transaction Request Register) Offset: 0x370 Default: 0x00000000 Access: DMACx->DMADSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-63: DMA Destination Software Transaction Request Register (DMADSREQ) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
10	CH2WE	W1S	0x0	CH2 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 enable. 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 enable. 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 enable. 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 enable. 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 enable. 0: Disable 1: Enable
2	CH2	RW	0x0	Channel 2 enable. 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 enable. 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 enable. 0: Disable 1: Enable

Table 26-64: DMA Source Software Transaction Single Request Register (DMASSSREQ) Layout

DMASSSREQ (DMA Source Software Transaction Single Request Register) Offset: 0x378 Default: 0x00000000 Access: DMACx->DMASSSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-65: DMA Source Software Transaction Single Request Register (DMASSSREQ) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
7	CH7	RW	0x0	Channel 7 enable. 0: Source burst request 1: Source single request
6	CH6	RW	0x0	Channel 6 enable. 0: Source burst request 1: Source single request
5	CH5	RW	0x0	Channel 5 enable. 0: Source burst request 1: Source single request
4	CH4	RW	0x0	Channel 4 enable. 0: Source burst request 1: Source single request
3	CH3	RW	0x0	Channel 3 enable. 0: Source burst request 1: Source single request
2	CH2	RW	0x0	Channel 2 enable. 0: Source burst request 1: Source single request
1	CH1	RW	0x0	Channel 1 enable. 0: Source burst request 1: Source single request
0	CH0	RW	0x0	Channel 0 enable. 0: Source burst request 1: Source single request

Table 26-66: DMA Destination Software Transaction Single Request Register (DMADSSREQ) Layout

DMADSSREQ (DMA Destination Software Transaction Single Request Register) Offset: 0x380 Default: 0x00000000 Access: DMACx->DMADSSREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
CH7WE	CH6WE	CH5WE	CH4WE	CH3WE	CH2WE	CH1WE	CH0WE
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

Table 26-67: DMA Destination Software Transaction Single Request Register (DMADSSREQ)
Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 enable. 0: Destination burst request 1: Destination single request
6	CH6	RW	0x0	Channel 6 enable. 0: Destination burst request 1: Destination single request
5	CH5	RW	0x0	Channel 5 enable. 0: Destination burst request 1: Destination single request
4	CH4	RW	0x0	Channel 4 enable. 0: Destination burst request 1: Destination single request
3	CH3	RW	0x0	Channel 3 enable. 0: Destination burst request 1: Destination single request

Bits	Field Name	Type	Reset	Description
2	CH2	RW	0x0	Channel 2 enable. 0: Destination burst request 1: Destination single request
1	CH1	RW	0x0	Channel 1 enable. 0: Destination burst request 1: Destination single request
0	CH0	RW	0x0	Channel 0 enable. 0: Destination burst request 1: Destination single request

Table 26-68: DMA Source Software Transaction Last Request Register (DMASSLREQ) Layout

DMASSLREQ (DMA Source Software Transaction Last Request Register) Offset: 0x388 Default: 0x00000000 Access: DMACx->DMASSLREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-69: DMA Source Software Transaction Last Request Register (DMASSLREQ) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
10	CH2WE	W1S	0x0	CH2 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 enable. 0: Not last transaction in current block 1: Last transaction in current block
6	CH6	RW	0x0	Channel 6 enable. 0: Not last transaction in current block 1: Last transaction in current block
5	CH5	RW	0x0	Channel 5 enable. 0: Not last transaction in current block 1: Last transaction in current block
4	CH4	RW	0x0	Channel 4 enable. 0: Not last transaction in current block 1: Last transaction in current block
3	CH3	RW	0x0	Channel 3 enable. 0: Not last transaction in current block 1: Last transaction in current block
2	CH2	RW	0x0	Channel 2 enable. 0: Not last transaction in current block 1: Last transaction in current block
1	CH1	RW	0x0	Channel 1 enable. 0: Not last transaction in current block 1: Last transaction in current block
0	CH0	RW	0x0	Channel 0 enable. 0: Not last transaction in current block 1: Last transaction in current block

Table 26-70: DMA Destination Software Transaction Last Request Register (DMADSLREQ) Layout

DMADSLREQ (DMA Destination Software Transaction Last Request Register) Offset: 0x390 Default: 0x00000000 Access: DMACx->DMADSLREQ							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-71: DMA Destination Software Transaction Last Request Register (DMADSLREQ) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
10	CH2WE	W1S	0x0	CH2 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable. 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 enable. 0: Not last transaction in current block 1: Last transaction in current block

Bits	Field Name	Type	Reset	Description
6	CH6	RW	0x0	Channel 6 enable. 0: Not last transaction in current block 1: Last transaction in current block
5	CH5	RW	0x0	Channel 5 enable. 0: Not last transaction in current block 1: Last transaction in current block
4	CH4	RW	0x0	Channel 4 enable. 0: Not last transaction in current block 1: Last transaction in current block
3	CH3	RW	0x0	Channel 3 enable. 0: Not last transaction in current block 1: Last transaction in current block
2	CH2	RW	0x0	Channel 2 enable. 0: Not last transaction in current block 1: Last transaction in current block
1	CH1	RW	0x0	Channel 1 enable. 0: Not last transaction in current block 1: Last transaction in current block
0	CH0	RW	0x0	Channel 0 enable. 0: Not last transaction in current block 1: Last transaction in current block

Table 26-72: DMA Enable Register (DMAEN) Layout

DMAEN (DMA Enable Register) Offset: 0x398 Default: 0x00000000							
Access: DMACx->DMAEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15	14	13	12	11	10	9	8
RESERVED							
7	6	5	4	3	2	1	0
RESERVED							EN

Table 26-73: DMA Enable Register (DMAEN) Description

Bits	Field Name	Type	Reset	Description
31:1	RESERVED_31_1	RO	0x0	Reserved.
0	EN	RW	0x0	DMA enable, If the global channel enable bit is cleared while any channel is still active, then DMAEN.EN still returns 1 to indicate that there are channels still active until hardware has terminated all activity on all channels, at which point the DMAEN.EN bit returns 0. 0: Disable 1: Enable

Table 26-74: DMA Channel Enable Register (DMACHEN) Layout

DMACHEN (DMA Channel Enable Register) Offset: 0x3A0 Default: 0x00000000 Access: DMACx->DMACHEN							
31	30	29	28	27	26	25	24
RESERVED							
23	22	21	20	19	18	17	16
RESERVED							
15 CH7WE	14 CH6WE	13 CH5WE	12 CH4WE	11 CH3WE	10 CH2WE	9 CH1WE	8 CH0WE
7 CH7	6 CH6	5 CH5	4 CH4	3 CH3	2 CH2	1 CH1	0 CH0

Table 26-75: DMA Channel Enable Register (DMACHEN) Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15	CH7WE	W1S	0x0	CH7 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
14	CH6WE	W1S	0x0	CH6 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
13	CH5WE	W1S	0x0	CH5 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
12	CH4WE	W1S	0x0	CH4 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
11	CH3WE	W1S	0x0	CH3 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.

Bits	Field Name	Type	Reset	Description
10	CH2WE	W1S	0x0	CH2 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
9	CH1WE	W1S	0x0	CH1 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
8	CH0WE	W1S	0x0	CH0 bit write enable 0: Disable 1: Enable. This bit is self-cleared to 0.
7	CH7	RW	0x0	Channel 7 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
6	CH6	RW	0x0	Channel 6 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
5	CH5	RW	0x0	Channel 5 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
4	CH4	RW	0x0	Channel 4 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
3	CH3	RW	0x0	Channel 3 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
2	CH2	RW	0x0	Channel 2 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
1	CH1	RW	0x0	Channel 1 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable
0	CH0	RW	0x0	Channel 0 enable, automatically cleared if DMA transfer finished. 0: Disable 1: Enable