A31G226, A31G224
User's Manual

32-bit Cortex-M0+ based General Purpose Microcontroller
Flash 128/256KB, SRAM 20KB, Data Flash 32KB, ADC, Comparator, LCD

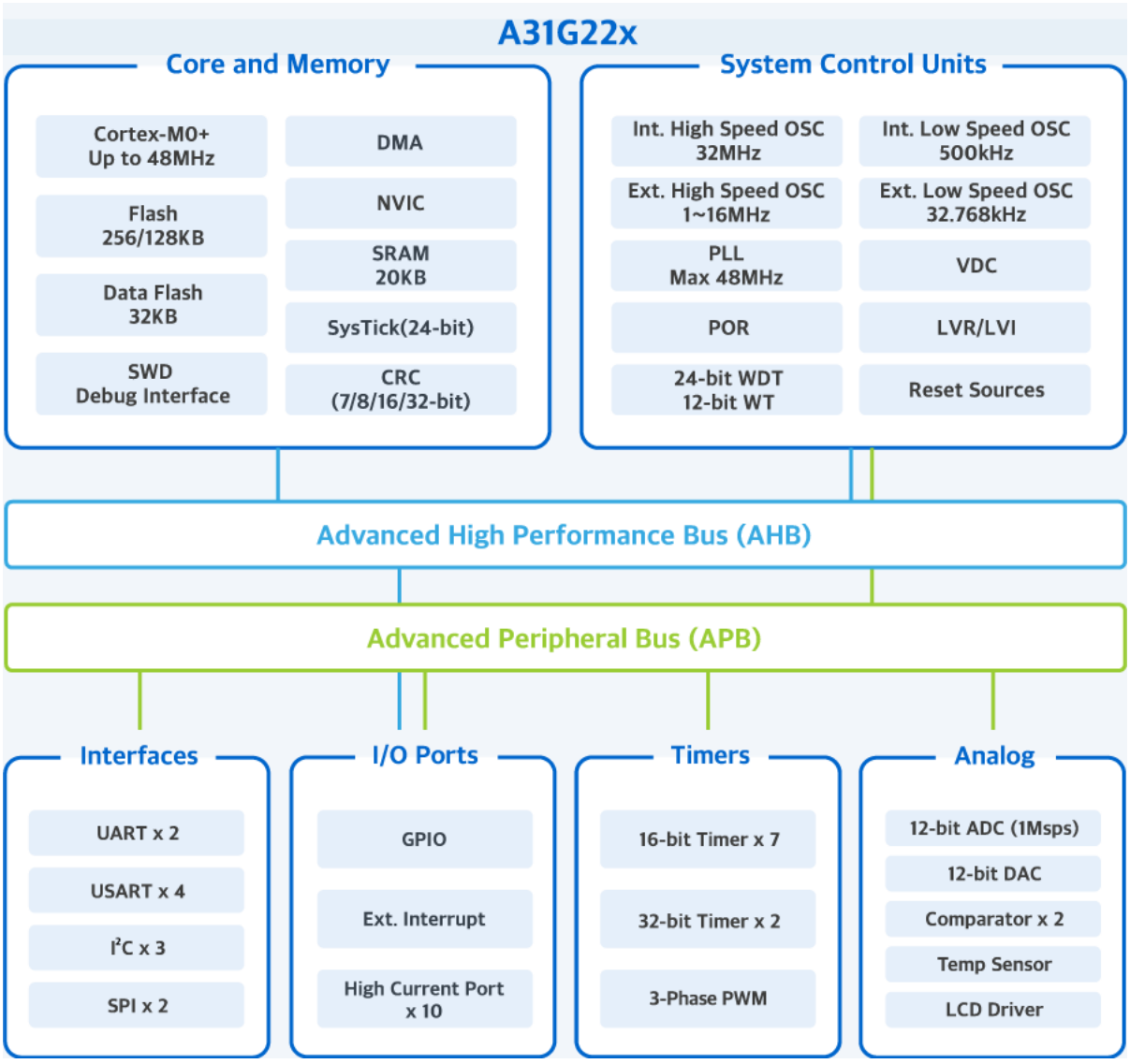
User's Manual Version 1.17

Introduction

This user's manual contains complete information for application developers who use A31G226 or A31G224 for their specific needs. A31G22x device is a 32-bit general purpose microcontroller for various appliances such as white goods and battery-powered equipment.

To meet the requirements for the complexity and high performance in consumer electronics, A31G22x is equipped with high performance low power Cortex-M0+ core, up to 256KB code flash memory, 32KB data flash memory, 20KB SRAM, and System Control Unit.

ARM®'s high-speed 32-bit Cortex®-M0+ retains full instruction set and tool compatibility, while further reducing energy consumption and increasing performance.



Reference document

- Document ‘DDI 0484C’ is provided by ARM and contains information of Cortex-M0+.

Contents

List of figures

List of tables

1 Description

A31G22x is a 32-bit general purpose microcontroller with up to 256 Kbytes of flash memory. It is a powerful microcontroller which provides effective solutions to various electrical appliances which require both low power consumption and high performance.

1.1 Features

In this section, features of A31G22x and peripheral counts are introduced.

Table 1. A31G22x Device Features and Peripheral Counts

Peripherals		Description
Core	CPU	• High Performance Low-Power Cortex-M0+ Core • 32-bit ARM Cortex-M0+ CPU • Uses general-purpose registers specified by the 32-bit Thumb®-2 instruction set • Data ordering format : Little-Endian • Von-Neumann Architecture • AHB-Lite / APB • 24-bit SYSTICK timer • SWD Debugger
	Interrupt	• NVIC (Nested-Vectored Interrupt Controller) • Up to 32 peripheral interrupts supported • Can be assigned with 4 different priority levels
Memory	Code flash	• A31G226: 256 Kbytes code flash memory • A31G224: 128 Kbytes code flash memory • Flash access wait — 0 clock wait: up to 20MHz — 1 clock wait: up to 40MHz — 2 clock wait: up to 48MHz • Supports 512B, 1KB, and 4KB erase • Supports bulk erase • Supports Read protection and Write Protection • Memory Bank Swap • Supports self-programming • Endurance: 10,000 Cycle at room temperature • Retention for 10 years

Table 2. A31G22x Device Features and Peripheral Counts

Peripherals		Description
Memory	Data flash	• 32 KB data flash memory • Supports 512B, 1KB, and 4KB sector erase • Supports bulk erase • Supports read protection • Supports write protection • Supports byte-unit (8-bit) programming • Supports word-unit (32-bit) programming • Endurance: 100,000 Cycle at room temperature • Retention for 10 years
	SRAM	• 20 Kbytes SRAM (0 wait) • Usable as a program work area • Supports code execution in SRAM
System Control Unit (SCU)	Operating frequency	• Up to 48 MHz
	Clock	• High speed internal oscillator (HSI) — 32MHz (±1.0% @ -20℃ to +85℃) — 32MHz (±1.5% @ -40℃ to +105℃) • Low speed internal oscillator (LSI) — 500kHz (±20% @ -40℃ to +105℃) • External main oscillator (HSE): 1MHz to 16MHz • External sub-oscillator (LSE): 32.768kHz • Phase-locked loop (PLL): up to 48 MHz
	Clock monitoring	• System Fail-Safe function by Clock Monitoring — External main oscillator (HSE) — External sub oscillator (LSE) — Main system clock (MCLK)
	Operating mode	• RUN mode • SLEEP mode • DEEP-SLEEP mode
	Reset	• nRESET pin reset • Core reset • Software reset • POR (Power On Reset) • LVR (Low Voltage Reset) • WTR (Watch Timer Reset) • WDTR (Watch Dog Timer Reset) • Reset due to clock oscillating error

Table 2. A31G22x Device Features and Peripheral Counts

Peripherals		Description
System Control Unit (SCU)	Wake-up	• Systick Timer (In SLEEP mode) • general-purpose input/output (GPIO) • Watch Dog Timer (WDT) • Watch Timer (WT) • Low-Voltage Indicator (LVI)
General Purpose I/O (GPIO)		• 75 Ports (PA[11:0], PB[15:0], PC[12:0], PD[5:0], PE[15:0], PF[11:0]): 80-Pin • 59 Ports (PA[11:0], PB[11:0], PC[6:0], PC[12:11], PD[5:0], PE[11:0], PF[7:0]) : 64-pin • 43 Ports (PA[8:0], PB[7:0], PC[4:0], PD[5:0], PE[7:0], PF[7:0]) : 48-Pin • 10-ch sink type high current output ports for driving high currents like LEDs • Strength ports for high speed SPI communication • The use of each pin can be set by setting the mux • Each pin can be configured as an external interrupt source, either the high/low level interrupt or the rising-/falling-edge interrupt • Pull-up/pull-down/debouncing can be set for each pin • Each pin bit can be individually set/reset • Supports wake-up events triggered by external asynchronous inputs
Direct Memory Access Controller (DMA)		• 8-ch direct memory access (DMA) support peripherals • UART0, UART1, CRC, SPI20, SPI21, ADC, DAC, USART10, USART11, USART12, USART13 • DMAC directions — Memory to Peripheral (Tx) — Peripheral to Memory (Rx) — Peripheral to- Peripheral (P2P)
Watch Timer (WT)		• 14-bit divider with extended 12-bit counter 1-ch • Wake up and periodic interrupts are supported
Watchdog Timer (WDT)		• 24-bit down counter timer: 1-ch • Reset, Wake up and periodic interrupts are supported
TIMER	Timer1x	• 16bit: 7-ch — Periodic timer mode, One-shot timer mode, PWM pulse mode, Capture mode • 12-bit prescaler • Synchronous start and clear function with TIMER30

Table 2. A31G22x Device Features and Peripheral Counts

Peripherals		Description
TIMER	Timer2x	• 32bit: 2-ch — Periodic timer mode, One-shot timer mode, PWM pulse mode, Capture mode • 12-bit prescaler
PWM	Timer30	• 16bit: 3-phase complementary PWM Outputs — Periodic timer mode, Back-to-Back mode, Capture mode — 12-bit prescaler — Synchronous start and clear function with TIMERn (n=10 ~ 16)
Comm. function	UART	• Up to 2-ch
	USART	• Up to 4-ch — 80-pin: 4-ch — 64-pin: 3-ch — 48-pin: 3-ch
	SPI	• 2-ch • USART10, USART11 Port Re-Mapping • Strength control for high speed communication
	I2C	• Up to 3-ch — 80-pin: 3-ch — 64-pin: 3-ch — 48-pin: 2-ch
ADC		• 12-bit ADC: 1Msps — 80-pin: 18-ch — 64-pin: 18-ch — 48-pin: 14-ch • ADC Trigger sources — TIMER1n (n=10 to 16) — TIMER30 — ADC Clock
DAC		• 12-bit DAC 1-ch — PA6: output pin
CRC calculator		• CRC-32 / CRC-16 / CRC-8 / CRC-7 • DMA transmission
Comparator		• 2-unit comparators • 1-ch reference input of each comparator — External Input channel — Internal Input • 1-ch output of each comparator

Table 2. A31G22x Device Features and Peripheral Counts

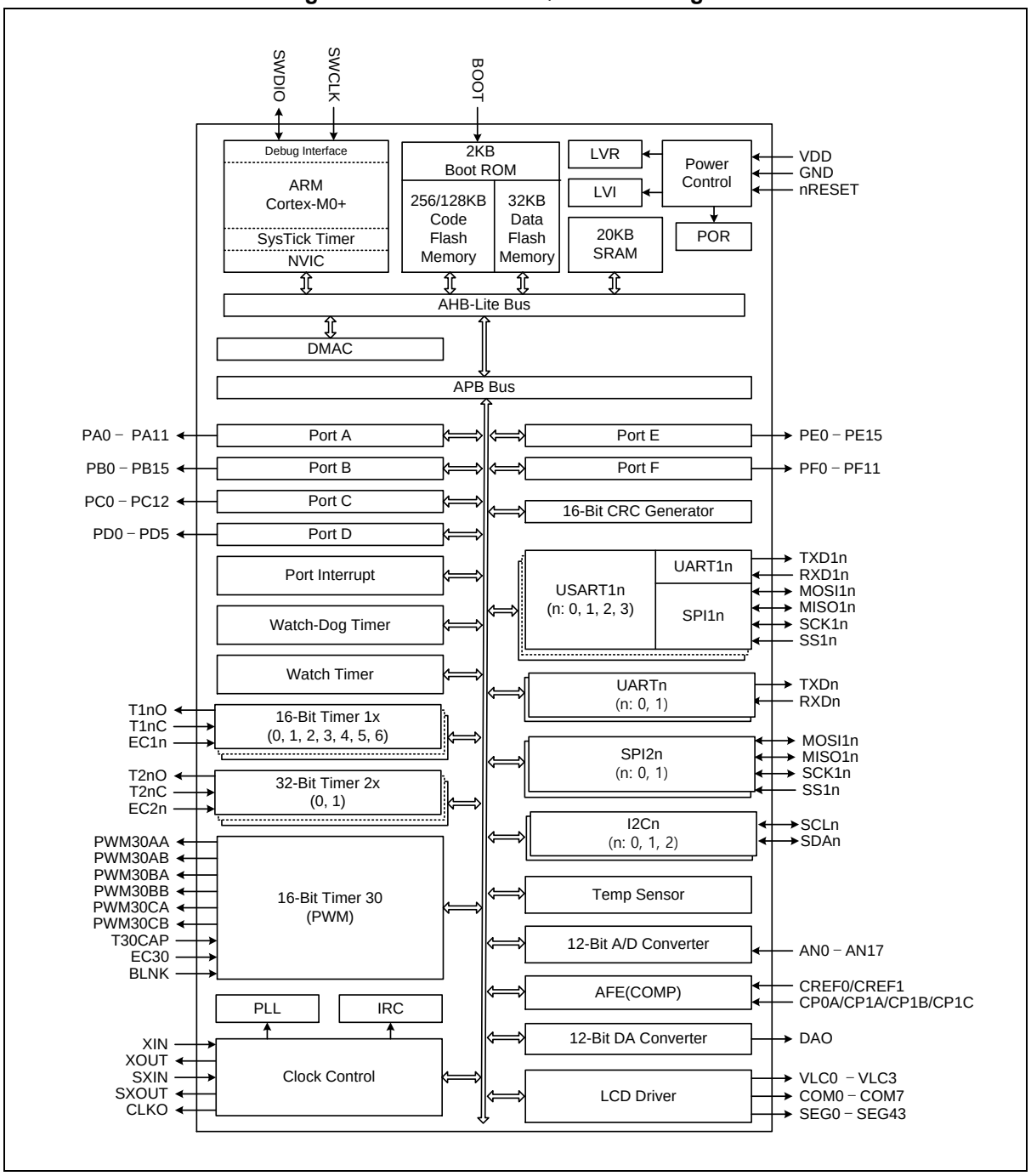
Peripherals	Description
Temperature Sensor	• 20-bit reference counter and 16-bit sensing counter • Internal oscillation for temperature sensing (LSITS)
LCD driver	• 80-pin — 8-COM x 37-SEG — 3-COM x 42-SEG • 64-pin — 8-COM x 29-SEG — 3-COM x 34-SEG • 48-pin — 8-COM x 21-SEG — 3-COM x 26-SEG • 3.3V external bias source voltage
Operating Voltage	• 1.8V to 5.5V
Operating temperature	• Commercial grade (-40℃ to +85℃) • Industrial grade (-40℃ to +105℃)
Package	• Three types of package options — 48-pin LQFP (0.50mm pitch) — 64-pin LQFP (0.50mm, 0.65mm pitch) — 80-pin LQFP (0.50mm, 0.65mm pitch)

1.2 Block diagram

In this section, A31G22x device with peripherals is described in a block diagram.

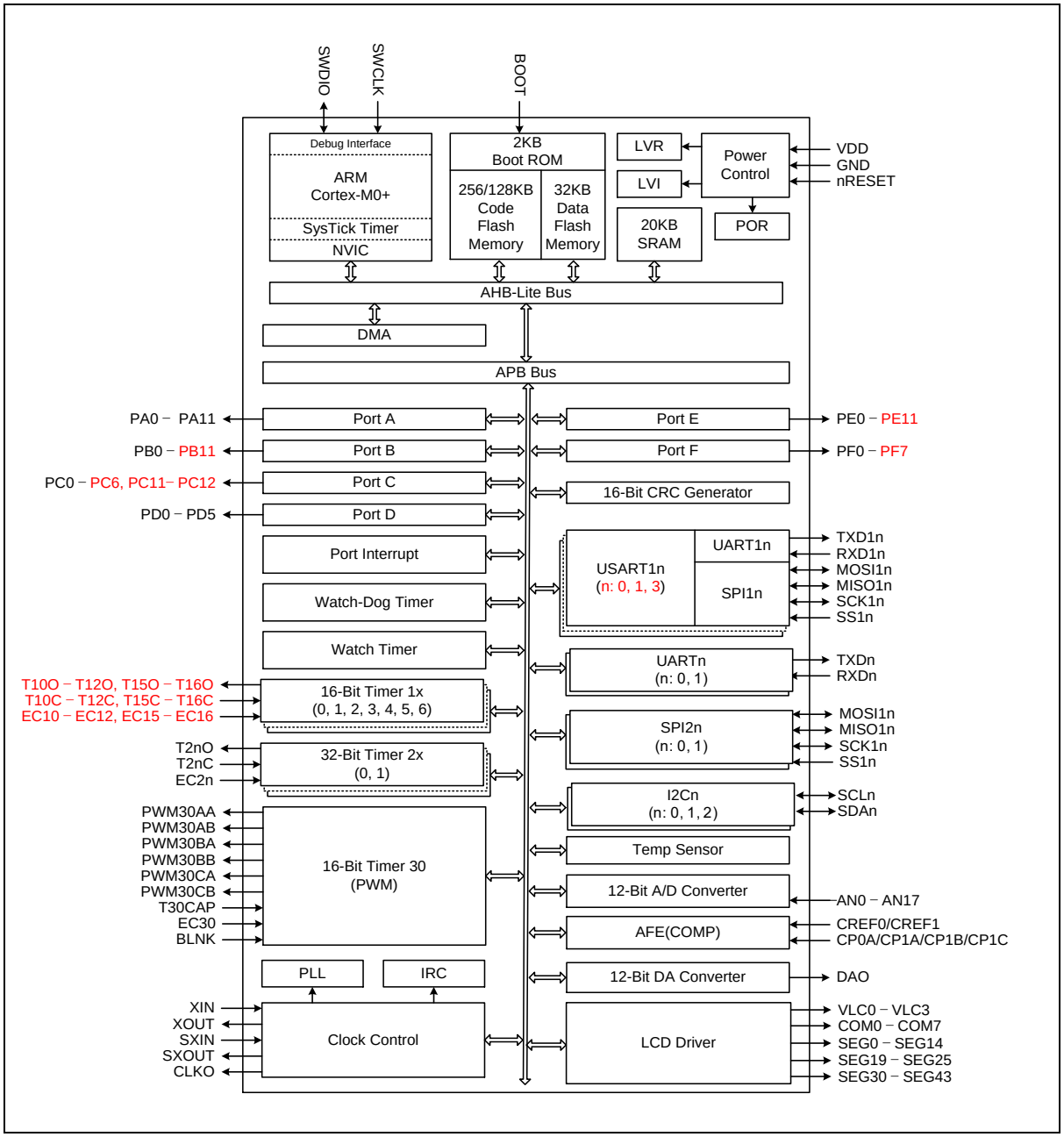
1.2.1 A31G226MM2N/ A31G226ML2N/ A31G224MM2N/ A31G224ML2N/ A31G226MMN/ A31G226MLN/ A31G224MMN/ A31G224MLN (80-LQFP)

Figure 1. A31G22x 80-LQFP Block Diagram



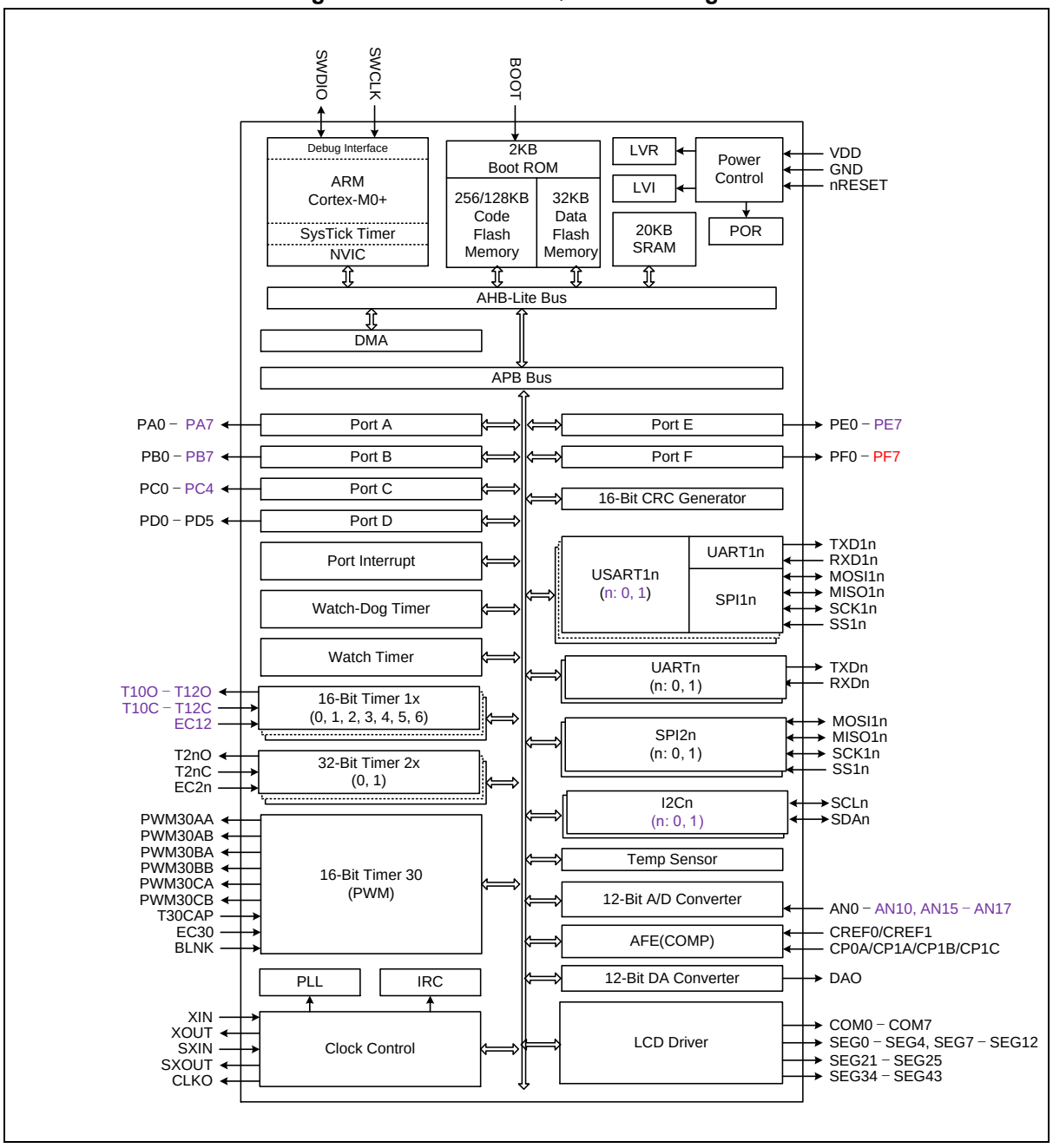
1.2.2 A31G226RM2N/ A31G226RL2N/ A31G224RM2N/ A31G224RL2N/ A31G226RMN/ A31G226RLN/ A31G224RMN/ A31G224RLN (64-LQFP)

Figure 2. A31G22x 64-LQFP Block Diagram



1.2.3 A31G226CL2N/ A31G224CL2N/ A31G226CLN/ A31G224CLN (48-LQFP)

Figure 3. A31G22x 48-LQFP Block Diagram



2 Pinouts and pin descriptions

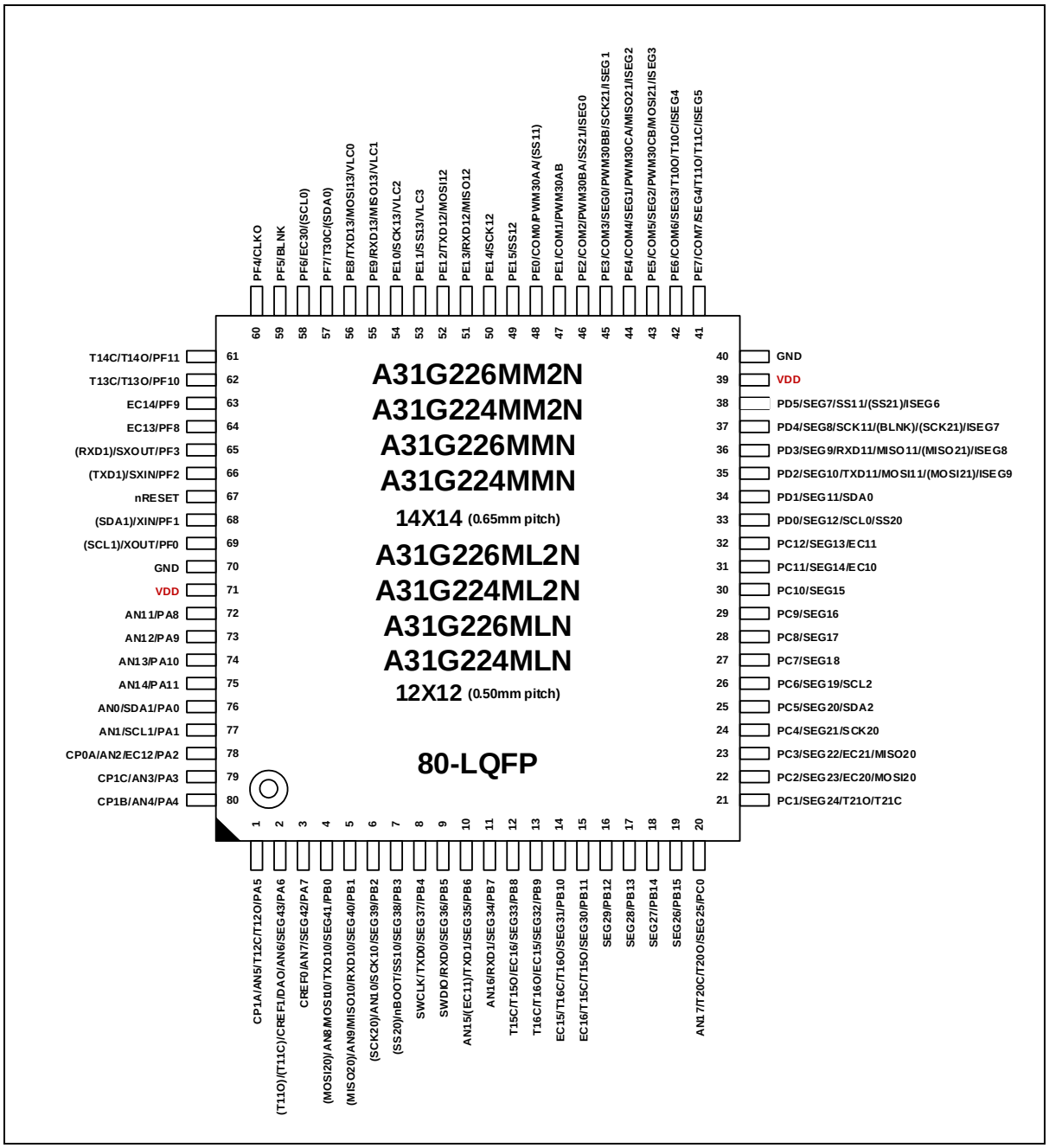
In this chapter, A31G22x devices' pinouts and pin descriptions are introduced.

2.1 Pinouts

2.1.1 A31G226MM2N/ A31G226ML2N/ A31G224MM2N/ A31G224ML2N/ A31G226MMN/ A31G226MLN/ A31G224MMN/ A31G224MLN (LQFP 80)

Pinouts of A31G22x 80-LQFP package are showed in Figure 4.

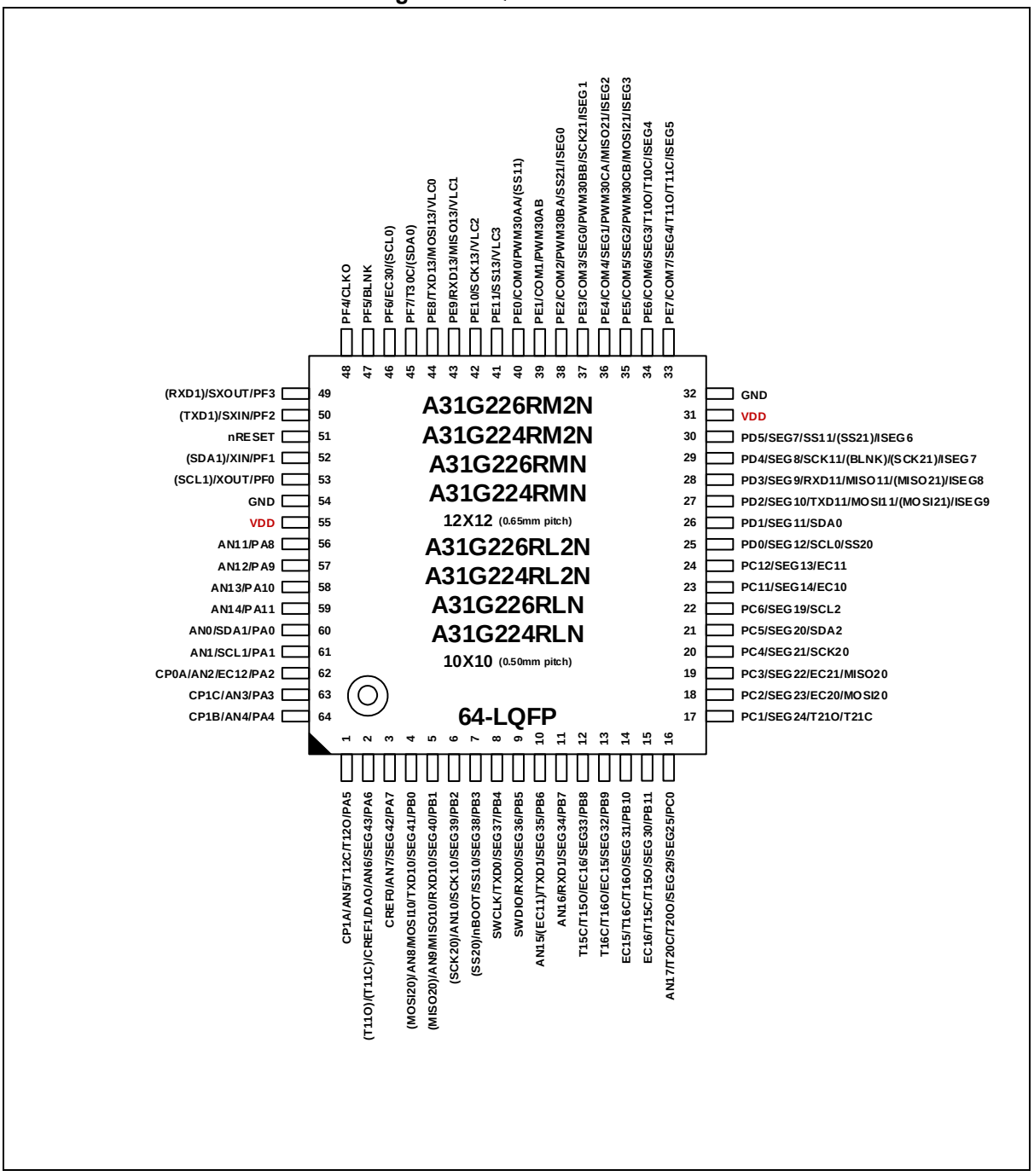
Figure 4. LQFP 80 Pinouts



2.1.2 A31G226RM2N/ A31G226RL2N/ A31G224RM2N/ A31G224RL2N/ A31G226RMN/ A31G226RLN/ A31G224RMN/ A31G224RLN (64-LQFP)

Pinouts of A31G22x 64-LQFP package are showed in Figure 5.

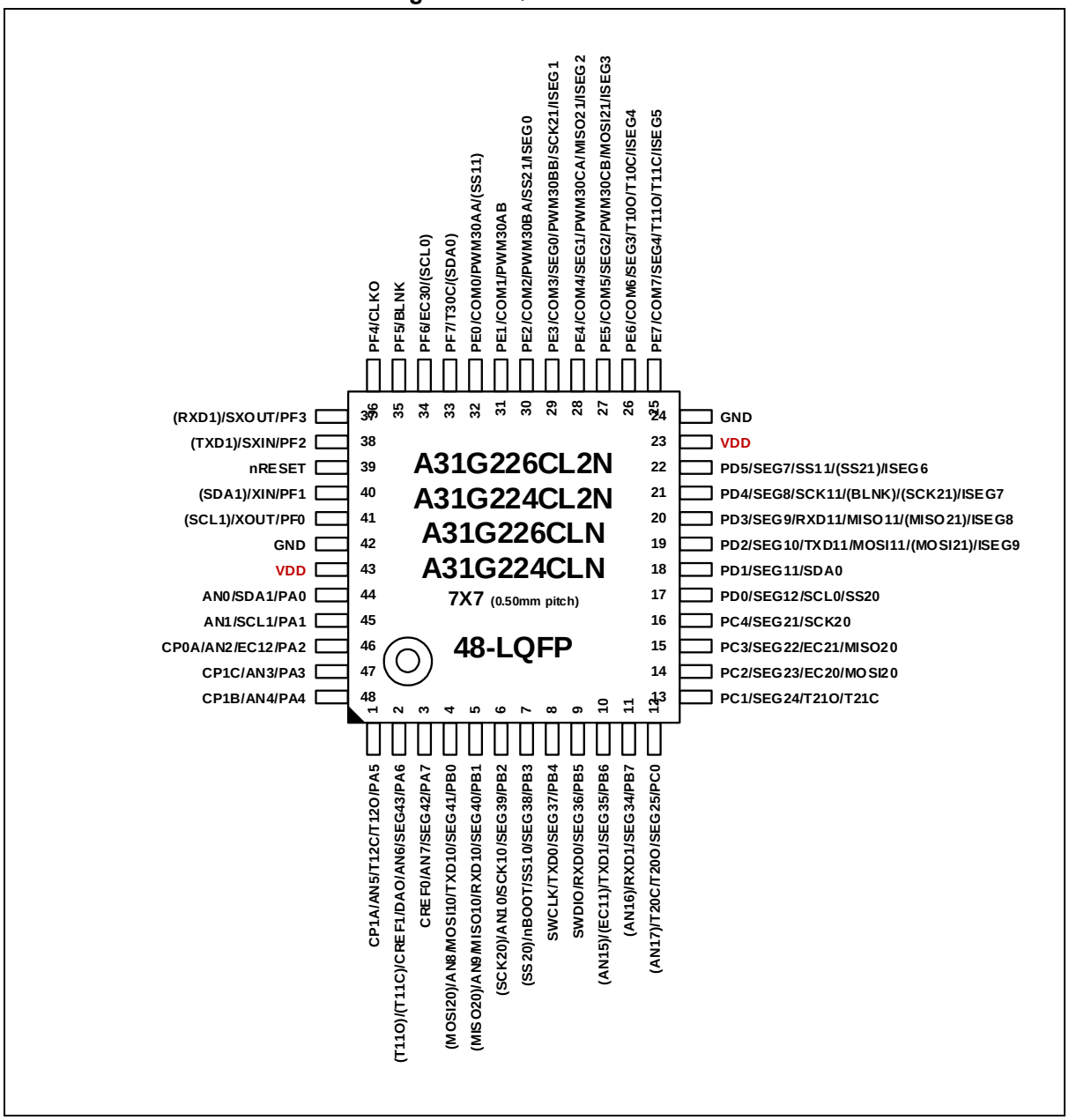
Figure 5. LQFP 64 Pinouts



2.1.3 A31G226CL2N/ A31G224CL2N/ A31G226CLN/ A31G224CLN (48-LQFP)

Pinouts of A31G22x 48-LQFP are showed in Figure 6.

Figure 6. LQFP 48 Pinouts



2.2 Pin description

Pin configuration information in Table 3 contains two pairs of power/ground and other dedicated pins. These multi-function pins have up to ten selections of functions including GPIO. Configuration including pin ordering can be changed without notice.

Table 3. Pin Description

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
1	1	1	PA5*	IOUDS	PORT A Bit 5 Input/Output	
			T12O	O	Timer 12 Output	
			T12C	I	Timer 12 Capture Input	
			AN5	IA	Analog Input 5	
			CP1A	IA	Comparator input 1A	
2	2	2	PA6*	IOUDS	PORT A Bit 6 Input/Output	
			SEG43	O	LCD Segment Signal 43 Output	
			T11O	O	Timer 11 Output	
			T11C	I	Timer 11 Capture Input	
			AN6	IA	Analog Input 6	
			DAO	OA	Digital to analog output	
			CREF1	IA	Comparator 1 Reference Input	
3	3	3	PA7*	IOUDS	PORT A Bit 7 Input/Output	
			SEG42	O	LCD Segment Signal 42 Output	
			AN7	IA	Analog Input 7	
			CREF0	IA	Comparator 0 Reference Input	
4	4	4	PB0	IOUDS	PORT B Bit 0 Input/Output	
			SEG41	O	LCD Segment Signal 41 Output	
			TXD10*	O	UART Channel 10 TxD Input	
			MOSI10	I/O	SPI Channel 10 Master Out / Slave In	
			MOSI20	I/O	SPI Channel 20 Master Out / Slave In	
			AN8	IA	Analog Input 8	
5	5	5	PB1	IOUDS	PORT B Bit 1 Input/Output	
			SEG40	O	LCD Segment Signal 40 Output	
			RXD10*	I	UART Channel 10 RxD Input	
			MISO10	I/O	SPI10 Master-Input/Slave-Output Data signal	
			MISO20	I/O	SPI20 Master-Input/Slave-Output Data signal	
			AN9	IA	Analog Input 9	
6	6	6	PB2*	IOUDS	PORT B Bit 2 Input/Output	
			SEG39	O	LCD Segment Signal 39 Output	
			SCK10	I/O	SPI10 Data Clock Input/Output	
			SCK20	I/O	SPI20 Data Clock Input/Output	
			AN10	IA	Analog Input 10	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
7	7	7	PB3*	IOUDS	PORT B Bit 3 Input/Output	
			SEG38	O	LCD Segment Signal 38 Output	
			SS10	I/O	SPI10 Slave Select signal	
			SS20	I/O	SPI20 Slave Select signal	
			nBOOT*	I	Boot Mode Selection Pin	Pull-up
8	8	8	PB4	IOUDS	PORT B Bit 4 Input/Output	
			SEG37	O	LCD Segment Signal 37 Output	
			TXD0	O	UART Channel 0 TxD Input	
			SWCLK*	I	SWD Clock Input	Pull-up
9	9	9	PB5	IOUDS	PORT B Bit 5 Input/Output	
			SEG36	O	LCD Segment Signal 36 Output	
			RXD0	I	UART Channel 0 RxD Input	
			SWDIO*	I/O	SWD Data Input/Output	Pull-up
10	10	10	PB6*	IOUDS	PORT B Bit 6 Input/Output	
			SEG35	O	LCD Segment Signal 35 Output	
			TXD1	O	UART Channel 1 TxD Input	
			EC11	I	Timer 11 Event Count Input	
			AN15	IA	Analog Input 15	
11	11	11	PB7*	IOUDS	PORT B Bit 7 Input/Output	
			SEG34	O	LCD Segment Signal 34 Output	
			RXD1	I	UART Channel 1 RxD Input	
			AN16	IA	Analog Input 16	
12	12	-	PB8*	IOUDS	PORT B Bit 8 Input/Output	
			SEG33	O	LCD Segment Signal 33 Output	
			EC16	I	Timer 16 Event Count Input	
			T15O	O	Timer 15 Output	
			T15C	I	Timer 15 Capture Input	
13	13	-	PB9*	IOUDS	PORT B Bit 9 Input/Output	
			SEG32	O	LCD Segment Signal 32 Output	
			EC15	I	Timer 15 Event Count Input	
			T16O	O	Timer 16 Output	
			T16C	I	Timer 16 Capture Input	
14	14	-	PB10*	IOUDS	PORT B Bit 10 Input/Output	
			SEG31	O	LCD Segment Signal 31 Output	
			T16O	O	Timer 16 Output	
			T16C	I	Timer 16 Capture Input	
			EC15	I	Timer 15 Event Count Input	
15	15	-	PB11*	IOUDS	PORT B Bit 11 Input/Output	
			SEG30	O	LCD Segment Signal 30 Output	
			T15O	O	Timer 15 Output	
			T15C	I	Timer 15 Capture Input	
			EC16	I	Timer 16 Event Count Input	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
16	-	-	PB12*	IOUDS	PORT B Bit 12 Input/Output	
			SEG29	O	LCD Segment Signal 29 Output	
17	-	-	PB13*	IOUDS	PORT B Bit 13 Input/Output	
			SEG28	O	LCD Segment Signal 28 Output	
18	-	-	PB14*	IOUDS	PORT B Bit 14 Input/Output	
			SEG27	O	LCD Segment Signal 27 Output	
19	-	-	PB15*	IOUDS	PORT B Bit 15 Input/Output	
			SEG26	O	LCD Segment Signal 26 Output	
20	16	12	PC0*	IOUDS	PORT C Bit 0 Input/Output	
			SEG25	O	LCD Segment Signal 25 Output	
			T20O	O	Timer 20 Output	
			T20C	I	Timer 20 Capture Input	
			AN17	IA	Analog Input 17	
21	17	13	PC1*	IOUDS	PORT C Bit 1 Input/Output	Pull-up
			SEG24	O	LCD Segment Signal 24 Output	
			T21O	O	Timer 21 Output	
			T21C	I	Timer 21 Capture Input	
22	18	14	PC2*	IOUDS	PORT C Bit 2 Input/Output	Pull-up
			SEG23	O	LCD Segment Signal 23 Output	
			EC20	I	Timer 20 Event Count Input	
			MOSI20	I/O	SPI Channel 20 Master Out / Slave In	
23	19	15	PC3*	IOUDS	PORT C Bit 3 Input/Output	
			SEG22	O	LCD Segment Signal 22 Output	
			EC21	I	Timer 21 Event Count Input	
			MISO20	I/O	SPI20 Master-Input/Slave-Output Data signal	
24	20	16	PC4*	IOUDS	PORT C Bit 4 Input/Output	
			SEG21	O	LCD Segment Signal 21 Output	
			SCK20	I/O	SPI20 Data Clock Input/Output	
25	21	-	PC5*	IOUDS	PORT C Bit 5 Input/Output	
			SEG20	O	LCD Segment Signal 20 Output	
			SDA2	O	I^2C Channel 2 SDA In/Out	
26	22	-	PC6*	IOUDS	PORT C Bit 6 Input/Output	
			SEG19	O	LCD Segment Signal 19 Output	
			SCL2	O	I^2C Channel 2 SCL In/Out	
27	-	-	PC7*	IOUDS	PORT C Bit 7 Input/Output	
			SEG18	O	LCD Segment Signal 18 Output	
28	-	-	PC8*	IOUDS	PORT C Bit 8 Input/Output	
			SEG17	O	LCD Segment Signal 17 Output	
29	-	-	PC9*	IOUDS	PORT C Bit 9 Input/Output	
			SEG16	O	LCD Segment Signal 16 Output	
30	-	-	PC10*	IOUDS	PORT C Bit 10 Input/Output	
			SEG15	O	LCD Segment Signal 15 Output	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
31	23	-	PC11*	IOUDS	PORT C Bit 11 Input/Output	
			SEG14	O	LCD Segment Signal 14 Output	
			EC10	I	Timer 10 Event Count Input	
32	24	-	PC12*	IOUDS	PORT C Bit 12 Input/Output	
			SEG13	O	LCD Segment Signal 13 Output	
			EC11	I	Timer 11 Event Count Input	
33	25	17	PD0*	IOUDS	PORT D Bit 0 Input/Output	
			SEG12	O	LCD Segment Signal 12 Output	
			SCL0	O	I^2C Channel 0 SCL In/Out	
			SS20	I/O	SPI Channel 20 Slave Select signal	
34	26	18	PD1*	IOUDS	PORT D Bit 1 Input/Output	
			SEG11	O	LCD Segment Signal 11 Output	
			SDA0	O	I^2C Channel 0 SDA In/Out	
			EC10	I	Timer 10 Event Count Input	
35	27	19	PD2*	IOUDS	PORT D Bit 2 Input/Output	
			SEG10	O	LCD Segment Signal 10 Output	
			TXD11	O	UART Channel 11 TxD Input	
			MOSI11	I/O	SPI Channel 11 Master Out / Slave In	
			MOSI21	I/O	SPI Channel 21 Master Out / Slave In	
			ISEG9	O	Sink Type High Current Output	
36	28	20	PD3*	IOUDS	PORT D Bit 3 Input/Output	
			SEG9	O	LCD Segment Signal 9 Output	
			RXD11	I	UART Channel 11 RxD Input	
			MISO11	I/O	SPI11 Master-Input/Slave-Output Data signal	
			MISO21	I/O	SPI21 Master-Input/Slave-Output Data signal	
			ISEG8	O	Sink Type High Current Output	
37	29	21	PD4*	IOUDS	PORT D Bit 4 Input/Output	
			SEG8	O	LCD Segment Signal 8 Output	
			BLNK	I	External Sync Signal Input for Timer 30 PWM	
			SCK11	I/O	SPI11 Data Clock Input/Output	
			SCK21	I/O	SPI21 Data Clock Input/Output	
			ISEG7	O	Sink Type High Current Output	
38	30	22	PD5*	IOUDS	PORT D Bit 5 Input/Output	
			SEG7	O	LCD Segment Signal 7 Output	
			SS11	I/O	SPI Channel 11 Slave Select signal	
			SS21	I/O	SPI Channel 21 Slave Select signal	
			ISEG6	O	Sink Type High Current Output	
39	31	23	VDD	P	VDD	
40	32	24	VSS	P	VS	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
41	33	25	PE7*	IOUDS	PORT E Bit 7 Input/Output	
			COM7	O	LCD Common Signal 7 Outputs	
			SEG4	O	LCD Segment Signal 4 Output	
			T11O	O	Timer 11 Output	
			T11C	I	Timer 11 Capture Input	
			ISEG5	O	Sink Type High Current Output	
42	34	26	PE6*	IOUDS	PORT E Bit 6 Input/Output	
			COM6	O	LCD Common Signal 6 Output	
			SEG3	O	LCD Segment Signal 3 Output	
			T10O	O	Timer 10 Output	
			T10C	I	Timer 10 Capture Input	
			ISEG4	O	Sink Type High Current Output	
43	35	27	PE5*	IOUDS	PORT E Bit 5 Input/Output	
			COM5	O	LCD Common Signal 5 Output	
			SEG2	O	LCD Segment Signal 2 Output	
			PWM30CB	O	Timer 30 PWM Output	
			MOSI21	I/O	SPI Channel 21 Master-Output / Slave-Input	
			ISEG3	O	Sink Type High Current Output	
44	36	28	PE4*	IOUDS	PORT E Bit 4 Input/Output	
			COM4	O	LCD Common Signal 4 Output	
			SEG1	O	LCD Segment Signal 1 Output	
			PWM30CA	O	Timer 30 PWM Output	
			MISO21	I/O	SPI21 Master-Input/Slave-Output Data signal	
			ISEG2	O	Sink Type High Current Output	
45	37	29	PE3*	IOUDS	PORT E Bit 3 Input/Output	
			COM3	O	LCD Common Signal 3 Output	
			SEG0	O	LCD Segment Signal 0 Output	
			PWM30BB	O	Timer 30 PWM Output	
			SCK21	I/O	SPI20 Data Clock Input/Output	
			ISEG1	O	Sink Type High Current Output	
46	38	30	PE2*	IOUDS	PORT E Bit 2 Input/Output	
			COM2	O	LCD Common Signal 2 Output	
			PWM30BA	O	Timer 30 PWM Output	
			SS21	I/O	SPI Channel 21 Slave Select signal	
			ISEG0	O	Sink Type High Current Output	
47	39	31	PE1*	IOUDS	PORT E Bit 1 Input/Output	
			COM1	O	LCD Common Signal 1 Output	
			PWM30AB	O	Timer 30 PWM Output	
48	40	32	PE0*	IOUDS	PORT E Bit 0 Input/Output	
			COM0	O	LCD Common Signal 0 Output	
			PWM30AA	O	Timer 30 PWM Output	
			SS11	I/O	SPI Channel 11 Slave Select signal	
49	-	-	PE15*	IOUDS	PORT E Bit 15 Input/Output	
			SS12	I/O	SPI Channel 12 Slave Select signal	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
50	-	-	PE14*	IOUDS	PORT E Bit 14 Input/Output	
			SCK12	I/O	SPI12 Data Clock Input/Output	
51	-	-	PE13*	IOUDS	PORT E Bit 13 Input/Output	
			RXD12	I	UART Channel 12 RxD Input	
			MISO12	I/O	SPI12 Master-Input/Slave-Output Data signal	
52	-	-	PE12*	IOUDS	PORT E Bit 12 Input/Output	
			TXD12	O	UART Channel 12 TxD Input	
			MOSI12	I/O	SPI Channel 12 Master Out / Slave In	
53	41	-	PE11*	IOUDS	PORT E Bit 11 Input/Output	
			SS13	I/O	SPI Channel 13 Slave Select signal	
			VLC3	IA	External LCD Voltage bias 3	
54	42	-	PE10*	IOUDS	PORT E Bit 10 Input/Output	
			SCK13	I/O	SPI13 Data Clock Input/Output	
			VLC2	IA	External LCD Voltage bias 2	
55	43	-	PE9*	IOUDS	PORT E Bit 9 Input/Output	
			RXD13	I	UART Channel 13 RxD Input	
			MISO13	I/O	SPI13 Master-Input/Slave-Output Data signal	
			VLC1	IA	External LCD Voltage bias 1	
56	44	-	PE8*	IOUDS	PORT E Bit 8 Input/Output	
			TXD13	O	UART Channel 13 TxD Input	
			MOSI13	I/O	SPI Channel 13 Master Out / Slave In	
			VLC0	IA	External LCD Voltage bias 0	
57	45	33	PF7*	IODS	PORT F Bit 7 Input/Output	Open-drain
			T30C	I	Timer 30 Capture Input	
			(SDA0)	O	I^2C Channel 0 SDA In/Out	
58	46	34	PF6*	IODS	PORT F Bit 6 Input/Output	Open-drain
			EC30	I	Timer 30 Event Count Input	
			(SCL0)	O	I^2C Channel 0 SCL In/Out	
59	47	35	PF5*	IODS	PORT F Bit 5 Input/Output	Open-drain
			BLNK	I	External Sync Signal Input for Timer 30 PWM	
60	48	36	PF4*	IOUDS	PORT F Bit 4 Input/Output	
			CLKO	O	System Clock Output	
61	-	-	PF11*	IOUDS	PORT F Bit 11 Input/Output	
			T14O	O	Timer 14 Output	
			T14C	I	Timer 14 Capture Input	
62	-	-	PF10*	IOUDS	PORT F Bit 10 Input/Output	
			T13O	O	Timer 13 Output	
			T13C	I	Timer 13 Capture Input	
62	-	-	PF10*	IOUDS	PORT F Bit 10 Input/Output	
			T13O	O	Timer 13 Output	
			T13C	I	Timer 13 Capture Input	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
63	-	-	PF9*	IOUDS	PORT F Bit 9 Input/Output	
			EC14	I	Timer 14 Event Count Input	
64	-	-	PF8*	IOUDS	PORT F Bit 8 Input/Output	
			EC13	I	Timer 13 Event Count Input	
65	49	37	PF3*	IOUDS	PORT F Bit 3 Input/Output	
			SXOUT	O	Sub Oscillator Output	
			(RXD1)	I	UART Channel 1 RxD Input	
66	50	38	PF2*	IOUDS	PORT F Bit 2 Input/Output	
			SXIN	I	Sub Oscillator Input	
			(TXD1)	O	UART Channel 1 TxD Input	
67	51	39	nRESET	IU	External Reset Input	Pull-up
68	52	40	PF1*	IOUDS	PORT F Bit 1 Input/Output	
			XIN	I	Main Oscillator Input	
			(SDA1)	O	I^2C Channel 2 SDA In/Out	
69	53	41	PF0*	IOUDS	PORT F Bit 0 Input/Output	
			XOUT	O	Main Oscillator Output	
			(SCL1)	O	I^2C Channel 2 SCL In/Out	
70	54	42	VSS	P	VSS	
71	55	43	VDD	P	VDD	
72	56	-	PA8	IOUDS	PORT A Bit 8 Input/Output	
			AN11	IA	Analog Input 11	
73	57	-	PA9*	IOUDS	PORT A Bit 9 Input/Output	
			AN12	IA	Analog Input 12	
74	58	-	PA10*	IOUDS	PORT A Bit 10 Input/Output	
			AN13	IA	Analog Input 13	
75	59	-	PA11*	IOUDS	PORT A Bit 11 Input/Output	
			AN14	IA	Analog Input 14	
76	60	44	PA0*	IOUDS	PORT A Bit 0 Input/Output	
			SDA1	O	I2C Channel 1 SDA In/Out	Open-drain
			AN0	IA	Analog Input 0	
77	61	45	PA1*	IOUDS	PORT A Bit 1 Input/Output	
			SCL1	O	I2C Channel 1 SCL In/Out	Open-drain
			AN1	IA	Analog Input 1	
78	62	46	PA2*	IOUDS	PORT A Bit 2 Input/Output	
			EC12	I	Timer 12 Event Count Input	
			AN2	IA	Analog Input 2	
			CP0A	IA	Comparator plus input 0	
79	63	47	PA3*	IOUDS	PORT A Bit 3 Input/Output	
			AN3	IA	Analog Input 3	
			CP1C	IA	Comparator plus input 1C	

Table 3. Pin Description (continued)

Pin no.			Pin no.	Type	Description	Remark
80-pin	64-pin	48-pin				
80	64	48	PA4*	IOUDS	PORT A Bit 4 Input/Output	
			AN4	IA	Analog Input 4	
			CP1B	IA	Comparator plus input 1B	

NOTES:

1. I=Input, O=Output, U=Pull-up, D=Pull-down, S=Schmitt-Trigger Input Type, C=CMOS Input Type, A=Analog, P=Power
2. The * means 'Selected pin function after reset condition'.
3. Pin order may be changed with revision notice.
4. PB3 (nBOOT), nRESET, PB5 (SWDIO), PB4 (SWCLK) are default pull-up pins.
5. PC1 and PC2 are default pull-up pins.

3 System and memory overview

3.1 System architecture

Main system of A31G22x series consists of the followings:

- ARM® Cortex® -M0+ core
- General purpose DMAC (Direct Memory Access Controller)
- Internal SRAM
- Internal Code and Data Flash memory
- Two AHB buses

3.1.1 Cortex-M0+ core

The ARM® Cortex®-M0+ processor is the most energy-efficient ARM processor available. It builds on the very successful Cortex-M0+ processor, retaining full instruction set and tool compatibility, while further reducing energy consumption and increasing performance. Document "DDI 0484C" from ARM provides detail information of Cortex-M0+.

3.1.2 Interrupt controller

Table 4. Interrupt Vector Map

Priority	Vector Address	Interrupt Source
-16	0x0000_0000	Stack Pointer
-15	0x0000_0004	Reset Address
-14	0x0000_0008	NMI Handler
-13	0x0000_000C	Hard Fault Handler
-12	0x0000_0010	Reserved
-11	0x0000_0014	
-10	0x0000_0018	
-9	0x0000_001C	
-8	0x0000_0020	
-7	0x0000_0024	
-6	0x0000_0028	
-5	0x0000_002C	SVCall Handler
-4	0x0000_0030	Reserved
-3	0x0000_0034	

Table 4. Interrupt Vector Map

Priority	Vector Address	Interrupt Source
-2	0x0000_0038	PenSV Handler
-1	0x0000_003C	SysTick Handler
0	0x0000_0040	LVI
1	0x0000_0044	SYSCLKFAIL
2	0x0000_0048	WDT
3	0x0000_004C	GPIOA, GPIOB
4	0x0000_0050	GPIOC, GPIOD
5	0x0000_0054	GPIOE
6	0x0000_0058	GPIOF
7	0x0000_005C	TIMER10
8	0x0000_0060	TIMER11
9	0x0000_0064	TIMER12
10	0x0000_0068	I2C0
11	0x0000_006C	USART10
12	0x0000_0070	WT
13	0x0000_0074	TIMER30
14	0x0000_0078	I2C1
15	0x0000_007C	TIMER20
16	0x0000_0080	TIMER21
17	0x0000_0084	USART11
18	0x0000_0088	ADC
19	0x0000_008C	UART0
20	0x0000_0090	UART1
21	0x0000_0094	TIMER13
22	0x0000_0098	TIMER14
23	0x0000_009C	TIMER15
24	0x0000_00A0	TIMER16
25	0x0000_00A4	I2C2, SPI20
26	0x0000_00A8	USART12, USART13, SPI21
27	0x0000_00AC	DAC
28	0x0000_00B0	TS
29	0x0000_00B4	Reserved
30	0x0000_00B8	Reserved
31	0x0000_00BC	CMP, CRC

NOTES:

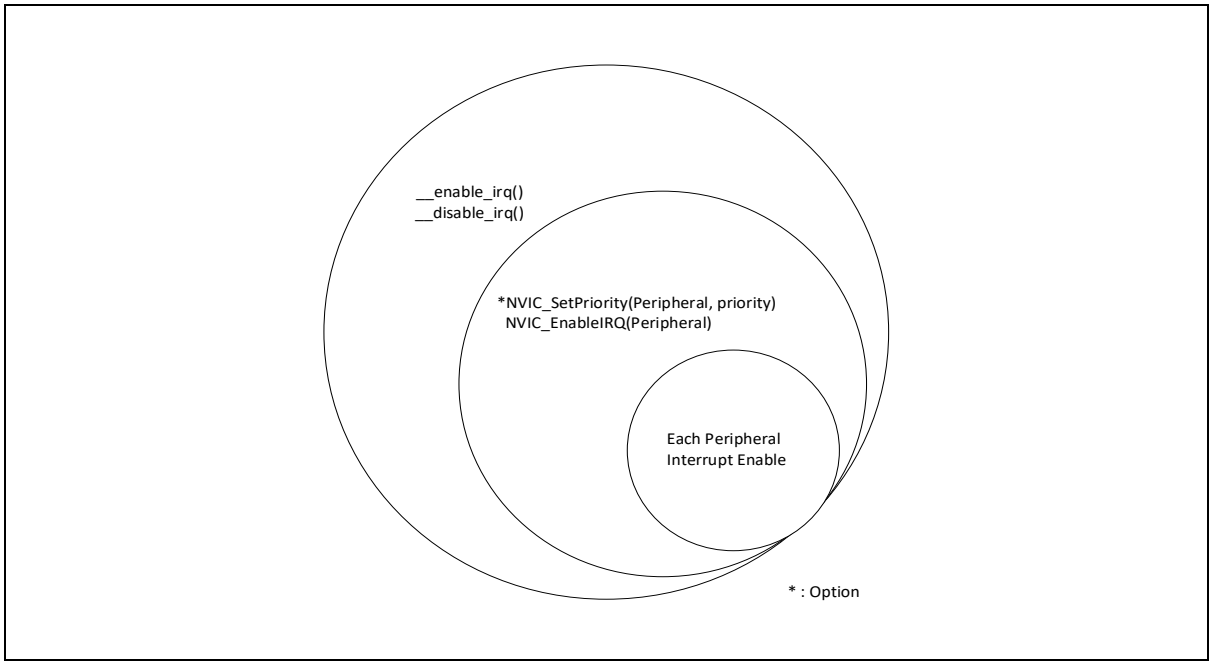
1. Each interrupt has an associated priority-level register. Each of them is 2 bits wide, occupying the two MSBs of the Interrupt Priority Level Registers. Each Interrupt Priority Level Register occupies 1 byte (8 bits).NVIC registers in the Cortex-M0+ processor can only be accessed using word-size transfers, so for each access, four Interrupt Priority Level Registers are accessed at the same time.

   ** __NVIC_PRIO_BITS = 2

2. Figure 7 is a caution when using Peripheral Interrupts. Interrupt don't work if only peripheral interrupts are enabled. Enable the function in core to enable interrupt.

   * __enable_irq > NVCI_EnableIRQ(Peripheral) > Each Peripheral Interrupt

Figure 7. Interrupt Block Diagram



3.2 Memory organization

Program memory, data memory, registers and I/O ports are organized in the same address space.

3.2.1 Register boundary address

Table 5 gives the boundary addresses of peripherals in A31G22x series.

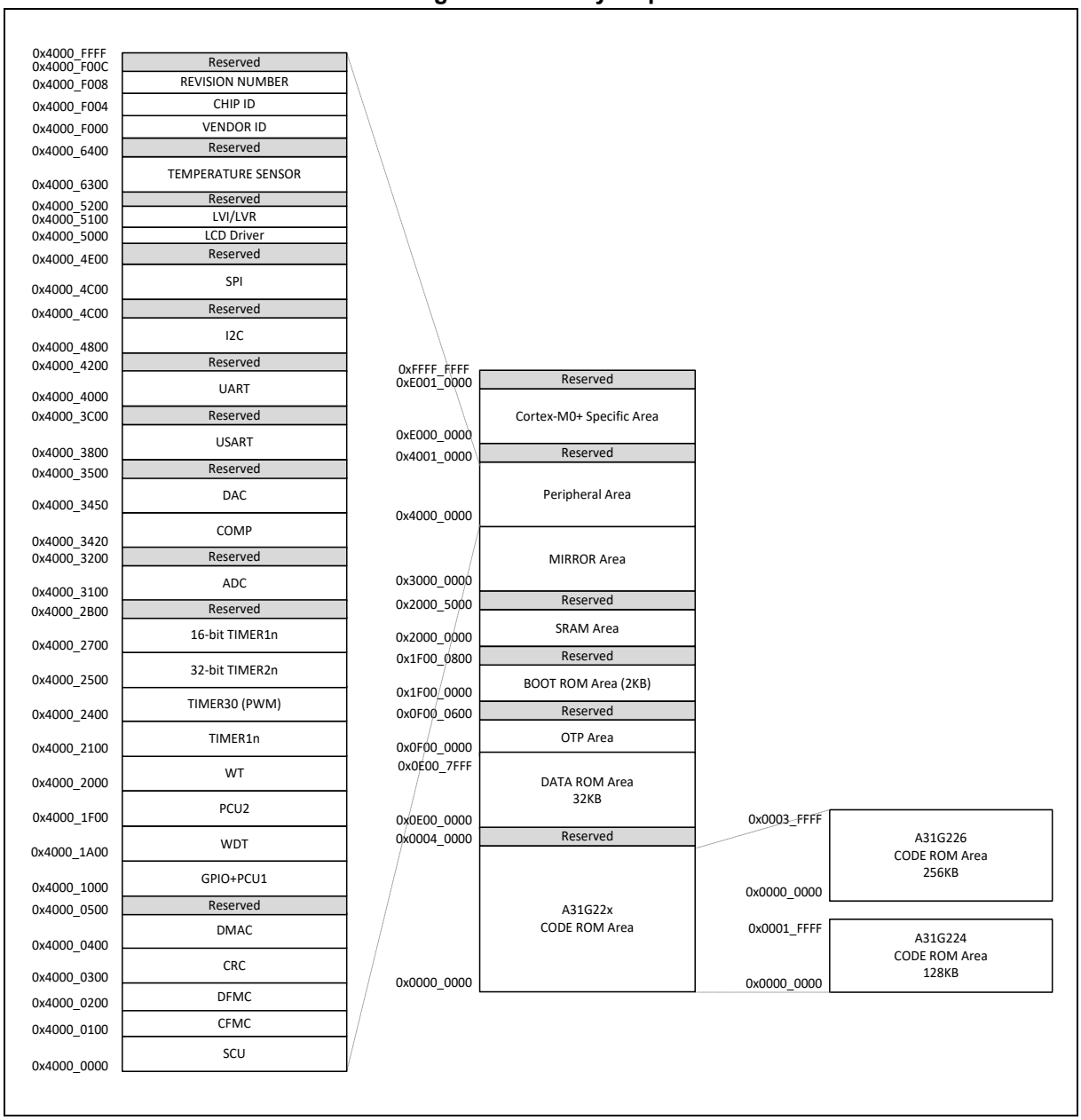
Table 5. A31G22x Memory Boundary Addresses

Boundary address	Memory area	Register description
0x4000_0000	SCU	4.5
0x4000_F000	SCUCC	4.5
0x4000_5100	SCULV	4.5
0x4000_1000/1100/1200/1300/1400/1500	PCU A/B/C/D/E/F	5.3
0x4000_0100	Code Flash Controller	6.1
0x4000_0200	Data Flash Controller	7.1
0x2000_0000	Internal SRAM	8
0x4000_0400/0410/0420/0430/0440/0450/0460/0470	DMAC 0/1/2/3/4/5/6/7	9.2
0x4000_1A00	WDT	10.2
0x4000_2000	WT	11.2
0x4000_2100/2200/2300/2700/2800/2900/2A00	Timer 10/11/12/13/14/15/16	12.2
0x4000_2500/2600	Timer 20/21	13.2
0x4000_2400	Timer 30	14.2
0x4000_3800/3900/3A00/3B00	USART 10/11/12/13	15.2
0x4000_4000/4100	UART 1/2	16.2
0x4000_4C00/4D00	SPI 20/21	17.2
0x4000_4800/4900/4A00	I2C 0/1/2	18.2
0x4000_3100	12-bit ADC	19.2
0x4000_3450	12-bit DAC	20.2
0x4000_3420	Comparator	21.2
0x4000_5000	LCD Driver	22.2
0x4000_0300	CRC	23.2
0x4000_6300	Temp sensor	24.2

3.2.2 Memory map

Figure 8 shows addressable memory space in memory map.

Figure 8. Memory Map



3.2.3 Embedded SRAM

A31G22x series has a block of 0-wait on-chip SRAM. The size of SRAM is 20KB and its base address is 0x2000_0000.

SRAM memory area is usually used for data memory and stack memory. Sometimes the code is dumped into the SRAM memory for fast operation or flash erase/programming operation. This device does not support memory remap strategy. So jump and return are required to perform the code in SRAM memory area.

3.2.4 Flash memory overview

A31G22x series provides internal 256KB code flash memory and its controller. This is enough to control the general system. Self-programming is available and ISP and SWD programming is also supported in boot or debugging mode.

Instruction and data cache buffer are present and overcome the low bandwidth flash memory. CPU can access flash memory with one wait state up to 48MHz bus frequency.

3.2.5 Boot mode

Pins for Boot mode

A31G22x has a Boot mode option to program internal flash memory. System enters in Boot mode by setting BOOT pin to 'L' at reset timing (Normal state is 'H').

In Boot mode, UART boot is available as well:

- USART10_TXD/USART10_RXD port is used in UART boot.

Pins for Boot mode are listed in Table 6.

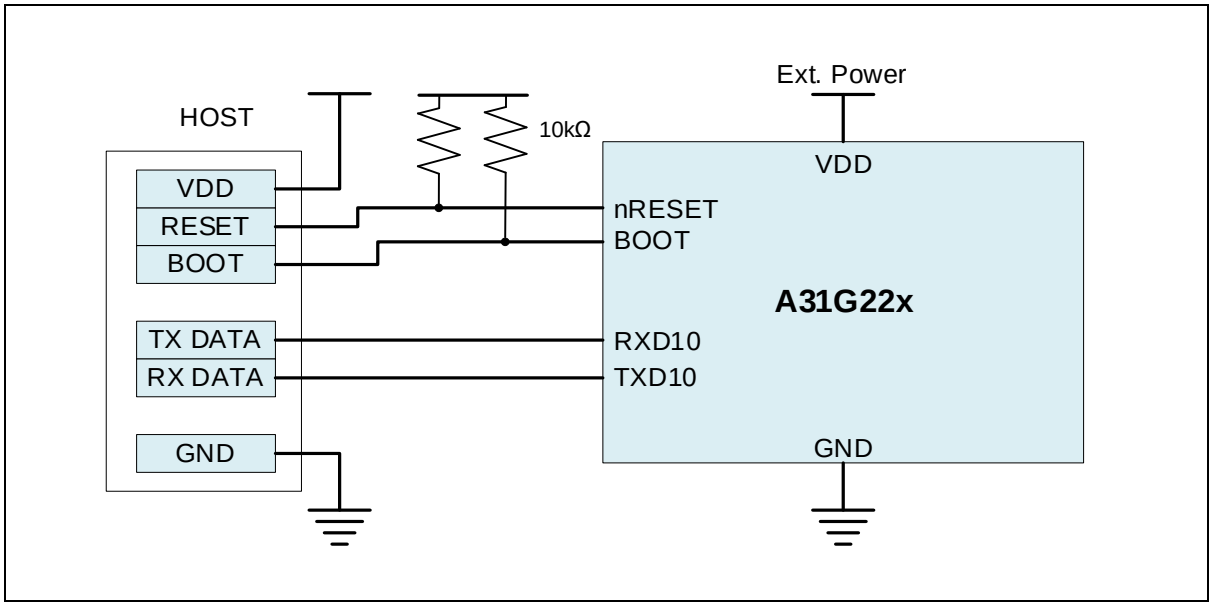
Table 6. Boot Mode Pin List

Block	Pin name	Dir	Description
SYSTEM	nRESET	I	Reset Input signal
	nBOOT	I	'Low' to enter Boot mode
UART mode of USART10	USART10_RXD/PB1	I	USART10 Boot Receive Data
	USART10_TXD/PB0	O	USART10 Boot Transmit Data

Connections for Boot mode

A user can design a target board using any of boot mode ports such as UART mode of USART10. A sample connection diagram of Boot mode is introduced in Figure 9:

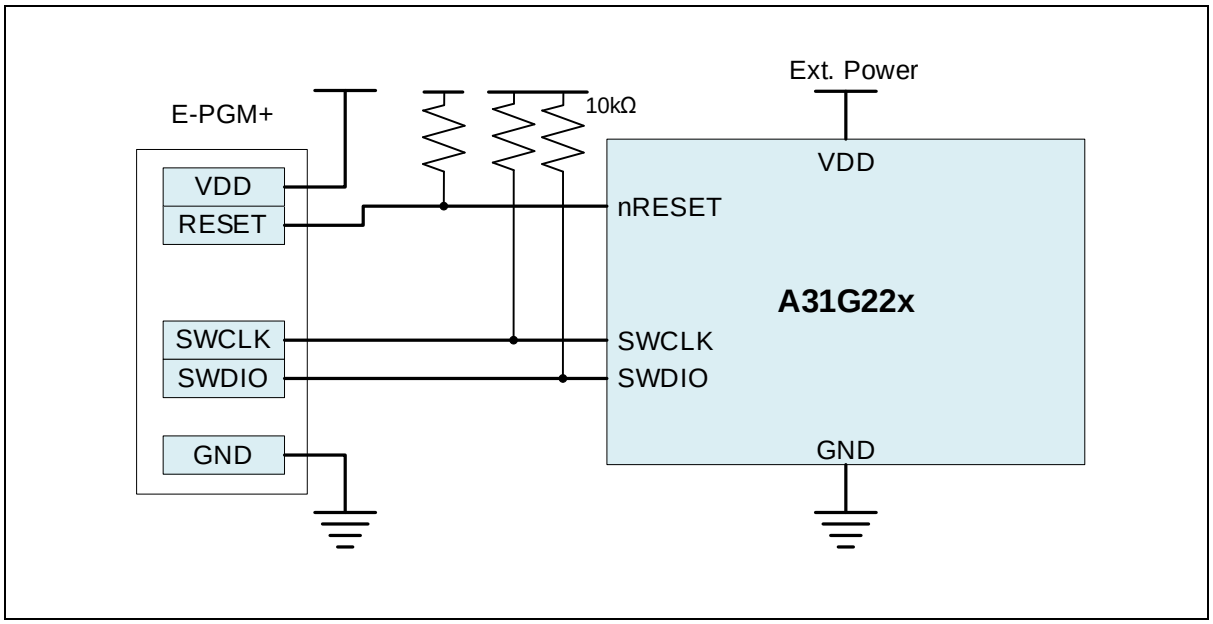
Figure 9. Connection Diagram of UART10 Boot



SWD mode connections

A user can use SWD mode for writing with E-PGM+ or A-Link (CMSIS-DAP) debugger. This mode can be used for writing & debugging.

Figure 10. Connection Diagram of E-PGM+ and SWD Port



4 System Control Unit (SCU)

A31G22x series has a built-in intelligent power control block which manages system analog blocks and operating modes. System Control Unit (SCU) block controls an internal reset and clock signals to maintain optimize system performance and power dissipation.

Four pins in Table 7 are assigned for SCU block.

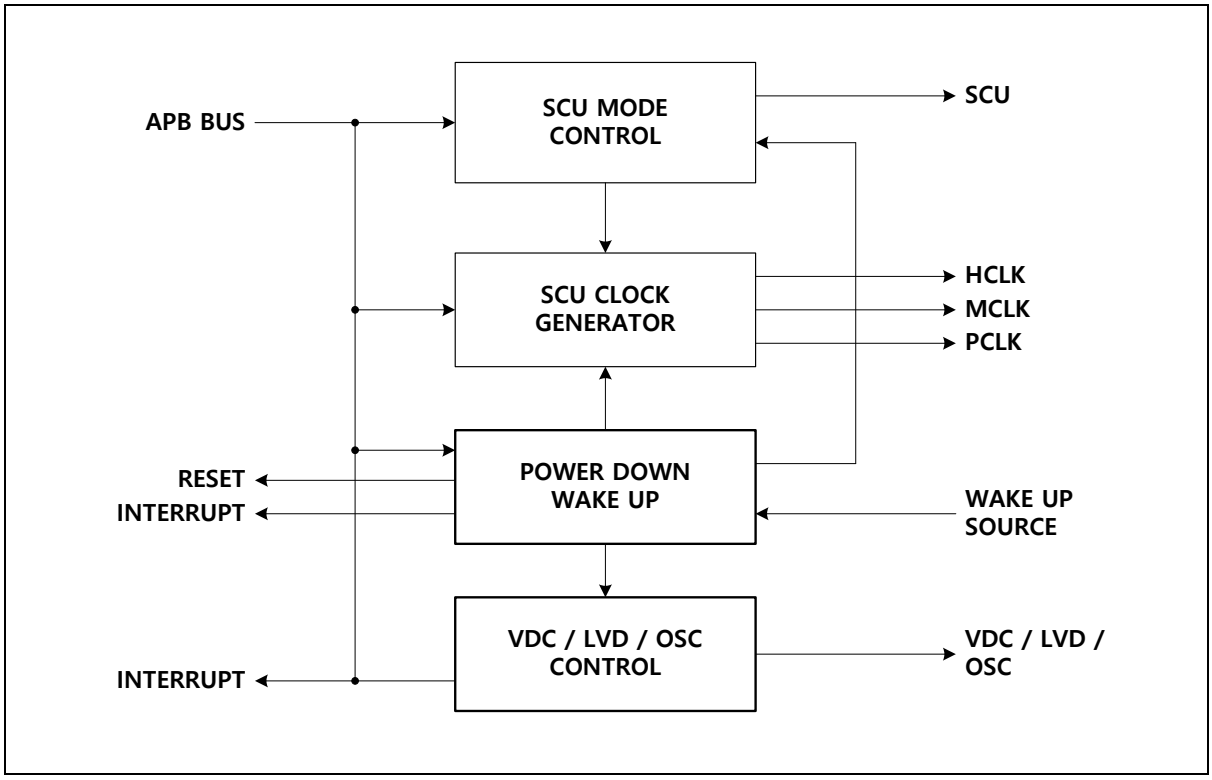
Table 7. SCU Pins

Pin name	Type	Description
nRESET	I	External Reset Input
XIN/XOUT	OSC	External Crystal Oscillator
SXIN/SXOUT	OSC	External sub-Crystal Oscillator
CLKO	O	Clock Output Monitoring Signal

4.1 SCU block diagram

SCU block diagram is introduced in Figure 11.

Figure 11. SCU Block Diagram

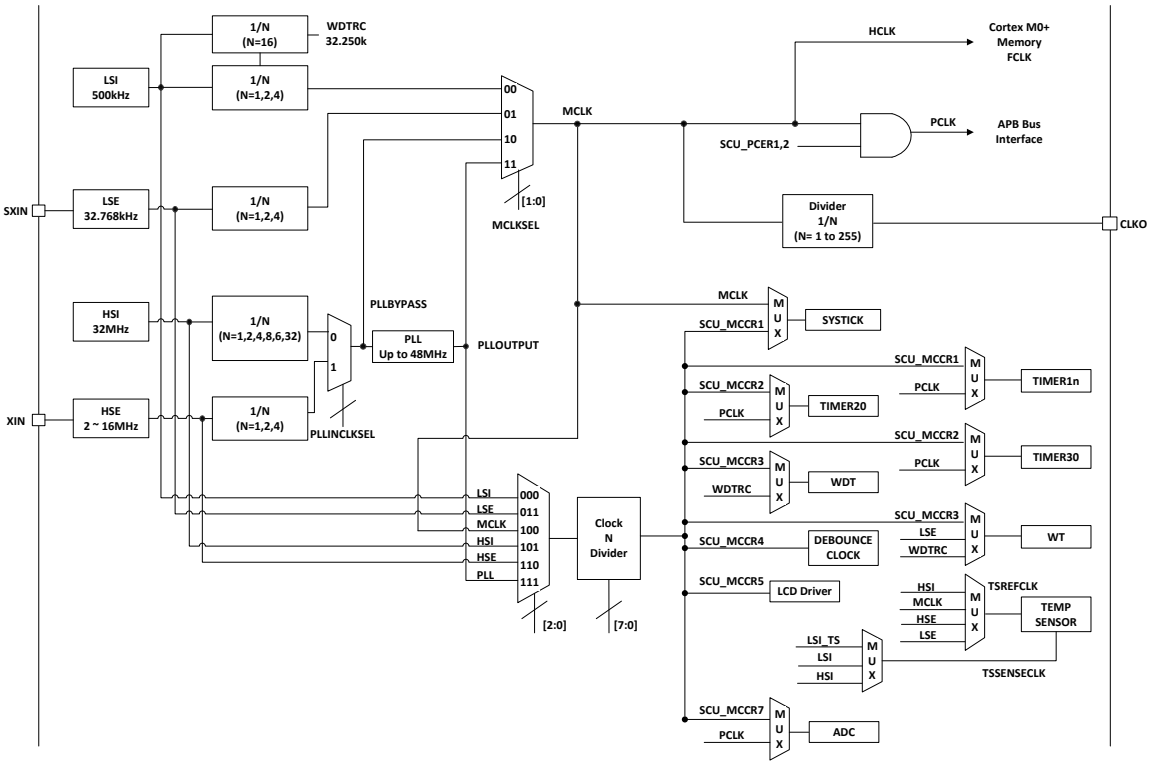


4.2 Clock system

A31G22x series has two main operating clocks. One is HCLK which produces a clock signal both for CPU and AHB bus system. The other is PCLK which produces a clock signal for peripheral systems.

A user can keep the clock system variation under software control. Through Figure 12 and Table 8, users learn about the clock system of A31G22x devices and clock sources.

Figure 12. Clock Tree Configuration



NOTES:

1. In the block diagram above, peripheral blocks not specified in the SCU_MCCRx register use PCLK as a reference clock by default.
2. If you want to use a peripheral clock based on LSI / LSE / HSI / HSE in your application, it is recommended to enable the clock source before enabling the peripheral.

All multiplexers switching clock sources have a glitch-free circuit in each. So a clock can be switched without glitch risks. When a user tries to change a clock mux control, both of clock sources must be alive. If one of them is not alive, clock change operation is stopped and system will be halted and not be recovered.

Table 8. Clock Sources

Clock name	Frequency	Description
HSE	1-16 MHz	High Speed External Oscillator
LSE	32.768 kHz	Low Speed External Oscillator
HSI	32 MHz	High Speed Internal OSC
LSI	500 kHz	Low Speed Internal OSC
LSITS	—	Internal OSC for temp sensor

4.2.1 HCLK clock domain

HCLK clock feeds the clock to the CPU and the AHB bus. Cortex-M0+ CPU requires 2 clocks related with FCLK and HCLK. FCLK is free running clock and it is always running except in power down mode. HCLK can be stopped in SLEEP mode and power down mode.

BUS system and memory systems are operated by MCLK clock. Maximum bus operating clock speed is 48MHz.

4.2.2 PCLK clock domain

PCLK is the master clock of all peripherals. Each peripheral clock is enabled by SCU_PCER1, and SCU_PCER2 registers can be used by each peripheral. Before enabling the PCLK input clock of each block, it can't be accessible even reading its registers. It can be stopped in power down mode.

4.2.3 Clock configuration procedure

After powering up, the default system clock is fed by LSI (500 kHz) clock. By default LSI is enabled at power up sequence. The other clock sources will be enabled by user controls with the LSI system clock.

HSI (32 MHz) clock can be enabled by SCU_CSCR register.

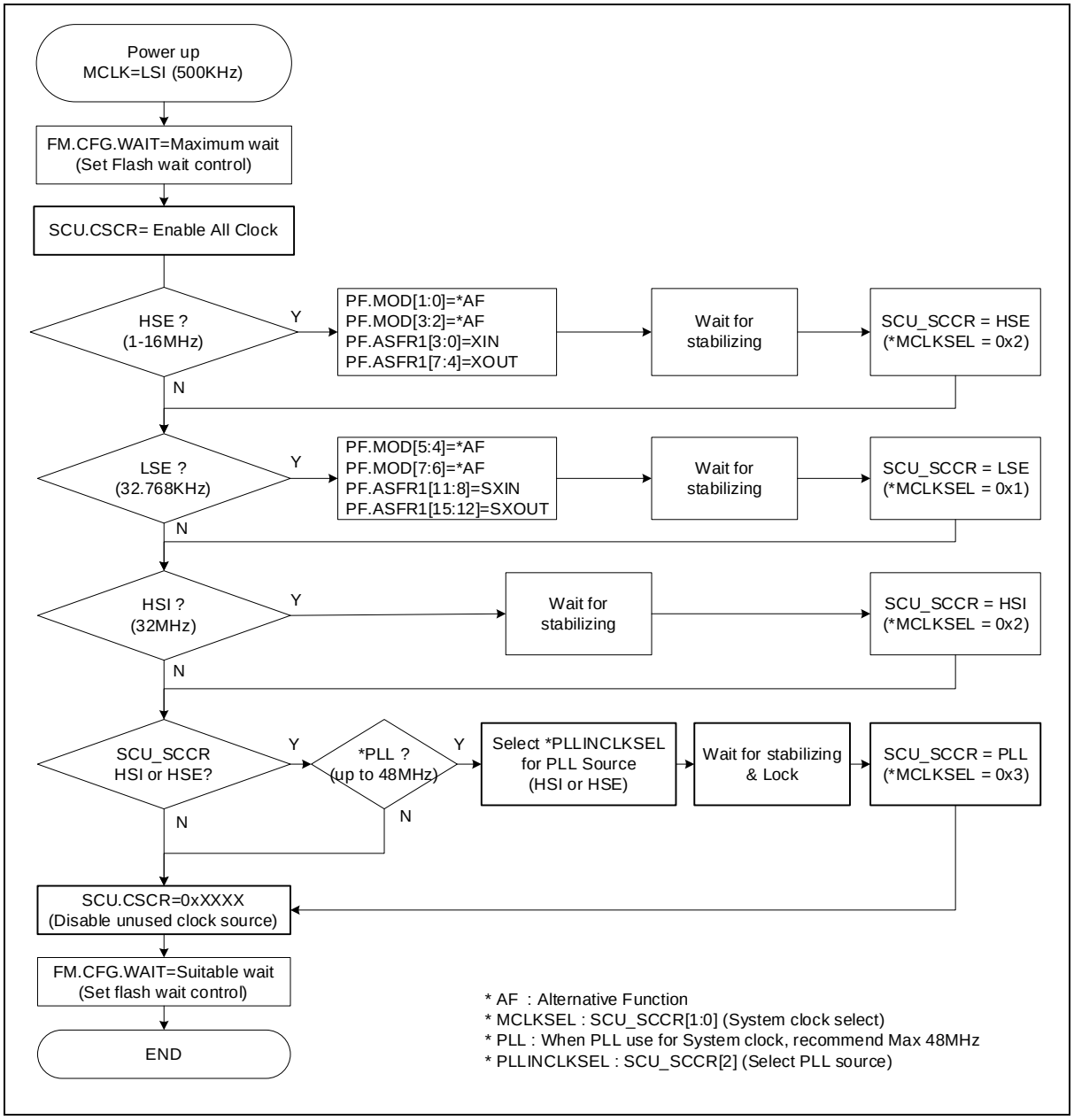
HSE (1 to 16 MHz) clock can be enabled by SCU_CSCR register. Prior to enable the HSE block, the pin mux configuration should be set for XIN, XOUT function. PF1 and PF0 pins are shared with HSE's XIN and XOUT function – PF_MOD and PF_AFSR1 registers should be configured properly. After enabling the HSE block, you must wait for more than Typ. 200ms time to ensure stable operation of crystal oscillation.

LSE (32.768 kHz) clock can be enabled by SCU_CSCR register. Prior to enable the LSE block, the pin mux configuration should be set for SXIN, SXOUT function. PF2 and PF3 pins are shared with LSE's SXIN and SXOUT function – PC_MOD and PC_AFSR1 registers should be configured properly.

After enabling the LSE block, you must wait for more than Typ. 2s time to ensure stable operation of crystal oscillation. You can change an MCLK by using the SCU_SCCR register.

You can find an example flow chart configuring the system clock in Figure 13.

Figure 13. Clock Change Procedure



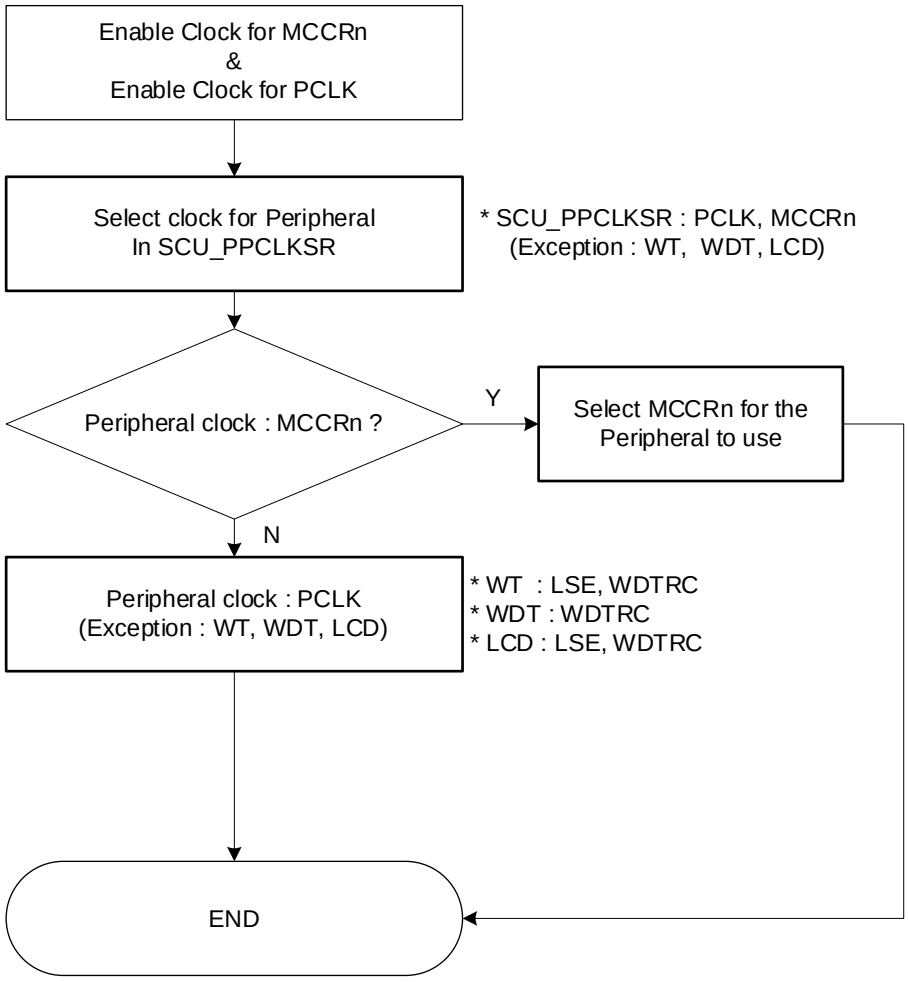
When you speed up the system clock to the maximum operating frequency, you should check flash wait control configuration. Flash read access time is one of limitation factor for performance. The wait control recommendation is suggested in Table 9.

Table 9. Flash Wait Control Recommendation

FM.CFG.WAIT	FLASH Access Wait	Available Max System clock frequency
00	0 clock wait	~20MHz
01	1 clock wait	~40MHz
10	2 clock wait	~48MHz

Figure 14 shows how to set the peripheral clock. Selecting a peripheral clock is a typical method of MCCRn and PCLK. Exceptionally WT, WDT, LCD use other clocks besides MCCRn and PCLK. (n = 1, 2, 3, 4, 5, 6 and 7).

Figure 14. Peripheral Clock Select (n: 1, 2, 3, 4, 5, 6 and 7)



NOTES:

1. The WDTRC clock outputs 31.250KHz frequency generated by dividing the 500KHz LSI clock by 16 in the internal logic.
2. This WDTRC clock can be the source clock of the WDT, WT, LCD and TIMER20 capture signals.

4.3 Reset

A31G22x series has two system reset options. One is to cold reset that is effective during power up or down sequence. The other is warm reset which is generated by several reset sources. A reset event makes a chip to turn to an initial state. Reset sources of the cold reset and the warm reset are listed in Table 10.

Table 10. Reset Sources of Cold Reset and Warm Reset

	Cold reset	Warm reset
Reset sources	• POR (Power On Reset) • LVR reset	• nRESET Pin • WDT reset • MCLK Fail reset • HSE Fail reset • S/W reset • CPU request reset

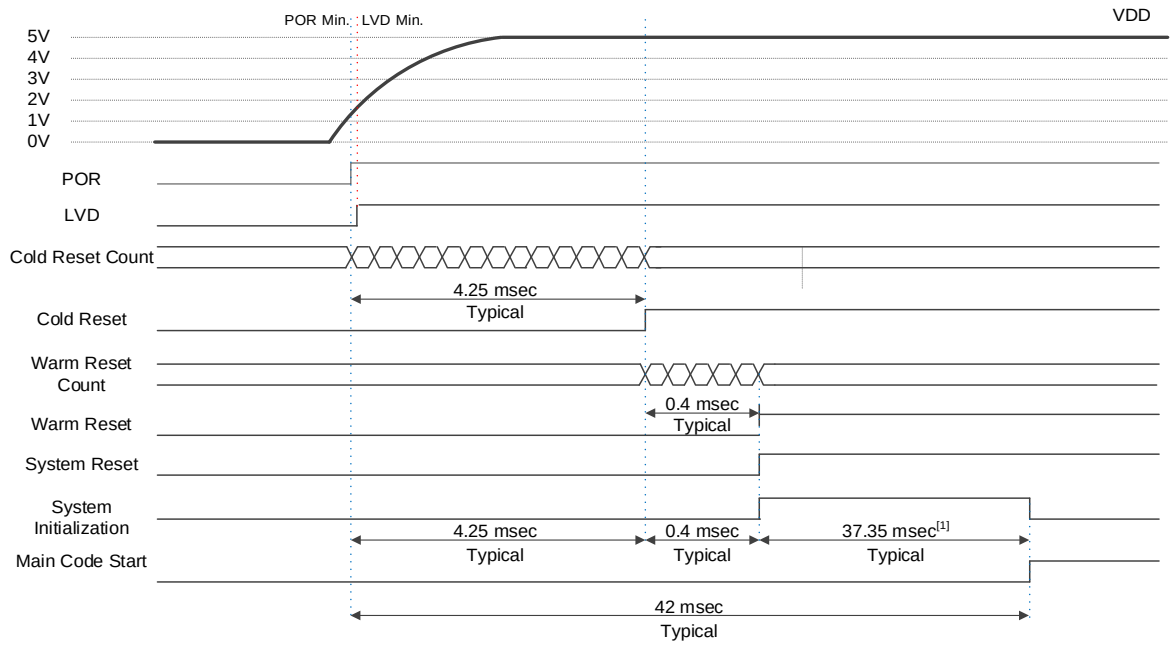
4.3.1 Cold reset

Cold reset is an important feature of a chip when power is up. This characteristic will affect overall system boot.

Internal VDC is enabled when VDD power is turn on. Internal POR trigger level is 1.2V of VDD voltage out level. At this time, boot operation is started. The LSI clock is enabled and counts 4ms time for internal VDC level stabilizing. In this time, VDD voltage level should be over than initial LVR level (1.56V). After 4ms counting, the cold reset is released and counts 0.4ms time for warm reset synchronizing. After releasing both cold and warm reset, BOOTROM and CPU are running.

Figure 15 shows power up sequence and internal reset waveforms.

Figure 15. Power up Procedure



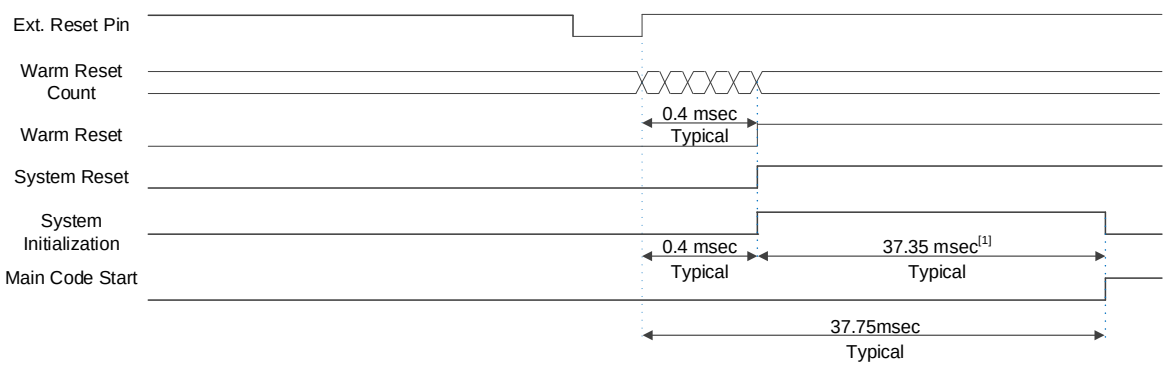
NOTE: [1] System Initialization time after warm reset is based on LSI clock operation.

4.3.2 Warm reset

Warm reset event has several reset sources and some parts of chip returns to an initial state when the warm reset condition is occurred.

The warm reset source is controlled by SCU_RSER register and the status is appeared in SCU_RSSR register. The reset for each peripheral blocks is controlled by SCU_PRER register. The reset can be masked independently.

Figure 16. Warm Reset Diagram



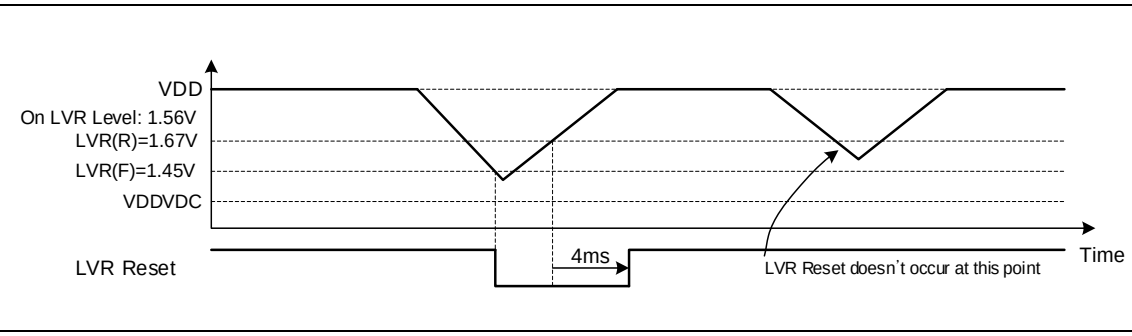
NOTE: [1] System Initialization time after warm reset is based on LSI clock operation.

4.3.3 LVR reset

Voltage level of LVR is set by low voltage reset configuration register (SCULV_LVRCNFG). Reset status of the LVR is shown in SCU_RSSR register.

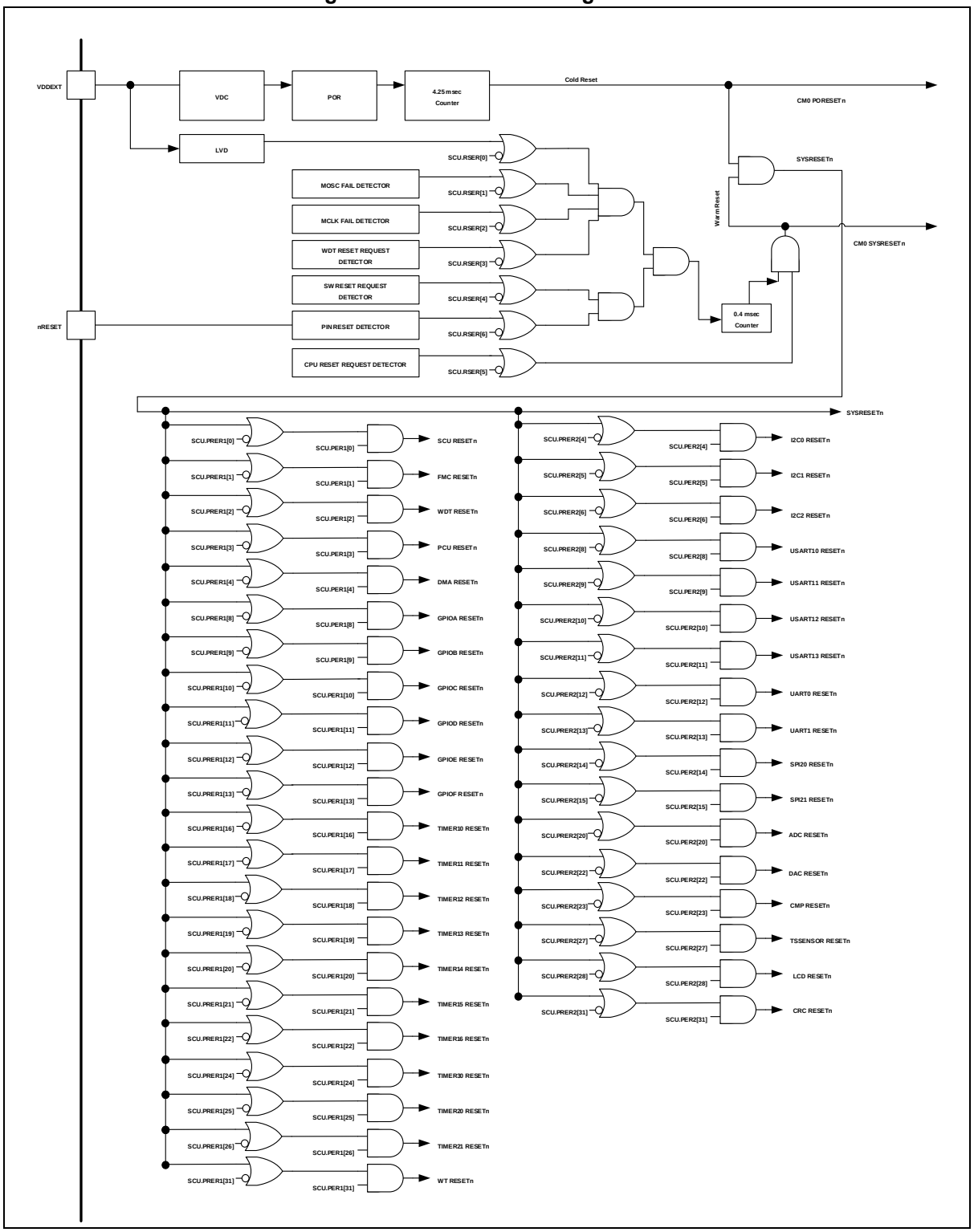
LVR (Low Voltage Reset) must be always enabled to guarantee the operation of the IC. The factory default LVR is set to 1.56V with a margin applied to the minimum operating voltage to guarantee the operation of the IC, but the Reset Detect Level can be changed with the SCU_LVRCNFIG register.

Figure 17. LVR Reset Timing Diagram



4.3.4 Reset tree

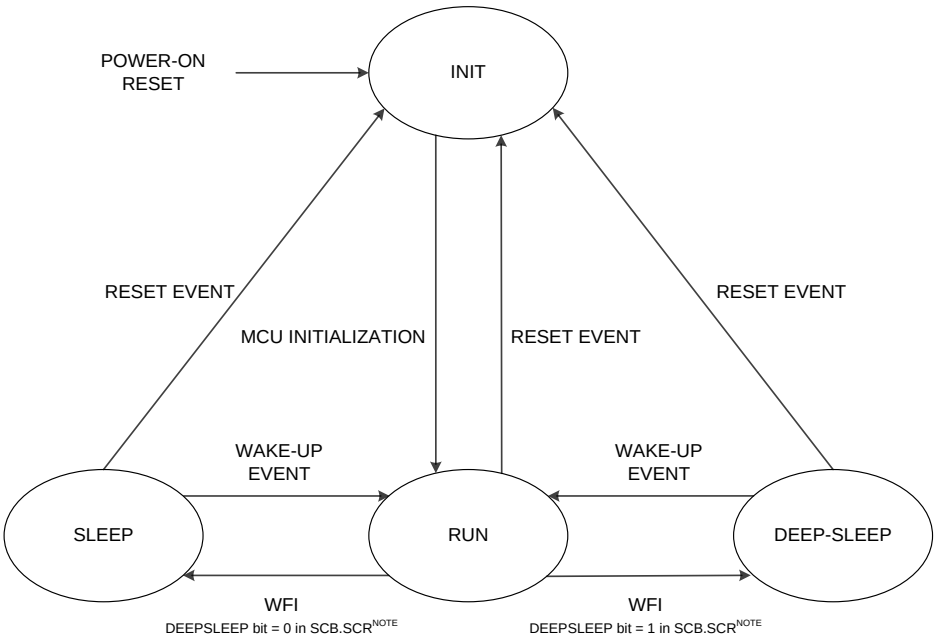
Figure 18. Reset Tree Configuration



4.4 Operation mode

INIT mode is an initial state of the chip when a reset is asserted. In RUN mode, CPU shows the maximum performance with high-speed clock system. SLEEP mode and DEEP-SLEEP mode can be used as a low power consumption mode. In the low power consumption mode, power is effectively managed to reduce power consumption by halting processor core and unused peripherals. Figure 19 describes transition between the operation modes.

Figure 19. Transition between Operation Modes



NOTE: SCB.SCR means System Control Register in System Control Block.

Table 11. Operation Mode

MODE	Condition	After the wake up event	After the reset event
RUN	POWER ON	N/A	INIT
SLEEP	WFI (Wait for Interrupt): SCB.SCR[2]*=0	RUN	INIT
DEEP-SLEEP	WFI (Wait for Interrupt): SCB.SCR[2]*=1	RUN	INIT

By default, SLEEP and DEEP-SLEEP modes automatically power off the LSE and LSI oscillation. As such, the peripherals fed with a clock signal from the LSE or LSI will stop operating in SLEEP or DEEP-SLEEP mode. To keep those peripherals running in this mode, the MCU can be configured in either of the following ways:

A. Set the LSI or LSE as the clock source to feed the peripherals that need to continue running in SLEEP or DEEP-SLEEP mode.

B. To continue using the LSE or LSI in SLEEP or DEEP-SLEEP mode, set the LSEAON or LSIAON bit in the SCU_SMR register to "1." LSEAON and LSIAON bits will prevent the LSE or LSI from automatically being powered off. Next, configure the SCU_SCCR register manually to disable the clock sources that are NOT in use in SLEEP or DEEP-SLEEP mode.

4.4.1 RUN mode

In RUN mode, CPU and the peripheral hardware operate with a high-speed clock. After a reset followed by INIT state, the system enters in the RUN mode.

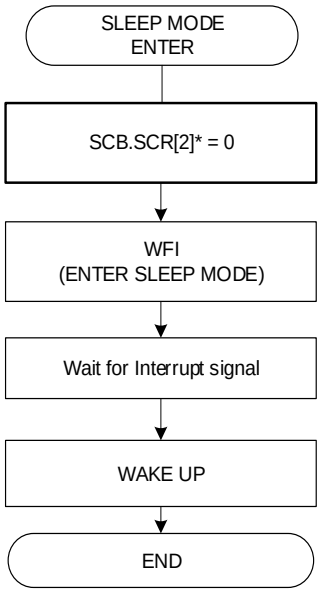
4.4.2 SLEEP mode

When the core goes under SLEEP mode using WFI instruction, the chip enters in IDLE state. In SLEEP mode, only CPU is stopped. Each peripheral function can be enabled by the function enable and clock enable bit in the PER and PCER register. When the WFI instruction is used to enter SLEEP mode, the device can exit from SLEEP mode by any peripheral interrupt that is acknowledged by the nested vectored interrupt controller (NVIC). When a wake-up event occurs based on the settings, the MCU will return to run mode after POR reset.

Table 12. SLEEP Mode Configuration

Mode	Condition	Wakeup source
SLEEP (Idle state)	WFI SCB.SCR[2]* = 0, VDCCON[18] = 0	Peripheral Interrupt

Figure 20. SLEEP Mode Operation Sequence



NOTE: SCB.SCR is System Control Register in System Control Block.

4.4.3 DEEP-SLEEP mode

When the core goes under DEEP-SLEEP using WFI instruction, the chip enters in power down state. In DEEP-SLEEP mode, all peripherals are stopped. Once power is supplied, internal SRAM and registers maintain their values.

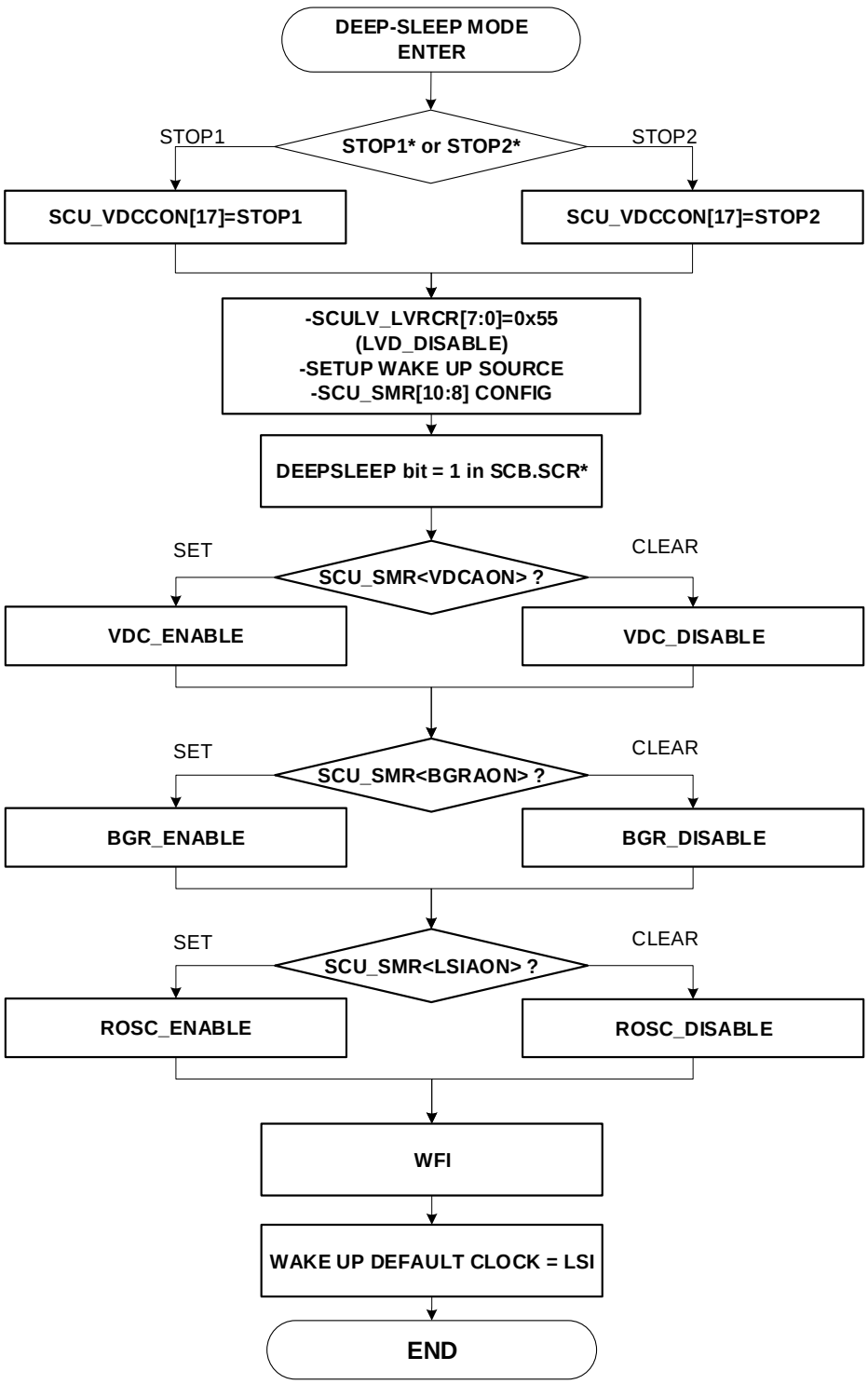
To wake up in DEEP-SLEEP mode, an interrupt request must be generated and can be selected in the SCU_WUER register. Stabilization time is 4ms after wakeup event occurs.

When entering in DEEP-SLEEP mode, LSI is selected as the MCLK, and HSE is selected as the PLL input clock. These are maintained after wakeup.

Table 13. DEEP-SLEEP Mode Configuration

Mode	Condition	Wakeup source
DEEP-SLEEP (Power down state)	WFI SCB.SCR[2]* = 1, VDCCON[18] = 0	Source included in WUER

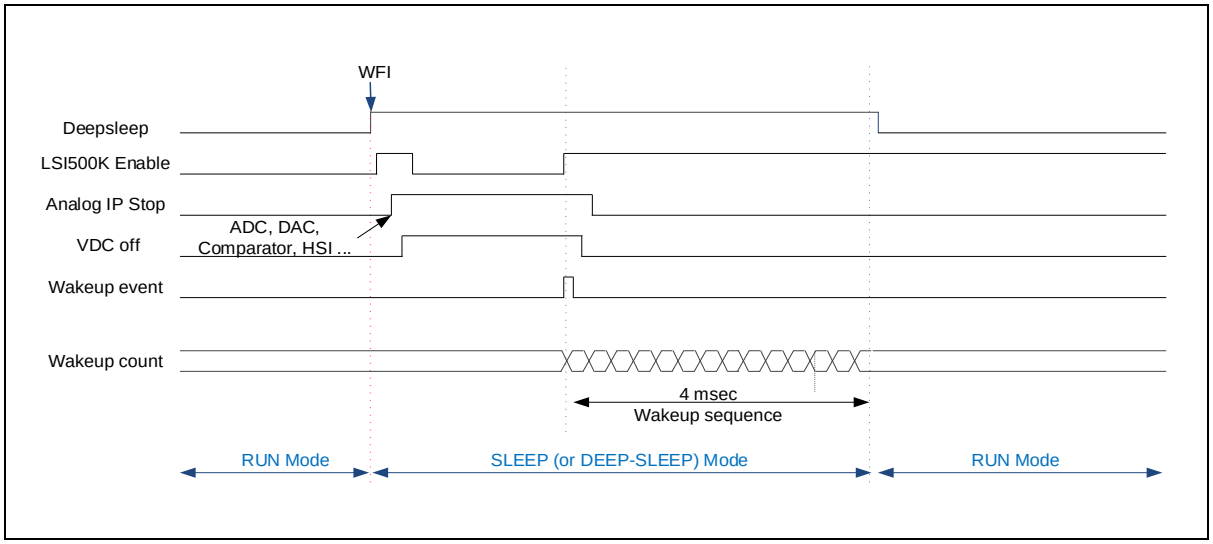
Figure 21. DEEP-SLEEP Mode Sequence



NOTES:

1. When STOP1, VDC load current is limited 500uA.
2. When STOP2, VDC load current is limited 10uA.
3. SCB.SCR is System Control Register in System Control Block.

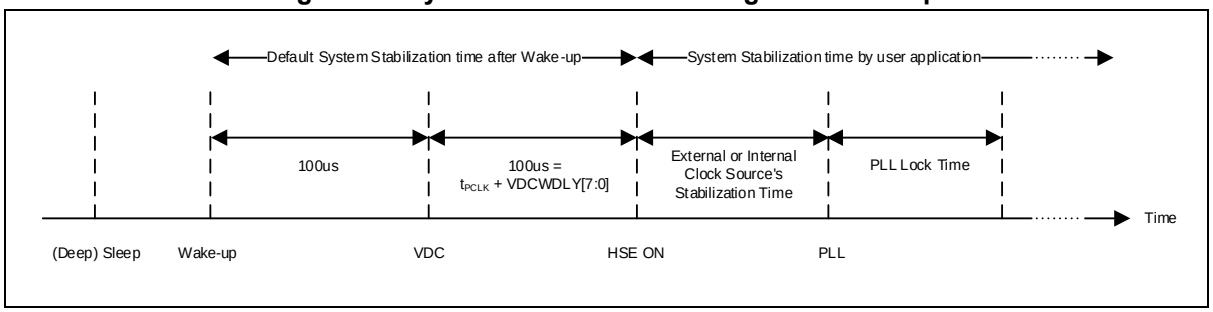
Figure 22. SLEEP & DEEP-SLEEP Mode Timing Diagram



4.4.4 Wake-up timing

Figure 23 shows stabilization timing of RUN mode after wake up from SLEEP mode or DEEP-SLEEP mode by configuring WUER register for wake-up sources.

Figure 23. System Stabilization Timing after wake-up



4.5 Registers

Base address of SCUCC (chip configuration) and register map are introduced in the followings:

Table 14. Base Address of SCUCC (Chip Configuration)

Name	Base address
SCUCC (Chip Configuration)	0x4000_F000

Table 15. SCU Register Map (Chip Configuration)

Name	Offset	Type	Description	Reset value	Ref.
SCUCC_VENDORID	0x0000	RO[NOTE]	Vendor Identification Register	0x4142_4F56	4.5.1
SCUCC_CHIPID	0x0004	RO[NOTE]	Chip Identification Register.	0x4D31_A00x	4.5.2
SCUCC_REVNR	0x0008	RO[NOTE]	Revision Number Register	0x0000_00xx	4.5.3

NOTE: 'RO' means 'Read Only'.

Base address of SCU and register map are introduced in the followings:

Table 16. Base Address of SCU

Name	Base address
SCU	0x4000_0000

Table 17. SCU Register Map

Name	Offset	Type	Description	Reset value	Ref.
SCU_SMR	0x0004	RW	System Mode Register	0x0000_0000	4.5.4
SCU_SCR	0x0008	RW	System Control Register	0x0000_0000	4.5.5
SCU_WUER	0x0010	RW	Wake up source enable register	0x0000_0000	4.5.6
SCU_WUSR	0x0014	RO	Wake up source status register	0x0000_0000	4.5.7
SCU_RSER	0x0018	RW	Reset source enable register	0x0000_0069	4.5.8
SCU_RSSR	0x001C	RW	Reset source status register	0x0000_00E8	4.5.9
SCU_PRER1	0x0020	RW	Peripheral reset enable register 1	0x877F_3F1F	4.5.10
SCU_PRER2	0x0024	RW	Peripheral reset enable register 2	0x98D0_FF70	4.5.11
SCU_PER1	0x0028	RW	Peripheral enable register 1	0x0000_000F	4.5.12
SCU_PER2	0x002C	RW	Peripheral enable register 2	0x2000_0100	4.5.13
SCU_PCER1	0x0030	RW	Peripheral clock enable register 1	0x0000_000F	4.5.14
SCU_PCER2	0x0034	RW	Peripheral clock enable register 2	0x2000_0100	4.5.15
SCU_PPCLKSR	0x0038	RW	Peripheral clock selection register	0x0000_0000	4.5.16
SCU_CSCR	0x0040	RW	Clock Source Control register	0x0000_0800	4.5.17
SCU_SCCR	0x0044	RW	System Clock Control register	0x0000_0000	4.5.18
SCU_CMR	0x0048	RW	Clock Monitoring register	0x0000_0090	4.5.19

Table 17. SCU Register Map (continued)

Name	Offset	Type	Description	Reset value	Ref.
SCU_NMIR	0x004C	RW	NMI control register	0x0000_0000	4.5.20
SCU_COR	0x0050	RW	Clock Output Control register	0x0000_000F	4.5.21
SCU_PLLCON	0x0060	RW	PLL Control register	0x0600_0000	4.5.22
SCU_VDCCON	0x0064	RW	VDC Control register	0x0000_007F	4.5.23
SCU_LSICON	0x006C	RW	Internal Ring OSC Control Register	0x0000_0000	4.5.24
SCU_EOSCR	0x0080	RW	External Oscillator control register	0x0000_1014	4.5.25
SCU_EMODR	0x0084	RW	External mode pin read register	0x0000_0001	4.5.26
SCU_RSTDBCR	0x0088	RW	Pin Reset Debounce Control Register	0x0000_0000	4.5.27
SCU_MCCR1	0x0090	RW	Misc. Clock Control register 1	0x0000_0000	4.5.28
SCU_MCCR2	0x0094	RW	Misc. Clock Control register 2	0x0000_0000	4.5.29
SCU_MCCR3	0x0098	RW	Misc. Clock Control register 3	0x0000_0000	4.5.30
SCU_MCCR4	0x009C	RW	Misc. Clock Control register 4	0x0000_0000	4.5.31
SCU_MCCR5	0x00A0	RW	Misc. Clock Control register 5	0x0000_0000	4.5.32
SCU_MCCR6	0x00A4	RW	Misc. Clock Control register 6	0x0000_0000	4.5.33
SCU_MCCR7	0x00A8	RW	Misc. Clock Control register 7	0x0000_0000	4.5.34

NOTES:
1. 'RO' means 'Read Only'.
2. 'RW' means 'Read and Write'.

Base address of LVI/LVR unit and register map are introduced in the followings:

Table 18. Base Address of LVI/LVR

Name	Base address
SCULV(LVI/LVR)	0x4000_5100

Table 19. LVI/LVR Register Map

Name	Offset	Type	Description	Reset value	Ref.
SCULV_LVICR	0x0000	RW	Low Voltage Indicator Control Register	0x0000_0000	4.5.35
SCULV_LVRCR	0x0004	RW	Low Voltage Reset Control Register	0x0000_0000	4.5.36
SCULV_LVRCNFIG	0x0008	RW	Configuration for Low Voltage Reset	0x0000_000F	4.5.37

NOTES:
1. "RO' means 'Read Only' and 'RW' means 'Read and Write'.
2. 'RC' means 'Read and Write 1 Clear'.

4.5.1 SCUCC_VENDORID: vendor ID register

SCUCC_VENDORID register shows the vendor identification information. This is a 32-bit read-only register.

SCUCC_VENDORID=0x4000_F000

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
VENDID			
0x4142_4F56			
RO			

Bits	Name	Function
31 0	VENDID	Vendor Identification bits. 0x4142_4F56

4.5.2 SCUCC_CHIPID: chip ID register

SCUCC_CHIPID register shows chip identification information. This is a 32-bit read-only register.

SCUCC_CHIPID=0x4000_F004

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
CHIPID			
0x4D31A00A or 0x4D31A00B			
RO			

Bits	Name	Function	
31 0	CHIPID	Chip Identification bits.	
		0x4D31A00A	256k bytes flash memory for program
		0x4D31A00B	128k bytes flash memory for program

4.5.3 SCUCC_REVNR: revision number register

Revision Number register is a 32-bit read-only register. This Register is available at 32/16/8-bit access.

SCUCC_REVNR=0x4000_F008

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			REVNO
0x000000			0xXX
-			RO

Bits	Name	Function
7 0	REVNO	Chip Revision Number. These bits are fixed by manufacturer.

4.5.4 SCU_SMR: system mode register

Current operating mode is shown in this SCU mode register. The previous operating mode will be saved in this register after reset event. There is a VDC On/Off control bit in power down mode. The previous operating mode stores the state of the operating mode before reset event in this register, and the previous mode can be confirmed by checking PREVMODE value after reset event.

SCU_SMR=0x4000_0004

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12	11	10	9	8	7 6	5 4	3 2 1 0
Reserved			LSEAON	LSIAON	BGRAON	VDCAON	Reserved	PREVMODE	Reserved
-			0	0	0	0	-	00	-
-			RW	RW	RW	RW	-	RO	-

Bits	Name	Function	
11	LSEAON	LSE Always on select bit in power down mode[NOTE]	
		0	LSE is automatically off entering power down mode
		1	LSE isn't automatically off entering power down mode
10	LSIAON	LSI Always on select bit in power down mode[NOTE]	
		0	LSI is automatically off entering power down mode
		1	LSI isn't automatically off entering power down mode
9	BGRAON	BGR Always on select bit in power down mode	
		0	BGR is automatically off entering power down mode
		1	BGR isn't automatically off entering power down mode
8	VDCAON	VDC Always on select bit in power down mode	
		0	VDC is automatically off entering power down mode
		1	VDC isn't automatically off entering power down mode
5 4	PREVMODE	Previous operating mode before current reset event	
		00	Previous operating mode was RUN mode RUN status sets after warm reset.
		01	Previous operating mode was SLEEP mode
		10	Previous operating mode was DEEP-SLEEP mode
		11	Previous operating mode was INIT mode INIT status sets after cold (Power-On Reset) reset.

4.5.5 SCU_SCR: system control register

It is possible to reset MCU as SWRST bit set. To reset the MCU with the software reset bit, the SWRST must be set with the 'Write Identification Key'.

SCU_SCR=0x4000_0008

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1	0
WTIDKY		Reserved		SWRST
0x0000		-		0
WO		-		RW

Bits	Name	Function	
31 16	WTIDKY	Write Identification Key	
		On writes, write 0x9EB3 to these bits, otherwise the write is ignored.	
0	SWRST	Internal soft reset activation bit (check RSER[4] for reset)	
		0	Normal operation
		1	Internal soft reset generated and auto cleared

4.5.6 SCU_WUER: wakeup source enable register

Enable wakeup source when the chip is in the DEEP-SLEEP mode. Wakeup sources which will be used the source of chip wakeup should be enabled in each bit field. If the source is used as a wakeup source, the corresponding bit should be written with '1'. If the source is not used as a wakeup source, the bit should be written with '0'.

SCU_WUER=-0x4000_0010

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14	13	12	11	10	9	8	7 6 5 4 3	2	1	0
Reserved		SYSTICKWUE	Reserved	GPIOFWUE	GPIOEWUE	GPIODWUE	GPIOCWUE	GPIOBWUE	GPIOAWUE	Reserved	WTWUE	WDTWUE	LVIWUE
-		0	-	0	0	0	0	0	0	-	0	0	0
-		RW	-	RW	RW	RW	RW	RW	RW	-	RW	RW	RW

Bits	Name	Function
16	SYSTICKWUE	Enable wakeup source of SYSTICK timer event only operating in SLEEP mode.*
		0 Not used for wakeup source
		1 Enable the wakeup event generation
13	GPIOFWUE	Enable wakeup source of GPIOF port pin change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
12	GPIOEWUE	Enable wakeup source of GPIOE change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
11	GPIODWUE	Enable wakeup source of GPIOD change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
10	GPIOCWUE	Enable wakeup source of GPIOC port pin change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
9	GPIOBWUE	Enable wakeup source of GPIOB port pin change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
8	GPIOAWUE	Enable wakeup source of GPIOA port pin change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
2	WTWUE	Enable wakeup source of watch timer event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
1	WDTWUE	Enable wakeup source of watchdog timer event
		0 Not used for wakeup source

		1	Enable the wakeup event generation
0	LVIWUE	Enable wakeup source of LVI event	
		0	Not used for wakeup source
		1	Enable the wakeup event generation
* **NOTE**: SYSTICK timer (SYSTICKWUE) wake-up works only in SLEEP mode.			

4.5.7 SCU_WUSR: wakeup source status register

When the system is waked up by any wakeup source, the wakeup source is identified by reading SCU_WUSR register. When the bit is set to 1, the corresponding wakeup source issues the wake-up signal to SCU. The bit will be cleared when the event source is cleared by the software.

SCU_WUSR=0x4000_0014

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8							7 6 5 4 3	2	1	0
Reserved		SYSTICKWU	Reserved	GPIOFWU	GPIOEWU	GPIODWU	GPIOCWU	GPIOBWU	GPIOAWU	Reserved	WTWU	WDTWU	LVIWU
-		0	-	0	0	0	0	0	0	-	0	0	0
-		RO	-	RO	RO	RO	RO	RO	RO	-	RO	RO	RO

Bits	Name	Function	
16	SYSTICKWU	Status of wakeup source of SYSTICK timer event.*	
		0	No wakeup event
		1	Wakeup event was generated
13	GPIOFWU	Status of wakeup source of GPIOF port pin change event	
		0	No wakeup event
		1	Wakeup event was generated
12	GPIOEWU	Status of wakeup source of GPIOE port pin change event	
		0	No wakeup event
		1	Wakeup event was generated
11	GPIODWU	Status of wakeup source of GPIOD port pin change event	
		0	No wakeup event
		1	Wakeup event was generated
10	GPIOCWU	Status of wakeup source of GPIOC port pin change event	
		0	No wakeup event
		1	Wakeup event was generated
9	GPIOBWU	Status of wakeup source of GPIOB port pin change event	
		0	No wakeup event
		1	Wakeup event was generated
8	GPIOAWU	Status of wakeup source of GPIOA port pin change event	
		0	No wakeup event
		1	Wakeup event was generated

2	WTWU	Status of wakeup source of watch timer event	
		0	No wakeup event
		1	Wakeup event was generated
1	WDTWU	Status of wakeup source of watchdog timer event	
		0	No wakeup event
		1	Wakeup event was generated
0	LVIWU	Status of wakeup source of LVR event	
		0	No wakeup event
		1	Wakeup event was generated

* **NOTE**: Systick timer wakeup flag bit occurs within 1-cycle and is automatically clear after waking-up from SLEEP mode. So this flag bit can't check in User Code. Therefore, The wakeup status by SYSTICK timer event from SLEEP mode, You should check the PREVMODE value in SCU_SMR register.

4.5.8 SCU_RSER: reset source enable register

A reset source to CPU can be selected by SCU_RSER register. When writing '1' in the bit field of each reset source, the reset source event will be transferred to reset generator. When writing '0' in the bit field of each reset source, the reset source event will be masked and not generate the reset event.

SCU_RSER=0x4000_0018

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved				PINRST	CPURST	SWRST	WDTRST	MCKFRST	HSEFRST	LVDRST
-				1	1	0	1	0	0	1
-				RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
6	PINRST	External pin reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
5	CPURST	CPU request reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
4	SWRST	Software reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
3	WDTRST	Watchdog Timer reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
2	MCKFRST	MCLK Clock fail reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
1	HSEFRST	HSE Clock fail reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
0	LVDRST	LVD reset (LVR) enable bit **NOTES:** 3. LVR (Low Voltage Detection Reset) must be always enabled to guarantee the operation of the IC. 4. The factory default LVR is set to 1.56V with a margin applied to the minimum operating voltage to guarantee the operation of the IC, but the Reset Detect Level can be changed with the SCU_LVRCNFIG register.
		0 Reset from this event is masked
		1 Reset from this event is enabled

4.5.9 SCU_RSSR: reset source status register

SCU_RSSR register shows reset source information when reset event is occurred. '1' means reset event was exist and '0' means reset event is not exist for the corresponding reset source.

SCU_RSSR=0x4000_001C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			PORST	PINRST	CPURST	SWRST	WDTRST	MCKFRST	HSEFRST	LVDRST
-			1	0	0	0	0	0	0	0
-			RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
7	PORST	Power on reset status bit
	0	Read : Reset from this event was not exist Write : no effect
	1	Read : Reset from this event was occurred Write : Clear the status
6	PINRST	External pin reset status bit
	0	Read : Reset from this event was not exist Write : no effect
	1	Read : Reset from this event was occurred Write : Clear the status
5	CPURST	CPU request reset status bit
	0	Read : Reset from this event was not exist Write : no effect
	1	Read : Reset from this event was occurred Write : Clear the status
4	SWRST	Software reset status bit
	0	Read : Reset from this event was not exist Write : no effect
	1	Read : Reset from this event was occurred Write : Clear the status
3	WDTRST	Watchdog Timer reset status bit
	0	Read : Reset from this event was not exist Write : no effect
	1	Read : Reset from this event was occurred Write : Clear the status
2	MCKFRST	MCLK Clock fail reset status bit
	0	Read : Reset from this event was not exist Write : no effect
	1	Read : Reset from this event was occurred Write : Clear the status
1	HSEFRST	HSE Clock fail reset status bit
	0	Read : Reset from this event was not exist

			Write : no effect
		1	Read : Reset from this event was occurred Write : Clear the status
0	LVDRST	LVD reset status bit	
		0	Read : Reset from this event was not exist Write : no effect
		1	Read : Reset from this event was occurred Write : Clear the status

NOTE: When reset source is founded, write '1' into the corresponding bit to clear the reset status.

4.5.10 SCU_PRER1: peripheral reset enable register 1

The reset of each peripheral by event reset, can be masked by user setting. SCU_PRER1/SCU_PRER2 register will control the enable of the event WRAM reset. If the corresponding bit is '1', the peripheral corresponded with this bit, accepts the reset event. Otherwise, the peripheral is protected from reset event and maintain current operation. To use warm reset event control of a peripheral, SCU_PRER1[0] SCU bit must be controlled with a corresponding peripheral bit.

SCU_PRER1=0x4000_0020

31	30 29 28 27 26	25	24	23	22	21	20	19	18	17	16	15 14	13	12	11	10	9	8	7 6 5	4	3	2	1	0
WT	Reserved	TIMER20	TIMER30	Reserved	TIMER16	TIMER15	TIMER14	TIMER13	TIMER12	TIMER11	TIMER10	Reserved	GPIOF	GPIOE	GPIOD	GPIOC	GPIOB	GPIOA	Reserved	DMA	PCU	WDT	FMC	SCU*
1	-	1	1	-	1	1	1	1	1	1	1	-	1	1	1	1	1	1	-	1	1	1	1	1
RW	-	RW	RW	-	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	RW

Bits	Name	Function
31	WT	WT reset mask
26	TIMER21	TIMER21 reset mask
25	TIMER20	TIMER20 reset mask
24	TIMER30	TIMER30 reset mask
22	TIMER16	TIMER16 reset mask
21	TIMER15	TIMER15 reset mask
20	TIMER14	TIMER14 reset mask
19	TIMER13	TIMER13 reset mask
18	TIMER12	TIMER12 reset mask
17	TIMER11	TIMER11 reset mask
16	TIMER10	TIMER10 reset mask
13	GPIOF	GPIOF reset mask
12	GPIOE	GPIOE reset mask
11	GPIOD	GPIOD reset mask
10	GPIOC	GPIOC reset mask
9	GPIOB	GPIOB reset mask

8	GPIOA	GPIOA reset mask
4	DMA	DMA reset mask
3	PCU	Port controller reset mask
2	WDT	Watchdog Timer reset mask
1	FMC	Flash memory controller reset mask
0	SCU*	System Management Unit reset mask

4.5.11 SCU_PRER2: peripheral reset enable register 2

The reset of each peripheral by event reset, can be masked by user setting. SCU_PRER1/SCU_PRER2 register will control the enable of the event WARM reset. If the corresponding bit is '1', the peripheral corresponded with this bit, accepts the reset event. Otherwise, the peripheral is protected from reset event and maintain current operation.

SCU_PRER2=0x4000_0024

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC	Reserved		LCD	TS	Reserved			CMP	DAC	Reserved	ADC	Reserved				SPI21	SPI20	UART1	UART0	USART13	USART12	USART11	USART10	Reserved	I2C2	I2C1	I2C0	Reserved			
1	-		1	1	-			1	1	-	1	-				1	1	1	1	1	1	1	1	-	1	1	1	-			
RW	-		RW	RW	-			RW	RW	-	RW	-				RW	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	-			

Bits	Name	Function
31	CRC	CRC reset enable
28	LCD	LCD reset enable
27	TS	Temperature sensor reset mask
23	CMP	CMP reset enable
22	DAC	DAC reset enable
20	ADC	ADC reset enable
15	SPI21	SPI21 reset enable
14	SPI20	SPI20 reset enable
13	UART1	UART1 reset enable
12	UART0	UART0 reset enable
11	USART13	USART13 reset enable
10	USART12	USART12 reset enable
9	USART11	USART11 reset enable
8	USART10	USART10 reset enable
6	I2C2	I2C2 reset enable
5	I2C1	I2C1 reset enable
4	I2C0	I2C0 reset enable

4.5.12 SCU_PER1: peripheral enable register1

Before using a peripheral unit, it must be activated by writing '1' to a corresponding bit in SCU_PER1 register. Before the activation, the peripheral will stay in reset state.

All peripherals are enabled by default. To disable the peripheral unit, write '0' to the corresponding bit in the SCU_PER1 register, and then the peripheral goes under reset state.

SCU_PER1=0x4000_0028

31	30 29 28 27	26	25	24	23	22	21	20	19	18	17	16	15 14	13	12	11	10	9	8	7 6 5	4	3 2 1 0
WT	Reserved	TIMER21	TIMER20	TIMER30	Reserved	TIMER16	TIMER15	TIMER14	TIMER13	TIMER12	TIMER11	TIMER10	Reserved	GPIOF	GPIOC	GPIOB	GPIOC	GPIOB	GPIOA	Reserved	DMA	Reserved
0	-	0	0	0	-	0	0	0	0	0	0	0	-	0	0	0	0	0	0	-	0	1111
RW	-	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	-	RW	RO

Bits	Name	Function
31	WT	WT function enable
26	TIMER21	TIMER21 function enable
25	TIMER20	TIMER20 function enable
24	TIMER30	TIMER30 function enable
22	TIMER16	TIMER16 function enable
21	TIMER15	TIMER15 function enable
20	TIMER14	TIMER14 function enable
19	TIMER13	TIMER13 function enable
18	TIMER12	TIMER12 function enable
17	TIMER11	TIMER11 function enable
16	TIMER10	TIMER10 function enable
13	GPIOF	GPIOF function enable
12	GPIOE	GPIOE function enable
11	GPIOD	GPIOD function enable
10	GPIOC	GPIOC function enable
9	GPIOB	GPIOB function enable
8	GPIOA	GPIOA function enable
4	DMA	DMA function enable

4.5.13 SCU_PER2: peripheral enable register 2

Before using a peripheral unit, it must be activated by writing '1' to a corresponding bit in SCU_PER2 register. Before the activation, the peripheral will stay in reset state.

All peripherals are enabled by default. To disable the peripheral unit, write '0' to the corresponding bit in the SCU_PER2 register, and then the peripheral goes under reset state.

SCU_PER2=0x4000_002C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC	Reserved		LCD	TS	Reserved			CMP	DAC	Reserved	ADC	Reserved				SPI21	SPI20	UART1	UART0	USART13	USART12	USART11	USART10	Reserved	I2C2	I2C1	I2C0	Reserved			
0	-		0	0	-			0	0	-	0	-				0	0	0	0	0	0	0	0	-	0	0	0	-			
RW	-		RW	RW	-			RW	RW	-	RW	-				RW	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	-			

Bits	Name	Function
31	CRC	CRC function enable
28	LCD	LCD function enable
27	TS	Temperature sensor function enable
23	CMP	CMP function enable
22	DAC	DAC function enable
20	ADC	ADC function enable
15	SPI21	SPI21 function enable
14	SPI20	SPI20 function enable
13	UART1	UART1 function enable
12	UART0	UART0 function enable
11	USART13	USART13 function enable
10	USART12	USART12 function enable
9	USART11	USART11 function enable
8	USART10	USART10 function enable
6	I2C2	I2C2 function enable
5	I2C1	I2C1 function enable
4	I2C0	I2C0 function enable

4.5.14 SCU_PCER1: peripheral clock enable register 1

To use peripheral unit, its clock should be activated by writing '1' to the corresponding bit in the SCU_PCER1 register. Without enabling the clock, the peripheral won't operate properly.

To stop the clock of the peripheral, write '0' to the corresponding bit in the SCU_PCER1 register, then the clock will be stopped.

SCU_PCER1=0x4000_0030

31	30 29 28 27	26	25	24	23	22	21	20	19	18	17	16	15 14	13	12	11	10	9	8	7 6 5	4	3 2 1 0
WT	Reserved	TIMER21	TIMER20	TIMER30	Reserved	TIMER16	TIMER15	TIMER14	TIMER13	TIMER12	TIMER11	TIMER10	Reserved	GPIOF	GPIOE	GPIOD	GPIOC	GPIOB	GPIOA	Reserved	DMA	Reserved
0	-	0	0	0	-	0	0	0	0	0	0	0	-	0	0	0	0	0	0	-	0	1111
RW	-	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	-	RW	RO

Bits	Name	Function
31	WT	WT clock enable
26	TIMER21	TIMER21 clock enable
25	TIMER20	TIMER20 clock enable
24	TIMER30	TIMER30 clock enable
22	TIMER16	TIMER16 clock enable
21	TIMER15	TIMER15 clock enable
20	TIMER14	TIMER14 clock enable
19	TIMER13	TIMER13 clock enable
18	TIMER12	TIMER12 clock enable
17	TIMER11	TIMER11 clock enable
16	TIMER10	TIMER10 clock enable
13	GPIOF	GPIOF clock enable
12	GPIOE	GPIOE clock enable
11	GPIOD	GPIOD clock enable
10	GPIOC	GPIOC clock enable
9	GPIOB	GPIOB clock enable
8	GPIOA	GPIOA clock enable
4	DMA	DMA clock enable

4.5.15 SCU_PCER2: peripheral clock register 2

To use peripheral unit, its clock should be activated by writing '1' to the corresponding bit in the SCU_PCER2 register. Without enabling the clock, the peripheral won't operate properly.

To stop the clock of the peripheral, write '0' to the corresponding bit in the SCU_PCER2 register, then the clock will be stopped.

SCU_PCER2=0x4000_0034

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRC	Reserved		LCD	TS	Reserved			CMP	DAC	Reserved	ADC	Reserved				SPI21	SPI20	UART1	UART0	USART13	USART12	USART11	USART10	Reserved	I2C2	I2C1	I2C0	Reserved			
0	-		0	0	-			0	0	-	0	-				0	0	0	0	0	0	0	1	-	0	0	0	-			
RW	-		RW	RW	-			RW	RW	-	RW	-				RW	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	-			

Bits	Name	Function
31	CRC	CRC clock enable
28	LCD	LCD clock enable
27	TS	Temperature sensor clock enable
23	CMP	CMP clock enable
22	DAC	DAC clock enable
20	ADC	ADC clock enable
15	SPI21	SPI21 clock enable
14	SPI20	SPI20 clock enable
13	UART1	UART1 clock enable
12	UART0	UART0 clock enable
11	USART13	USART13 clock enable
10	USART12	USART12 clock enable
9	USART11	USART11 clock enable
8	USART10	USART10 clock enable
6	I2C2	I2C2 clock enable
5	I2C1	I2C1 clock enable
4	I2C0	I2C0 clock enable

4.5.16 SCU_PPCLKSR: peripheral clock selection register

SCU_PPCLKSR register is a 32-bit register. This register is available at 32/16/8-bit access.

SCU_PPCLKSR=0x4000_0038

31 30 29 28 27 26 25 24	23	22	21	20	19	18	17	16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved		T1xCLK	Reserved	T20CLK	Reserved		T30CLK		Reserved	LCDCLK		Reserved	WTCLK		Reserved		WDTCLK
-		0	-	0	-		0		-	00		-	00		-		0
-		RW	–	RW	-		RW		-	RW		–	RW		-		RW

Bits	Name	Function
22	T1xCLK	Timer 1x Clock Selection bit
		0 SCU_MCCR1 Timer1x clock
		1 PCLK clock
20	T20CLK	Timer 20 Clock Selection bit
		0 SCU_MCCR2 Timer20 clock
		1 PCLK clock
17	T30CLK	Timer 30 Clock Selection bit
		0 SCU_MCCR2 Timer30 clock
		1 PCLK clock
7 6	LCDCLK	LCD Clock Selection bit
		00 SCU_MCCR5 LCD clock
		01 LSE clock
		10 WDTRC* clock
		11 Reserved
4 3	WTCLK	Watch Timer Clock Selection bit
		00 SCU_MCCR3 WT clock
		01 LSE clock
		10 WDTRC* clock
		11 Reserved
		NOTE: These bits should be changed during the WTEN bit of watch timer control register (WTCR) is "0b".
0	WDTCLK	Watch-dog Timer Clock Selection bit
		0 WDTRC* clock
		1 SCU_MCCR3 WDT clock

*** NOTE:** The WDTRC clock outputs 32.250KHz frequency generated by dividing the 500kHz LSI clock by 16 in the internal logic. This WDTRC clock can be the source clock of the WDT, WT, LCD and TIMER20 capture signals.

4.5.17 SCU_CSCR: clock source control register

A31G22x series has multiple clock sources to generate internal operating clocks. Each clock source can be controlled by SCU_CSCR register.

SCU_CSCR=0x4000_0040

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8		7 6 5 4 3 2 1 0	
WTIDKY		LSECON	LSICON	HSICON	HSECON
0x0000		0000	1000	0000	0000
WO		RW	RW	RW	RW

Bits	Name	Function	
31 16	WTIDKY	Write Identification Key On writes, write 0xA507 to these bits, otherwise the write is ignored.	
15 12	LSECON	External crystal sub oscillator control **NOTES:** 1. Divided LSE(/2 /4) cannot use as MCLK with clock monitoring block. For Divided LSE(/2, /4) is used as MCLK, Clock monitoring must be turn off before clock change. 2. The LSECON bit can be enabled only when PF2 and PF3 pins are selected as SXIN and SXOUT.	
		0XXX	Disable external sub crystal oscillator
		1000	Enable external sub crystal oscillator
		1001	Enable external sub crystal oscillator divide by 2
		1010	Enable external sub crystal oscillator divide by 4
		Other	Reserved
11 8	LSICON	Low speed internal oscillator control	
		0XXX	Disable low speed internal oscillator
		1000	Enable low speed internal oscillator
		1001	Enable low speed internal oscillator divide by 2
		1010	Enable low speed internal oscillator divide by 4
		Other	Reserved
7 4	HSICON	High speed internal oscillator control	
		0XXX	Disable high speed internal oscillator
		1000	Enable high speed internal oscillator
		1001	Enable high speed internal oscillator divide by 2
		1010	Enable high speed internal oscillator divide by 4
		1011	Enable high speed internal oscillator divide by 8
		1100	Enable high speed internal oscillator divide by 16
		1101	Enable high speed internal oscillator divide by 32
		1111	Reserved

3 0	HSECON	External crystal main oscillator control	
		0XXX	Disable external main crystal oscillator
		1000	Enable external main crystal oscillator
		1001	Enable external main crystal oscillator divide by 2
		1010	Enable external main crystal oscillator divide by 4
		Other	Reserved

4.5.18 SCU_SCCR: system clock control register

Selected system clock source in SCU_SCCR becomes MCLK. Before changing clock, clock sources have to be alive by SCU_CSCR register.

SCU_SCCR=0x4000_0044

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3	2	1 0
WTIDKY		Reserved		PLLINCLKSE	MCLKSEL
0x0000		-		0	00
WO		-		RW	RW

Bits	Name	Function	
31 16	WTIDKY	Write Identification Key	
		On writes, write 0x570A to these bits, otherwise the write is ignored.	
2	PLLINCLKSEL	PLL input source select register	
		0	HSI clock is used as PLLINCLK clock
		1	HSE clock is used as PLLINCLK clock
1 0	MCLKSEL	System clock select register	
		00	Internal ring oscillator(500kHz)
		01	LSE XTAL (32KHz)
		10	PLL bypassed clock
		11	PLL output clock

NOTE: When change MCLKSEL, both the currently running clock source and the clock source to be changed must be enabled. If the current clock source is turned OFF, MCLK FAIL may occur, and If the main clock is changed while the clock source to be changed is disabled, IC malfunctions.

4.5.19 SCU_CMR: clock monitoring register

The MCLK, LSE, HSE operating clocks of the A31G22x are monitored by the LSI for operational stability. Each MCLK, LSE, HSE oscillation status monitoring and interrupt function by oscillation error are supported. Monitoring function runs based on the LSI clock source.

SCU_CMR=0x4000_0048

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		MCLKREC	Reserved			LSEMNT	LSEIE	LSEFAIL	LSESTS	MCLKMNT	MCLKIE	MCLKFAIL	MCLKSTS	HSEMNT	HSEIE	HSEFAIL	HSESTS
-		0	-			0	0	0	0	1	0	0	1	0	0	0	0
-		RW	-			RW	RW	RC	RO	RW	RW	RC	RO	RW	RW	RC	RO

Bits	Name	Function	
15	MCLKREC	MCLK fail auto recovery	
		0	MCLK is changed to LSI by default when MCLKFAIL issued
		1	MCLK auto recovery is disabled
11	LSEMNT	External sub oscillator monitoring enable	
		0	External sub oscillator monitoring disabled
		1	External sub oscillator monitoring enabled
10	LSEIE	External sub oscillator fail interrupt enable	
		0	External sub oscillator fail interrupt disabled
		1	External sub oscillator fail interrupt enabled
9	LSEFAIL	External sub oscillator fail interrupt	
		0	External sub oscillator fail interrupt not occurred
		1	Read : External sub oscillator fail interrupt is pending Write : Clear pending interrupt
8	LSESTS	External sub oscillator status	
		0	Not oscillate
		1	External sub oscillator is working normally
7	MCLKMNT	MCLK monitoring enable	
		0	MCLK monitoring disabled
		1	MCLK monitoring enabled
6	MCLKIE	MCLK fail interrupt enable	
		0	MCLK fail interrupt disabled
		1	MCLK fail interrupt enabled
5	MCLKFAIL	MCLK fail interrupt	
		0	MCLK fail interrupt not occurred
		1	Read : MCLK fail interrupt is pending Write : Clear pending interrupt
4	MCLKSTS	MCLK clock status	
		0	No clock is present on MCLK

		1	Clock is present on MCLK
3	HSEMNT	External main oscillator monitoring enable	
		0	External main oscillator monitoring disabled
		1	External main oscillator monitoring enabled
2	HSEIE	External main oscillator fail interrupt enable	
		0	External main oscillator fail interrupt disabled
		1	External main oscillator fail interrupt enabled
1	HSEFAIL	External main oscillator fail interrupt	
		0	External main oscillator fail interrupt not occurred
		1	Read : External main oscillator fail interrupt is pending Write : Clear pending interrupt
0	HSESTS	External main oscillator status	
		0	Not oscillate
		1	External main oscillator is working normally

4.5.20 SCU_NMIR: NMI control register

SCU_NMIR is a non-maskable interrupt configuration register which can be set by software. NMI interrupts is connected to the microcontroller's processor directly and are used when there is a serious possibility of an external fault. There are six kinds of interrupt sources from WDT and SCU.

It will jump to NMI handler if a selected NMI event occurred and it must check peripheral event status. For clearing occurred status, it should clear the interrupt flags of that peripheral occurred. Otherwise, the NMI handler is pending until the event flag in the peripheral status register is cleared.

Write access key is required 0xA32C on SCU_NMIR[31:16] when write register.

SCU_NMIR=0x4000_004C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACCESSCODE		Reserved		SWAPFAILST	CMPINTSTS	T30INTSTS	WDTINTSTS	MCLKFAILST	LVDSTS	Reserved		SWAPFAILE	CMPINTEN	T30IEN	WDTINTEN	MCLKFAILEN	LVREN
-		-		0	0	0	0	0	0	-		0	0	0	0	0	0
WO		-		RO	RO	RO	RO	RO	RO	-		RW	RW	RW	RW	RW	RW

Bits	Name	Function
31 16	ACCESSCODE	This field enables writing access to this register. Writing 0xA32C is to enable writing.
13	SWAPFAILSTS	WDT Interrupt condition status bit This bit can't invoke NMI interrupt without enable bit 0 Not occurred 1 Event occurred **NOTE:** If the SWAPFAILSTS NMI Interrupt occurs, write access to the Bank Swap register is NOT possible.
12	CMPINTST	Comparator Interrupt condition status bit This bit can't invoke NMI interrupt without enable bit 0 Not occurred 1 Event occurred
11	T30INTSTS	TIMER30 (PWM) BLNK Interrupt condition status bit When T30IEN bit is enabled, the NMI interrupt flag is set by the high impedance interrupt occurred by the BLNK pin of TIMER30. 0 Not occurred 1 Event occurred
10	WDTINTSTS	WDT Interrupt condition status bit This bit can't invoke NMI interrupt without enable bit 0 Not occurred 1 Event occurred To clear the WDTINTSTS NMI status bit, clear the corresponding event flag in the WDT_SR register.
9	MCLKFAILSTS	MCLK Fail condition status bit This bit can't invoke NMI interrupt without enable bit 0 Not occurred

		1	Event occurred
8	LVDSTS	LVD reset condition status bit This bit can't invoke NMI interrupt without enable bit	
		0	Not occurred
		1	Event occurred To clear the LVDSTS NMI status bit, clear the corresponding event flag in the SCULV_LVICR register.
2	WDTINTEN	WDT Interrupt condition enable for NMI interrupt	
		0	Disable
		1	Enable
1	MCLKFAILEN	MCLK Fail condition enable for NMI interrupt	
		0	Disable
		1	Enable
0	LVREN	LVR Fail condition enable for NMI interrupt	
		0	Disable
		1	Enable

4.5.21 SCU_COR: clock output register

A31G22x series can drive a clock from internal MCLK clock with a dedicated post divider. To use CLKO output function, it should be set as CLKO that has output mode in PF4 Pin Mux. SCU_COR register is an 8-bit register.

SCU_COR=0x4000_0050

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5	4	3 2 1 0
Reserved				CLKOEN	CLKODIV
-				0	1111
-				RW	RW

Bits	Name	Function	
4	CLKOEN	Clock output enable	
		0	CLKO is disabled and stay "L" output
		1	CLKO Is enabled
3 0	CLKODIV	Clock output divider value $\mathrm{CLKO} = \mathrm{MCLK} \quad (CLKODIV = 0)$ or, $\mathrm{CLKO} = \frac{MCLK}{2 * (\mathrm{CLKODIV} + 1)} \quad (CLKODIV > 0)$	

4.5.22 SCU_PLLCON: PLL control register

Integrated PLL will synthesize high speed clock for extremely high performance of CPU. The PLL is controlled by setting the register. PLL Control Register is 32-bit register.

SCU_PLLCON=0x4000_0060

31	30 29 28 27 26 25 24	23	22	21	20	19	18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3 2 1 0
PLLLOCK	Reserved	PLLRSTB	PLLEN	BYPASSB	PLLMODE	Reserved	PREDIV	POSTRIV1	POSTDIV2	OUTDIV
0	-	0	0	0	0	-	000	00000000	0000	0000
RO	-	RW	RW	RW	RW	-	RW	RW	RW	RW

Bits	Name	Function
31	PLLLOCK	PLLLOCK status
		0 PLL is not locked
		1 PLL is locked
23	PLLRSTB	PLL reset **NOTES:** 1. When entering in DEEP-SLEEP mode under PLL state, this bit is automatically set to 'L' state. 2. To wake up from DEEP-SLEEP mode, this bit must be set to 'H'. If this bit maintains H' state in DEEP-SLEEP mode, current consuming may occur because of the oscillation of VCO in PLL block.
		0 PLL reset is asserted
		1 PLL reset is negated
22	PLLEN	PLL enable
		0 PLL is disabled
		1 PLL is enabled
21	BYPASSB	PLLINCLK bypass
		0 FOUT is bypassed as PLLINCLK
		1 FOUT is PLL output
20	PLLMODE	PLL VCO mode (D)
		0 VCO frequency is the same with FOUT
		1 VCO frequency is double of FOUT
18 16	PREDIV	PLLINCLK pre-divider (R)
		0~7 PLLINCLK divided by (PREDIV + 1), (PLLINCLK/1 to PLLINCLK/8)
15 8	POSTDIV1	Feedback control 1 (N1)
		0x00 N1 = 0 (N1 + 1)
		0xFF N1 = 255 (N1 + 1)
7 4	POSTDIV2	Feedback control 1 (N2)
		0x0 N2 = 0 (N2 + 1)
		0xF N2 = 15 (N2 + 1)

3 0	OUTDIV	Output divider control (P)	
		0x0	P = 0 (P+1)
		0xF	P = 15 (P+1)

NOTES:

1. When PLLEN is set to '1', PLLRSTB is set to '1' and PLL Divider value after at least 1us.
2. Wait more than 190us for PLL Lock Time, then check the PLLLOCK flag (Bit 31).
3. At power down(Stop/Standby) mode, set PLLRSTB to '0', PLLEN to '0'.
4. Output calculation formula is as followings:

$$F_{OUT} = \frac{PLLINCLK \times (N_1 + 1)}{(R + 1) \times (N_2 + 1) \times (P + 1)} * (D + 1)$$

Symbol	Description
R	Pre Divider Counter Value
N_1	Post Divider1 Counter Value
N_2	Post Divider2 Counter Value
P	Output Divider Counter Value
D	Frequency Doubler

Calculating PLL output frequency value

PLL of A31G22x series can accurately set the output frequency, F_{OUT}, in 1MHz increments. The formula for the F_{IN} input to the F_{VCO} input of the PLL is as follows, and the input range of the F_{IN} frequency is between 1MHz and 3MHz. However, it is recommended to set the input frequency (FIN) up to 2MHz as much as possible.

$$FIN = \frac{PLLINCLK}{(R + 1)}, \quad \text{Where } 1MHz \le FIN \le 3MHz \ (\text{Recommanded } FIN = 2MHz)$$

At this time, the range of F_{VCO} output frequency should be set to 200MHz or less, and the calculation formula is as follows.

$$F_{VCO} = FIN \times (N_1 + 1), \quad VCO \le 200MHz$$

SCU_PLLCON also supports the Doubler function which can double the F_{VCO} output through the bit setting of VCOMODE. When using this doubler function, the output of F_{VCOx2} should be set to 250MHz or less.

$$V_{VCOX2} = VCO \times (D + 1), \quad VCOx2 \le 250MHz \ if \ D = 1$$

As a result, the final frequency of PLL, F_{OUT}, can be obtained from the formula below using the formula above.

$$F_{OUT} = \frac{PLLINCLK \times (N_1 + 1)}{(R + 1) \times (N_2 + 1) \times (P + 1)} \times (D + 1) = \frac{FIN \times (N_1 + 1)}{(N_2 + 1) \times (P + 1)} \times (D + 1)$$

4.5.23 SCU_VDCCON: VDC control register

On chip VDC control register. VDCTRIM is used for the trim value of VDC output. VDCWDLY value can be written with writing '1' to VDCWDLY_WEN bit simultaneously.

SCU_VDCCON=0x4000_0064

31 30 29 28 27 26 25 24	23 22 21 20	19	18	17	16	15	14 13 12 11 10 9	8	7 6 5 4 3 2 1 0
Reserved	VDC_WTIDKY	VDC_PDBGR	Reserved	VDC_STOP	VDC_IDLE	VDC_LOCK	Reserved	VDCWDLY_WEN	VDCWDLY
-	0000	0	-	0	0	0	-	0	0111_1111
-	WO	RW	-	RW	RW	RW	-	WO	RW

Bits	Name	Function
23 20	VDC_WTIDKY	VDC15 Write Identification Key On writes, write 0x5 to these bits, otherwise the write is ignored.
19	VDC_PDBGR	Power down mode selection signal when CPU enters DEEP-SLEEP mode *In BGR on → off, VDCLOCK should be 0
		0 BGR/Buffer ON (RUN / IDLE / STOP1)
		1 BGR/Buffer OFF (STOP2)
17	VDC_STOP	Power down mode selection signal when CPU enters DEEP-SLEEP mode
		0 Stop Mode
		1 Standby Mode
16	VDC_IDLE	VDC IDLE Mode Control Signal
		0 no IDLE Mode (RUN or STOP1 or STOP2 Mode)
		1 IDLE Mode
15	VDC_LOCK	VDC_LOCK Control Signal for *BGR Stabilization *In BGR off → on Sequence, VDCLOCK should be 0 during 120usec
		0 VDC Using BGR Reference Voltage
		1 VDC Using BMR Reference Voltage
8	VDCWDLY_WEN	VDCWDLY value write enable. VDCWDLY value can be written with writing '1' to VDCWDLY_WEN bit simultaneously.
7 0	VDCWDLY	VDC warm-up delay count value. When SCU is waked up from power down mode, the warm-up delay is inserted for VDC output being stabilized. The amount of delay basically be defined with this field 0x7F : 4ms

NOTE: Reserved bits should never be modified.

4.5.24 SCU_LSICON: low speed internal OSC control register

SCU_LSICON=0x4000_006C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1	0
Reserved				SKIP_LS
-				1
-				RW

Bits	Name	Function	
0	SKIP_LS	Internal Level Shifter control signal	
		0	Enable
		1	Disable

4.5.25 SCU_EOSCR: external oscillator control register

External main crystal oscillator has two characteristics. For the noise immunity, NMOS amp type is recommended and for the low power characteristic, INV amp type is recommended.

SCU_EOSCR=0x4000_0080

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13 12	11 10 9	8	7	6	5 4	3 2	1	0
Reserved		ESEN	Reserved	ESISEL	Reserved	ESNCBYPS	EMEN	Reserved	ISEL	NCOPT	Reserved	NCSKIP
-		0	-	01	-	0	0	-	01	01	-	0
-		W	-	RW	-	RW	W	-	RW	RW	-	RW

Bits	Name	Function	
15	ESEN	Write enable for External LSE	
		0	Write access disabled
		1	Write access enabled
13 12	ESISEL	Select current for External LSE	
		00	1.57uA
		01	1.79uA (Default)
		10	1.93uA
		11	2.04uA
8	ESNCBYPS	Noise canceling for LSE oscillation	
		0	Use noise canceling (Default)
		1	Not use noise canceling
7	EMEN	Write enable for External HSE	
		0	Write access disabled
		1	Write access enabled
5 4	ISE	Select current for External HSE	
		00	Typ. 6.72mA, 1 to 16MHz
		01	Typ. 6.41mA, 1 to 12MHz
		10	Typ. 5.78mA, 1 to 8MHz
		11	Typ. 2.71mA, 1 to 4MHz
3 2	NCOPT	Noise Cancel delay Option for External HSE	
		00	Typ. 23ns, 1 to 4MHz
		01	Typ. 18ns, 1 to 8MHz
		10	Typ. 13ns, 1 to 12MHz
		11	Typ. 8ns, 1 to 16MHz
0	NCSKIP	Noise canceling for HSE oscillation	
		0	Use noise canceling (Default)
		1	Not use noise canceling

NOTE: The HSE's ISE(current option), NCOPT(noise cancel option) and NCSKIP(use or no use noise canceling) must be set along with the EMEN bit (allow writes to external HSE).

4.5.26 SCU_EMODR: external mode status register

SCU_EMODR register shows external mode pin status while booting.

SCU_EMODR=0x4000_0084

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			BOOT
-			0
-			RO

Bits	Name	Function
0	BOOT	BOOT pin (PB3) level
		0 BOOT pin (PB3) is low
		1 BOOT pin (PB3) is high

4.5.27 SCU_RSTDBCR: pin reset debounce control register

Pin Reset Debounce Control Register.

SCU_ RSTDBCR=0x4000_0088

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
WTIDKY		Reserved	CLKCNT	Reserved	EN
0x0000		-	0x00	-	0
WO		-	RW	-	RW

Bits	Name	Function
31 16	WTIDKY	Write Identification Key
		On writes, write 0x0514 to these bits, otherwise the writing value is ignored.
13 8	CLKCNT	Noise cancel delay option for external HSE
		N N clock checking for debounce by LSI (500kHz)
0	EN	Pin reset debounce enable bit
		0 Disable
		1 Enable

NOTE: If a user wants to operate pin reset debounce, the user must enable LSI (500kHz). Because pin reset debounce uses LSI for clock source.

4.5.28 SCU_MCCR1: miscellaneous clock control register 1

A31G22x series can drive a clock from internal MCLK clock with dedicated post divider. STCSEL bits and STCDIV bits of MCCR1 are used as SYSTICK external clock source. TEXT1CSEL bits and TEXT1DIV bits of SCU_MCCR1 are used as TIMER1n external clock source. Register SCU_MCCR1 is a 32-bit register.

SCU_MCCR1=0x4000_0090

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved / TEXT1CSEL	TEXT1DIV	Reserved / STCSEL	SYSTICKDIV
- / 0x0	0x00	- / 0x0	0x00
- / RW	RW	- / RW	RW

Bits	Name	Function	
26 24	TEXT1CSEL	TIMER1n external clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI 32MHz
		110	HSE
		111	PLL Clock
23 16	TEXT1DIV	TIMER1n external clock N divider **NOTE:** To change the value, set 0x0 first without changing TEXT1CSEL.	
		0	Disabled
		N	(Selected clock) / N
10 8	STCSEL	SYSTICK external clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI 32MHz
		110	HSE
		111	PLL Clock
7 0	STDIV	SYSTICK clock N divider **NOTE:** To change the value, set 0x0 first without changing STCSEL. When the STDIV value is set to 0, the SYSTICK clock input is stopped.	
		0	Disabled
		N	(Selected clock) / N

4.5.29 SCU_MCCR2: miscellaneous clock control register 2

A31G22x series can drive a clock from internal MCLK clock with a dedicated post divider. TEXT2CSEL bits and TEXT2DIV bits of MCCR2 are used as a TIMER20 external clock source. TEXT3CSEL bits and TEXT3DIV bits of SCU_MCCR2 are used as a TIMER30 external clock source. Register SCU_MCCR2 is a 32-bit register.

SCU_MCCR2=0x4000_0094

31 30 29 28 27	26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11	10 9 8	7 6 5 4 3 2 1 0
Reserved	TEXT3CSEL	TEXT3DIV	Reserved	TEXT2CSEL	TEXT2DIV
-	0x0	0x00	-	0x0	0x00
-	RW	RW	-	RW	RW

Bits	Name	Function	
26 24	TEXT3CSEL	TIMER 30 EXT Clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
23 16	TEXT3DIV	TIMER30 EXT Clock N divider **NOTE:** To change the value, set 0x0 first without changing TEXT3CSEL	
		0	Disabled
		N	(Selected clock) / N
10 8	TEXT2CSEL	TIMER20 EXT Clock source select bit*	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
7 0	TEXT2DIV	TIMER20 EXT Clock N divider **NOTE:** To change the value, set 0x0 first without changing TEXT2CSEL	
		0	Disabled
		N	(Selected clock) / N

4.5.30 SCU_MCCR3: miscellaneous clock control register 3

A31G22x series can drive a clock from an internal MCLK clock with a dedicated post divider. WDTCSEL bits and WDTDIV bits of a MCCR3 are used as WDT external clock source. WTEXTCSEL bits and WTEXTCDIV bits of SCU_MCCR3 are used as a WT external clock source. This register is a 32-bit register.

SCU_MCCR3=0x4000_0098

31 30 29 28	27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4 3 2 1 0
Reserved	WTEXTCSEL	WTEXTCDIV	Reserved	WDTCSEL	WDTDIV
-	0x0	0x00	-	0x0	0x00
-	RW	RW	-	RW	RW

Bits	Name	Function	
26 24	WTEXTCSEL	WT external Clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
23 16	WTEXTCDIV	WT external clock N divider **NOTE:** To change the value, set 0x0 first without changing WTEXTCSEL	
		0	Disabled
		N	(Selected clock) / N
10 8	WDTCSEL	WDT external clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
7 0	WDTDIV	WDT external clock N divider **NOTE:** To change the value, set 0x0 first without changing WDTCSEL	
		0	Disabled
		N	(Selected clock) / N

4.5.31 SCU_MCCR4: miscellaneous clock control register 4

A31G22x series can drive a clock from an internal MCLK clock with a dedicated post divider. PD0CSEL bits and PD0DIV bits of SCU_MCCR4 are used as PA, PB, and PC debounce clock source. PD1CSEL bits and PD1DIV bits of SCU_MCCR4 are used as PD, PE, and PF debounce clock source. This register is a 32-bit register.

SCU_MCCR4=0x4000_009C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	PD1CSEL	PD1DIV	Reserved	PD0CSEL	PD0DIV
-	000	0000_0000	-	000	0000_0000
-	RW	RW	-	RW	RW

Bits	Name	Function	
26 24	PD1CSEL	Debounce Clock for PORT source select bit (PD, PE, PF)	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
23 16	PD1DIV	PORT Debounce Clock N divider (PD, PE, PF) **NOTE:** 1. To change the value, set 0x0 first without changing PD1CSEL 2. Clock is not activated during PD1DIV bit is disabled.	
		0	Disabled
		N	(Selected clock) / N
10 8	PD0CSEL	Debounce Clock for PORT source select bit (PA, PB, PC)	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
7 0	PD0DIV	PORT Debounce Clock N divider (PA, PB, PC) **NOTE:** 1. To change the value, set 0x0 first without changing PD0CSEL 2. Clock is not activated during PD0DIV bit is disabled.	
		0	Disabled
		N	(Selected clock) / N

4.5.32 SCU_MCCR5: miscellaneous clock control register 5

A31G22x series can drive a clock from an internal MCLK clock with a dedicated post divider. LCDCSEL bits and LCDDIV bits of MCCR5 are used as LCD clock source. TSREFCLKSEL, TSSENSECLKSEL and TSLSIEN bits of SCU_MCCR5 are used as temperature sensor clock source.

SCU_MCCR5=0x4000_00A0

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	LCDCSEL	LCDDIV	TSRCSEL	TSSCSEL	TSLSITSEN	Reserved
-	000	0000_0000	00	00	0	-
-	RW	RW	RW	RW	RW	-

Bits	Name	Function	
26 24	LCDCSEL	LCD clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
23 16	LCDDIV	LCD clock N divider **NOTE:** To change the value, set 0x0 first without changing LCDCSEL. Divider value (N) must be larger than '0' and the divided LCD clock must be slower than the system clock.	
		0	Disabled
		N	Divided LCD clock source by divider N (Clock source/N)
15 14	TSRCSEL	Temperature sensor's reference clock select bit **NOTE:** When reference clock is LSE, reference and sense clock are changed to each other.	
		00	HSI (Output clock set by HSI from SCU_CSCR)
		01	MCLK (Bus Clock)
		10	HSE
		11	LSE (32.768KHz)
13 12	TSSCSEL	Temperature sensor's sensing clock select bit	
		00	LSITS (Internal temperature sensing oscillator)
		01	LSI (500KHz)
		10	Reserved
		11	HSI (Output clock set by HSI from SCU_CSCR)
11	TSLSITSEN	Enable temperature sensing oscillator (LSITS)	
		0	Disable
		1	Enable

4.5.33 SCU_MCCR6: miscellaneous clock control register 6

SCU_MCCR6 is a reserved register.

SCU_MCCR6=0x4000_00A4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																															
-																															
-																															

Bits	Name	Function
31 0	Reserved	Reserved

4.5.34 SCU_MCCR7: miscellaneous clock control register 7

A31G22x series can drive a clock from an internal MCLK clock with a dedicated post divider. ADCCSEL bits and ADCCDIV bits of MCCR7 are used as an ADCI clock source.

SCU_MCCR2=0x4000_00A4

31 30 29 28 27	26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	ADCCSEL	ADCCDIV	Reserved	
-	000	0000_0000	-	
-	RW	RW	-	

Bits	Name	Function	
26 24	ADCCSEL	ADC clock source select bit	
		0xx	LSI (500KHz)
		011	LSE (32.768KHz)
		100	MCLK (Bus clock)
		101	HSI (32MHz)
		110	HSE
		111	PLL Clock
16	ADCCDIV	ADC clock N divider **NOTE:** To change the value, set 0x0 first without changing ADCCSEL. Divider value (N) must be larger than '0' and the divided LCD clock must be slower than the system clock.	
		0	Disabled
		N	Divided ADC clock source by divider N (Clock source/N)

4.5.35 SCULV_LVICR: low voltage indicator control register

The LVD (Low Voltage Detection) block detects low voltage by 11 levels. When it detects a specified level of low voltage which was predefined in SCULV_LVICR register, an LVI interrupt and a flag are generated. This function prevents system malfunction caused by unstable voltage supply and steadily supports to drive user applications.

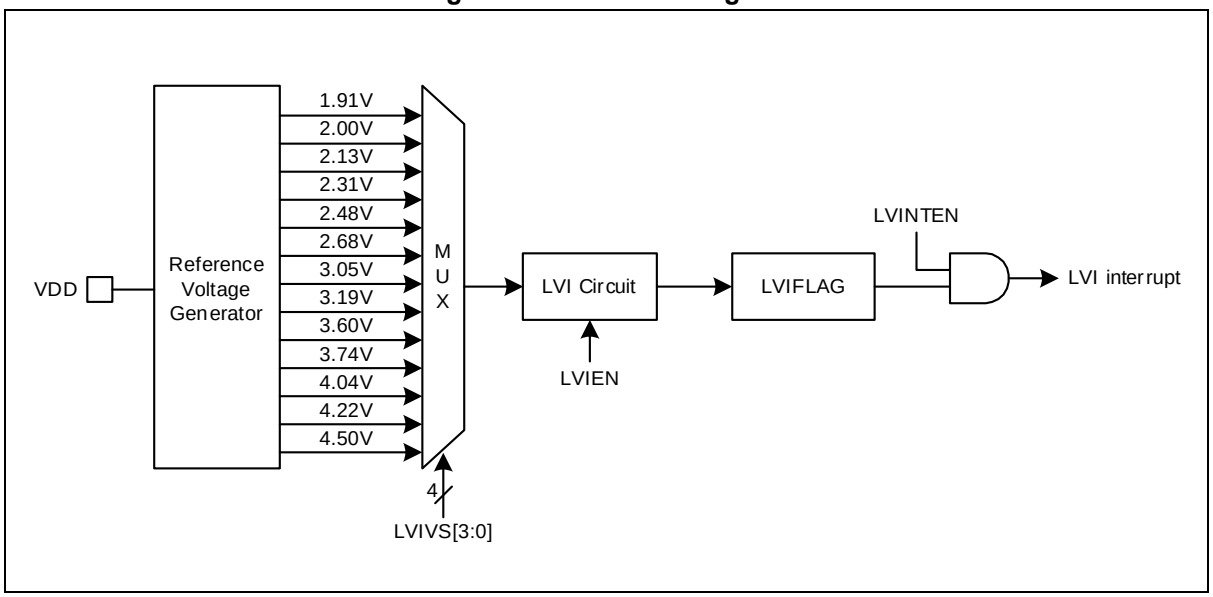
SCULV_LVICR=0x4000_5100

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3 2 1 0
Reserved			LVIFLAG	LVIEN	Reserved	LVINTEN	LVIST	LVIVS
-			0	0	-	0	0	0000
-			RW	RW	-	RW	RO	RW

Bits	Name	Function	
8	LVIFLAG	LVI interrupt flag bit	
		0	No request occurred
		1	Request occurred. To release the LVI interrupt and to clear the LVI flag bit, the supply voltage of the system must be higher than the LVI level set in the SCUCR_LVICR register.
7	LVIEN	LVI Enable bit.	
		0	Disable low voltage indicator.
		1	Enable low voltage indicator.
5	LVINTEN	LVI Interrupt Enable bit.	
		0	Disable low voltage indicator interrupt.
		1	Enable low voltage indicator interrupt.
4	LVIST	LVI Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred. This bit is automatically cleared by system operating voltage level.
3 0	LVIVS	LVI Voltage Selection bits.	
		0000	Reserved
		0001	Reserved
		0010	Reserved
		0011	Reserved
		0100	Reserved
		0101	2.13V
		0110	2.31V
		0111	2.48V
		1000	2.68V
		1001	3.05V

	1010	3.19V
	1011	3.60V
	1100	3.74V
	1101	4.04V
	1110	4.22V
	1111	4.50V

Figure 24. LVI Block Diagram



4.5.36 SCULV_LVRCR: low voltage reset control register

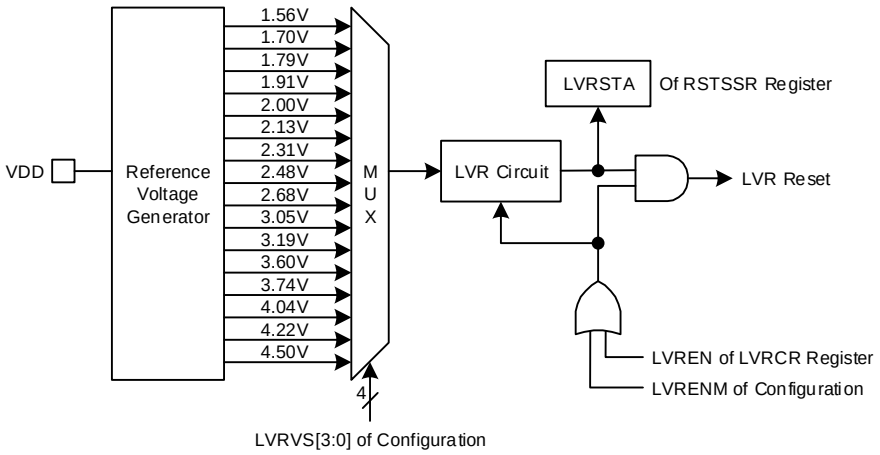
SCULV_LVRCR register can enable a system reset function of the LVD block as shown in Figure 25. LVD reset function can be defined by configuring SCULV_LVRCNFIG register.

SCULV_LVRCR=0x4000_5104

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			LVREN
-			0x00
-			RW

Bits	Name	Function	
7 0	LVREN	LVR Enable bits. These bits are cleared to 0x00 by only POR and retained by other reset signals.	
		0x55	Disable low voltage reset.
		Others	Enable low voltage reset.

Figure 25. LVR Block Diagram



NOTE: The voltage level of LVR is set by the low voltage reset configuration register (LVRCNFG).

4.5.37 SCULV_LVRCNFIG: configuration for low voltage reset

The LVD (Low Voltage Detection) block detects low voltage by 16 levels. When it detects a specified level of low voltage which was predefined in SCULV_LVRCNFIG register, an LVD reset is generated. This function prevents system malfunction caused by unstable voltage supply and steadily supports to drive user applications.

To release a system reset using the LVD reset, the system operating voltage must be higher than the LVR level defined in the SCUCR_LVRCR register.

SCULV_LVRCNFIG =0x4000_5108

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3 2 1 0
WTIDKY		LVRENM	Reserved	LVRVS
0x0000		0x00	-	0xF
WO		RW	-	RW

Bits	Name	Function	
31 24	WTIDKY	Write Identification Key	
		On writes, write 0x72A5 to these bits, otherwise the write is ignored.	
15 8	LVRENM	LVR Reset Operation Control Master Configuration	
		0xAA	LVR operation is decided by the LVREN of LVRCR register
		Others	Master enable LVR operation
3 0	LVRVS	LVR Voltage Selection bits.	
		1111	1.56V
		1110	1.70V
		1101	1.79V
		1100	1.91V
		1011	2.00V
		1010	2.13V
		1001	2.31V
		1000	2.48V
		0111	2.68V
		0110	3.05V
		0101	3.19V
		0100	3.60V
		0011	3.74V
		0010	4.04V
		0001	4.22V
		0000	4.50V

5 PCU and GPIO

Port Control Unit (PCU) configures and controls external I/Os as listed in the followings:

- External signal directions of each pins
- Interrupt trigger mode for each pins
- Internal pull-up/down register control and open drain control
- Remaps function to use USART1n channel as SPI2n channel. (n = 0, 1)
- Controls sink-type LED ports for high current driving (SEG0 to SEG9)
- SPI20 port strength configuration for high speed communication.

General Purpose Input/Output (GPIO) is corresponding to the most pins except dedicated-function pins. GPIO ports are controlled by GPIO block as listed in the followings:

- Output signal level (H/L) select
- External interrupt interface (Level, Rising edge, Falling edge, Both edge)
- Pull up/down enable or disable

Up to 75 pins in Table 20 are assigned for PCU and GPIO blocks.

Table 20. PCU and GPIO pins

Pin name	Type	80-pin: 75 ports	64-pin: 59 ports	48-pin: 43 ports
PA	IO	PA0 to PA11	PA0 to PA11	PA0 to PA8
PB	IO	PB0 to PB15	PB0 to PB11	PB0 to PB7
PC	IO	PC0 to PC12	PC0 to PC7 PC11 to PC12	PC0 to PC4
PD	IO	PD0 to PD5	PD0 to PD5	PD0 to PD5
PE	IO	PE0 to PE15	PE0 to PE11	PE0 to PE7
PF	IO	PF0 to PF11	PF0 to PF7	PE0 to PF7

5.1 PCU and GPIO block diagram

Figure 26 describes PCU in block diagram.

Figure 26. PCU Block Diagram

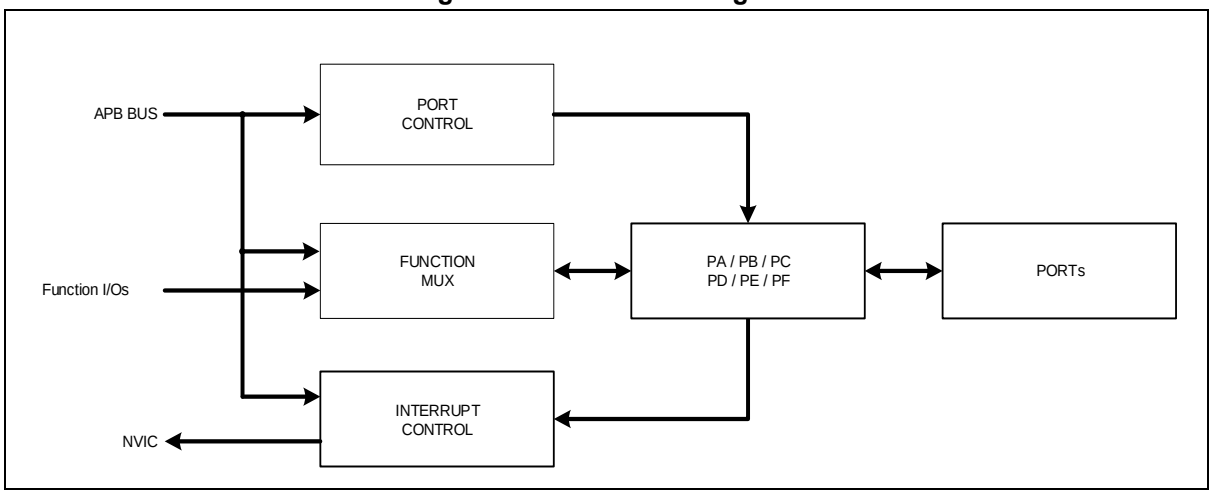


Figure 27 describes GPIO in block diagram, and Figure 28 introduces external interrupt I/O pins.

Figure 27. GPIO Block Diagram

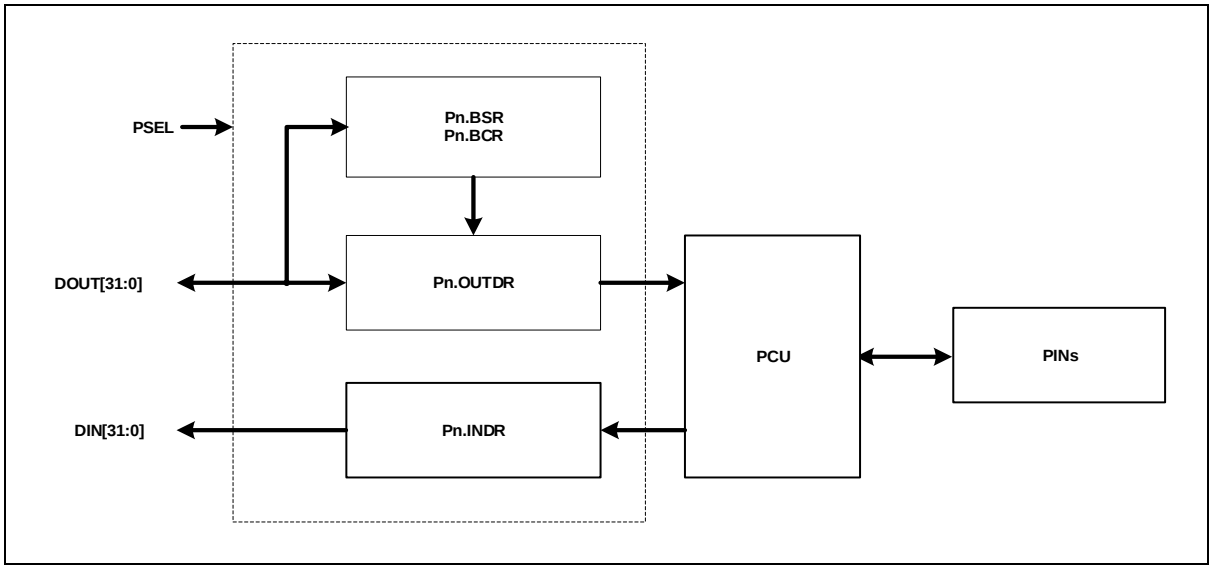


Figure 28. I/O Port Block Diagram (External Interrupt I/O Pins)

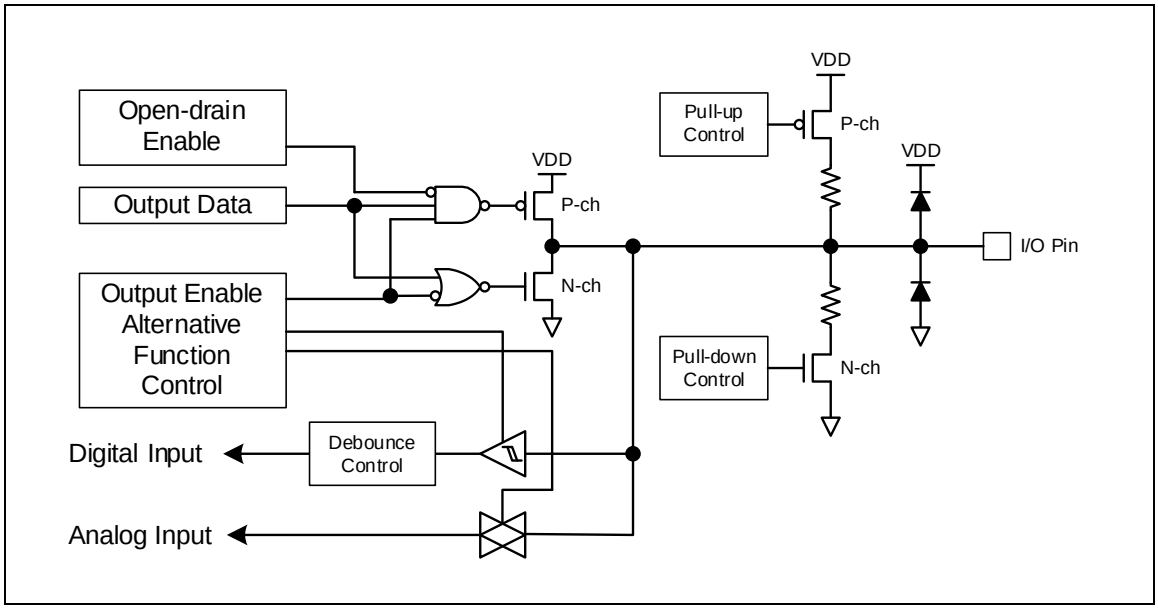
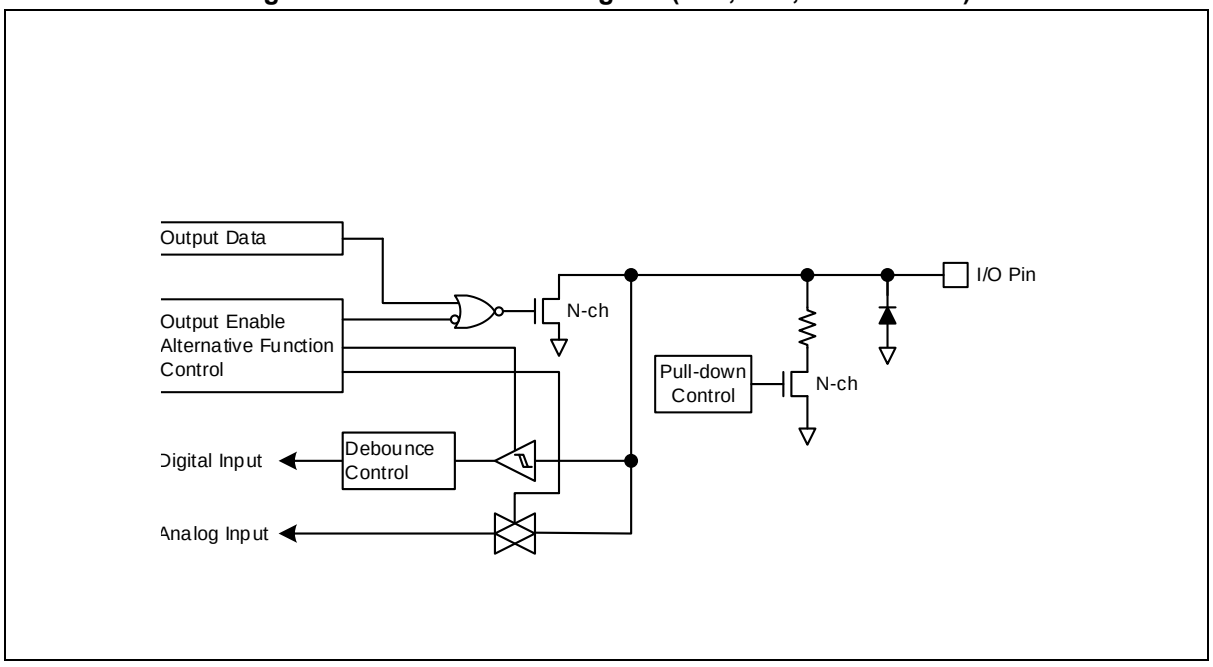


Figure 29. I/O Port Block Diagram (PF5, PF6, PF7 I/O Pins)



5.2 Pin multiplexing

GPIO pins have alternative function pins.

Table 21 shows pin multiplexing information.

Table 21. GPIO Alternative Function

Pin name	Alternative function				
	AF0	AF1	AF2	AF3	AF4
PA0		I2C1_SDA1		ADC_AN0	
PA1		I2C1_SCL1		ADC_AN1	
PA2		TIMER12_EC12		ADC_AN2/ CP0A	
PA3				ADC_AN3/ CP1C	
PA4				ADC_AN4/ CP1B	
PA5		TIMER12_T12O	TIMER12_T12C	ADC_AN5/ CP1A	
PA6	LCD_SEG43	TIMER11_T11O+	TIMER11_T11C+	ADC_AN6/ CREF1/ DAO	
PA7	LCD_SEG42			ADC_AN7/ CREF0	
PA8				ADC_AN11	
PA9				ADC_AN12	
PA10				ADC_AN13	
PA11				ADC_AN14	
PB0	LCD_SEG41	USART10_TXD10	USART10_MOSI10/ SPI20_MOSI20	ADC_AN8	
PB1	LCD_SEG40	USART10_RXD10	USART10_MISO10/ SPI20_MISO20	ADC_AN9	
PB2	LCD_SEG39		USART10_SCK10/ SPI20_SCK20	ADC_AN10	
PB3	LCD_SEG38	nBOOT	USART_SS10/ SPI20_SS20		
PB4	LCD_SEG37	UART0_TXD0	SWD_SWCLK		
PB5	LCD_SEG36	UART0_RXD0	SWD_SWDIO		
PB6	LCD_SEG35	UART1_TXD1	T11_EC11+	ADC_AN15	
PB7	LCD_SEG34	UART1_RXD1		ADC_AN16	
PB8	LCD_SEG33	T15_T15O	T15_T15C	T16_EC16	
PB9	LCD_SEG32	T16_T16O	T16_T16C	T15_EC15	
PB10	LCD_SEG31	T16_T16C	T15_EC15	T16_T16O	
PB11	LCD_SEG30	T15_T15C	T16_EC16	T15_T15O	
PB12	LCD_SEG29				
PB13	LCD_SEG28				
PB14	LCD_SEG27				
PB15	LCD_SEG26				
PC0	LCD_SEG25	T20_T20O	T20_T20C	ADC_AN17	
PC1	LCD_SEG24	T20_T21O	T20_T21C		
PC2	LCD_SEG23	T20_EC20	SPI20_MOSI20		
PC3	LCD_SEG22	T21_EC21	SPI20_MISO20		

Table 21. GPIO Alternative Function

Pin name	Alternative function				
	AF0	AF1	AF2	AF3	AF4
PC4	LCD_SEG21		SPI20_SCK20		
PC5	LCD_SEG20	I2C2_SDA2			
PC6	LCD_SEG19	I2C2_SCL2			
PC7	LCD_SEG18				
PC8	LCD_SEG17				
PC9	LCD_SEG16				
PC10	LCD_SEG15				
PC11	LCD_SEG14	T10_EC10			
PC12	LCD_SEG13	T11_EC11			
PD0	LCD_SEG12	I2C0_SCL0	SPI20_SS20		
PD1	LCD_SEG11	I2C0_SDA0	T10_EC10		
PD2	LCD_SEG10	USART11_TXD11	USART11_MOSI11/ SPI21_MOSI21		LED_ISEG9
PD3	LCD_SEG9	USART11_RXD11	USART11_MISO11/ SPI21_MISO21		LED_ISEG8
PD4	LCD_SEG8	T30_BLNK	USART11_SCK11/ SPI21_SCK21		LED_ISEG7
PD5	LCD_SEG7		USART11_SS11/ SPI21_SS21		LED_ISEG6
PE0	LCD_COM0	TIMER30_PWM30AA	USART11_SS11		
PE1	LCD_COM1	TIMER30_PWM30AB			
PE2	LCD_COM2	TIMER30_PWM30BA	SPI21_SS21		LED_ISEG0
PE3	LCD_COM3/ LCD_SEG0	TIMER30_PWM30BB	SPI21_SCK21		LED_ISEG1
PE4	LCD_COM4/ LCD_SEG1	TIMER30_PWM30CA	SPI21_MISO21		LED_ISEG2
PE5	LCD_COM5/ LCD_SEG2	TIMER30_PWM30CB	SPI21_MOSI21		LED_ISEG3
PE6	LCD_COM6/ LCD_SEG3	TIMER10_T10O	TIMER10_T10C		LED_ISEG4
PE7	LCD_COM7/ LCD_SEG4	TIMER11_T11O	TIMER11_T11C		LED_ISEG5
PE8		USART13_TXD13	USART13_MOSI13	LCD_VLC0	
PE9		USART13_RXD13	USART13_MISO13	LCD_VLC1	
PE10			USART13_SCK13	LCD_VLC2	
PE11			USART_SS13	LCD_VLC3	
PE12		USART12_TXD12	USART12_MOSI12		
PE13		USART12_RXD12	USART12_MISO12		
PE14			USART12_SCK12		
PE15			USART_SS12		

Table 21. GPIO Alternative Function

Pin name	Alternative function				
	AF0	AF1	AF2	AF3	AF4
PF0		I2C1_SCL1		XOUT	
PF1		I2C1_SDA1		XIN	
PF2		UART1_TXD1		SXIN	
PF3		UART1_RXD1		SXOUT	
PF4		CLKO			
PF5		TIMER30_BLNK			
PF6		TIMER30_EC30	I2C0_SCL0		
PF7		TIMER30_T30C	I2C1_SDA0		
PF8		TIMER13_EC13			
PF9		TIMER14_EC14			
PF10		TIMER13_T13O	TIMER13_T13C		
PF11		TIMER14_T14O	TIMER14_T14C		

NOTE: On connection with debugger host, SWCLK and SWDIO pins are always for SW-DP pins. So, the corresponding bits of PB_MOD/PB_TYP/PB_AFSR1/PB_PUPD registers may not be written by software.

5.3 Registers

Base address of GPIO is introduced in the followings:

Table 22. Base Address of PCU

Name	Base address	Description
PA	0x4000_1000	General Port A
PB	0x4000_1100	General Port B
PC	0x4000_1200	General Port C
PD	0x4000_1300	General Port D
PE	0x4000_1400	General Port E
PF	0x4000_1500	General Port F

Table 23. PCU and GPIO Register Map

Name	Offset	Type	Description	Reset value	Ref.
Pn_MOD	0x0000	RW	Port n Mode Register	PA:0x00FF_FFFF PB:0xFFFF_FABF PC:0x03FF_FFC3 PD:0x0000_0FFF PE:0xFFFF_FFFF PF:0x00FF_FFFF	5.3.1
Pn_TYP	0x0004	RW	Port n Output Type Selection Register	0x0000_0000	5.3.2
Pn_AFSR1	0x0008	RW	Port n Alternative Function Selection Register 1	Pn:0x0000_0000 PB:0x0022_1000	5.3.3
Pn_AFSR2	0x000C	RW	Port n Alternative Function Selection Register 2	0x0000_0000	5.3.4
Pn_PUPD	0x0010	RW	Port n Pull-up/down Resistor Selection Register	Pn:0x0000_0000 PB:0x0000_0540 PC:0x0000_0014	5.3.5
Pn_INDR	0x0014	RO	Port n Input Data Register	0x0000_XXXX	5.3.6
Pn_OUTDR	0x0018	RW	Port n Output Data Register	0x0000_0000	5.3.7
Pn_BSR	0x001C	WO	Port n Output Bit Set Register	0x0000_0000	5.3.8
Pn_BCR	0x0020	WO	Port n Output Bit Clear Register	0x0000_0000	5.3.9
Pn_OUTDMSK	0x0024	RW	Port n Output Data Mask Register	0x0000_0000	5.3.10
Pn_DBCR	0x0028	RW	Port n Debounce Control Register	0x0000_0000	5.3.11
Pn_IER	0x002C	RW	Port n interrupt enable register	0x0000_0000	5.3.12
Pn_ISR	0x0030	RW	Port n interrupt status register	0x0000_0000	5.3.13
Pn_ICR	0x0034	RW	Port n interrupt control register	0x0000_0000	5.3.14
PF_PLSR	0x0038	RW	Port F Level Selection Register	0x0000_0000	5.3.15
Pn_STR	0x0040	RW	Port Strength Configuration Register	0x0000_0000	5.3.16

NOTE: Where n = A, B, C, D, E and F

Base address of PCU1 is introduced in the followings:

Table 24. Base Address of PCU1

Name	Base address	Description
PCU1	0x4000_1544	Port Control Unit 1

Table 25. PCU1 Register Map

Name	Offset	Type	Description	Reset value	Ref.
PCU1_SPI2PMR	0x0000	RW	SPI2n Pin-Re-Map Register	0x0000_0000	5.3.17

Base address of PCU2 is introduced in the followings:

Table 26. Base Address of PCU2

Name	Base address	Description
PCU2	0x4000_1F00	Port Control Unit 2

Table 27. PCU2 Register Map

Name	Offset	Type	Description	Reset value	Ref.
PCU2_ISEGKEY	0x0000	RW	LED ISEG Port Enable Register	0x0000_0000	5.3.18
PCU2_ISEGR	0x0004	RW	LED ISEG Port Register	0x0000_03FF	5.3.19
PCU2_ISEGIR	0x0010	RW	LED ISEG Inversion Port Register	0x0000_0000	5.3.20
PCU2_PORTEN	0x00F0	WO	Port Access Enable Register	0x0000_0000	0

5.3.1 Pn_MOD: port n mode register

Input or output control of each port pin. Each pin can be configured as an input pin, an output pin or an alternative function pin. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_MOD=0x4000_1000, PB_MOD=0x4000_1100
PC_MOD=0x4000_1200, PD_MOD=0x4000_1300
PE_MOD=0x4000_1400, PF_MOD=0x4000_1500

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
MODE15	MODE14	MODE13	MODE12	MODE11	MODE10	MODE9	MODE8	MODE7	MODE6	MODE5	MODE4	MODE3	MODE2	MODE1	MODE0
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
2x+1 2x	MODEx	Port n Mode Selection bits, x:0 to 15	
		00	Input mode
		01	Output mode
		10	Alternative function mode
		11	Reserved

NOTES:
1. For exception, the reset value is, PA_MOD: 0x00FF_FFF, PB_MOD: 0xFFFF_FABF, PC_MOD: 0x03FF_FFC3, PD_MOD: 0x0000_0FFF, PE_MOD: 0xFFFF_FFFF, PF_MOD: 0x00FF_FFFF.
2.

5.3.2 Pn_TYP: port n output type selection register

Pn_TYP selects control option from a Push-pull output and Open-drain output for each port pin. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_MOD=0x4000_1004, PB_MOD=0x4000_1104
PC_MOD=0x4000_1204, PD_MOD=0x4000_1304
PE_MOD=0x4000_1404, PF_MOD=0x4000_1504

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		TYP15	TYP14	TYP13	TYP12	TYP11	TYP10	TYP9	TYP8	TYP7	TYP6	TYP5	TYP4	TYP3	TYP2	TYP1	TYP0
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
x	TYPx	Port n Output Type Selection bits, x:0 to 15	
		0	Push-pull output
		1	Open-drain output

5.3.3 Pn_AFSR1: port n alternative function selection register 1

Pn_AFSR1 registers must be set properly before using this port. Otherwise the port cannot be guaranteed for its functionality. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_AFSR1=0x4000_1008, PB_AFSR1=0x4000_1108
PC_AFSR1=0x4000_1208, PD_AFSR1=0x4000_1308
PE_AFSR1=0x4000_1408, PF_AFSR1=0x4000_1508

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
AFSB7	AFSB6	AFSB5	AFSB4	AFSB3	AFSB2	AFSB1	AFSB0
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
4x+3 4x	AFSBx	Port n Alternative Function Selection bits, x:0 to 7	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		Others	Reserved

NOTES:

1. When HSE is used as the system clock (MCLK), the AFSBx bits for PF0 (XOUT), PF1 (XIN) must be configured as AF3 before changing the system clock and the value should not be changed.
2. If the software controls to use a port supported by an analog block (ADC, DAC, CMP, XTAL, SXTAL, etc.) for other purposes, such as GPIO, the software need to disable that analog peripheral.

5.3.4 Pn_AFSR2: port n alternative function selection register 2

Pn_AFSR2 registers must be set properly before using this port. Otherwise the port cannot be guaranteed for its functionality. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_AFSR2=0x4000_100C, PB_AFSR2=0x4000_110C
PC_AFSR2=0x4000_120C, PD_AFSR2=0x4000_130C
PE_AFSR2=0x4000_140C, PF_AFSR2=0x4000_150C

31 30 29 28 27 26 25 24		23 22 21 20 19 18 17 16		15 14 13 12 11 10 9 8		7 6 5 4 3 2 1 0	
AFSB15	AFSB14	AFSB13	AFSB12	AFSB11	AFSB10	AFSB9	AFSB8
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
4(x-8)+3 4(x-8)	AFSRx	Port n Alternative Function Selection bits, x:8 to 15	
		0000	Alternative Function 0 (AF0)
		0001	Alternative Function 1 (AF1)
		0010	Alternative Function 2 (AF2)
		0011	Alternative Function 3 (AF3)
		0100	Alternative Function 4 (AF4)
		Others	Reserved

NOTES:

1. When LSE is used as the system clock (MCLK) the AFSBx bits for PF2(SXIN), PF3(SXOUT) must be configured as AF3 before changing the system clock and the value should not be changed.
2. If the software controls to use a port supported by an analog block (ADC, DAC, CMP, XTAL, SXTAL, etc.) for other purposes, such as GPIO, the software need to disable that analog peripheral.

5.3.5 Pn_PUPD: port n pull-up/down resistor selection register

Each pin of the ports has on-chip pull-up/down resistor which can be configured by Pn_PUPD registers. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_PUPD=0x4000_1010, PB_PUPD=0x4000_1110
PC_PUPD=0x4000_1210, PD_PUPD=0x4000_1310
PE_PUPD=0x4000_1410, PF_PUPD=0x4000_1510

31 30 29 28 27 26 25 24				23 22 21 20 19 18 17 16				15 14 13 12 11 10 9 8				7 6 5 4 3 2 1 0			
PUPD15	PUPD14	PUPD13	PUPD12	PUPD11	PUPD10	PUPD9	PUPD8	PUPD7	PUPD6	PUPD5	PUPD4	PUPD3	PUPD2	PUPD1	PUPD0
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
2x+1 2x	PUPDx	Port n Pull-up/down Resistor Selection bits, x:0 to 15	
		00	Disable pull-up/down resistor
		01	Enable pull-up resistor
		10	Enable pull-down resistor
		11	Reserved

NOTES:

1. The pull-up resistors for the SWD pin (PB4: SWCLK, PB5: SWDIO) are initially enabled.
2. For exception, the reset value of PB_PUPD, PC_PUPD register is 0x00000540, 0x00000014 respectively.
3. The pull-up resistors for PC1 and PC2 are initially enabled.

5.3.6 Pn_INDR: port n input data register

Each pin level status can be read in the Pn_INDR register. Even if a pin is alternative mode except analog mode and output in alternative mode, the pin level can be detected in the Pn_INDR register.

Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_INDR=0x4000_1014, PB_INDR=0x4000_1114
PC_INDR=0x4000_1214, PD_INDR=0x4000_1314
PE_INDR=0x4000_1414, PF_INDR=0x4000_1514

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		INDR15	INDR14	INDR13	INDR12	INDR11	INDR10	INDR9	INDR8	INDR7	INDR6	INDR5	INDR4	INDR3	INDR2	INDR1	INDR0
-		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
-		RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

Bits	Name	Function
x	INDRx	Port n Input Data bit, x:0 to 15

5.3.7 Pn_OUTDR: port n output data register

Pn_OUTDR registers define output level of a pin when the pin is set as output and GPIO mode. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_OUTDR=0x4000_1018, PB_OUTDR=0x4000_1118
PC_OUTDR=0x4000_1218, PD_OUTDR=0x4000_1318
PE_OUTDR=0x4000_1418, PF_OUTDR=0x4000_1518

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		OUTDR15	OUTDR14	OUTDR13	OUTDR12	OUTDR11	OUTDR10	OUTDR9	OUTDR8	OUTDR7	OUTDR6	OUTDR5	OUTDR4	OUTDR3	OUTDR2	OUTDR1	OUTDR0
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
x	OUTDRx	Port n Output Data bit, x:0 to 15 The OUTDR bits can be individually set/cleared by writing to the Pn_BSR/Pn_BCR register

5.3.8 Pn_BSR: port n output bit set register

Pn_BSR registers control each bit of Pn_OUTDR register. Writing '1' into the specific bit field will set a corresponding bit of Pn_OUTDR to '1'. Writing '0' has no effect.

Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_BSR=0x4000_101C, PB_BSR=0x4000_111C
PC_BSR=0x4000_121C, PD_BSR=0x4000_131C
PE_BSR=0x4000_141C, PF_BSR=0x4000_151C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		BSR15	BSR14	BSR13	BSR12	BSR11	BSR10	BSR9	BSR8	BSR7	BSR6	BSR5	BSR4	BSR3	BSR2	BSR1	BSR0
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-		WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO

Bits	Name	Function	
x	BSRx	Port n Output Set bit, x: 0 to 15. These bits are always read to 0x00	
		0	No effect
		1	Set the corresponding OUTDRx bit (automatically cleared to 0)

5.3.9 Pn_BCR: port n output bit clear register

Pn_BCR registers control each bit of Pn_OUTDR register. Writing '1' into the specific bit field will set a corresponding bit of Pn_OUTDR to '0'. Writing '0' has no effect.

Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_BCR=0x4000_1020, PB_BCR=0x4000_1120
PC_BCR=0x4000_1220, PD_BCR=0x4000_1320
PE_BCR=0x4000_1420, PF_BCR=0x4000_1520

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		BCR15	BCR14	BCR13	BCR12	BCR11	BCR10	BCR9	BCR8	BCR7	BCR6	BCR5	BCR4	BCR3	BCR2	BCR1	BCR0
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-		WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO

Bits	Name	Function
x	BCRx	Port n Output Clear bit, x: 0 to 15. These bits are always read to 0x00
		0 No effect
		1 Clear the corresponding OUTDRx bit (automatically cleared to 0)

5.3.10 Pn_OUTDMSK: port n output data mask register

Pn_OUTDMSK registers protect each bit of Pn_OUTDR registers. Writing '1' into the specific bit field will protect a corresponding bit of Pn_OUTDR. Writing '0' is unmasked.

Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_OUTDMSK=0x4000_1024, PB_OUTDMSK=0x4000_1124
PC_OUTDMSK=0x4000_1224, PD_OUTDMSK=0x4000_1324
PD_OUTDMSK=0x4000_1324, PF_OUTDMSK=0x4000_1524

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		OUTDMSK15	OUTDMSK14	OUTDMSK13	OUTDMSK12	OUTDMSK11	OUTDMSK10	OUTDMSK9	OUTDMSK8	OUTDMSK7	OUTDMSK6	OUTDMSK5	OUTDMSK4	OUTDMSK3	OUTDMSK2	OUTDMSK1	OUTDMSK0
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
x	OUTDMSKx	Port n Output Data Mask bit, x: 0 to 15.
		0 Unmask. The corresponding OUTDR bit can be changed.
		1 Mask. The corresponding OUTDRx bit is protected.

5.3.11 Pn_DBCR: port n debounce control register

Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PA_DBCR=0x4000_1028, PB_DBCR=0x4000_1128
PC_DBCR=0x4000_1228, PD_DBCR=0x4000_1328
PE_DBCR=0x4000_1428, PF_DBCR=0x4000_1528

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		DBEN15	DBEN14	DBEN13	DBEN12	DBEN11	DBEN10	DBEN9	DBEN8	DBEN7	DBEN6	DBEN5	DBEN4	DBEN3	DBEN2	DBEN1	DBEN0
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
x	DBENx	Port n Debounce Enable bit, x: 0 to 15. Port debounce length = selected debounce clock period * (4 to 5)	
		0	Disable debounce filter
		1	Enable debounce filter

NOTES:

1. It needs to check clock source when using the Port n debounce function.
   The clock source is selected by SCU_MCCR4 register. (The selected clock should be enabled.)
   LSI, LSE, MCLK, HSI, HSE and PLL are selectable as clock source through this SCU_MCCR4 register.(Refer 4.5.31)
2. If a level is not detected on an enabled pin three or more times in a row at the sampling clock, the signal is eliminated as noise.
3. The port debounce should be disabled before Power Down mode.

5.3.12 Pn_IER: port n interrupt enable register

Each pin of A31G22x can be an external interrupt source. In this case, both of edge trigger interrupt and level trigger interrupt are supported. Pn_IER registers can configure the interrupt mode (n = A, B, C, D, E and F).

PA_IER=0x4000_102C, PB_IER=0x4000_112C
PC_IER=0x4000_122C, PD_IER=0x4000_132C
PE_IER=0x4000_142C, PF_IER=0x4000_152C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIE15		PIE14		PIE13		PIE12		PIE11		PIE10		PIE9		PIE8		PIE7		PIE6		PIE5		PIE4		PIE3		PIE2		PIE1		PIE0	
00		00		00		00		00		00		00		00		00		00		00		00		00		00		00		00	
RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW	

Bits	Name	Function
2x+1 2x	PIEx	Port n Pin interrupt Enable Selection bits, x:0 to 15
		00 Disable Interrupt
		01 Enable interrupt as level trigger mode
		10 Reserved
		11 Enable interrupt as edge trigger mode

5.3.13 Pn_ISR: port n interrupt status register

When an interrupt is delivered to CPU, the interrupt status can be detected by reading Pn_ISR registers. Pn_ISR registers will report a source pin of the interrupt and a type of the interrupt (n = A, B, C, D, E and F).

PA_ISR=0x4000_1030, PB_ISR=0x4000_1130
PC_ISR=0x4000_1230, PD_ISR=0x4000_1330
PE_ISR=0x4000_1430, PF_ISR=0x4000_1530

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIS15		PIS14		PIS13		PIS12		PIS11		PIS10		PIS9		PIS8		PIS7		PIS6		PIS5		PIS4		PIS3		PIS2		PIS1		PIS0	
00		00		00		00		00		00		00		00		00		00		00		00		00		00		00		00	
RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW	

Bits	Name	Function
2x+1 2x	PISx	Port n Pin interrupt Status bits, x:0 to 15
		00 No interrupt event
		01 Low level interrupt or falling edge interrupt event is present.
		10 High level interrupt or rsing edge interrupt event is present.
		11 Both of rising and falling interrupt event is present in edge trigger interrupt mode. Not available in level trigger interrupt mode.

5.3.14 Pn_ICR: port n interrupt control register

Pn_ICR registers control interrupt mode of port pins (n = A, B, C, D, E and F).

PA_ICR=0x4000_1034, PB_ICR=0x4000_1134
PC_ICR=0x4000_1234, PD_ICR=0x4000_1334
PD_ICR=0x4000_1434, PF_ICR=0x4000_1534

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16	15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
PIC15	PIC14	PIC13	PIC12	PIC11	PIC10	PIC9	PIC8	PIC7	PIC6	PIC5	PIC4	PIC3	PIC2	PIC1	PIC0
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
2x+1 2x	PICx	Port n Pin interrupt Control bits, x:0 to 15	
		00	Prohibit external interrupt
		01	Low level interrupt or falling edge interrupt mode
		10	High level interrupt or rising edge interrupt mode
		11	Both of rising and falling edge interrupt mode Not support for level trigger interrupt mode.

5.3.15 PF_PLSR: port F level select register

PF_PLSR register is Port F input level select register. PF5, PF6, PF7 are open-drain only pins. If user uses PF5, PF6, PF7 pins for 1.8V level instead VDD level, these bits must be set. Size of this register is 32-bit, and it is able to do 32/16/8-bit access (n = A, B, C, D, E and F).

PF_PLSR=0x4000_1538

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3	2	1	0
Reserved				PF7LSB	PF6LSB	PF5LSB
-				0	0	0
-				RW	RW	RW

Bits	Name	Function	
2	PF7LSB	PF7 input level select bit.	
		0	VDD level (Default)
		1	1.8V level
1	PF6LSB	PF6 input level select bit.	
		0	VDD level (Default)
		1	1.8V level
0	PF5LSB	PF5 input level select bit.	
		0	VDD level (Default)
		1	1.8V level

NOTE: PF5, PF6, PF7 pins are used open-drain only.

5.3.16 Pn_STR: port n strength configuration register

Pn_STR register can configure the drive strength of the PC[4:2] and PB[2:0] pin only of A31G22x. The drive strength configuration affects the speed of the SPI20 interface and the current consumption of the system.

PB_STR=0x4000_1140, PC_STR=0x4000_1240

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6	5 4	3 2	1 0
Reserved			PST4	PST3	PST2	PST1	PST0
-			00	00	00	00	00
-			RW	RW	RW	RW	RW

Bits	Name	Function	
2x+1 2x	PSTx	Pin Strength Configuration, x:0 to 4	
		00	Disabled pin strength
		01	Enabled pin strength
		10	Reserved
		11	Reserved

5.3.17 PCU1_SPI2PMR: SPI2n pin re-map register

PCU1_SPI2PMR register redirects the pin map to use the SPI interface on the SPI2n or SPI21 pins instead of the USART10_SPI10 or USART11_SPI11 channels on USART1n. If SPI2n pin map redirection of the SPI20 or SPI21 is enable, the USART10 or USART11 SPI interface will operate on the SPI20 or SPI21 channels and will not operate on the USART10 or USART11 channel. On the contrary, if SPI2n pin re-map of the SPI20 or SPI21 is disable, SPI interface will operate on USART10 or USART11 channel. The write access of PCU1_SPI2PMR register is affected by the setting value of PCU2_PORTEN.

PF_PLSR=0x4000_1544

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2	1	0
Reserved				SPI21_PRM	SPI20_PRM
-				0	0
-				RW	RW

Bits	Name	Function	
1	SPI21_PRM	Enable pin re-map of SPI21 channel	
		0	PE[5:2] (Default)
		1	PD[5:2]
0	SPI20_PRM	Enable pin Re-map of SPI20 channel	
		0	PC[4:2], PD[0] (Default)
		1	PB[3:0]

5.3.18 PCU2_ISEGPEN: ISEG LED port enable register

The PCU2_ISEGPEN register accepts key values to activate access to the PCU2_ISEGR or PCU2_ISEGIR registers, which can control high current output pots of A31G22x. When the '0x0702' key value is entered into the PCU2_ISEGPEN register, the setting value update of the PCU2_ISEGR or PCU2_ISEGIR register is activated.

PCU2_ISEGEN=0x4000_1F00

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		ISEGENKEY	
-		0x0000	
-		RW	

Bits	Name	Function	
15 0	ISEGENKEY	Write 0x0702 to the operation key register for operating each of the ISEG and the ISEGIR output ports.	
		0x0702	Enabled the access of high current output register.
		Others	Disabled the access of high current output register.

5.3.19 PCU2_ISEGR: ISEG LED port control register

The A31G22x ISEG[9:0] (PE [7: 2] and PD [5: 2]) pins are designed to enable high current output settings. This high current output function provides enough current to drive an LED or a system that requires tens of mA of current to the user application. This register bit is controlled by inverting the value set on each pin in the PCU_ISEGIR register.

PCU2_ISEGR=0x4000_1F04

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10	9	8	7	6	5	4	3	2	1	0
Reserved			ISEG9	ISEG8	ISEG7	ISEG6	ISEG5	ISEG4	ISEG3	ISEG2	ISEG1	ISEG0
-			1	1	1	1	1	1	1	1	1	1
-			RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
9	ISEG9	LED ISEG9 signal control of PD2 port	
		0	High current low output
		1	Floating
8	ISEG8	LED ISEG8 signal control of PD3 port	
		0	High current low output
		1	Floating
7	ISEG7	LED ISEG7 signal control of PD4 port	
		0	High current low output
		1	Floating
6	ISEG6	LED ISEG6 signal control of PD5 port	
		0	High current low output
		1	Floating
5	ISEG5	LED ISEG5 signal control of PE7 port	
		0	High current low output
		1	Floating
4	ISEG4	LED ISEG4 signal control of PE6 port	
		0	High current low output
		1	Floating
3	ISEG3	LED ISEG3 signal control of PE5 port	
		0	High current low output
		1	Floating
2	ISEG2	LED ISEG2 signal control of PE4 port	
		0	High current low output
		1	Floating
1	ISEG1	LED ISEG1 signal control of PE3 port	
		0	High current low output
		1	Floating

0	ISEG0	LED ISEG0 signal control of PE2 port	
		0	High current low output
		1	Floating

5.3.20 PCU2_ISEGIR: ISEG LED Inversion port control register

The A31G22x PE [7: 2] and PD [5: 2] pins are designed to enable high current output settings by inverting each pin value. This register is controlled by inverting the value set on each pin in the PCU2_ISEGR register

PCU2_ISEGIR=0x4000_1F10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10	9	8	7	6	5	4	3	2	1	0
Reserved			ISEG9	ISEG8	ISEG7	ISEG6	ISEG5	ISEG4	ISEG3	ISEG2	ISEG1	ISEG0
-			0	0	0	0	0	0	0	0	0	0
-			RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
9	ISEG9_INV	LED ISEG9 inverted signal control of PD2 port	
		0	Floating
		1	High current low output
8	ISEG8_INV	LED ISEG8 inverted signal control of PD3 port	
		0	Floating
		1	High current low output
7	ISEG7_INV	LED ISEG7 inverted signal control of PD4 port	
		0	Floating
		1	High current low output
6	ISEG6_INV	LED ISEG6 inverted signal control of PD5 port	
		0	Floating
		1	High current low output
5	ISEG5_INV	LED ISEG5 inverted signal control of PE7 port	
		0	Floating
		1	High current low output
4	ISEG4_INV	LED ISEG4 inverted signal control of PE6 port	
		0	Floating
		1	High current low output
3	ISEG3_INV	LED ISEG3 inverted signal control of PE5 port	
		0	Floating
		1	High current low output
2	ISEG2_INV	LED ISEG2 inverted signal control of PE4 port	
		0	Floating
		1	High current low output
1	ISEG1_INV	LED ISEG1 inverted signal control of PE3 port	
		0	Floating
		1	High current low output

0	ISEG0_INV	LED ISEG0 inverted signal control of PE2 port	
		0	Floating
		1	High current low output

5.3.21 PCU2_PORTEN: port access enable

PCU_PORTEN register enables the register writing permission of all PCU registers. Refer to Table 28 for control of each pin.

PCU_PORTEN=0x4000_1FF0

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			PORTEN
-			-
-			WO

7 0	PORTEN	Writing the sequence of 0x15 and 0x51 in this register enables writing to PCU registers, and writing other values protects all PCU registers from writing.

NOTE: Refer to the followings to use PORTEN. Refer to Table 28.

```
PORTEN=0x15;PORTEN=0x51;  // enable PORTEN
(Pn_MOD, Pn_TYP, Pn_AFSR1,2, Pn_PUPD. Pn_DBCR, Pn_IER, Pn_ICR, Pn_STR,
PCU1_SPI2PMR registers are enabled the write access)
PORTEN=0;                 // disable PORTEN
```

Table 28 The PCUx Registers Affected by the PORTEN Settings.

Register Name	Offset	Access Type	Write enable registers by PORTEN
Pn_MOD	0x00	RW	Affected.
Pn_TYP	0x04	RW	Affected.
Pn_AFSR1	0x08	RW	Affected.
Pn_AFSR2	0x0C	RW	Affected.
Pn_PUPD	0x10	RW	Affected.
Pn_INDR	0x14	RO	Not Affected.
Pn_OUTDR	0x18	RW	Not Affected.
Pn_BSR	0x1C	WO	Not Affected.
Pn_BCR	0x20	WO	Not Affected.
Pn_OUTDMSK	0x24	RW	Not Affected.
Pn_DBCR	0x28	RW	Affected.
Pn_IER	0x2C	RW	Affected.
Pn_ISR	0x30	RW	Affected.
Pn_ICR	0x34	RW	Affected.
PF_PLSR	0x38	RW	Not Affected.
Pn_STR	0x40	RW	Affected.
PCU1_SPI2PMR	0x1544	RW	Affected.
PCU2_ISEGKEY	0x1F00	RW	Not Affected.
PCU2_ISEGR	0x1F04	RW	Not Affected.
PCU2_ISEGIR	0x1F10	RW	Not Affected.

5.4 Functional description

If an input function of a certain I/O port is used by Pin Control Register, an output function of the I/O port is disabled. Function of each port can be different in accordance with an Alternative Function Selection Register. Input Data Register captures current data of the I/O pin or debounced input data at every GPIO clock cycle.

Figure 30. Port Diagram

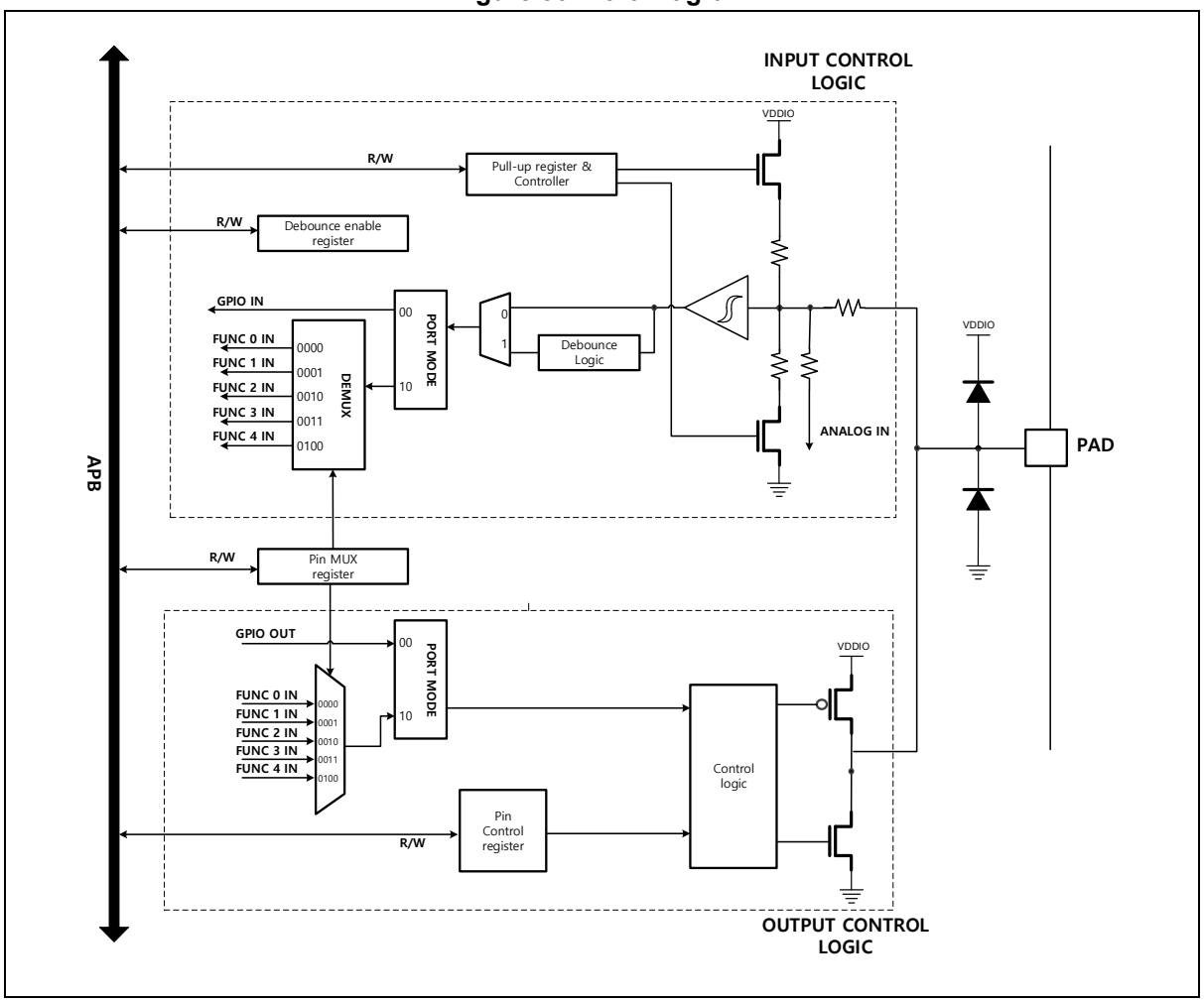
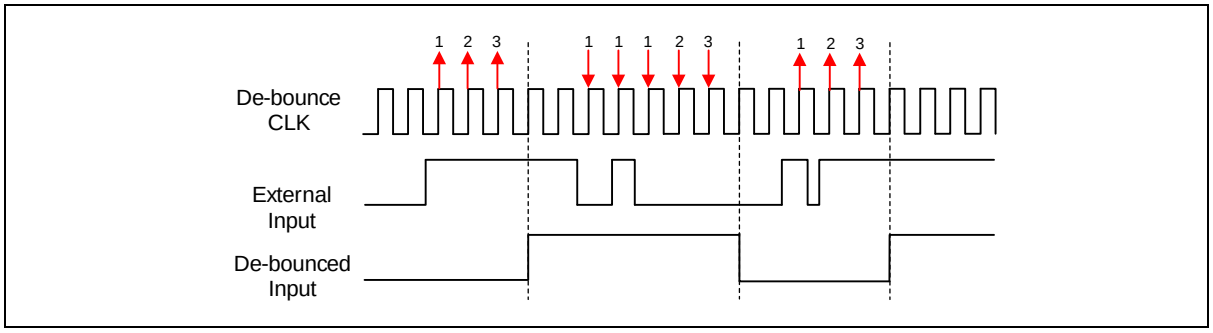
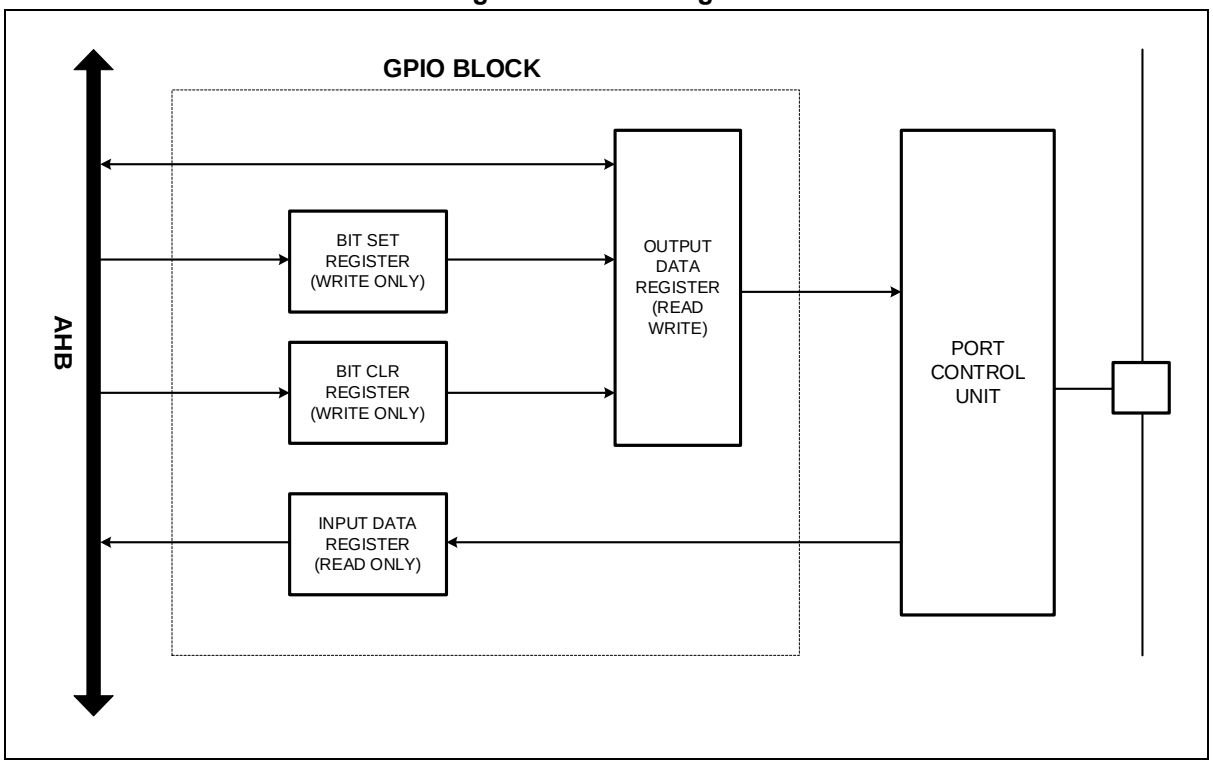


Figure 31. Debounce Function Timing Diagram



- When configured as output, the value written to the GPIO Output Data Register is output on the I/O Pin.
- When setting the Bit Set Register, GPIO Output Data Register set the high.
- When setting the Bit Clr Register, GPIO Output Data Register set the Low.
- The Input Data Register captures the data present on the I/O pin or Debounced input data at every GPIO Clock cycle.

Figure 32. GPIO Diagram



6 Code Flash memory controller (CFMC)

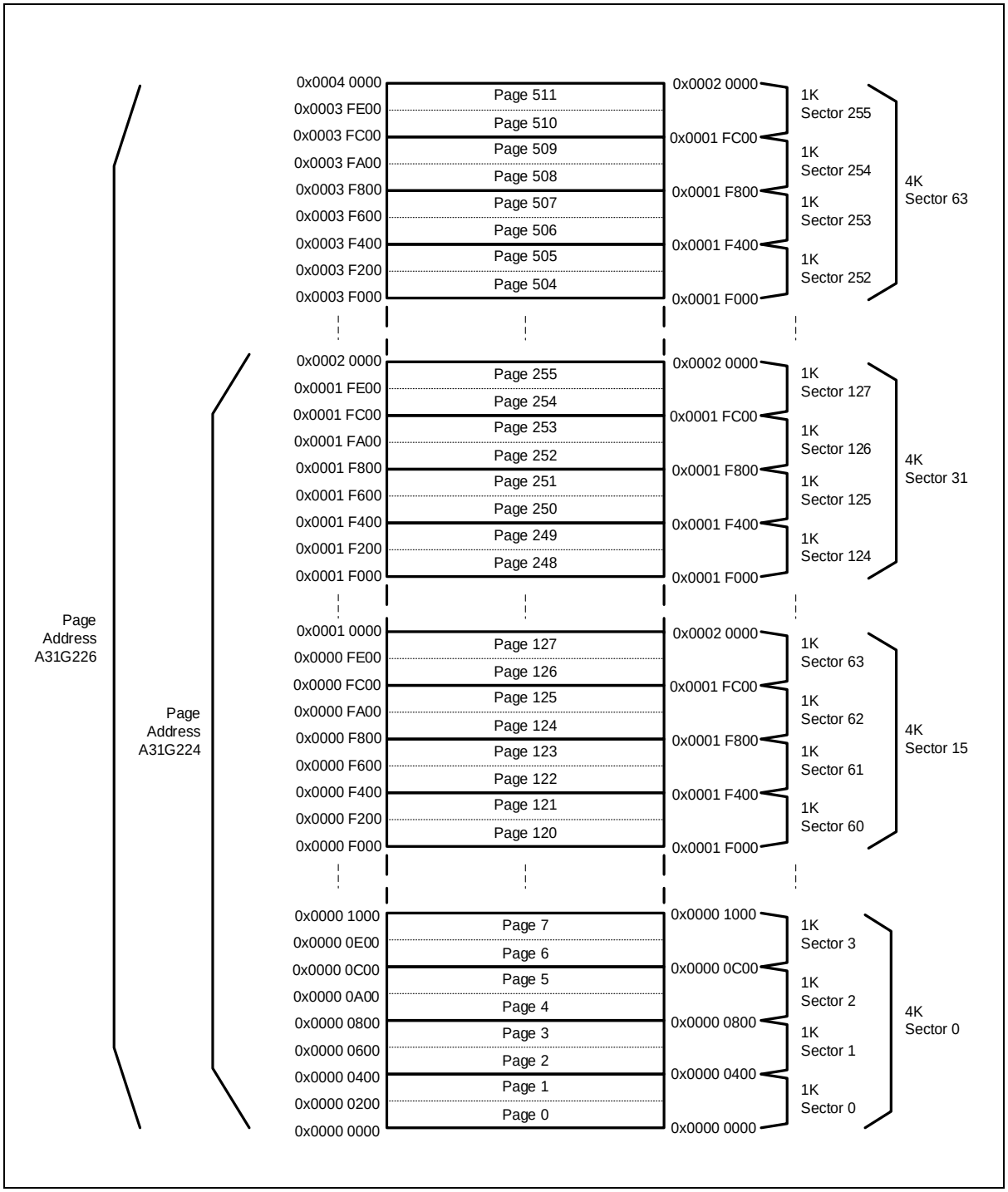
Code Flash Memory Controller is an internal flash memory interface controller, and includes following features as shown below:

- 256KB or 128KB code flash memory
- Wait, 0-wait to 6-wait (default : 5-wait)
- Read protection support
- Write protection support
  - Write protection with 16KB-unit
  - Write protection in 1KB-unit of the first region of the first region of each bank for the user boot loader
- Self-program and Self-erase on code flash memory
- Memory swap functions
- Endurance: 10,000 Cycles
- Data Retention: 10 Years

Table 29. Code Flash Memory Controller Features

Item	Description	
Size	256KB (A31G226)	128KB (A31G224)
Start Address	0x0000_0000	0x0000_0000
End Address	0x0003_FFFF	0x0001_FFFF
Page Size	512-byte	512-byte
Total Page Count	512 pages	256 pages
PGM Unit	1-word (4-byte) Self-PGM	1-word (4-byte) Self-PGM
Erase Unit	512-byte/ 1KB/ 4KB/ bulk	512-byte/ 1KB/ 4KB/ bulk

Figure 33. Code Flash Memory Map (A31G22x Code Flash)



6.1 Registers

Base address of code flash memory controller is introduced in the followings:

Table 30. Base Address of Flash Memory Controller

Name	Base address
Code Flash controller	0x4000_0100

Table 31. CFMC Register Map

Name	Offset	Type	Description	Reset value	Ref.
CFMC_MR	0x0004	R/W	Code Flash Memory Mode Select Register	0x0100_0000	6.1.1
CFMC_CR	0x0008	R/W	Code Flash Memory Control Register	0x0000_0000	6.1.2
CFMC_AR	0x000C	R/W	Code Flash Memory Address Register	0x0000_0000	6.1.3
CFMC_DR	0x0010	R/W	Code Flash Memory Data Register	0x0000_0000	6.1.4
CFMC_BUSY	0x0018	R/O	Code Flash Memory Write Busy Status Register	0x0000_0000	6.1.5
CFMC_CRC	0x0020	R/W	Code Flash Memory CRC-CCITT check Register	0x0000_FFFF	6.1.6
CFMC_CFG	0x0030	R/W	Code Flash Memory Configuration Register	0x0000_8500	6.1.7
CFMC_WPROT	0x0034	R/W	Code Flash Memory Write Protection Register	0xFFFF_FFFF	6.1.8
CFMC_RPROT	0x003C	R/W	Code Flash Memory Read Protection Register	0x0000_00FF	6.1.9
CFMC_PWIN	0x0040	WO	Code Flash Memory Password Input Register	-	6.1.10
CFMC_PWPRST	0x0044	WO	Code Flash Password Preset Register	-	6.1.11
CFMC_BCR	0x0048	R/W	Code Flash Bank Control Register	0x0000_0000	6.1.12
CFMC_BSR	0x004C	R/W	Code Flash Bank Status Register	0x0000_0000	6.1.13
CFMC_ABWPROT	0x0050	R/W	Code Flash Memory Active Bootloader Area Write Protection Register	0x0000_FFFF	6.1.14
CFMC_NBWPROT	0x0054	R/W	Code Flash Memory Non-active Bootloader Area Write Protection Register	0x0000_FFFF	6.1.15

6.1.1 CFMC_MR: code flash memory mode register

CFMC_MR is an internal flash memory mode register. Size of this register is 32-bit.

CFMC_MR=0x4000_0104

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			ACODE
-			0x00
-			RW

Bits	Name	Function	Bits
7 0	ACODE	Code flash entry code	
		0x5A to 0xA5	Flash mode entry
		0xA5 to 0x5A	Trim mode entry
		0x81 to 0x28	AMBA mode entry
		0x66 to 0x99	PROT mode entry

6.1.2 CFMC_CR: code flash memory control register

CFMC_CR is an internal flash memory control register.

CFMC_CR=0x4000_0108

31 30 29 28	27 26	25	24	23	22 21 20 19 18 17 16	15	14 13 12	11 10 9	8	7	6	5	4	3	2	1	0
BBLOCK WTIDKY	Reserved	BSAEN	RPAEN	SELFPGM	Reserved	IFEN		Reserved	BBLOCK	MAS	SECT4K	SECT1K	PMODE	WADCK	PGM	ERS	HVEN
0000	-	0	0	0	-	0		-	0	0	0	0	0	0	0	0	0
RW	-	RW	RW	RW	-	RW		-	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
31 28	BBLOCK WTIDKY	Write protection key value '0xA' for the BBLOCK(Boot Block lock)	
		0xA	Write protection key for the BBLOCK(Boot Block lock)
25	BSAEN	Bank Selection block access enable	
		0	Disable bank selection block access
		1	Enable bank selection block access
24	RPAEN	Read Protection block access enable.	
		0	Disable read protection block access
		1	Enable read protection block access
23	SELFPGM	When this bit is set ("1"), PGM/ERS/HVEN will be cleared automatically after WRBUSY falling edge. It also enable CPU wait control when HVEN bit is set(1) (start of program or erase operation). It also affects to PMODE bit operation.	
12	IFEN	Enable the access of user option 0/1/2 block in code flash access	
		0	Disable
		1	Enable the access of user option 0/1/2 block (It works with OTP3EN to OTP1EN for PMODE operation).
8	BBLOCK	Whether to enable or disable Boot Block lock (0x0000 – 0xFFFF) This setting applies to Mass(bulk) erase only. The locked area can be erased by other erase commands (SECT1 K, 4 K and page erases).) **NOTE:** To update the AB_WPROT value, the AB_KEY value must be written together.	
		0	Boot Block (1st 4KB) not protected from Mass(Bulk) Erase
		1	Boot Block (1st 4KB) protection enable from Mass(bulk) erase
7	MAS	Enable mass erase (bulk) – All data of code flash area is erased.	
		0	Mass (bulk) erase disable
		1	Mass (bulk) erase enable.
6	SECT4K	4KB-units sector erase	
		0	Sector 4K erase disable
		1	Sector 4K erase enable
5	SECT1K	1KB-units sector erase	
		0	Sector 1K erase disable
		1	Sector 1K erase enable

4	PMODE	The type of program mode of code flash (Flash Address path is connected with FMAR)	
		0	Normal mode
		1	PMODE enable **NOTE**: PMODE only valid when SELFPGM bit = 0
3	WADCK	Enable Program/Erase address data latch clock	
		0	Program/Erase address data latch clock disable
		1	Program/Erase address data latch clock enable, this bit assert for one system clock period so user cannot read
2	PGM	Enable program mode of code flash memory	
		0	Program mode disable
		1	Program mode enable
1	ERS	Enable erase mode of code flash	
		0	Erase mode disable
		1	Erase mode enable
0	HVEN	Enable High voltage cycle	
		0	High voltage cycle disable
		1	High voltage cycle enable (start program or erase cycle) **NOTES**: User must set and clear this bit when PMODE=1. In SELFPGM mode, user must set HVEN then HVEN will be cleared automatically after WRBUSY goes low.

6.1.3 CFMC_AR: code Flash memory address register

The CFMC_AR register designates the word-based (32-bit) memory address of the code flash memory to be programmed or erased.

CFMC_DR=0x4000_010C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		FADDR	
-		0x0000	
-		RW	

Bits	Name	Function
15 0	FADDR	Word (32-bit) base address: 64K-word address for 256KB Flash. Auto Incremental after WADCK trigger (after latching of target address). **NOTE:** If an address is written to CFMC_AR register, the address value must be a 2-bit left shift value.

6.1.4 CFMC_DR: code flash memory data input register

The CFMC_DR register gets the word-sized (32-bit) input data to be programmed or erased to code flash memory.

CFMC_DR=0x4000_0110

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
FDATA			
0x00000000			
RW			

Bits	Name	Function
31 0	FDATA	Word size(32-bit)

6.1.5 CFMC_BUSY: code flash memory write busy status register

CFMC_BUSY is a flash write (program/erase) busy status monitor register. This register is a 1-bit read only register.

CFMC_BUSY=0x4000_0118

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1	0
Reserved	WRBUSY
-	0
-	RO

Bits	Name	Function
0	WRBUSY	Write busy status bit FLBUSY bit goes high after set HVEN bit (in CTRL register). FLBUSY bit goes low when WRBUSY becomes low after program (or erase) complete.

6.1.6 CFMC_CRC: code flash memory CRC check register

CFMC_CRC represents the 16-bit CRC-CCITT calculated value of all data in the code flash memory area. CFMC_CRC is read only register which enabled by CRCEN bit of CFMC_CFG register. To obtain CRC-CCITT calculation results for data in the code flash memory area, perform at least 16 times (16-word) of CFMC_CRC register read access.

CFMC_CRC=0x4000_0120

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CRC-CCITT	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	CRC- CCITT	CRC- CCITT result register polynomial: $(1 + x^5 + x^{12} + x^{16})$ At least 16-word read to get a CRC value.

6.1.7 CFMC_CFG: code flash memory configuration register

The CFMC_CFG register supports access timing configuration and the CRC-CCITT operation of the code flash memory area. The input identification key is required in the CFMC_CFG register to write the setting value.

CFMC_CFG=0x4000_0130

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5 4 3 2 1 0
WTIDKY		Reserved	WAIT	CRCINIT	CRCEN	Reserved
0x7858		-	101	0	0	-
WO		-	RW	RW	RW	-

Bits	Name	Function	
31 16	WTIDKY		Write identification key. On writes, write 0x7858 to these bits, otherwise the write is ignored.
10 8	WAIT		This bits only be written in AMBA mode and MSB 16-bit (bit [31:16]) must be 0x7858
		000	WAIT is 000, flash access in 1 cycle (0-wait)
		001	WAIT is 001, flash access in 2 cycles (1-wait)
		010	WAIT is 010, flash access in 3 cycles (2-wait) – default
		011	WAIT is 011, flash access in 4 cycles (3-wait)
		100	WAIT is 100, flash access in 5-cycles (4-wait)
		101 1xx	WAIT is 101, 11x flash access in 6-cycles (5-wait) – default
7	CRCINIT	0	When this bit is set('1'), CRC register will be initialized It should be reset again before read flash to generate CRC-CCITT calculation (Initial value of CFMC_CRC is 0xFFFF)
6	CRCEN	0	CRC-CCITT enable CRC value will be calculated at every flash read timing

NOTES:

1. In order to operate user application with optimized performance, the proper access time of the code flash memory is required. Refer to Table 32 for the settings of wait-time for the code flash memory of A31G22x.
2. When read access to user option area, wait time needs to be adjusted based on min. 100ns. It is recommend to change the flash bus speed to 5-wait (6 cycles) accessing the user option area.

Table 32. The wait-time setting table of A31G22x code flash memory

APB speed (HCLK)	Wait-time	Access freq.	Access time	Recommended access time of code flash area	Recommended access time of user option area
48MHz	5 (6-cycles)	8.0MHz	125.0ns	50ns	100ns
48MHz	4 (5-cycles)	9.6MHz	104.1ns	50ns	100ns
48MHz	3 (4-cycles)	12.0MHz	83.0ns	50ns	100ns
48MHz	2 (3-cycles)	16.0MHz	62.5ns	50ns	100ns
48MHz	1 (2-cycles)	24.0MHz	41.6ns	50ns	100ns
48MHz	0 (1-cycles)	48.0MHz	20.8ns	50ns	100ns
32MHz	5 (6-cycles)	5.3MHz	187.0ns	50ns	100ns
32MHz	4 (5-cycles)	6.4MHz	156.0ns	50ns	100ns
32MHz	3 (4-cycles)	8.0MHz	125.0ns	50ns	100ns
32MHz	2 (3-cycles)	10.6MHz	93.7ns	50ns	100ns
32MHz	1 (2-cycles)	16.0MHz	62.5ns	50ns	100ns
32MHz	0 (1-cycles)	32.0MHz	31.2ns	50ns	100ns
20MHz	5 (6-cycles)	3.3MHz	300.0ns	50ns	100ns
20MHz	4 (5-cycles)	4.0MHz	250.0ns	50ns	100ns
20MHz	3 (4-cycles)	5.0MHz	200.0ns	50ns	100ns
20MHz	2 (3-cycles)	6.7MHz	150.0ns	50ns	100ns
20MHz	1 (2-cycles)	10.0MHz	100.0ns	50ns	100ns
20MHz	0 (1-cycles)	20.0MHz	50.0ns	50ns	100ns
500KHz	5 (6-cycles)	83.3KHz	12.0us	50ns	100ns
500KHz	4 (5-cycles)	100.0KHz	10.0us	50ns	100ns
500KHz	3 (4-cycles)	125.0KHz	8.0us	50ns	100ns
500KHz	2 (3-cycles)	166.6KHz	6.0us	50ns	100ns
500KHz	1 (2-cycles)	250KHz	4.0us	50ns	100ns
500KHz	0 (1-cycles)	500KHz	2.0us	50ns	100ns

6.1.8 CFMC_WPROT: code flash memory write protection register

CFMC_WPROT register supports write protection function for internal code flash memory. Each bit of the CFMC_WPROT register controls the write protection of a 16KB sized segment which comprises the memory.

Since the write protection for the internal code flash memory is enabled at system initialization by default, a user needs to disable the write protection of the corresponding segment before erasing or programming on a certain area of the code flash memory.

When using memory bank swap, write protection coverage changes depending on the swap status, so you need to adjust the WPROT value. For more information on this, see 6.2.5.

CFMC_WPROT=0x4000_0134

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	WPAS	WPROT (A31G226)	
-	0000	0xFFFF	
-	RW	RW	
Reserved	WPAS	Reserved	WPROT (A31G224)
-	0000	-	0xFF
-	RW	-	RW

[A31G226]

Bits	Name	Function
19 16	WPAS	Protection area selection (Reserved) 000 Reserved (Default)
15 0	WPROT	Write protection each 16 KB segments of code flash for whole memory address **NOTES:** 1. Protection Range = WPROT[n] : (n*16kB) ~ (16kB * (n+1) - 1) 2. The FM_WPROT register can only be modified in PROT (FM_MR = 66-> 99) mode.

[A31G224]

Bits	Name	Function
19 16	WPAS	Protection area selection 000 Reserved (Default)
8 0	WPROT	Write protection each 16 KB segments of code flash for whole memory address **NOTES:** 1. Protection Range = WPROT[n] : (n*16kB) ~ (16kB * (n+1) - 1) 2. The FM_WPROT register can only be modified in PROT (FM_MR = 66-> 99) mode.

6.1.9 CFMC_RPORT: code flash memory read protection register

CFMC_RPORT register provides read protection function for internal code flash memory. When the read protection is enabled, the read protection is applied to the whole area of the code flash memory. Access operations to the flash and debugging will be limited depending on the protection level.

The read protection can be disabled by using one of the following two process:

1. When a user enters a password in FMC_PWIN register which matches to the predefined value, the read protection is disabled. This is available when both the write protection and pre-set function of FMC_PWPRST register are applied.
2. If a user erases whole code flash memory with the user option area for read protection by using chip (mass) erase, the read protection can be disabled.

CFMC_RPROT=0x4000_013C

31	30	29 28 27	26	25	24	23 22 21 20 19 18	17	16	15 14 13 12 11 10	9	8	7 6 5 4 3 2 1 0
DBGMOD	SRBOOT	Reserved	PWMATCH	RPBERSD	CERSD	Reserved	LVL2_STS	LVL1_STS	Reserved	LVL2_EN	LVL1_EN	RPROT
0	0	-	0	0	0	-	0	0	-	0	0	0x0000_00FF
RO	RO	-	RO	RO	RO	-	RO	RO	-	RO	RO	RW

Bits	Name	Function	
31	DBGMOD	Debug operating status bit	
		0	Not operating
		1	Operating
30	SRBOOT	SRAM boot mode status bit	
		0	Normal mode
		1	SRAM boot mode
26	PWMATCH	Password match flag bit	
		0	Not matched
		1	Password preset value and user password input value are matched.
25	RPBERSD	Chip erase and read protection block erase done flag bit	
		0	Not occurred
		1	Chip erase and read protection block erase done
24	CERSD	Chip erase done flag bit	
		0	Not occurred
		1	Chip erase done
17	LVL2_STS	Read protection level-2 status bit (raw data)	
		0	Normal status
		1	Protection level-2 status
16	LVL1_STS	Read protection level-1 status bit (raw data)	
		0	Normal status

		1	Protection level-1 status
9	LVL2_EN	Read protection level-2 enable status bit	
		0	Disable
		1	Enable
8	LVL1_EN	Read protection level-1 enable status bit	
		0	Disable
		1	Enable
7 0	RPROT	Read Protection Control bit	
		0xFF	Unprotection
		0x39	Read protection level-1 The data of the code flash is read as '0x55AA55AA' Debugger is connected.
		0xB9	Read protection level-1 with password
		0x0b'7	Read Protection Level-2 Any other value with bit-7 is '0'. The data of the code flash is read as '0x55AA55AA' Debugger is NOT connected.
		0x1b'7	Read Protection Level-2 with Password Any other value with bit-7 is '1'. The data of the code flash is read as '0x55AA55AA' Debugger is NOT connected.

NOTES:

1. When the memory is in either read protection mode, setting the 8th bit to 1 enables the password function for the read protection mode.

   Ex 1) 0x80 : Read protection Level-2 with password mode

   Ex 2) 0x00 : Read protection Level-2 mode
2. The CFMC_RPROT register can only be modified in PROT (CFMC_MR = 66-> 99) mode.

6.1.10 CFMC_PWIN: code flash memory password input register

A user enters a password in CFMC_PWIN register to release the password function, if FMC_PWPRST register has a pre-set password. If FMC_ PWPRST register has not a pre-set password, the value entered in this register will be ignored.

If FMC_RPROT register enabled both flash read protection and password, a user must enter a password in CFMC_PWIN, which was predefined in CFMC_PWPRST register, to generate a PWMATCH flag and to access the flash.

After finishing user operation on the code flash memory, the user enters the password in CFMC_PWIN register again to recover the read protection.

If the user enters different password to the pre-set value in CFMC_PWPRST register, the valued entered in this register will be ignored. To release the password, the user must reset system.

CFMC_PWIN=0x4000_0140

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
PWIN			
-			
WO			

Bits	Name	Function
31 0	PWIN	Password input data bit Writing is done twice to the register in the order from LSB to MSB. Rewriting is restricted once the register is written. Ex) Entering 0x1234_5678 as a password. CFMC_PWIN = 0x1234_5678; CFMC_PWIN = 0x1234_5678;

NOTE: CFMC_PWIN inaccessible after two write access.

6.1.11 CFMC_PWPRST: code flash memory password preset register

CFMC_PWPRST is a password pre-set register which is used to define a system password. Password is set as a system password if it is entered in this register twice. If a password is set while processing user code, it is initialized by system reset. To maintain the pre-set password even after the system initialization, a user must define the pre-set value in User option area at the system initialization stage by using User Bootloader.

CFMC_PWPRST register supports to release the read protection temporarily. If the flash read protection is applied and FMC_PWPRST register has a pre-set password, a user can release the read protection by entering a password in FMC_PWIN.

In the same condition, if a system reset occurs, the read protection is recovered.

CFMC_PWPRST=0x4000_0144

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
PWPRST			
-			
WO			

Bits	Name	Function
31 0	PWPRST	Password preset data bit Writing is done twice to the register in the order from LSB to MSB. Rewriting is restricted once the register is written. Ex) Presetting 0x1234_5678 as a password. CFMC_PWPRST = 0x1234_5678; CFMC_PWPRST = 0x1234_5678;

NOTES:

1. CFMC_PWPRST inaccessible after two write access.
2. CFMC_PWPRST register cannot write access in SRAM boot or Debug boot mode.

6.1.12 CFMC_BCR: code flash memory bank control register

CFMC_BCR register is used to control Banks. To utilize Bank SWAP function, a user needs to use Active Bank and Non-active Bank by enabling BSE function first. Then BSWC bit needs to be set.

To change values in this register, WTIDKY key must be set together. This register CFMC_BCR must be accessed from BootROM area. Access from other area than the BootROM generates CFMC_BSR<BSERR> error flag and maintains the previous state.

CFMC_BCR=0x4000_0148

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5	4	3 2 1	0
WTIDKY		Reserved		BSE	Reserved	BSW
0x0000		-		0	-	0
WO		-		RW	-	RW

Bits	Name	Function
31 16	WTIDKY	Write identification key Write WTIDKY in this field to set the bank control register bits. If a value other than WTIDKY is entered, it is ignored.
		0xAA91 Write identification key to set the bank control register.
4	BSE	Memory Bank Selection Enable bit for swap operation. When this bit is set to 1, Code flash is used as active and non-active banks of consecutive memory addresses. **NOTE:** This bit should be set with WTIDKY
		0 Disable memory bank control bit to swap [A31G226] Active Bank = 256KB (0x0000_0000~0x0003_FFFF) [A31G224] Active Bank = 128KB (0x0000_0000~0x0001_FFFF)
		1 Enable memory bank control bit to swap [A31G226] Active Bank = 128KB (0x0000_0000~0x0001_FFFF) Non-active Bank = 128KB (0x0002_0000~0x0003_FFFF) [A31G224] Active Bank = 64KB (0x0000_0000~0x0000_FFFF) Non-active Bank = 64KB (0x0001_0000~0x0001_FFFF)
0	BSW	Bank Swap (Switching active and non-active bank) **NOTE:** This bit must be set with WTIDKY and the BSE bit set to '1'
		0 The logical address '0x0000_0000' starts from physical address 0x0000_0000.
		1 [A31G226] The logical address '0x0000_0000' starts from physical address 0x0002_0000. [A31G224] The logical address '0x0000_0000' starts from physical address 0x0001_0000.

6.1.13 CFMC_BSR: code flash memory bank status register

CFMC_BSR enables to generate a flag which is used to check status of memory bank operation. By checking the flag provided by this register, a user can recognize specific memory area to execute code at system initialization stage. This register CFMC_BSR must be accessed from BootROM area. Access from other area than the BootROM generates CFMC_BSR<BSERR> error flag and maintains the previous state.

CFMC_BSR=0x4000_014C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12	11 10 9	8	7 6 5	4	3 2 1	0
Reserved		BSERR	Reserved	CBF	Reserved	BSST	Reserved	BSWST
-		0	-	0	-	0	-	0
-		RW	-	RW	-	RO	-	RO

Bits	Name	Function
12	BSERR	Error handling flag bit for bank selection If the bank selection malfunctions, BSERR bit is set to "1". To clear the error flag bit, Write "1" to this bit. **NOTES:** 1. The bank control register must be controlled at SRAM or BootROM. Otherwise, the SWAPERR flag is raised. 2. If CFMC_BCR<BSE> bit is set to '1' and error occurs during SWAP operation, BSERR flag is generated and the previous state is maintained. 3. If Swap operation error occurs (BSERR = 1), a user must enable SWAPFAILEN bit of SCU_NMIR register to process corresponding interrupts.
		0 NOT occurred Bank Selection Error
		1 Occurred Bank Selection Error
8	CBF	Current boot bank flag bit **NOTE:** This bit is used to set or release a state flag of Bank Swap operation. This bit can be used in the form of OTP Command, and is set to '0' if the flag is not used.
		0 Current boot area is bank 0
		1 Current boot area is bank 1
4	BSST	Bank selection status bit **NOTE:** BSST bit is activated when BSWST bit is enabled that indicates current active area is bank 1.
		0 Disabled memory bank selection [A31G226] Active bank = 256KB (0x0000_0000~0x0003_FFFF) [A31G224] Active bank = 128KB (0x0000_0000~0x0001_FFFF)
		1 Enabled memory bank selection [A31G226] Active bank = 128KB (0x0000_0000~0x0001_FFFF) Non-active bank = 128KB (0x0002_0000~0x0003_FFFF) [A31G224] Active bank = 64KB (0x0000_0000~0x0000_FFFF)

			Non-active bank = 64KB (0x0001_0000~0x0001_FFFF)
0	BSWST	Bank swap status bit	
		0	NOT occurred bank swap
		1	Occurred bank swap

6.1.14 CFMC_ABWPROT: code flash memory active bootloader write protection register

CFMC_ABWPROT register provides write protection function which is applied to 16KB User Bootloader of Active Bank. Each bit of this register enables or disables the write protection on corresponding 1KB segment.

The write protection to the whole code flash memory is enabled at system initialization stage. Before updating data by erasing or programming a certain code flash area, the write protection to the corresponding segment must be released.

CFMC_ABWPROT=0x4000_0150

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
AB_KEY	AB_WPAS	AB_WPROT	
0xAAB	0000	0xFFFF	
WO	RW	RW	

[A31G226]

Bits	Name	Function
31 20	AB_KEY	Write protection key value '0xAAB' for the active bank bootloader area. **NOTE:** The FMC_ABWPROT register can only be modified in PROT mode. (FM_MR = 66-> 99)
		0xAAB Write protection key for the active bank bootloader area
19 16	WPAS	Protection area selection
		0000 Reserved (Default)
15 0	AB_WPROT	Write protection each 1KB segments for whole memory address **NOTE:** To update the AB_WPROT value, the AB_KEY value must be written together.

6.1.15 CFMC_NBWPROT: code flash memory non-active bootloader write protection register

CFMC_NBWPROT register provides write protection function which is applied to 16KB User Bootloader of Not-active Bank. Each bit of this register enables or disables the write protection on corresponding 1KB segment.

The write protection to the whole code flash memory is enabled at system initialization stage. Before updating data by erasing or programming a certain code flash area, the write protection to the corresponding segment must be released.

CFMC_NBWPROT=0x4000_0154

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
NB_KEY	NB_WPAS	NB_WPROT	
0x55B	0000	0xFFFF	
WO	RW	RW	

[A31G226]

Bits	Name	Function
31 20	NB_KEY	Write protection key value '0x55B' for the non-active bank bootloader area. **NOTE:** The FMC_NBWPROT register can only be modified in PROT mode. (FM_MR = 66-> 99) 0xAAB Write protection key for the non-active bank bootloader area
19 16	WPAS	Protection area selection 0000 Reserved (Default)
15 0	NB_WPROT	Write protection each 1KB segments for whole memory address **NOTE:** To update the NB_WPROT value, the NB_KEY value must be written together.

6.1.16 Code flash memory swap flag store register

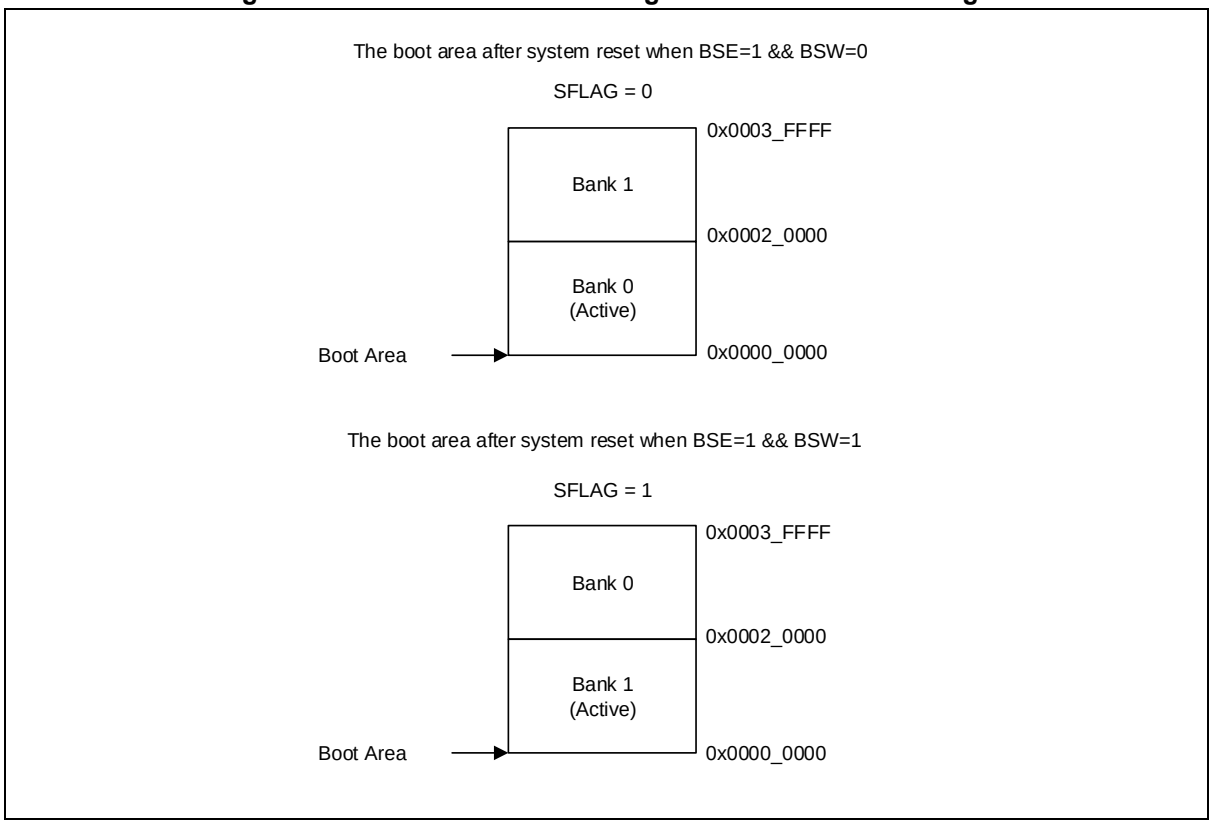
This is a nonvolatile register. To utilize Bank SWAP function, a user needs to set a bit of CFMC_BCR<BSE> to be active, and a SWAP flag is stored in this register in accordance with a value of CFMC_BCR<BSW>. In BOOTROM mode, corresponding bit is checked and BCR is updated. In BOOTROM mode, Bank to be processed is selected by SFLAG value. If SFLAG = 0, then program of Bank0 is executed. If SFLAG = 1, program of the Bank1 is executed.

CFMC_SFSR=0x0F01_0800

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			SFLAG
-			-
-			RW

Bits	Name	Function	
0	SFLAG	Swap status flag	
		0	Boot from bank 0 in BOOTROM mode
		1	Boot from bank 1 in BOOTROM mode

Figure 34 The Boot Area according to the SFLAG Bit Settings



6.2 Functional description

6.2.1 Code Flash erase and program examples

The basic step of Flash memory program consists with following three steps. Minimum Program or Erase unit is a Page and 128-word (512-byte) become a page.

- Page erase
- Page program
- Self page erase
- Self page program

For all of erase operations, pre-program operation is required to prevent over erase of flash memory cells.

Erase example

1. Enable flash mode to write CFMC_MR register (write 0x5A and then write 0xA5 into CFMC_MR).
2. Set target Page address in CFMC_AR.
3. Set PMODE bit first.
4. Set ERS, WADCK, HVEN bits of CFMC_CR.
5. Wait until IDLE bit of CFMC_BUSY register becomes "0" after erase.
6. Clear ERS, HVEN bits of CFMC_CR.
7. Set 0x80 to CFMC_BUSY.
8. Clear CFMC_CR.
9. Clear Flash mode (write 0x00 and then write 0x00 into CFMC_MR).

Program example

1. Enable flash mode to write CFMC_CR register (write 0x5A and then write 0xA5 into CFMC_MR).
2. Set PMODE bit first.
3. Set target Page address in CFMC_AR.
4. Set PGM bits of CFMC_CR.

5. Write word (32-bit) data into CFMC_DR, address increased automatically based on word address.
6. Set WADCK, HVEN bits of CFMC_CR.
7. Wait until IDLE bit of CFMC_BUSY register becomes "0" after program.
8. Clear HVEN bits of CFMC_CR.
9. Set 0x80 to CFMC_ BUSY.
10. Clear PGM bits of CFMC_CR.
11. Clear CFMC_CR.
12. Clear Flash mode (write 0x00 and then write 0x00 into CFMC_MR).

Self-Erase example

1. Enable flash mode to write CFMC_CR register (write 0x5A and then write 0xA5 into CFMC_MR).
2. Set SELFPGM, ERS bits of CFMC_CR.
3. Wait 5 clocks.
4. Write "0xFFFFFFFF" into target address. Target address must be aligned by 512 Bytes address space
5. Wait 5 clocks.
6. Clear ERS bits of CFMC_CR.
7. Clear CFMC_CR.
8. Wait 5 clocks.
9. Clear flash mode (write 0x00 and then write 0x00 into CFMC_MR).

Self-Program example

1. Enable flash mode to write CFMC_CR register (write 0x5A and then write 0xA5 into CFMC_MR).
2. Set SELFPGM, PGM bits of CFMC_CR.
3. Wait 5 clocks.
4. Write word (32-bit) data into target address. Address must be aligned by WORD (32-bit) address space. To write another word, go to step 4.
5. Wait 5 clocks.
6. Clear PGM bits of CFMC_CR.
7. Clear CFMC_CR.
8. Wait 5 clocks.
9. Clear flash mode (write 0x00 and then write 0x00 into CFMC_MR).

6.2.2 Write protection Summary

The protected areas of WPROT, ABWPROT, and NBWPROT are different according to bank swap. Refer to Table 34.

Table 33. Bank Swap Selection Enable

Case	BSE (CFMC_BCR[4])	BCW (CFMC_BCR[0])	WPROT	ABWPROT	NBWPROT
1	0b	-	O	O	X
2	1b	0b	O	O	O
3	1b	1b	O	O	O

Case 1) If bank selection(BSE) is disabled, WPROT and ABWPROT used, but NBWPROT don't used.

Case 2) If Bank selection is Enable and Active Bank (logical address 0x0000_0000) is physical address 0x0000_0000, WPROT, ABWPROT, and NBWPROT used.

Case 3) If Bank selection is Enable and Active Bank (logical address 0x0000_0000) is physical address 0x0002_0000, ABWPROT and NBWPROT follow the logical address, but for WPROT it follows the physical address.

Ex) If CFMC_WPROT is 0x0000_0100, WPROT physical address is 0x0002_0000 ~ 0x0002_4000, so write protection is available to logical address 0x0000_0000 ~ 0x0000_4000. Refer to Figure 36

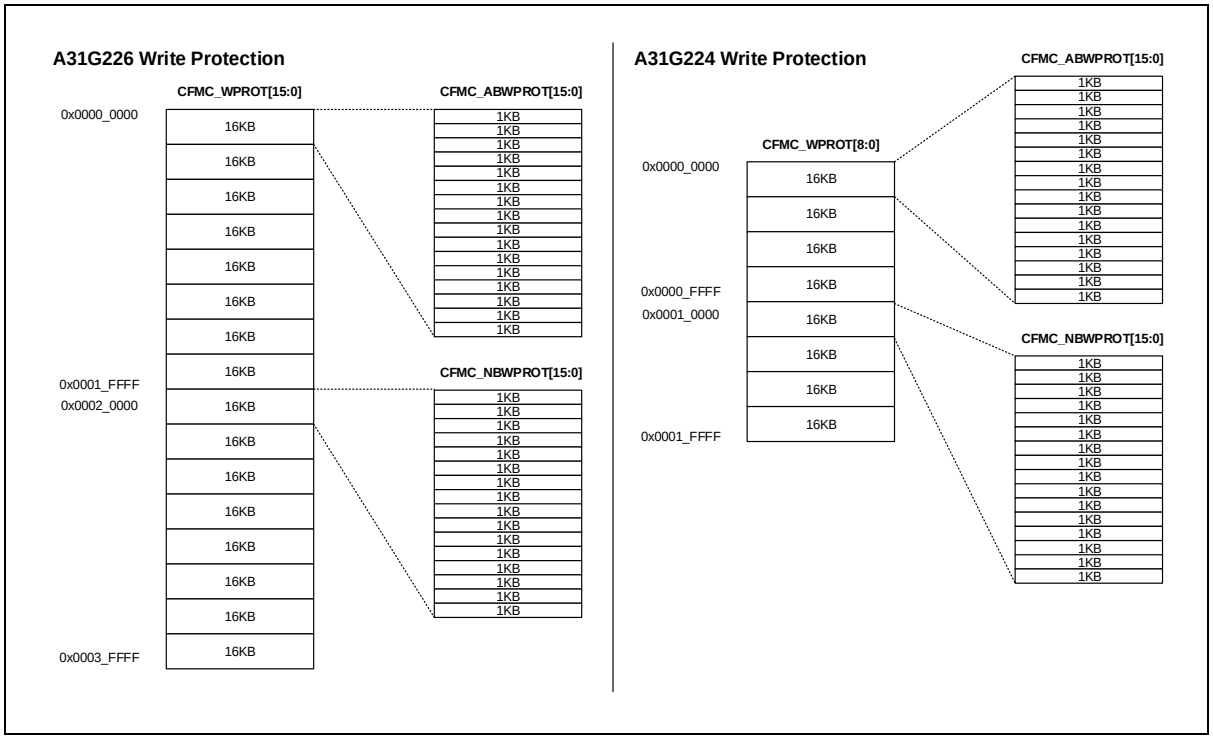
6.2.3 Write protection of code flash memory

A31G22x provides 16KB or 1KB segment-write protection for the internal code flash memory. Since the write protection does not allow any access of programming or erasing, if it is applied to the code flash memory, user data in the code flash memory is safely protected.

When rebooting the system, the write protection is enabled in whole area of the code flash memory by default. To update data in the code flash memory, a user needs to follow the procedure listed below:

1. Enter in PROT mode by entering 0x66 and 0x99 in <ACODE> of CFMC_MR register.
2. Set the corresponding bit of CFMC_WPROT, CFMC_ABWPROT, and CFMC_NBWPROT registers to '0'. The CFMC_WPROT register controls 16KB segment-write protection, while CFMC_ABWPROT and CFMC_NBWPROT control 1KB segment-write protection.
3. Start to program or erase on the area where the write protection is disabled.

Figure 35. The write protection coverage of code flash



6.2.4 Write protection of user bootloader area in code flash memory

If 1KB segment-write protection is used, a user can assign the internal code flash memory to User Bootloader area and User Data area. The user can assign the User Bootloader area to the address from the starting address of the code flash memory and up to 16KB, and protect data in this area from programming and erasing by using the 1KB segment-write protection. The rest area not set as the User Bootloader area can be used as User Data area.

To use Memory Swap, if a user assign the User Bootloader area and the User Data area to both Active Bank and Non-active Bank in the same size, it is able to do Boot Swap on the User Bootloader.

Table 34. Write Protection Address of User Bootloader Area with Swap Function (A31G226)

A31G226 – Used Bootloader Area					
Active Bank			Non-active Bank		
ABWPROT	Start addr.	End addr.	NBWPROT	Start addr.	End addr.
0	0x00000	0x003FF	0	0x203FF	0x20000
1	0x00400	0x007FF	1	0x207FF	0x20400
2	0x00800	0x00BFF	2	0x20BFF	0x20800
3	0x00C00	0x00FFF	3	0x20FFF	0x20C00
4	0x01000	0x013FF	4	0x213FF	0x21000
5	0x01400	0x017FF	5	0x217FF	0x21400
6	0x01800	0x01BFF	6	0x21BFF	0x21800
7	0x01C00	0x01FFF	7	0x21FFF	0x21C00
8	0x02000	0x023FF	8	0x223FF	0x22000
9	0x02400	0x027FF	9	0x227FF	0x22400
10	0x02800	0x02BFF	10	0x22BFF	0x22800
11	0x02C00	0x02FFF	11	0x22FFF	0x22C00
12	0x03000	0x033FF	12	0x233FF	0x23000
13	0x03400	0x037FF	13	0x237FF	0x23400
14	0x03800	0x03BFF	14	0x23BFF	0x23800
15	0x03C00	0x03FFF	15	0x23FFF	0x23C00

Table 35. Write Protection Address of User Bootloader Area with Swap Function (A31G224)

A31G224 – Used Bootloader Area					
Active Bank			Non-active Bank		
ABWPROT	Start addr.	End addr.	NBWPROT	Start addr.	End addr.
0	0x00000	0x003FF	0	0x10000	0x103FF
1	0x00400	0x007FF	1	0x10400	0x107FF
2	0x00800	0x00BFF	2	0x10800	0x10BFF
3	0x00C00	0x00FFF	3	0x10C00	0x10FFF
4	0x01000	0x013FF	4	0x11000	0x113FF
5	0x01400	0x017FF	5	0x11400	0x117FF
6	0x01800	0x01BFF	6	0x11800	0x11BFF
7	0x01C00	0x01FFF	7	0x11C00	0x11FFF
8	0x02000	0x023FF	8	0x12000	0x123FF
9	0x02400	0x027FF	9	0x12400	0x127FF
10	0x02800	0x02BFF	10	0x12800	0x12BFF

11	0x02C00	0x02FFF	11	0x12C00	0x12FFF
12	0x03000	0x033FF	12	0x13000	0x133FF
13	0x03400	0x037FF	13	0x13400	0x137FF
14	0x03800	0x03BFF	14	0x13800	0x13BFF
15	0x03C00	0x03FFF	15	0x13C00	0x13FFF

6.2.5 Write protection with memory bank swap

The write protection function provided by the CFMC_WPROT register has a different scope depending on the swap status of the memory bank. In applications that require memory area data update while user code is being executed, write protection function must be released so that data update can only be done in non-active bank area according to memory swap status. Refer to 6.2.2

Write protected coverage of Non-active block before Memory Swap

- CFMC_WPROT =0x00FF in A31G226
  - Non-active block (0x20000 to 0x3FFFF) is write/erase accessible
- CFMC_WPROT = 0x000F in A31G224
  - Non-active block (0x10000 to 0x1FFFF) is write/erase accessible

Write protected coverage of Non-active block after Memory Swap

- CFMC_WPROT =0x00FF in A31G226
  - Non-active block (0x20000 to 0x3FFFF) is write/erase inaccessible
- CFMC_WPROT = 0x00F0 in A31G224
  - Non-active block (0x10000 to 0x1FFFF) is write/erase inaccessible

Figure 36. The write protection coverage of code flash when using memory swap

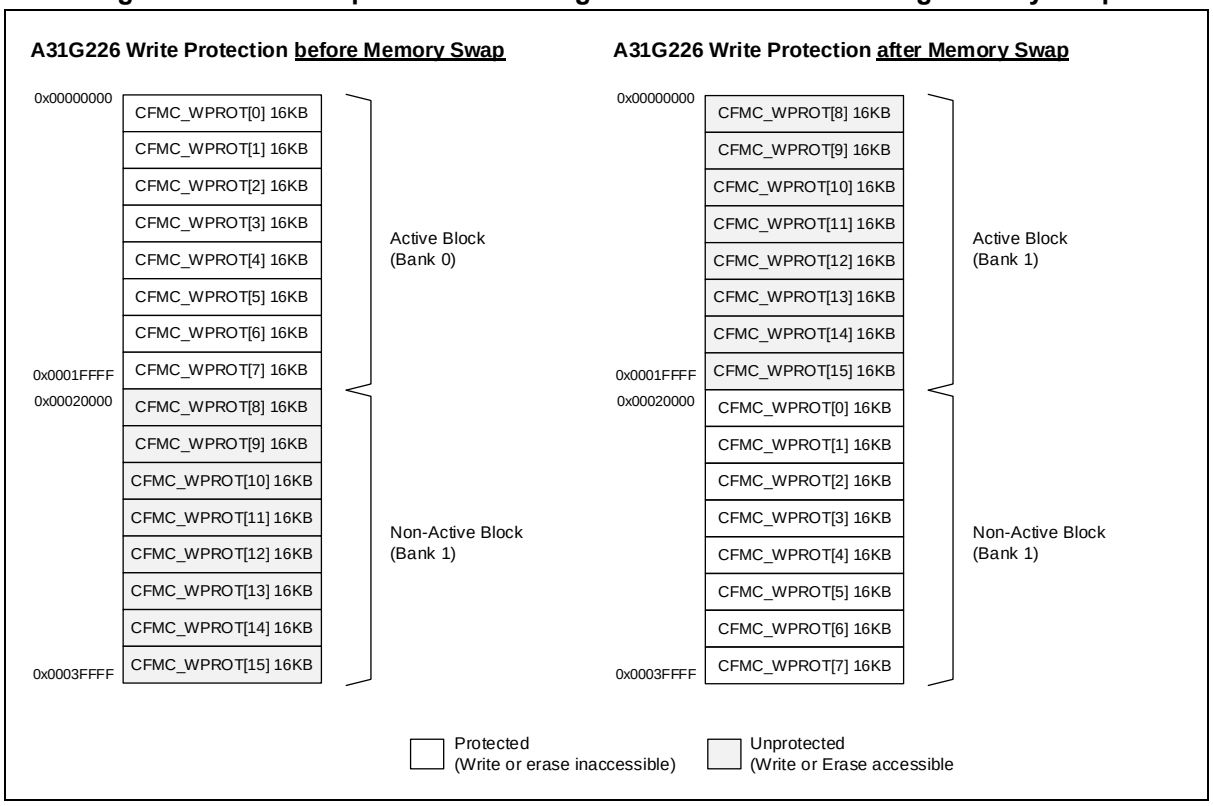
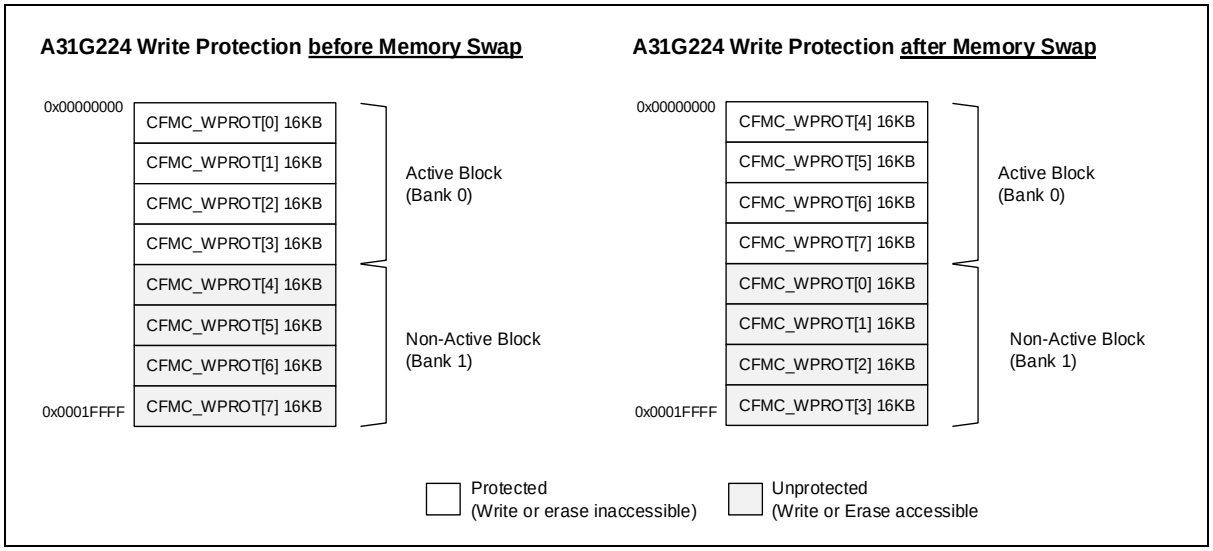


Figure 36. The write protection coverage of code flash when using memory swap (continued)



6.2.6 Read protection of code flash memory

Read protection settings

A31G22x provides two methods how to set the read protection as listed below:

1. In the Read Protection OTP area, access to CFMC_RPROT register and update corresponding code which configures the read protection in the form of OTP Command. When the code in the form of OTP Command is completed to update, if a user resets the system, BootROM code is executed at the initial stage of the booting process and the updated read protection is executed.
2. Access to the Read Protection register directly and set CFMC_RPROT<RPROT> to Level-1 or Level-2. This applies the read protection immediately. However using this method, the read protection is enabled temporarily and disabled after system reset. For permanent use of the read protection, it is recommended to use the 1st method introduced previously.

Read protection modes

The read protection operates in three modes, and it is irrespective of BOOT mode.

- Normal: Unlimited access to the internal flash memory
- Level-1: Debug connection is available. When the internal memory is accessed from outside, read output is 0x55AA55AA
- Level-2: Debug connection is not allowed. When the internal memory is accessed indirectly, read output is 0x55AA55AA.

A31G22x provides 2 levels of read protection function which are Level-1 and Level-2. The protection level can move from low (Level-1) to high (Level-2), and level settings to move from high to low is ignored.

Table 36. Read Protection Mode in Code Flash Memory

Read protection level	Mode	Code value	Descriptions
Unlock	Normal	0xFF	Debug and read access are available.
Level-1	Always	0x39	Debug is available, but read access is not allowed.
	Password	0xB9	Debug is available, but read output is 0x55AA55AA. In case of password matching, read access is available.
Level-2	Always	Other than Bit-7=0	Debug and read access are not allowed.
	Password	Other than Bit-7=1	Debug and read access are not allowed. In case of password matching, debug and read access are available.

Operation of the read protection level

Table 37. Operating Mode of Each Protection Level (Boot Mode = Normal Booting Mode)

Normal mode (unprotected)		Destination				• Debug is available. • Unlimited access to the whole memory. — R: Read — W: Write/Program — ME: Macro Erase — SE: Sector Erase — (E): Erase only when bulk erasing of the main FlashROM.
		BOOTROM	CODE	SRAM	OTP	
Source	BOOT/ CODE / SRAM	R	ALL	ALL	ALL	
	OTP	X	X	X	X	
	DMA	X	X	R/W	X	
	DEBUG	R	ALL	ALL	ALL	

Read protection LEVEL-1		Destination				• Debug is available. • Read output is 0x55AA55AA. • MICOM operation is not available during debug access. • To protect code from disclosing due to DMA settings, FlashRead through DMA is not available. • (E): Erase available only when bulk erasing of the main FlashROM.
		BOOTROM	CODE	SRAM	OTP	
Source	BOOT/ CODE/ SRAM	R	ALL	ALL	R/W/(E)	
	OTP	X	X	X	X	
	DMA	X	X	R/W	X	
	DEBUG	X	X	R/W	-/W/(E)	

Read protection LEVEL-2		Destination				• Debug is not available. • To protect code from disclosing due to DMA settings, FlashRead through DMA is not available. (E): Erase available only when bulk erasing of the main FlashROM.
		BOOTROM	CODE	SRAM	OTP	
Source	BOOT/ CODE/ SRAM	R	ALL	ALL	R/W/(E)	
	OTP	X	X	X	X	
	DMA	X	X	R/W	X	
	DEBUG	X	X	X	X	

BOOT mode = UART ISP mode

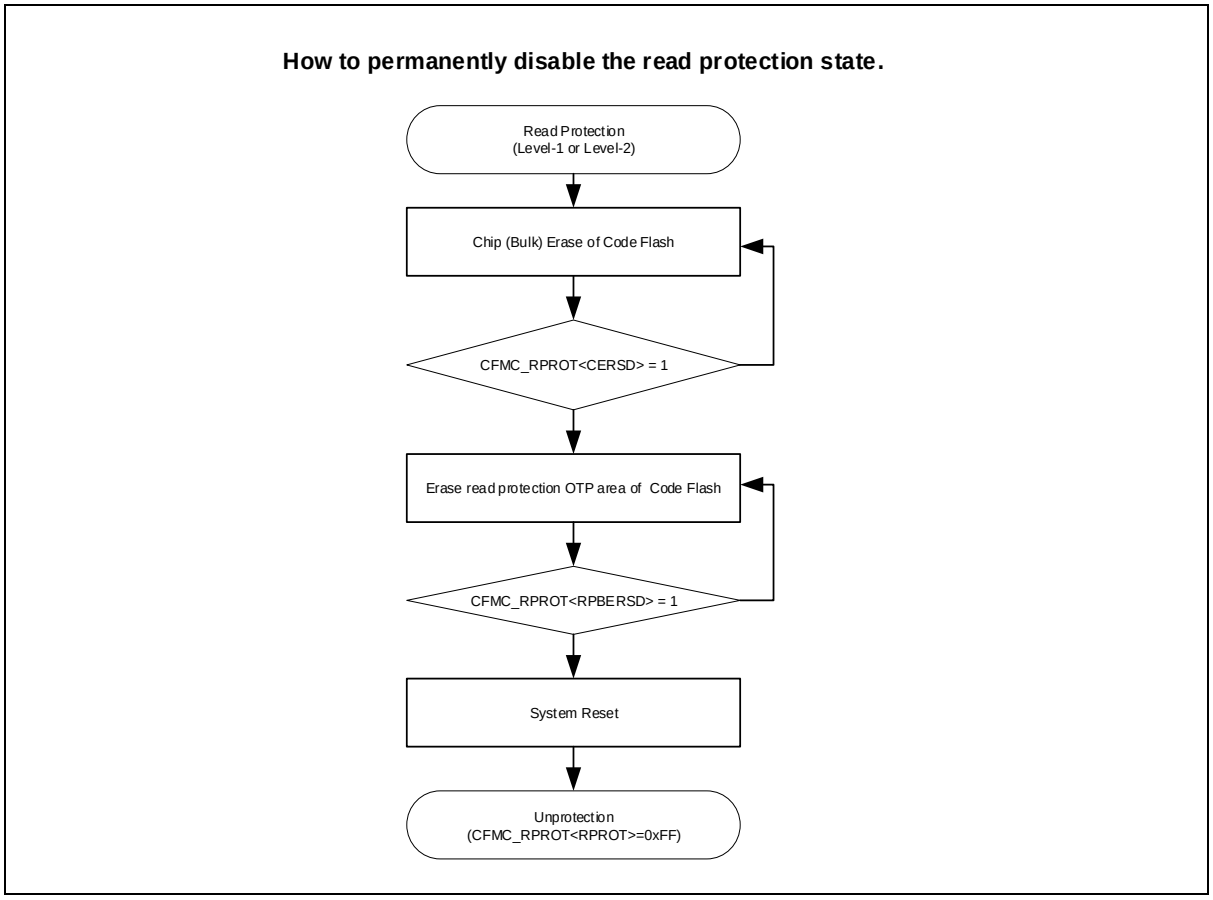
Read protection LEVEL-2		Destination				• Depending RP mode, it is decided for the debug access to be allowed or not. • Main flash cannot be read even from SRAM. • Content integrity is validated by CRC. • (E): Erase available only when bulk erasing of the main FlashROM.
		BOOTROM	CODE	SRAM	OTP	
Source	BOOT/ CODE/ SRAM	R	-/W/ ME/ SE	ALL	-/W/(E)	
	OTP	X	X	X	X	
	DMA	X	X	R/W	X	
	DEBUG	X	X	X	X	

Read protection release

To disable the read protection, a user must reset the system after removing values in Read Protection OTP area. In the system where the read protection Level-1 or Level-2 is applied, the internal code flash memory must be cleared by Chip (Bulk) Erase.

The Chip Erase enables CFMC_RPROT<CERSD> bit of Read Protection register, and the enabled bit maintains its state until the system reset occurs. That is, the code flash memory protected by the read protection cannot be accessed by debug or external interfaces such as UART. Since the read protection cannot be disabled until the whole code flash memory is cleared, user data in the memory is safely protected.

Figure 37. Sequence Disabling Read Protection Permanently: Bank Swap of Code Flash



Password and the read protection

When a user configures OTA or User Bootloader in the code flash memory where the read protection is applied, it is frequently required to reflect requirements for the code updates to the system. For this, A31G22x provides a password function to release the read protection temporarily.

For the password function, Preset register has predefined password, and Input register receives new password from a user and compares it and the Preset register. The Preset register is specified only once after the system initialization, and input values after the specification is ignored.

When the specified password in the Preset register is identical to the input value in the Input register, CFMC_RPROT<PWMATCH> flag is set to '1' and whole code flash memory enters in Normal mode temporarily to allow debug and external interfaces such as UART.

In this Normal mode when the read protection is temporarily disabled, to enable the read protection again, a user can use one of the following methods:

- System reset
- Enter the level to CFMC_RPROT<RPROT> field directly.
- Enter the password identical to the value of Preset register in Password Input register

Protection boot

When the read protection and the password function are applied to the code flash memory, the OTP Commands programmed in User Data OTP area and Read Protection OTP area are executed in the BootROM states which is initial stage of the system booting. This allows to use the protection boot function.

For example, a user can program to set the password in the Preset register and to enter the input value received through external URAT channel in the Password Input register of the User Data OTP area. In addition, the user can program OTP Commands to set Level-1 or Level-2 in the Read Protection OTP area. Then if the system reset occurs, the read protection is applied.

Now, the reset occurs by setting a nBOOT Pin (PB3) to low and the system enters in Boot mode. This is the sequence of receiving a password through UART channel. In this sequence, the password from the UART channel goes to the Password Input register and is compared to the value in the Preset register. If the two values match, the read protection is released. When the read protection is released to enter in Normal mode temporarily, a user can access to the code flash memory to update F/W. If the two values does not match, the system must be reset again and receive new password through the UART channel during Boot mode.

6.2.7 Bank swap of code flash memory

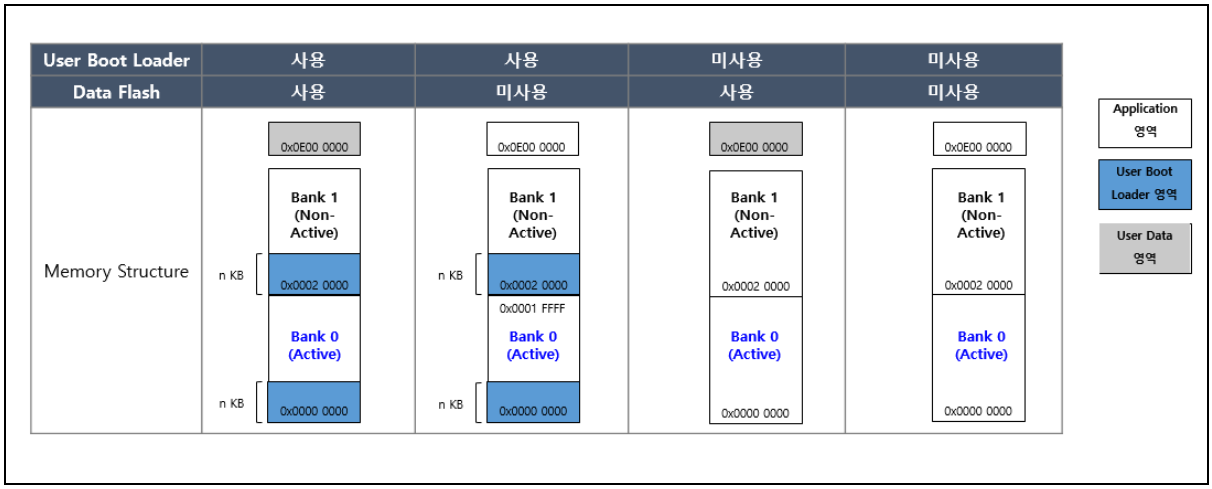
Bank Swap and memory structure

A31G22x provides a Flash Memory Bank Selection function for user applications and stable update and execution of User Bootloader.

To enable Memory Swap function in A31G226, a user needs to divide 256KB flash memory into two of 128KB and develop them for programming. Similar to A31G226, it is recommended to divide 128KB flash memory into two of 64KB to utilize Memory Swap function in A31G224.

To use the Memory Swap function in user applications, a user is required to acquire information of Bank Selection registers and memory structures.

Figure 38. A31G226 Memory Structure with Bank Swap Available

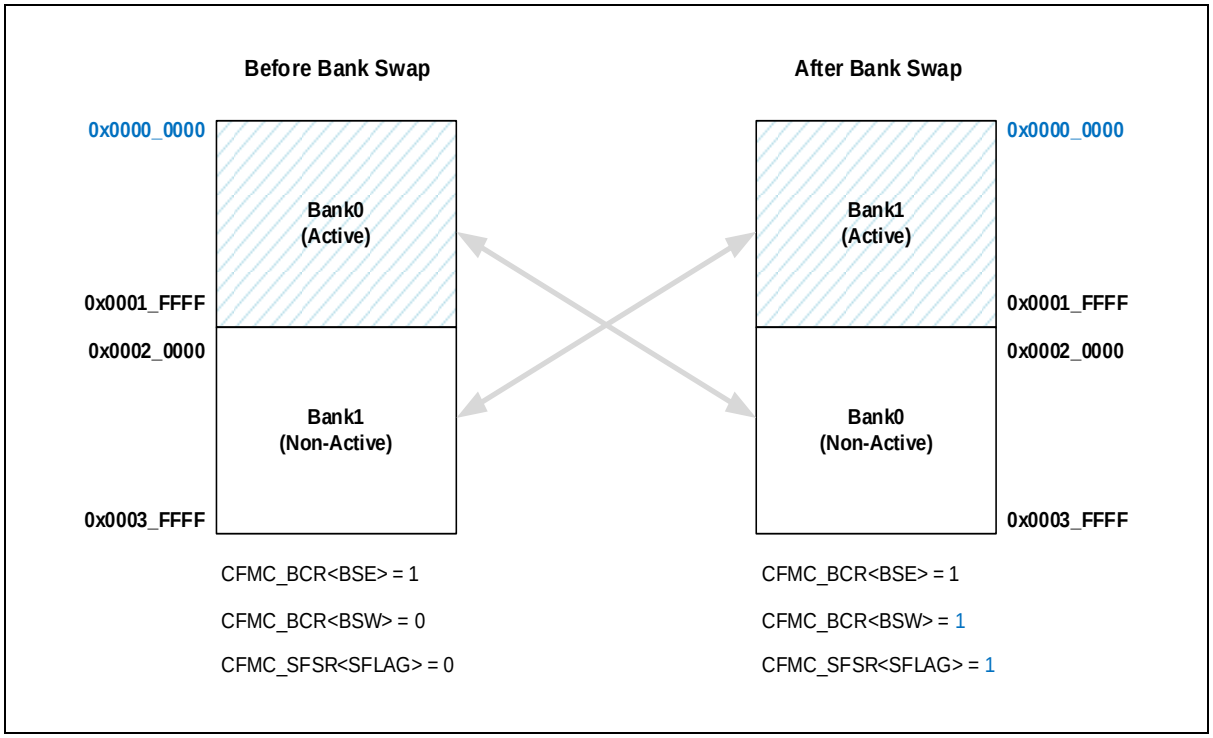


A31G22x Bank Swap operation

When the Memory Bank Swap function is enabled, memory block starting with address 0x0000_0000 is considered as Active block, where code will be executed after the BootROM operation. Next to the Active block, Non-active block is located where code will be updated or is used for back up operation.

When the Memory Bank Swap function is enabled, certain code is updated in the Non-active area, Swap control bit is set, and the system restarts successively. Then value of Swap Flag (SFLAG) is monitored in BootROM and switching between the Active block and the Non-active block occurs. By the Swap operation, the Non-active block where the code was updated switches to the Active block, while the Active block where the code was executed before the update operation switches to the Non-active block.

Figure 39. Logical Address Change of the Code Flash Memory by Bank Swap Settings



A31G22x memory structure when Bank Swap is used

Figure 40. A31G226 Memory Address when enabling Bank Swap

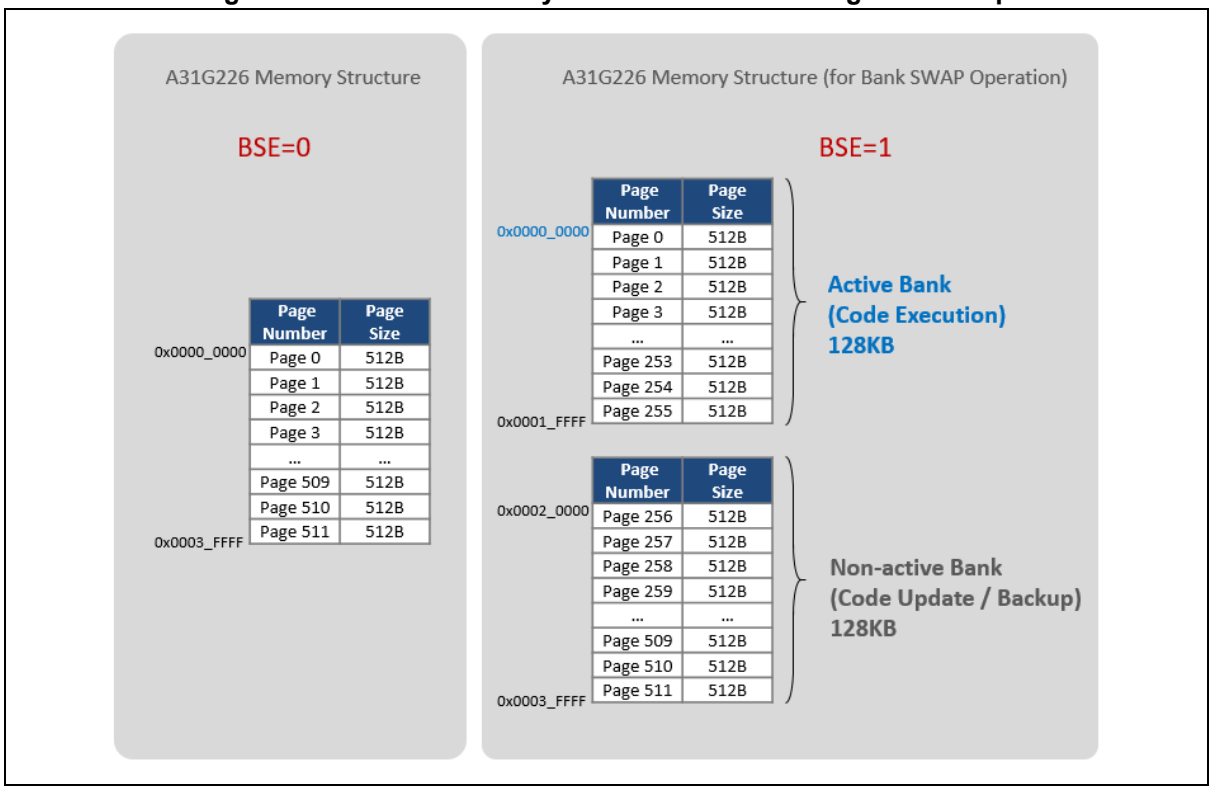
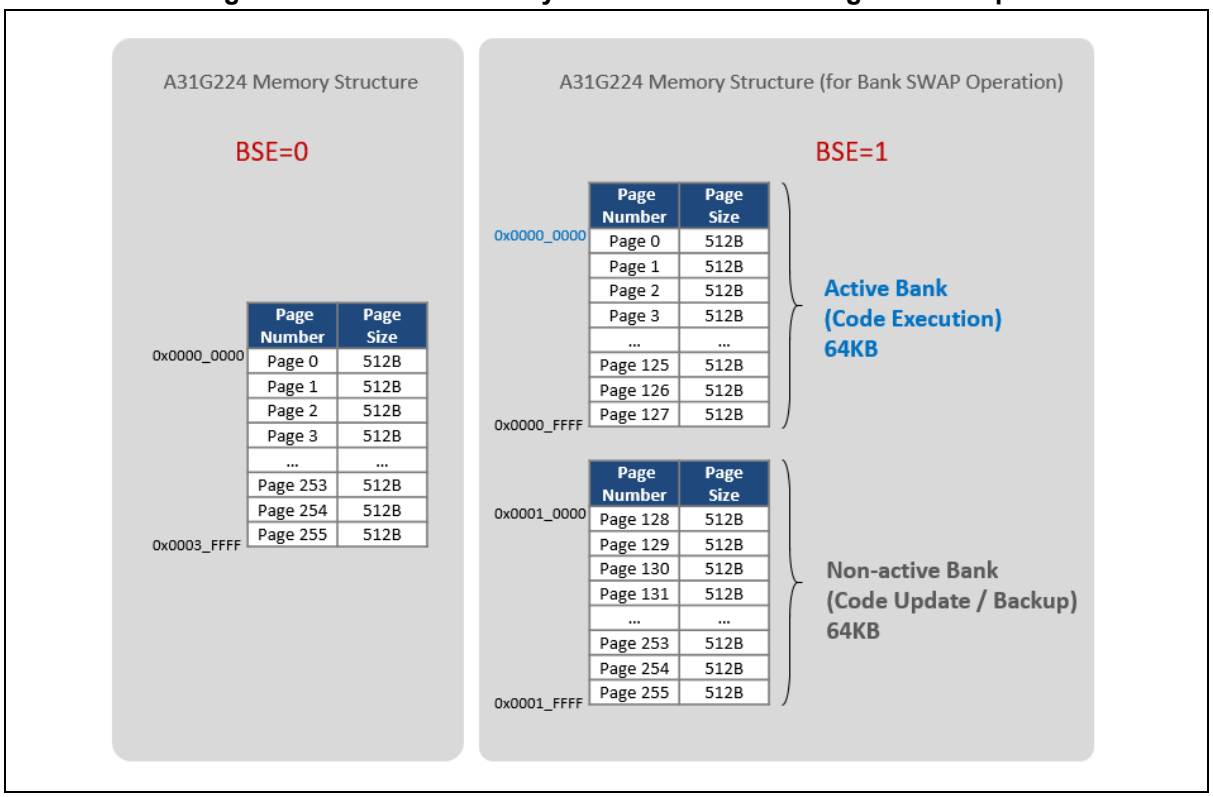


Figure 41. A31G224 Memory Address when enabling Bank Swap



Limitations and recommendations

- When Bank Swap is enabled, a user must use Self-PGM or Self-Erase to access the internal code flash memory. Sector Erase (Page Erase) and Sector Program in SRAM are also available.
- When a main program is running in Bank1 area, where the read protection is applied, if Chip (Mass) Erase is executed to disable the read protection permanently, user data in whole area of Bank0 and Bank1 is cleared. After system reset, the read protection is disabled. However, since Bank Swap function is still active, program in Bank1 is supposed to be executed but Bank1 doesn't have any command at all.

6.2.8 Sequence of Bank Swap

Figure 42. Sequence of Bank Swap: Bank0 ➔ Bank1

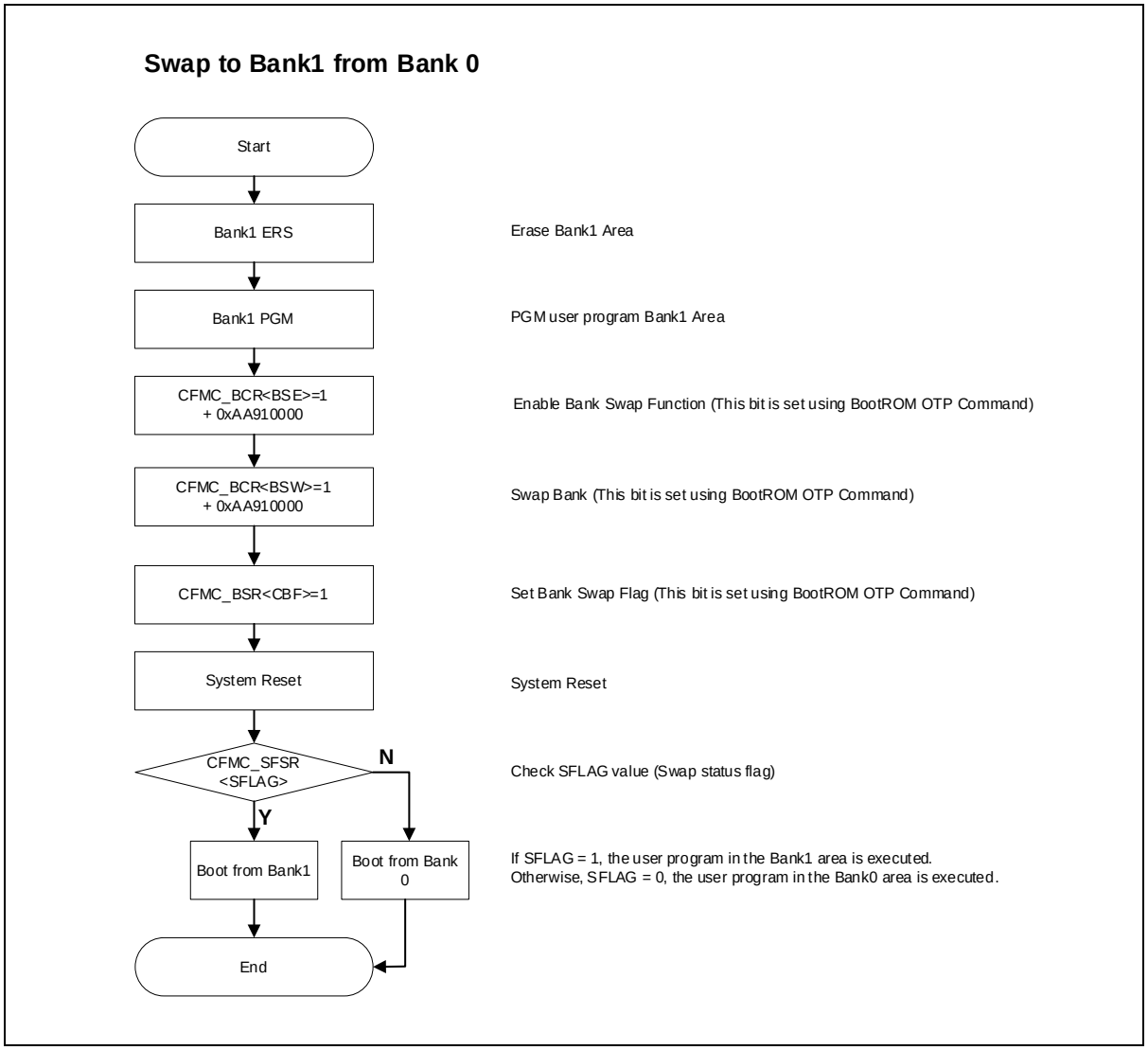
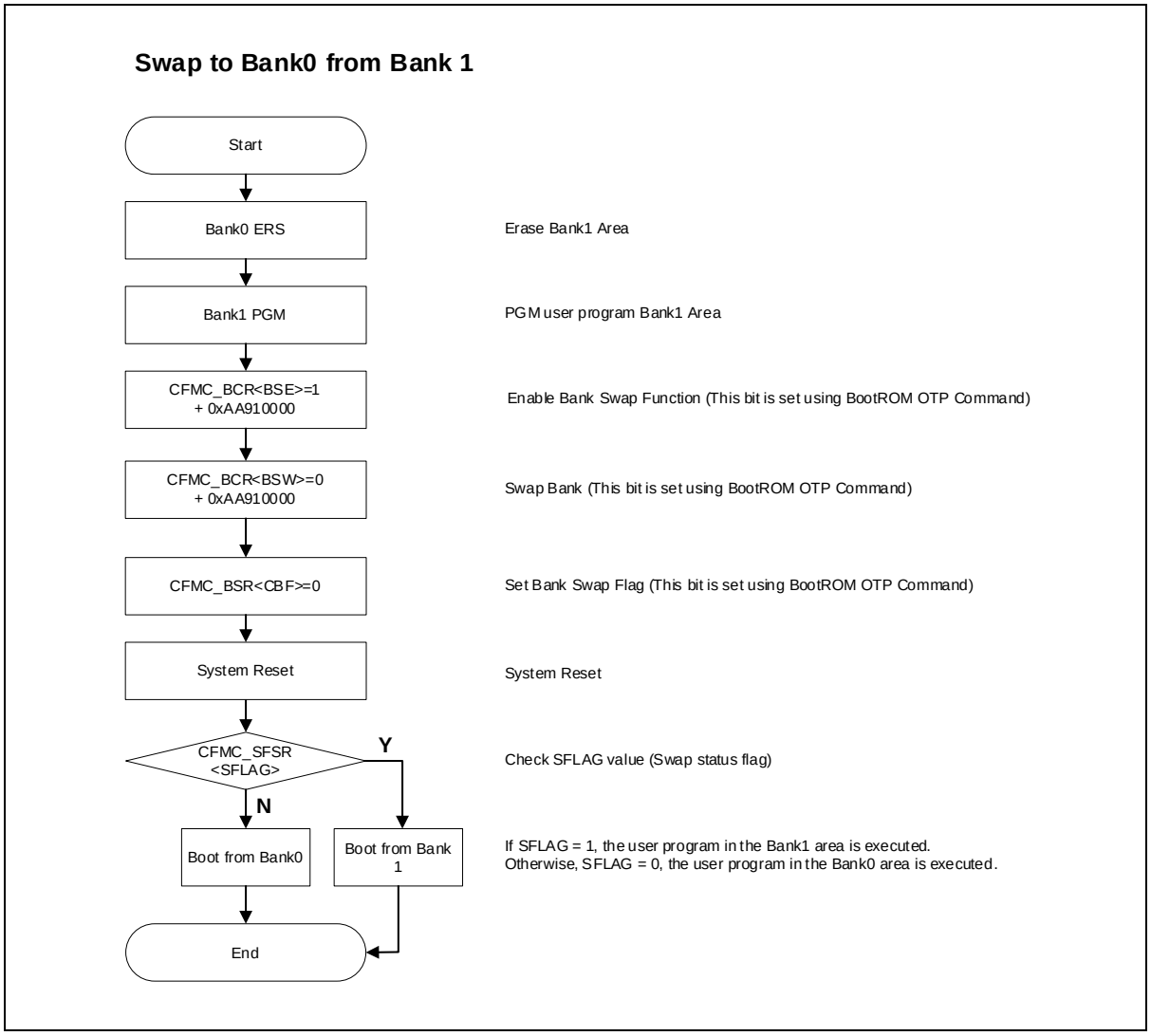


Figure 43. Sequence of Bank Swap: Bank1 ➔ Bank0



7 Data Flash memory controller (DFMC)

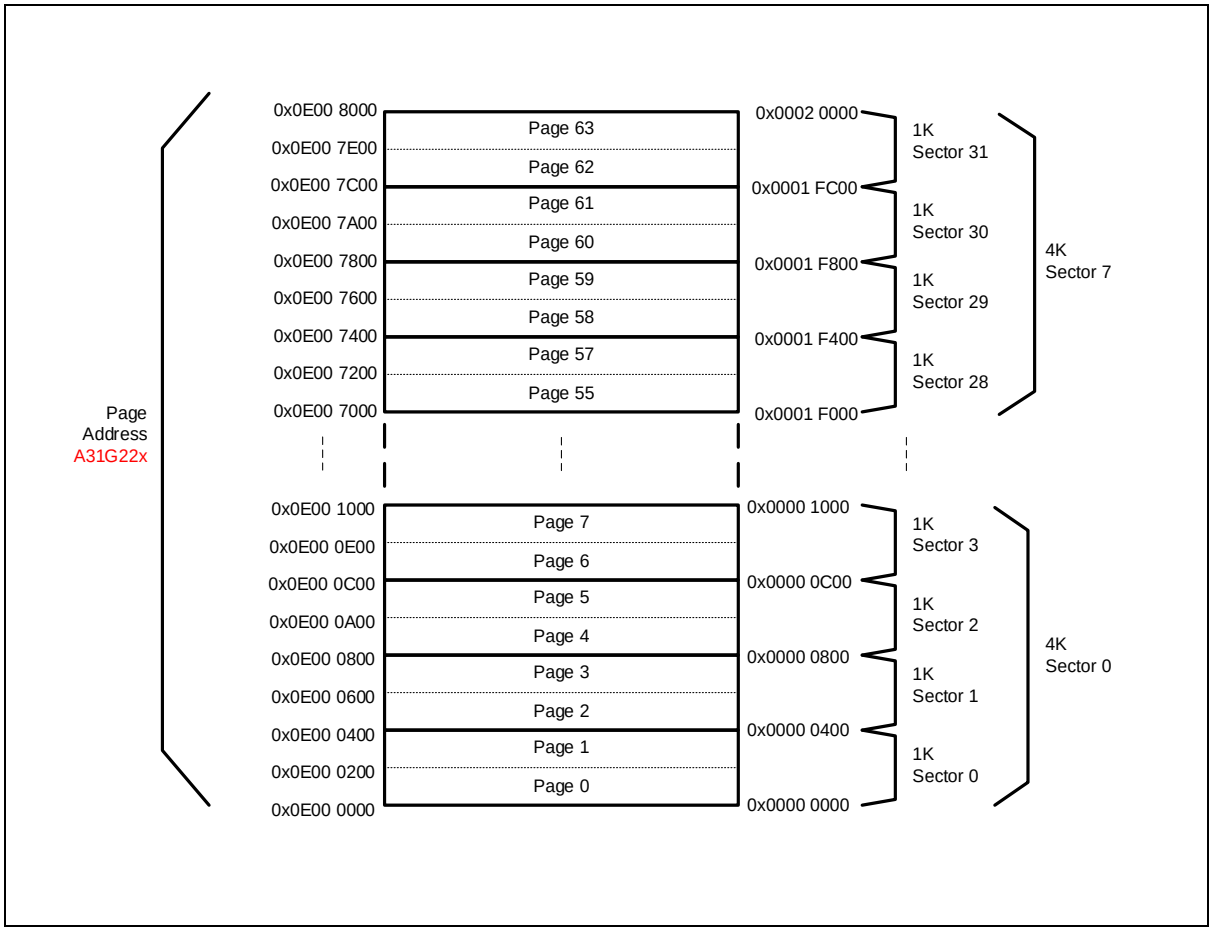
Data Flash Memory Controller is an internal flash memory interface controller, and includes following features as shown below:

- 32KB data flash memory
- Wait, 0-wait to 6-wait (default : 5-wait)
- Read protection support
- Write protection support with 1KB-unit
- A byte (8-bit) unit program and word (32-bit) unit program support
- Endurance: 100,000 Cycles
- Data Retention: 10 Years

Table 38. Data Flash Memory Controller Features

Item	Description
Size	32KB (A31G22x)
Start Address	0x0E00_0000
End Address	0x0E00_7FFF
Page Size	512-byte
Total Page Count	64 pages
PGM Unit	Byte PGM : 1-byte Word PGM : 1-word (4-byte)
Erase Unit	512-byte/ 1KB/ 4KB/ bulk

Figure 44. Data Flash Memory Map (A31G22x Data Flash)



7.1 Registers

Base address of data flash memory controller is introduced in the followings:

Table 39. Base Address of Flash Memory Controller

Name	Base address
Data Flash controller	0x4000_0200

Table 40. DFMC Register Map

Name	Offset	Type	Description	Reset value	Ref.
DFMC_MR	0x0004	R/W	Data Flash Memory Mode Select Register	0x0000_0000	7.1.1
DFMC_CR	0x0008	R/W	Data Flash Memory Control Register	0x0000_0000	7.1.2
DFMC_AR	0x000C	R/W	Data Flash Memory Address Register	0x0000_0000	7.1.3
DFMC_DR	0x0010	R/W	Data Flash Memory Data Register	0x0000_0000	7.1.4
DFMC_BUSY	0x0018	R/O	Data Flash Memory Write Busy Status Register	0x0000_0000	7.1.5
DFMC_CRC	0x0020	R/W	Data Flash Memory CRC-CCITT check Register	0x0000_FFFF	7.1.6
DFMC_CFG	0x0030	R/W	Data Flash Memory Configuration Register	0x0000_8500	7.1.7
DFMC_WPROT	0x0034	R/W	Data Flash Memory Write Protection Register	0xFFFF_FFFF	7.1.8
DFMC_RPROT	0x003C	R/W	Data Flash Memory Read Protection Register	0x0000_00FF	7.1.9
DFMC_PWIN	0x0040	WO	Data Flash Memory Password Input Register	-	7.1.10
DFMC_PWPRST	0x0044	WO	Data Flash Password Preset Register	-	7.1.11

7.1.1 DFMC_MR: data flash memory mode register

DFMC_MR is an internal flash memory mode register. Size of this register is 32-bit.

DFMC_MR=0x4000_0204

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			ACODE
-			0x00
-			RW

Bits	Name	Function	
7 0	ACODE	Data flash entry code.	
		5A → A5	Flash mode entry
		A5 → 5A	Trim mode entry
		81 → 28	AMBA mode entry
		66 → 99	PROT mode entry

7.1.2 DFMC_CR: data flash memory control register

DFMC_CR is an internal flash memory control register.

DFMC_CR=0x4000_0208

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	RPAEN	SELFPGM	Reserved	WPGMEN	Reserved	IFEN	Reserved	MAS	SECT4K	SECT1K	PMODE	WADCK	PGM	ERS	HVEN
-	0	0	-	0	-	0	-	0	0	0	0	0	0	0	0
-	RW	RW	-	RW	-	RW	-	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
24	RPAEN	Read Protection block access enable.	
		0	Disable read protection block access
		1	Enable read protection block access
23	SELFPGM	When this bit is set to '1', 1-byte data flash writing is possible. This bit is affected by the PGM bit and PMOME bit.	
16	WPGMEN	When the 'PMODE' bit is set to '1', 1 word (4 Bytes) data flash writing is possible.	
		0	8-bit write
		1	32-bit write
12	IFEN	Enable the access of user option block in data flash access	
		0	Disable
		1	Enable the access of user option 0/1/2 block (It works with OTP3EN to OTP1EN for PMODE operation).
7	MAS	Enable mass erase (bulk) – All data of data flash area is erased.	
		0	Mass (bulk) erase disable
		1	Mass (bulk) erase enable.
6	SECT4K	4KB-units sector erase	
		0	Sector 4K erase disable
		1	Sector 4K erase enable
5	SECT1K	1KB-units sector erase	
		0	Sector 1K erase disable
		1	Sector 1K erase enable
4	PMODE	The type of program mode of data flash memory. (Flash Address path is connected with FMAR)	
		0	Normal mode
		1	PMODE enable **NOTE:** PMODE only valid when SELFPGM bit = 0
3	WADCK	Enable Program/Erase address data latch clock	
		0	Program/Erase address data latch clock disable
		1	Program/Erase address data latch clock enable, this bit assert for one system clock period so user cannot read

2	PGM	Enable program mode of data flash	
		0	Program mode disable
		1	Program mode enable
1	ERS	Enable erase mode of data flash	
		0	Erase mode disable
		1	Erase mode enable
0	HVEN	Enable High voltage cycle	
		0	High voltage cycle disable
		1	High voltage cycle enable (start program or erase cycle) **NOTES:** 1. User must set and clear this bit when PMODE=1. 2. In SELFPGM mode, user must set HVEN then HVEN will be cleared automatically after WRBUSY goes low.

7.1.3 DFMC_AR: data flash memory address register

This register sets the output address of the flash memory and is used in the flash memory program mode. When setting the data flash memory address value, use a width of up to 16 bits. When data flash is programed in 1 Byte unit, the values of A0 and A1 in the memory address set in this register can be '1' or '0'. When If the data flash is programmed in 4 Bytes (word) units with the DFMC_CR <WPGMEN> bit set to '1', the A0 and A1 values set in this register are always 0.

DFMC_AR=0x4000_020C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2	1	0
Reserved		FADDR		A1	A0
-		0x0000		0	0
-		RW		RW	RW

Bits	Name	Function
15 0	FADDR	Word (32-bit) base address Auto Incremental after WADCK trigger (After latching of target address).
1	A1	16-bit mode address A1 (for Data Flash)
0	A0	Byte (8-bit) mode address A0 (for Data Flash)

NOTES:

1. DFMC_CR<WPGMEN>=0, Byte (8-bit) unit address (Address+1)
2. DFMC_CR<WPGMEN>=1, Word (32-bit) unit address, (Address+4)

7.1.4 DFMC_DR: data flash memory data input register

The DFMC_DR register gets the byte-sized (8-bit) or word-sized (32-bit) input data to be programmed or erased to data flash memory.

DFMC_DR=0x4000_0210

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
FDATA			
0xFFFF_FFFF			
RW			
Reserved			FDATA
-			0x00
-			RW

[Word-PGM mode], DFMC_CR<WPGMEN>=1

Bits	Name	Function
32 0	FDATA	Word size (32-bit)

[Byte-PGM mode], DFMC_CR<WPGMEN>=0

Bits	Name	Function
8 0	FDATA	Byte size (8-bit)

7.1.5 DFMC_BUSY: data flash memory write busy status register

DFMC_BUSY is a flash write (program/erase) busy status monitor register. This register is a 1-bit read only register.

DFMC_BUSY=0x4000_0218

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1	0
Reserved	WRBUSY
-	0
-	RO

Bits	Name	Function
0	WRBUSY	Write busy status bit FLBUSY bit goes high after set HVEN bit (in CTRL register). FLBUSY bit goes low when WRBUSY becomes low after program (or erase) complete.

7.1.6 DFMC_CRC: data flash memory CRC check register

DFMC_CRC represents the 16-bit CRC-CCITT calculated value of all data in the data flash memory area. DFMC_CRC is read only register which enabled by CRCEN bit of DFMC_CFG register. To obtain CRC-CCITT calculation results for data in the data flash memory area, perform at least 16 times (16-word) of DFMC_CRC register read access.

CFMC_CRC=0x4000_0220

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
Reserved	CRC-CCITT
-	0xFFFF
-	RW

Bits	Name	Function
15 0	CRC- CCITT	CRC- CCITT result register polynomial: $(1 + x^5 + x^{12} + x^{16})$ At least 16-word read to get a CRC value. Data width: 32 (The first serial bit is D[31])

7.1.7 DFMC_CFG: data flash memory configuration register

DFMC_CFG register supports access timing configuration and the CRC-CCITT operation of the data flash memory area. The input identification key is required in the DFMC_CFG register to write the setting value.

DFMC_CFG=0x4000_0230

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5 4 3 2 1 0
WTIDKY		Reserved	WAIT	CRCINIT	CRCEN	Reserved
0x7858		-	101	0	0	-
WO		-	RW	RW	RW	-

Bits	Name	Function	
31 16	WTIDKY		Write identification key. On writes, write 0x7858 to these bits, otherwise the write is ignored.
10 8	WAIT		This bits only be written in AMBA mode and MSB 16-bit (bit [31:16]) must be 0x7858
		000	WAIT is 000, flash access in 1 cycle (0-wait)
		001	WAIT is 001, flash access in 2 cycles (1-wait)
		010	WAIT is 010, flash access in 3 cycles (2-wait)
		011	WAIT is 011, flash access in 4 cycles (3-wait)
		100	WAIT is 100, flash access in 5 cycles (4-wait)
		101 11x	WAIT is 101, 11x flash access in 6-cycles (5-wait) – default
7	CRCINIT	0	When this bit is set('1'), CRC register will be initialized It should be reset again before read flash to generate CRC-CCITT calculation (Initial value of DFMC_CRC is 0xFFFF)
6	CRCEN	0	CRC-CCITT enable CRC value will be calculated at every flash read timing

NOTES:

1. In order to operate user application with optimized performance, the proper access time of the data flash memory is required. Refer to Table 41 for the settings of wait-time for the data flash memory of A31G22x.
2. When read access to user option area, wait time needs to be adjusted based on min. 100ns. It is recommend to change the flash bus speed to 5-wait (6 cycles) accessing the user option area.

Table 41. The wait-time setting table of A31G22x data flash memory

APB speed (HCLK)	Wait-time	Access freq.	Access time	Recommended access time of data flash area	Recommended access time of user option area
48MHz	5 (6-cycles)	8.0MHz	125.0ns	50ns	100ns
48MHz	4 (5-cycles)	9.6MHz	104.1ns	50ns	100ns
48MHz	3 (4-cycles)	12.0MHz	83.0ns	50ns	100ns
48MHz	2 (3-cycles)	16.0MHz	62.5ns	50ns	100ns
48MHz	1 (2-cycles)	24.0MHz	41.6ns	50ns	100ns
48MHz	0 (1-cycles)	48.0MHz	20.8ns	50ns	100ns
32MHz	5 (6-cycles)	5.3MHz	187.0ns	50ns	100ns
32MHz	4 (5-cycles)	6.4MHz	156.0ns	50ns	100ns
32MHz	3 (4-cycles)	8.0MHz	125.0ns	50ns	100ns
32MHz	2 (3-cycles)	10.6MHz	93.7ns	50ns	100ns
32MHz	1 (2-cycles)	16.0MHz	62.5ns	50ns	100ns
32MHz	0 (1-cycles)	32.0MHz	31.2ns	50ns	100ns
20MHz	5 (6-cycles)	3.3MHz	300.0ns	50ns	100ns
20MHz	4 (5-cycles)	4.0MHz	250.0ns	50ns	100ns
20MHz	3 (4-cycles)	5.0MHz	200.0ns	50ns	100ns
20MHz	2 (3-cycles)	6.7MHz	150.0ns	50ns	100ns
20MHz	1 (2-cycles)	10.0MHz	100.0ns	50ns	100ns
20MHz	0 (1-cycles)	20.0MHz	50.0ns	50ns	100ns
500KHz	5 (6-cycles)	83.3KHz	12.0us	50ns	100ns
500KHz	4 (5-cycles)	100.0KHz	10.0us	50ns	100ns
500KHz	3 (4-cycles)	125.0KHz	8.0us	50ns	100ns
500KHz	2 (3-cycles)	166.6KHz	6.0us	50ns	100ns
500KHz	1 (2-cycles)	250KHz	4.0us	50ns	100ns
500KHz	0 (1-cycles)	500KHz	2.0us	50ns	100ns

7.1.8 DFMC_WPROT: data flash memory write protection register

DFMC_WPROT register supports the write protection function for the internal data flash memory. A31G22x provides the write protection for 1KB segment of the data flash memory by configuring a bit of this register.

The write protection is applied to whole area of the data flash memory at system initialization. If a user wants to update data in this area, the write protection for the corresponding segment must be disabled before starting erasing or programming.

DFMC_WPROT=0x4000_0234

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
WPROT			
0xFFFF_FFFF			
RW			

[A31G226]

Bits	Name	Function
32 0	WPROT	Write protection each 1KB segments for whole memory address **NOTES:** 1. Protection Range = WPROT[n] : (n*1kB) ~ (1kB * (n+1) - 1), n=0~31 2. The FM_WPROT register can only be modified in PROT (FM_MR = 66-> 99) mode.

7.1.9 DFMC_RPROT: data flash memory read protection register

DFMC_RPROT register supports the read protection function for the internal data flash memory. A31G22x provides the read protection for whole area of the data flash memory and restricts flash access and debug operations by enabling the read protection.

To disable this read protection, a user can use one of the following two methods:

1. When a user enters a password in DFMC_PWIN register which matches to the predefined value, the read protection is disabled. This is available when both the write protection and pre-set function of DFMC_PWPRST register are applied.
2. If a user erases whole data flash memory using Chip Erase, the read protection can be disabled.

The read protection of the data flash memory operates together with the read protection of the data e flash memory. Its protection level is same as the Level-1 of the da flash memory, and Level-2 is not available.

Before applying the read protection to the code flash memory or to the data flash memory, a user must check following cases:

1. When the read protection is enabled only in the code flash memory, the read protection is applied to the data flash memory too. In this condition, a user must disable the read protection in the code flash memory before accessing to the data flash memory.
2. If the read protection is enabled in the data flash memory and is applied to the code flash memory together, a user must disable the read protection first in the data flash memory and secondly in the code flash memory before accessing to the data flash memory.

DFMC_RPROT=0x4000_023C

31	30	29 28 27	26	25	24	23 22 21 20 19 18 17	16	15 14 13 12 11 10 9	8	7 6 5 4 3 2 1 0
DBGMOD	SRBOOT	Reserved	PWMATCH	RPBERSD	CERSD	Reserved	RPROT_STS	Reserved	RPROT_EN	RPROT
0	0	-	0	0	0	-	0	-	0	0x0000_00FF
RO	RO	-	RO	RO	RO	-	RO	-	RO	RW

Bits	Name	Function	
31	DBGMOD	Debug operating status bit	
		0	Not operating
		1	Operating
30	SRBOOT	SRAM boot mode status bit	
		0	Normal mode
		1	SRAM boot mode
26	PWMATCH	Password match flag bit	
		0	Not matched
		1	Password preset value and user password input value are matched.

25	RPBERSD	Chip erase and read protection block erase done flag bit	
		0	Not occurred
		1	Chip erase and read protection block erase done
24	CERSD	Chip erase done flag bit	
		0	Not occurred
		1	Chip erase done
		0	Normal status
		1	Protection level-2 status
16	RPROT_STS	Read protection level-1 status bit (raw data)	
		0	Normal status
		1	Protection level-1 status
		0	Disable
		1	Enable
8	RPROT_EN	Read protection level-1 enable status bit	
		0	Disable
		1	Enable
7 0	RPROT	Read Protection Control bit	
		0xFF	Unprotected
		Others	Read Protection Level-1 with Password
NOTE: The DFMC_RPROT register can only be modified in PROT (DFMC_MR = 66-> 99) mode.			

Table 42. Available operating modes by read protection level

Protection level	Operation mode	Data		Read Protection OTP	
		read	write	read	write
UNPROT	Normal mode	O	O	O	O
	Debug mode	O	O	O	O
	Boot mode	O	O	O	O
RPROT (Password)	Normal mode	X	X	X	X
	Debug mode	X	X	X	X
	Boot mode	X	X	X	X

NOTES:

1. The priority levels of protection mode are as follows:
2. During read protection, a key value must be input to be readable (modified so that all FLASH, debug, and SRAM are not readable)
3. A higher protection level cannot transition to a lower level without the chip erase and read protection block erase executed.

(Ex) Procedure for transitioning to UNRPROT from RPROT

A. Chip erase
B. DFMC_RPROT.CERSD flag check
C. Read protection erase
D. DFMC_RPROT.RPBERSD flag check
E. Write 0xFF to DFMC_RPROT.RPROT to transition to UNPROT mode

7.1.10 DFMC_PWIN: data flash memory password input register

A user enters a password in DFMC_PWIN register to release the password function, if DFMC_PWRRST register has a pre-set password. If DFMC_PWRRST register has not a pre-set password, the value entered in this register will be ignored.

If DFMC_RPROT register enabled both flash read protection and password, a user must enter a password in DFMC_PWIN, which was predefined in DFMC_PWPRST register, to access the data flash memory. After finishing user operation on the data flash memory, the user enters the password in DFMC_PWIN register again to recover the read protection.

DFMC_PWIN=0x4000_0240

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
PWIN			
-			
WO			

Bits	Name	Function
31 0	PWIN	Password input data bit Writing is done twice to the register in the order from LSB to MSB. Rewriting is restricted once the register is written. Ex) Entering 0x1234_5678 as a password. DFMC_PWIN = 0x1234_5678; DFMC_PWIN = 0x1234_5678;

NOTE: DFMC_PWIN inaccessible after two write access.

7.1.11 DFMC_PWPRST: data flash memory password preset register

DFMC_PWPRST is a password pre-set register which is used to define a system password. Password is set as a system password if it is entered in this register twice. If a password is set while processing user code, it is initialized by system reset. To maintain the pre-set password even after the system initialization, a user must define the pre-set value in User Data area at the system initialization stage by using User Bootloader.

If the flash read protection is applied and DFMC_PWPRST register has a pre-set password, a user can release the read protection by entering a password in DFMC_PWIN.

DFMC_PWPRST=0x4000_0244

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWPRST																															
-																															
WO																															

Bits	Name	Function
31 0	PWPRST	Password preset data bit Writing is done twice to the register in the order from LSB to MSB. Rewriting is restricted once the register is written. Ex) Presetting 0x1234_5678 as a password. DFMC_PWPRST = 0x1234_5678; DFMC_PWPRST = 0x1234_5678;

NOTES:

1. DFMC_PWPRST inaccessible after two write access.
2. DFMC_PWPRST register cannot write access in SRAM boot or Debug boot mode.

7.2 Functional description

7.2.1 Data flash erase and program examples

The basic step of Flash memory program consists with following three steps. Minimum Program or Erase unit is a Page and 128-word (512-byte) become a page.

- Page erase
- Page program

For all of erase operations, pre-program operation is required to prevent over erase of flash memory cells.

Erase example

1. Enable flash mode to write DFMC_MR register (write 0x5A and then write 0xA5 into DFMC_MR).
2. Set target Page address in DFMC_AR.
3. Set PMODE bit first.
4. Set ERS, WADCK, HVEN bits of DFMC_CR.
5. Wait until IDLE bit of DFMC_BUSY register becomes "0" after erase.
6. Clear ERS, HVEN bits of DFMC_CR.
7. Set 0x80 to DFMC_BUSY.
8. Clear DFMC_CR.
9. Clear Flash mode (write 0x00 and then write 0x00 into DFMC_MR).

Program example

1. Enable flash mode to write DFMC_CR register (write 0x5A and then write 0xA5 into DFMC_MR).
2. Set PMODE bit first.
3. Set target Page address in DFMC_AR.
4. Set PGM bits of DFMC_CR.
5. Write word (32-bit) data into DFMC_DR, address increased automatically based on word address.
6. Set WADCK, HVEN bits of DFMC_CR.
7. Wait until IDLE bit of DFMC_BUSY register becomes "0" after program.
8. Clear HVEN bits of DFMC_CR.
9. Set 0x80 to DFMC_ BUSY.
10. Clear PGM bits of DFMC_CR.
11. Clear DFMC_CR.
12. Clear Flash mode (write 0x00 and then write 0x00 into DFMC_MR).

7.2.2 Write protection of data flash memory

A31G22x provides 1KB segment-write protection for the 32KB size of the data flash memory. Since the write protection does not allow any access of programming or erasing, if it is applied to the data flash memory, user data in the data flash memory is safely protected.

When rebooting the system, the write protection is enabled in whole area of the data flash memory by default. To update data in the data flash memory, a user needs to follow the procedure listed below:

1. Enter in PROT mode by entering 0x66 0x99 in <ACODE> of DFMC_MR register.
2. Set the corresponding bit of DFMC_WPROT register to '0' which controls 1KB segment-write protection.
3. Start to program or erase on the area where the write protection is disabled.

7.2.3 Read protection of data flash memory

Read protection settings

A31G22x provides three methods how to set the read protection for the data flash memory as listed below:

1. In the Read Protection OTP area, access to DFMC_RPROT register and update corresponding code which configures the read protection in the form of OTP Command. When the code in the form of OTP Command is completed to update, if a user resets the system, BootROM code is executed at the initial stage of the booting process and the updated read protection is executed.
2. Access to the Read Protection register directly and set DFMC_RPROT<RPROT> to Level-1. This applies the read protection immediately. However using this method, the read protection is enabled temporarily and disabled after system reset.
3. If the read protection is enabled in the code flash memory, it is applied to the data flash memory too. To disable the read protection at this condition, the password match in the code flash memory or elimination of Read Protection OTP area is required. If the read protection was enabled in the code flash memory and the data flash memory respectively, it needs to be disabled in each area respectively.

Read protection modes

The read protection operates in two modes, and it is irrespective of BOOT mode.

- Normal: Unlimited access to the internal flash memory
- Level-1: Debug connection and password function are available. When the internal memory is accessed from outside, read output is 0x55AA55AA.

Table 43. Read Protection Mode of Data Flash Memory

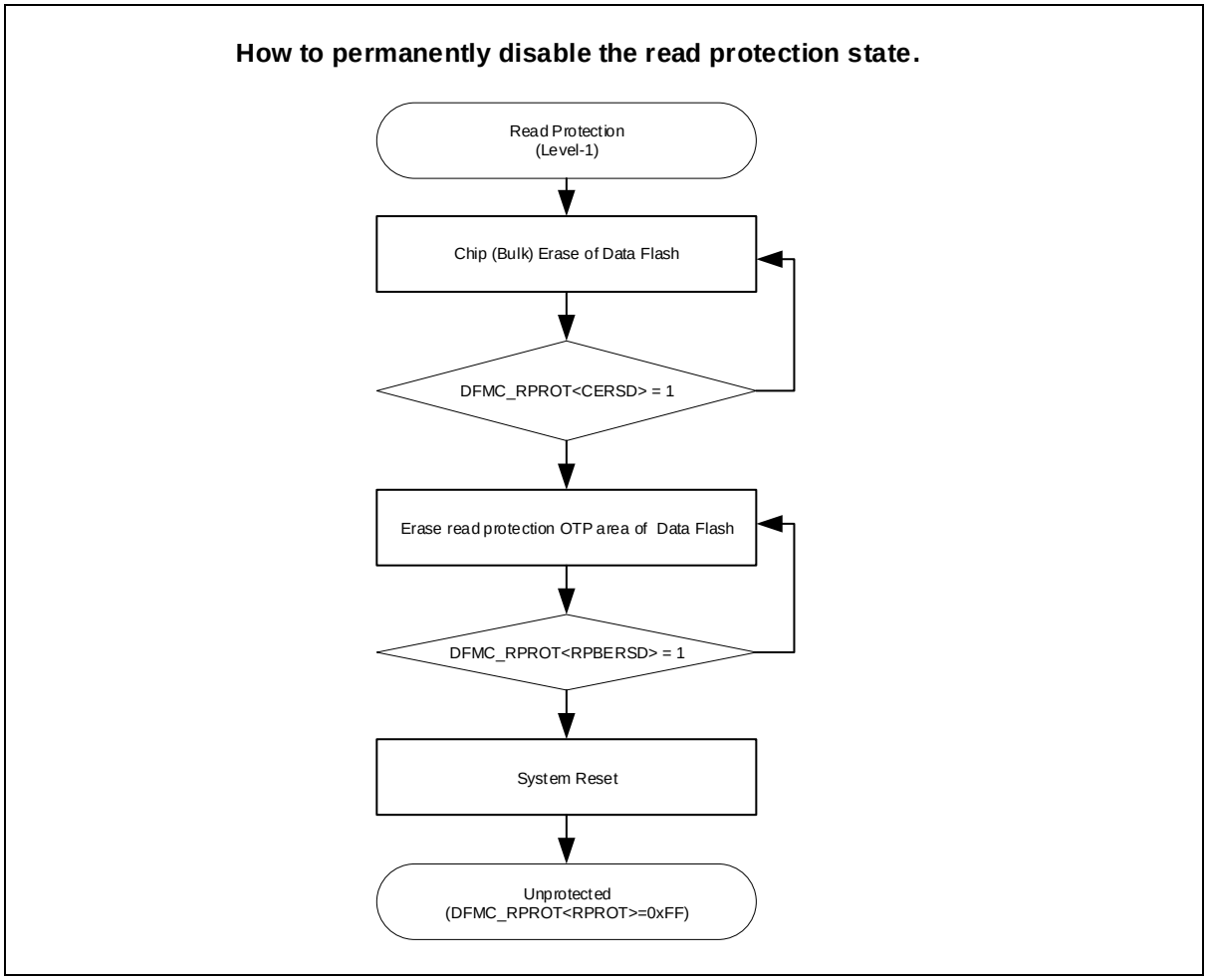
Read Protection Mode	Mode	Code value	Description
Unlock	-	0xFF	Debug and read access are available.
Level-1	Password	Others	Debug is available, but read output is 0x55AA55AA. In case of password matching, read access is available.

Read protection release

To disable the read protection, a user must reset the system after removing values in Read Protection OTP area. In the system where the read protection Level-1 is applied, the internal data flash memory must be cleared by Chip (Bulk) Erase.

The Chip Erase enables DFMC_RPROT<CERSD> bit of Read Protection register, and the enabled bit maintains its state until the system reset occurs. That is, the data flash memory protected by the read protection cannot be accessed by debug or external interfaces such as UART. Since the read protection cannot be disabled until the whole data flash memory is cleared, user data in the memory is safely protected.

Figure 45. Sequence Disabling Read Protection Permanently



Password and the read protection

Even after applying the read protection, it is frequently required to reflect requirements for the code updates to the system. For this, A31G22x provides a password function to release the read protection temporarily.

For the password function, Preset register has predefined password, and Input register receives new password from a user and compares it and the Preset register. The Preset register is specified only once after the system initialization, and input values after the specification are ignored.

When the specified password in the Preset register is identical to the input value in the Input register, DFMC_RPROT<PWMATCH> flag is set to '1' and whole data flash memory enters in Normal mode temporarily to allow debug and external interfaces such as UART.

In this Normal mode when the read protection is temporarily disabled, to enable the read protection again, a user can use one of the following methods:

- System reset
- Enter the level to DFMC_RPROT<RPROT> field directly.
- Enter the password identical to the value of Preset register in Password Input register

Protection boot

If a user wants to store or update user application data in the data flash memory where the read protection is applied already, the user can disable the read protection temporarily by entering preset password. The preset password is predefined in the Preset register.

Before doing this, the preset password and the read protection level must be configured in the form of OTP Commands in Read Protection OTP area of the data flash memory.

8 Internal SRAM

The A31G22x has a on-chip SRAM. The size of SRAM is 20KB.

The SRAM memory area is usually used for data memory and stack memory. Sometimes the code is dumped into the SRAM memory for fast operation or flash erase/program operation.

These SRAM can be accessed as a word (32-bit) alignment. The memory can be addressed at maximum system clock frequency without wait state and thus by both CPU and DMA.

Table 44. Base Address of SRAM

Name	Base address
SRAM	0x2000_0000

9 Direct Memory Access Controller (DMAC)

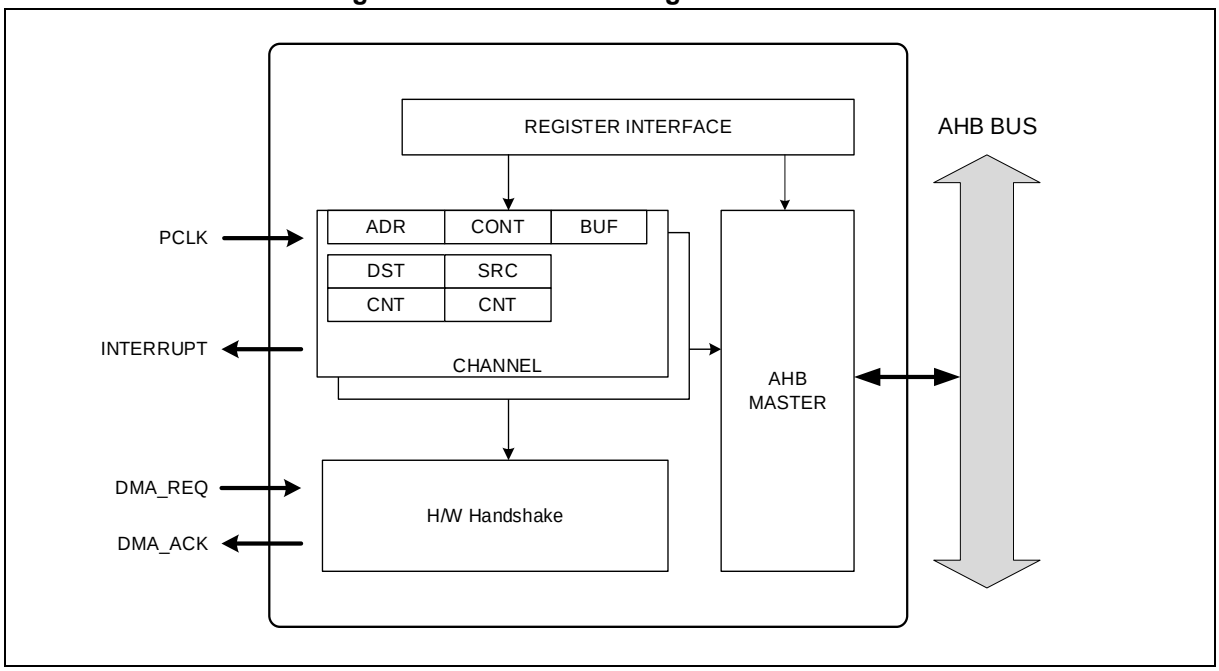
The direct memory access (DMA) controller is used for high-speed data transfers between peripherals and memories. DMA enables quick data transfers having memory to memory copying or moving of data within memory.

- 8 channels
- Single transfer only
- Support 8/16/32-bit data size
- Support multiple buffer with same size
- Interrupt condition is transferred through a peripheral interrupt.
- DMAC peripherals : UARTn Rx/Tx, USART1n Rx/Tx. CRC, SPI2n Rx/Tx, ADC, DAC
- DMAC directions
  - Memory to Peripheral (Tx)
  - Peripheral to Memory (Rx)
  - Peripheral to- Peripheral (P2P)

9.1 Block diagram

In this section, DMAC block diagram is introduced in Figure 46.

Figure 46. DMAC block diagram of A31G22x



9.2 Registers

Base address of DMAC is introduced in the followings:

Table 45. Base Address of DMAC

Name	Base address
DMA0	0x4000_0400
DMA1	0x4000_0410
DMA2	0x4000_0420
DMA3	0x4000_0430
DMA4	0x4000_0440
DMA5	0x4000_0450
DMA6	0x4000_0460
DMA7	0x4000_0470

Table 46. DMAC Register Map (n = 0 to 7)

Name	Offset	Type	Description	Reset value	Ref.
DMAn_CR	0x0000	RW	DMA Channel n Control Register	0x0000_0000	9.2.1
DMAn_SR	0x0004	RW	DMA Channel n Status Register	0x0000_0080	9.2.2
DMAn_PAR	0x0008	RO	DMA Channel n Peripheral Address	0x0000_0000	9.2.3
DMAn_MAR	0x000C	RW	DMA Channel n Memory Address	0x2000_0000	9.2.4

9.2.1 DMAn_CR: DMA controller configuration register

DMAn_CR registers are DMA operation control registers, and the register size is 32-bit.

DMA0_CR=0x4000_0400 , DMA1_CR=0x4000_0410
DMA2_CR=0x4000_0420 , DMA3_CR=0x4000_0430
DMA4_CR=0x4000_0440 , DMA5_CR=0x4000_0450
DMA6_CR=0x4000_0460 , DMA7_CR=0x4000_0470

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0			
Reserved	TRANSCNT	Reserved	PERISEL	Reserved	SIZE	DIR	Reserved
-	0x000	-	0	-	00	0	-
-	RW	-	RW	-	RW	RW	-

Bits	Name	Function	
27 16	TRANSCNT	Number of DMA transfer remained Required transfer number should be written before enable DMA transfer.	
		0	DMA transfer is done.
		N	N transfers are remained.
12 8	PERISEL	Peripheral selection	
		N	Associated peripheral selection. Refer to DMA Peripheral connection Table 47
3 2	SIZE	Bus transfer size.	
		00	DMA transfer is byte size transfer
		01	DMA transfer is half word size transfer
		10	DMA transfer is word size transfer
		11	Reserved
1	DIR	Select transfer direction.	
		0	Transfer direction is from memory to peripheral. (TX)
		1	Transfer direction is from peripheral to memory (RX)

NOTE:

1. A DMA channel will be connected with selected peripheral. Table 47 shows peripheral selection numbers. This PERISEL field should be set with proper number of peripheral which will be connected with DMA interface.
2. PERISEL[12:8] cannot have the same value in different channels. If the same PERISEL value is written in more than one channel, proper operation cannot be guaranteed. Unused channel should have CHANNEL IDLE value in PERISEL bit positions.

Table 47. DMAC PERISEL Selection of A31G22x

PERISEL[12]	PERISEL2[11:8]	Associated peripheral	PERISEL[12]	PERISEL[11:8]	associated peripheral
0	0	CHANNEL IDLE	1	0	CHANNEL IDLE
0	1	UART0 RX	1	1	USART10 Rx
0	2	UART0 TX	1	2	USART10 Tx
0	3	UART1 RX	1	3	USART11 Rx
0	4	UART1 TX	1	4	USART11 Tx
0	5	CRC (RAM Only)	1	5	USART12 Rx
0	6	N/A	1	6	USART12 Tx
0	7	N/A	1	7	USART13 Rx
0	8	N/A	1	8	USART13 Tx
0	9	SPI20 RX	1	9	N/A
0	10	SPI20 TX	1	10	N/A
0	11	SPI21 RX	1	11	N/A
0	12	SPI21 TX	1	12	N/A
0	13	ADC	1	13	N/A
0	14	DAC	1	14	N/A
0	15	N/A	1	15	N/A

9.2.2 DMAn_SR: DMA controller status register

DCn.SR registers represent current status of DMA Controller, and enable DMA function. The register size is 8-bit.

DMA0_SR=0x4000_0404 , DMA1_SR=0x4000_0414
DMA2_SR=0x4000_0424 , DMA3_SR=0x4000_0434
DMA4_SR=0x4000_0444 , DMA5_SR=0x4000_0454
DMA6_SR=0x4000_0464 , DMA7_SR=0x4000_0474

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved	EOT	Reserved		DMARC	Reserved			DMAEN
-	1	-			-			0
-	RO	-		WC	-			RW

Bits	Name	Function	
7	EOT	End of transfer.	
		0	Data to be transferred is existing. TRANSCNT shows non zero value
		1	All data is transferred. TRANSCNT shows now 0
4	DMARC	DMA request clear (Auto clear)	
		0	None
		1	Initializes DMA
0	DMAEN	DMA Enable	
		0	DMA is in stop or hold state.
		1	DMA is running or enabled.

NOTE: If you are using a DMA channel, you must set the DMA request clear bit to '1' to initialize the DMA channel.

```
DMAn_SR<DMARC[4]> = 1     // DMA initialization
DMAn_SR<DMAEN[0]> = 1     // DMA Enable
```

9.2.3 DMAn_PAR: DMA controller peripheral address register

DMAn_PAR registers represent peripheral addresses.

DMA0_PAR=0x4000_0408 , DMA1_PAR=0x4000_0418
DMA2_PAR=0x4000_0428 , DMA3_PAR=0x4000_0438
DMA4_PAR=0x4000_0448 , DMA5_PAR=0x4000_0458
DMA6_PAR=0x4000_0468 , DMA7_PAR=0x4000_0478

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Peripheral Base address Offset		PAR	
0x4000		0x0000	
RO		RW	

Bits	Name	Function
15 0	PAR	Target Peripheral address of transmit buffer or receive buffer. User must set exact target peripheral buffer address in this field. If DMAn_CR<DIR[0]> is "0", this address is destination address of data transfer. If DMAn_CR<DIR[0]> is "1", this address is source address of data transfer.

9.2.4 DMAn_MAR: DMA controller memory address register

DMAn_MAR registers represent the memory addresses.

DMA0_MAR=0x4000_040C , DMA1_MAR=0x4000_041C
DMA2_MAR=0x4000_042C , DMA3_MAR=0x4000_043C
DMA4_MAR=0x4000_044C , DMA5_MAR=0x4000_045C
DMA6_MAR=0x4000_046C , DMA7_MAR=0x4000_047C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Memory Base address Offset		MAR	
0x2000		0x0000	
RO		RW	

Bits	Name	Function
15 0	MAR	Target memory address of data transfer. Address is automatically incremented according to SIZE bits when each transfer is done. If DMAn_CR<DIR[0]> is "0", this address is source address of data transfer. If DMAn_CR<DIR[0]> is "1", this address is destination address of data transfer.

NOTE: To use Peri-to-Peri DMA, enter the address of the write-accessible peripheral's register to the MAR[31:0].

9.3 Functional description

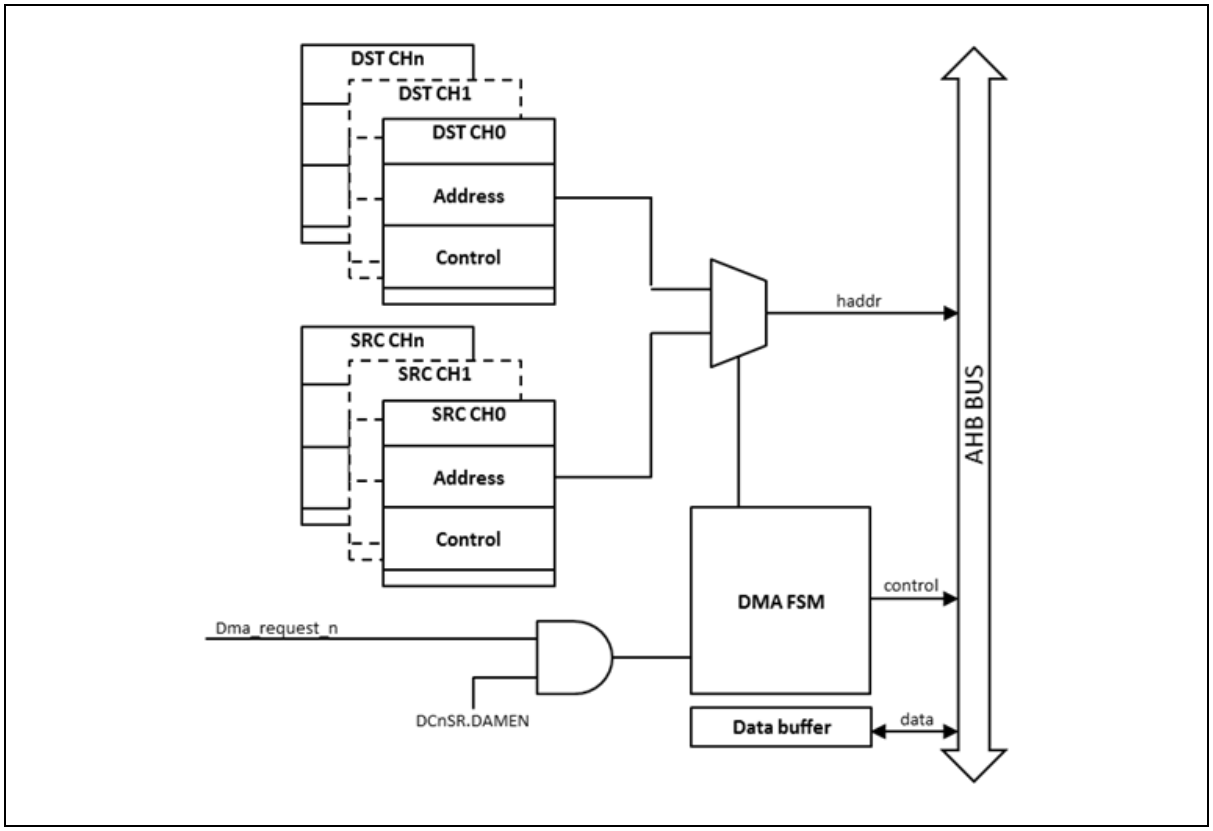
A DMA controller performs direct memory transfer by sharing the system bus with CPU core. The system bus is shared by two AHB (Advanced High-performance Bus) masters following the round-robin priority strategy. So the DMA controller can share the half of system bandwidth.

The DMA controller can be triggered only by a peripheral request. When a peripheral requests a transfer to the DMA controller, a corresponding channel is activated and the bus is accessed to transfer the requested data from memory to peripheral data buffer or vice versa.

Basic steps to trigger DMAC data transfer consist of following 10 steps:

1. Set both of peripheral address and memory address.
2. Configure DMA operation mode and transfer count.
3. Enable a DMA channel.
4. DMA request is occurred from the peripheral.
5. DMA activates the channel which was requested.
6. DMA reads data from source address and saves in internal buffer.
7. DMA writes the buffered data to destination address.
8. Transfer count number is decreased by '1'.
9. When the transfer count is '0', EOT flag is set and noticed to peripheral to issue the interrupt
10. DMA does not have interrupt sources, and the interrupt related DMA status can be shown from assigned peripheral interrupt.

Figure 47. DMAC Functional Block Diagram



9.3.1 DMA operation

A user can start DMA operation by following the procedure introduced below:

1. Set DMAn_CR registers of DMA.
   - [27:16]: Set a number of data to transfer to DMA.
   - [12:8]: Select a peripheral to connect with DMA.
   - [3:2]: Select a buffer size to transfer.
   - [1]: Set a transfer type from TX and RX.
2. Set MAR register of DMA (memory address to which DMA accesses).
3. Set PAR register of DMA (peripheral address to which DMA accesses).
4. Check the EOT flag of DMA start and DMAn_SR register.
5. Check the DMA flag through status register of each peripheral.
6. DMA stops.

A sequence of DMAC operation is described in Figure 48.

Figure 48. DMAC Operation Sequence

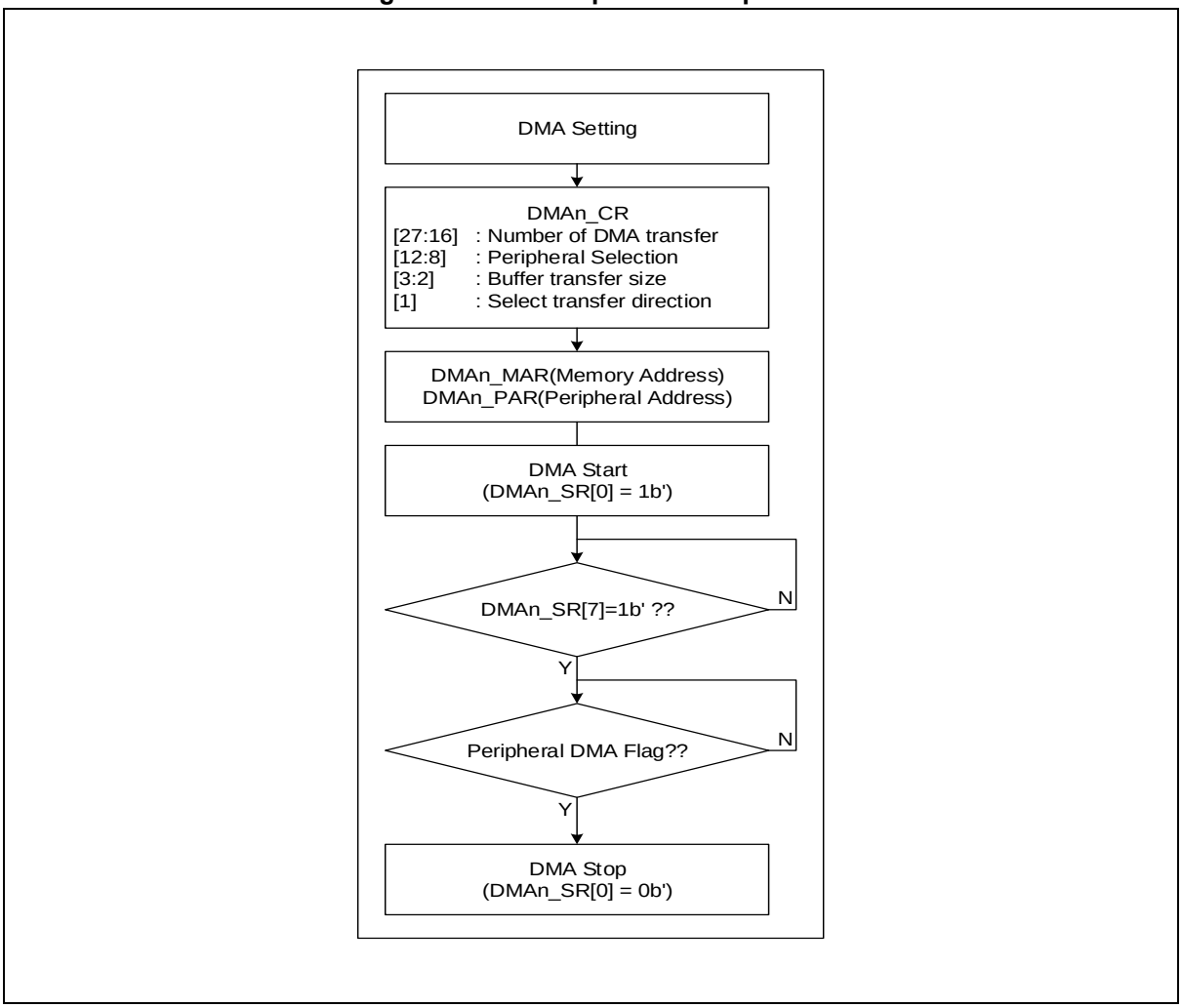


Figure 49 shows the functional timing diagram of DMAC. Transfer request from a certain peripheral is pended internally and it will invoke source data read transfer on the AHB bus. The read data from the source address is stored in the internal buffer. Then this data will be transferred to the destination address when the AHB bus is available.

Figure 49 introduces the timing diagram for a DMA transfer from peripheral to memory. There is 4-clock cycle latencies during accessing the peripheral. If the bus is occupied by different bus master, there are amount of bus waiting cycles.

Figure 49. Timing Diagram of DMAC Transfer from Peripheral to Memory

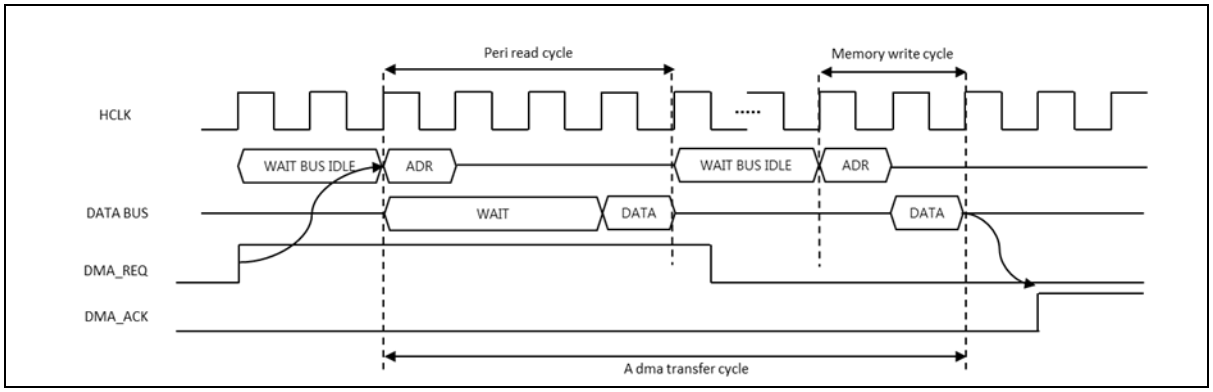


Figure 50 introduces the timing diagram for a DMA transfer from memory to peripheral. There is 4-clock cycle latencies during accessing the peripheral. If the bus is occupied by different bus master, there are amount of bus waiting cycles.

Figure 50. Timing Diagram of DMAC Transfer from Memory to Peripheral

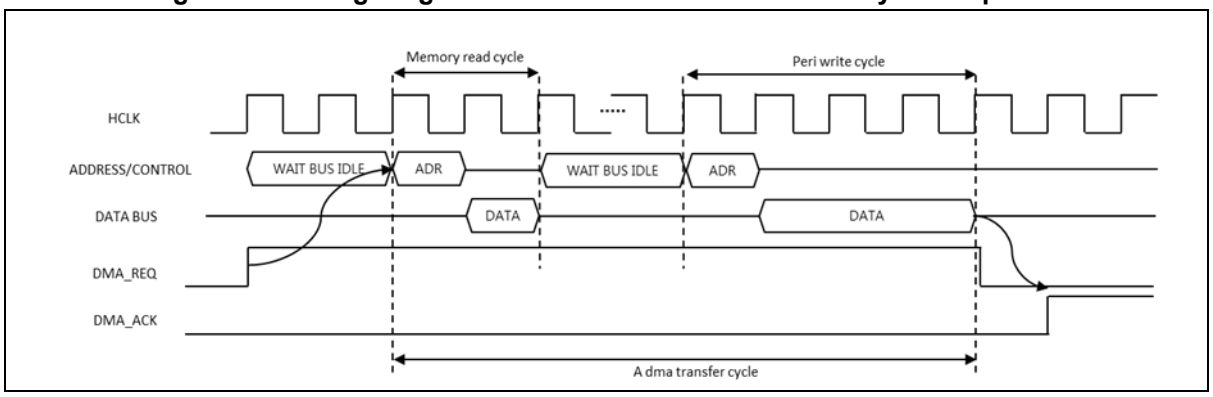
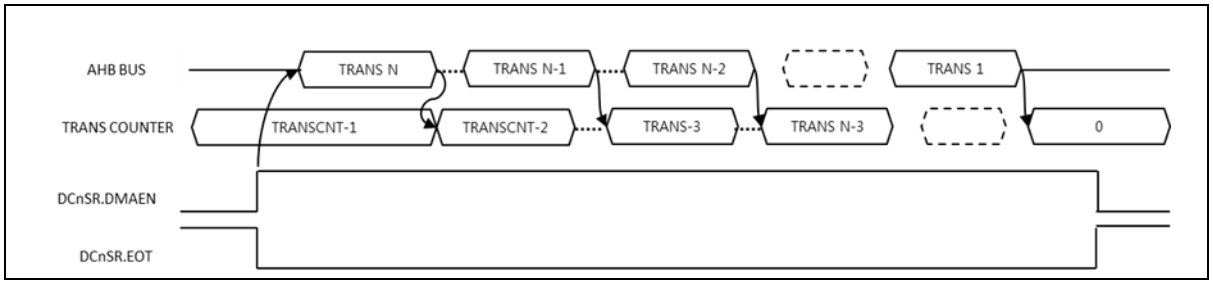


Figure 51 introduces an example of N data transfers with DMA. The DMA transfer is started when DMAn_SR<DMAEN> bit is set. When all the number of transfer is completed, the DMAn_SR<DMAEN> bit will be cleared.

Figure 51. Timing Diagram Example of N DMAC Transfer



10 Watchdog Timer (WDT)

Watchdog timer (WDT) rapidly detects CPU's malfunction such as endless looping caused by noise, and resumes the CPU to the normal state. WDT signal for malfunction detection can be used as either a CPU reset or an interrupt request. When WDT is not being used for malfunction detection, it can be used as a timer generating an interrupt at fixed intervals.

When WDT_CNT value is reached to WDT_WINDR value, a watchdog interrupt can be generated. The underflow time of WDT can be set by WDT_DR. If the underflow occurs, an internal reset is generated. WDT operates on the WDTRC embedded RC oscillator clock.

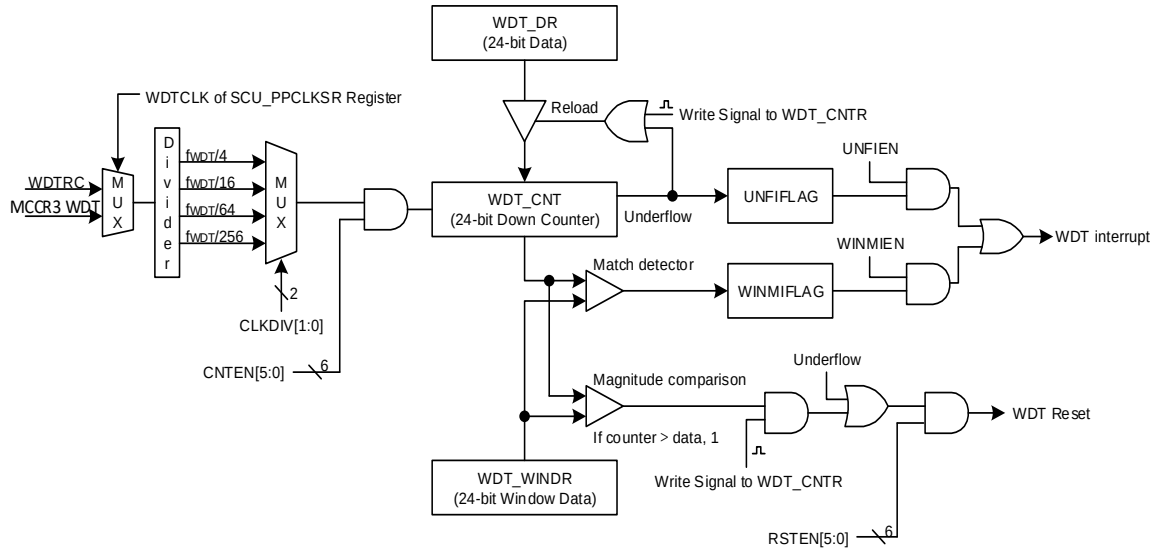
WDT of A31G22x series features followings:

- 24-bit down counter (WDT_CNT)
- Select reset or periodic interrupt
- Count clock selection
- Watchdog overflow output signal
- Counter window function

10.1 WDT block diagram

In this section, WDT block diagram is introduced in Figure 52.

Figure 52. WDT Block Diagram



NOTE: If a reload signal is generated, the WDT_DR value is reloaded to WDT_CNT register after 1 clock delay with the clock which is selected by the CLKDIV bits.

10.2 Registers

Initial watchdog time-out period is set to 2,000-milisecond. Base address of WDT is introduced in the followings:

Table 48. Base Address of WDT

Name	Base address
WDT	0x4000_1A00

Table 49. WDT Register Map

Name	Offset	Type	Description	Reset value	Ref.
WDT_CR	0x0000	RW	Watch-dog Timer Control Register	0x0000_0000	10.2.1
WDT_SR	0x0004	RW	Watch-dog Timer Status Register	0x0000_0080	10.2.2
WDT_DR	0x0008	RW	Watch-dog Timer Data Register	0x0000_3D09	10.2.3
WDT_CNT	0x000C	RO	Watch-dog Timer Counter Register	0x0000_3D09*	10.2.4
WDT_WINDR	0x0010	RW	Watch-dog Timer Window Data Register	0x0000_FFFF	10.2.5
WDT_CNTR	0x0014	WO	Watch-dog Timer Counter Reload Register	0x0000_0000	10.2.6

NOTE: The initial value of WDT_CNT is set equal to the initial value of the WDT DR register, but WDT is enabled by default in the system boot procedure, reading the WDT_CNT register counter value immediately after booting may result in different initial values.

10.2.1 WDT_CR: Watchdog timer control register

WDT module should be configured properly before running. WDT module can make reset event or assert interrupt signal to system. Size of this register is 32-bit.

WDT_CR=0x4000_1A00

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3	2	1 0
WTIDKY		RSTEN	CNTEN	WINMIEN	UNFIEN	CLKDIV
0x0000		000000	000000	0	0	00
WO		RW	RW	RW	RW	RW

Bits	Name	Function	
31 16	WTIDKY	Write identification key.	
		On writes, write 0x5A69 to these bits, otherwise the write is ignored.	
15 10	RSTEN	Watch-dog timer reset enable bits.	
		0x25	Disable watch-dog timer reset.
		Others	Enable watch-dog timer reset.
9 4	CNTEN	Watch-dog timer counter enable bits.	
		0x1A	Disable watch-dog timer counter.
		Others	Enable window data match interrupt.
3	WINMIEN	Watch-dog timer window match interrupt enable bit.	
		0	Disable window data match interrupt.
		1	Enable window data match interrupt.
2	UNFIEN	Watch-dog timer underflow interrupt enable bit.	
		0	Disable watch-dog timer underflow interrupt.
		1	Enable watch-dog timer underflow interrupt.
1 0	CLKDIV	Watch-dog timer clock divider bits, The clock which is selected by SCU_PPCLKSR[0].	
		00	f_{WDT}/4
		01	f_{WDT}/16
		10	f_{WDT}/64
		11	f_{WDT}/256

10.2.2 WDT_SR: Watchdog timer status register

WDT_SR register is 32-bit size, and able to do 32/16/8-bit access.

WDT_SR=0x4000_1A04

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6 5 4 3 2	1	0
Reserved			DBGCNTEN	Reserved	WINMIFLAG	UNFIFLAG
-			1	-	0	0
-			RW	-	RW	RW

Bits	Name	Function
7	DBGCNTEN	Watch-dog timer counter enable bit When the core is halted in the debug mode.
	0	The watch-dog timer counter continues even if the core is halted.
	1	The watch-dog timer counter is stopped when the core is halted.
		NOTE: This bit is set to "1b" by POR reset.
1	WINMIFLAG	Watch-dog timer window match interrupt flag bit.
	0	No request occurred.
	1	Request occurred, This bit is cleared to '0' when write '1'.
0	UNFIFLAG	Watch-dog timer underflow interrupt flag bit.
	0	No request occurred.
	1	Request occurred, This bit is cleared to '0' when write '1.
NOTE: Window match flag of WDT is recommended for System Clock(fx) ≥ 2*WDT Clock(WDT)		

10.2.3 WDT_DR: Watchdog timer data register

WDT_DR register is used to update WDT_CNT register. Size of this register is 32-bit.s

WDT_DR=0x4000_1A08

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	DATA		
-	0x003D09		
-	RW		

Bits	Name	Function
23 0	DATA	Watch-dog timer data bits. The range is 0x000000 to 0xFFFFFF.
NOTE: Once any value is written to this data register, the register can't be changed until a system reset.		

10.2.4 WDT_CNT: Watchdog timer counter register

WDT_CNT register represents the current count value of 32-bit down counter. When the counter value reach to 0, the interrupt or reset will be asserted. Size of this register is 32-bit.

WDT_CNT=0x4000_1A0C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	CNT		
-	0x003D09		
-	RO		

Bits	Name	Function
23 0	CNT	Watch-dog timer counter bits.

10.2.5 WDT_WINDR: Watchdog timer window data register

WDT_WINDR register is used to compare with WDT_CNT for window function. If the WDT_CNT value equals the WDT_WINDR value, WDT window match interrupt is occurred. If WDT_CNT is updated by WDT_CNTR when WDT_CNT value is bigger than WDT_WINDR value, WDT Reset is occurred.

WDT_WINDR=0x4000_1A10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	WDATA		
-	0x00FFFF		
-	RW		

Bits	Name	Function
23 0	WDATA	Watch-dog timer data bits. The range is 0x000000 to 0xFFFFFF.

NOTE: Once any value is written to this data register, the register can't be changed until a system reset.

10.2.6 WDT_CNTR: Watchdog timer counter reload register

WDT_CNTR register is used to make reload signal. If reload signal is I, WDT_DR Value is reloaded to WDT_CNT. Size of this register is 32-bit.

WDT_CNTR=0x4000_1A14

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			CNTR
-			0x00
-			WO

Bits	Name	Function	
7 0	CNTR	Watch-dog timer counter reload bits.	
		0x6A	Reload the WDTDR value to watch-dog timer counter and re-start. (Automatically cleared to "0x00" after operation)
		Others	No effect

10.3 Functional description

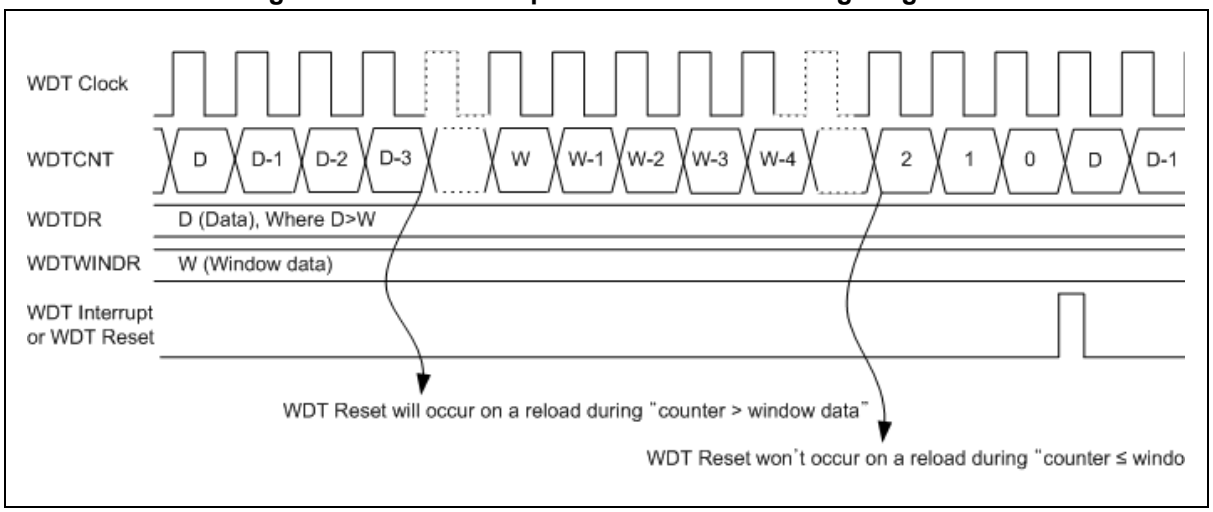
WDT counter can be enabled by CNTEN (WDT_CR[9:4]). Corresponding bit field of the register CNTEN is set with any value other than 0x1A. As WDT activates, a down counter will start counting from loaded value.

If RSTEN (WDT_CR[15:10]) is set with any value other than 0x25, WDT reset would be asserted when the WDT counter value reaches to '0' (underflow event) from WDT_DR value. Before WDT counter reaches to '0', software can write 0x6A to WDT_CNTR register in order to reload WDT counter when the counter value is less than or equal to the value of window data register. WDT reset may be asserted if the reload occurs when counter is greater than window data.

10.3.1 Timing diagram

In this section, WDT interrupt and reset timing diagram is introduced in Figure 53.

Figure 53. WDT Interrupt and WDT Reset Timing Diagram



10.3.2 Prescale table

WDT includes a 24-bit down counter with programmable pre-scaler to define different time-out intervals.

Clock sources of WDT can be WDT_RC or PCLK. PCLK can be selected by WDTCLK (SCU_PPCLKSR[0]) which is set to '1', then the WDTCNFIG[2] bit of configure option page 1 is cleared to logic "0b".

To configure a WDT counter as a base clock, user can control 2-bit pre-scaler CLKDIV [1:0] in the WDT_CR register, and the maximum pre-scaled value is "clock source frequency/256". The pre-scaled WDT counter clock frequency values are listed in the following table.

Selectable clock source (40KHz to 48MHz) and time-out interval at a single count

Time-out period = (Load Value + 1) * (1/pre-scaled WDT counter clock frequency)

*Time out period (when the Load Value reaches 0, underflow flag is set to '1')

Table 50. Pre-scaled WDT Counter Clock Frequency

Clock source	WDTCLKIN	WDTCLKIN/4	WDTCLKIN/16	WDTCLKIN/64	WDTCLKIN/256
WDTRC	31.250KHz	7.8125KHz	1.953125KHz	488.28125Hz	122.0703125Hz
MCCR3 WDT	MCCR3 WDT	MCCR3 WDT/4	MCCR3 WDT/16	MCCR3 WDT/64	MCCR3 WDT/256

11 Watch Timer (WT)

Watch Timer (WT) has functions for RTC (Real Time Clock) operation. It is generally used for RTC design. WT consists of a clock source select block, a timer counter block, an output select block and watch timer control registers.

Prior to operate WT, a user needs to determine an input clock source and output interval, and to set WTEN as '1' in watch timer control register (WT_CR). It is able to execute simultaneously or individually. To stop or reset WT, clear the WTEN bit in WT_CR register.

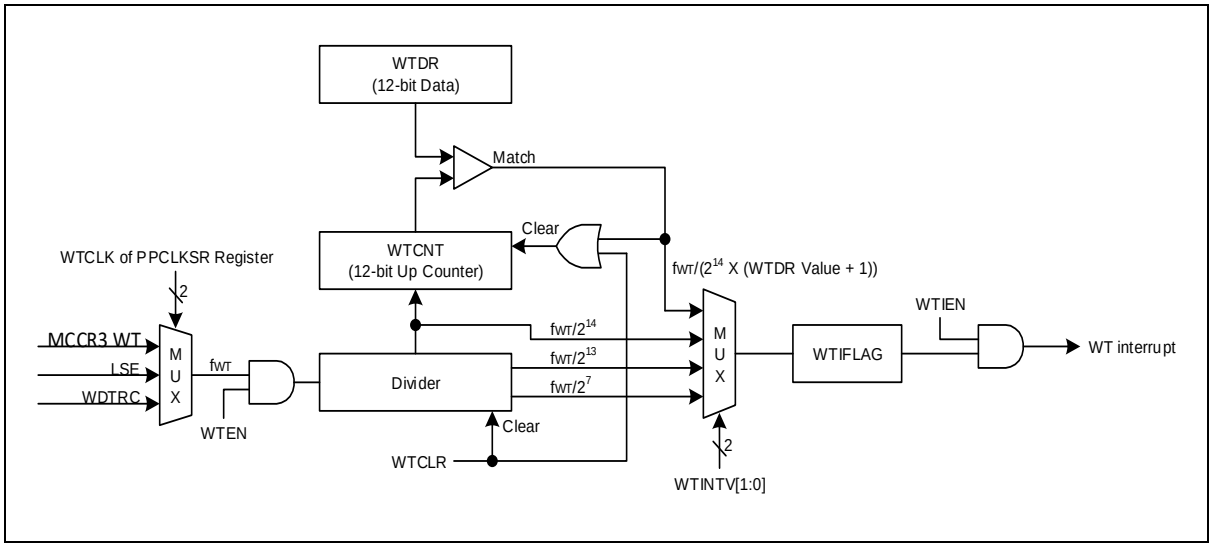
Watch timer counter block corporates a 12-bit up counter. In WT control register, it can control WT clear and set interval value at write time, and it can read 12-bit WT counter value at read time.

11.1 WT block diagram

As shown in Figure 54, WT of A31G22x series has the following blocks:

- 14-bit divider
- 12-bit up-counter
- RTC function

Figure 54. Watch Timer Block Diagram



11.2 Registers

Base address of WT is introduced in the followings:

Table 51. Base Address of WT

Name	Base address
WT	0x4000_2000

Table 52. WT Register Map

Name	Offset	Type	Description	Reset value	Ref.
WT_CR	0x0000	RW	Watch Timer Control Register	0x0000_0000	11.2.1
WT_DR	0x0004	RW	Watch Timer Data Register	0x0000_0FFF	11.2.2
WT_CNT	0x0008	RO	Watch Timer Counter Register	0x0000_0000	11.2.3

11.2.1 WT_CR: watch timer control register

WT_CR is a 32-bit register, and able to do 32/16/8-bit access.

WT_CR=0x4000_2000

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5 4	3	2	1	0
Reserved				WTEN	WTINTV	WTIEN	Reserved	WTIFLAG	WTCLR
-				0	00	0	-	0	0
-				RW	RW	RW	-	RW	RW

Bits	Name	Function	
7	WTEN	Watch timer operation enable bit.	
		0	Disable watch timer operation.
		1	Enable watch timer operation.
5 4	WTINTV	Watch Timer Interval Selection bits.	
		00	$fWT/2^7$ (Overflow Interrupt)
		01	$fWT/2^{13}$
		10	$fWT/2^{14}$
		11	$fWT/(2^{14}x(WTDR\ value + 1))$
		NOTE: These bits should be changed during WTEN bit is "0b".	
3	WTIEN	Watch timer interrupt enable bit.	
		0	Disable watch timer interrupt.
		1	Enable watch timer interrupt.
1	WTIFLAG	Watch timer interrupt flag bit.	
		0	No request occurred.
		1	Request occurred, this bit is cleared to '0' when write '1'.
0	WTCLR	Watch timer counter and divider clear bit.	
		0	No effect.
		1	Clear the counter and divider (Automatically cleared to "0b" after operation)

11.2.2 WT_DR: watch timer data register

WT_DR is a 32-bit register.

WT_DR=0x4000_2004

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			WTDATA
-			0xFFF
-			RW

Bits	Name	Function
11 0	WTDATA	Watch timer data bits. The range is 0x001 to 0xfff.

11.2.3 WT_CNT: watch timer counter register

WT_CNT is a 32-bit register.

WT_CNT=0x4000_2008

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			CNT
-			0x000
-			RO

Bits	Name	Function
11 0	CNT	Watch timer counter bits.

12 16-bit timer

16-bit timer block comprises 7 channels of 16-bit general purpose timers. Each channel has an independent 16-bit counter and a dedicated prescaler that feeds a counting clock. 16-bit timer supports periodic timer, PWM pulse, one-shot and capture mode. In addition, one more optional free-run timer is provided. Main purpose of 16-bit timer is a periodical tick timer.

16-bit timer of A31G22x series features the followings:

- 16-bit up-counter
- Periodic timer mode
- One-shot timer mode
- PWM pulse mode
- Capture mode
- 12-bit prescaler
- Synchronous start and clear function with TIMER30

Table 53 introduces pins assigned for 16-bit timer.

Table 53. Pin Assignment of 16-bit Timer: External Pins

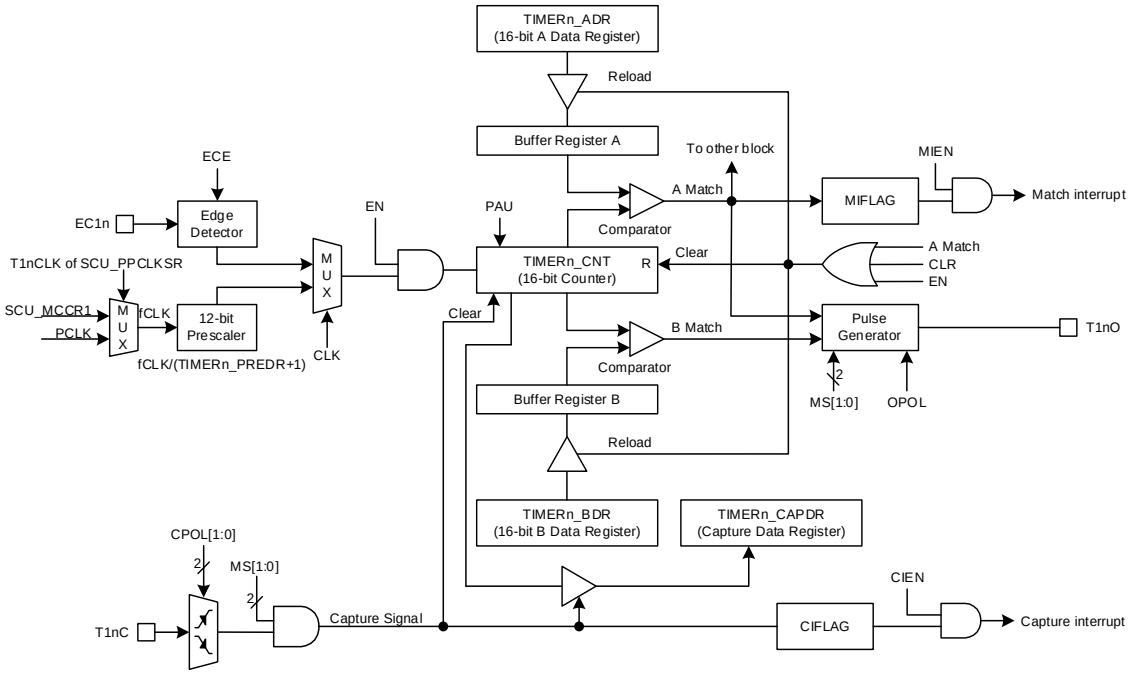
Pin name	Type	Description
EC1n	I	Timer 1n External Clock input
T1nC	I	Timer 1n Capture input
T1nO	O	Timer 1n Output

NOTE: n = 0, 1, 2, 3, 4, 5 and 6

12.1 16-bit timer block diagram

In this section, 16-bit timer is described in a block diagram in Figure 55.

Figure 55. 16-bit Timer 1n Block Diagram (n=0, 1, 2, 3, 4, 5 and 6)



NOTES:

1. The TnEN is automatically cleared to logic "0" after one pulse is generated at a PPG one-shot mode.
2. After the TnEN bit is set, the counting of TIMERn_CNT may be delayed 2 clocks or more.

12.2 Registers

Base address of 16-bit timer is introduced in the followings:

Table 54. Base Address of 16-bit Timer

Name	Base address
TIMER10	0x4000_2100
TIMER11	0x4000_2200
TIMER12	0x4000_2300
TIMER13	0x4000_2700
TIMER14	0x4000_2800
TIMER15	0x4000_2900
TIMER16	0x4000_2A00

Table 55. TIMER 1n Register Map

Name	Offset	Type	Description	Reset value	Ref.
TIMER1n_CR	0x0000	RW	Timer/Counter 1n Control Register	0x0000_0000	12.2.1
TIMER1n_ADR	0x0004	RW	Timer/Counter 1n A Data Register	0x0000_FFFF	12.2.2
TIMER1n_BDR	0x0008	RW	Timer/Counter 1n B Data Register	0x0000_FFFF	12.2.3
TIMER1n_CAPDR	0x000C	RO	Timer/Counter 1n Capture Data Register	0x0000_0000	12.2.4
TIMER1n_PREDR	0x0010	RW	Timer/Counter 1n Prescaler Data Register	0x0000_0FFF	12.2.5
TIMER1n_CNT	0x0014	RO	Timer/Counter 1n Counter Register	0x0000_0000	12.2.6

NOTE: n = 0, 1, 2, 3, 4, 5 and 6

12.2.1 TIMER1n_CR: timer/counter 1n control register

Timer module should be configured properly before running. When a target purpose is defined, the timer can be configured in the TIMER1n_CR register. After configuring TIMER1n_CR, a user can start or stop the timer function by using TIMER1n_CR.

TIMER1n_CR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0, 1, 2, 3, 4, 5 and 6)

TIMER10_CR=0x4000_2100, TIMER11_CR=0x4000_2200
TIMER12_CR=0x4000_2300, TIMER13_CR=0x4000_2700
TIMER14_CR=0x4000_2800, TIMER15_CR=0x4000_2900
TIMER16_CR=0x4000_2A00

31	30	29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13 12	11	10 9	8	7 6	5	4	3	2	1	0
SSYNC	CSYNC	Reserved		EN	CLK	MS	ECE	Reserved	OPOL	CPOL	MIEN	CIEN	MIFLAG	CIFLAG	PAU	CLR
0	0	-		0	0	00	0	-	0	00	0	0	0	0	0	0
RW	RW	-		RW	RW	RW	RW	-	RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
31	SSYNC	Synchronized start counter with TIMER30 See 12.3.5 for information about SSYNC features.
		0 Single counter mode
		1 Synchronized counter start mode
30	CSYNC	Synchronization clear counter with TIMER30 See 12.3.5 for information about CSYNC features.
		0 Single counter mode
		1 Synchronized counter clear mode
15	EN	Timer 1n operation enable bit.
		0 Disable timer 1n operation.
		1 Enable timer 1n operation. (Counter clear and start)
14	CLK	Timer 1n clock selection bit.
		0 Select an internal prescaler clock.
		1 Select an external clock. (EC1n)
		NOTE: This bit should be changed during T1nEN bit is "0b".
13 12	MS	Timer 1n Operation Mode Selection bits.
		00 Timer/Counter mode. (T1nO: Toggle at A-match)
		01 Capture mode. (The A-match interrupt can occur)
		10 PPG one-shot mode. (T1nO: Programmable pulse output)
		11 PPG repeat mode. (T1nO: Programmable pulse output)
		NOTE: This bit should be changed during TnEN bit is "0b".
11	ECE	Timer 1n external clock edge selection bit. (Input by EC1n pin)
		0 Select falling edge of external clock.
		1 Select rising edge of external clock.
8	OPOL	T1nO polarity selection bit.

		0	Start high. (T1nO is low level at disable)
		1	Start low. (T1nO is high level at disable)
7 6	CPOL	Timer 1n capture polarity selection bits.	
		00	Capture on falling edge.
		01	Capture on rising edge.
		10	Capture on both of falling and rising edge.
		11	Reserved.
5	MIEN	Timer 1n match interrupt enable bit.	
		0	Disable timer 1n match interrupt.
		1	Enable timer 1n match interrupt.
4	CIEN	Timer 1n capture interrupt enable bit.	
		0	Disable timer 1n capture interrupt.
		1	Enable timer 1n capture interrupt.
3	MIFLAG	Timer 1n match interrupt flag bit.	
		0	No request occurred.
		1	Request occurred, this bit is cleared to '0' when write '1'.
2	CIFLAG	Timer 1n capture interrupt flag bit.	
		0	No request occurred.
		1	Request occurred, this bit is cleared to '0' when write '1'.
1	PAU	Timer 1n counter temporary pause control bit.	
		0	Continue counting.
		1	Temporary pause.
0	CLR	Timer 1n counter and prescaler clear bit.	
		0	No effect.
		1	Clear timer 1n counter and prescaler. (Automatically cleared to "0b" after operation)

12.2.2 TIMER1n_ADR: timer/counter 1n A data register

TIMER1n_ADR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0, 1, 2, 3, 4, 5 and 6)

TIMER10_ADR=0x4000_2104, TIMER11_ADR =0x4000_2204
TIMER12_ADR =0x4000_2304, TIMER13_ADR =0x4000_2704
TIMER14_ADR=0x4000_2804, TIMER15_ADR =0x4000_2904
TIMER16_ADR =0x4000_2A04

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		ADATA	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	ADATA	Timer/Counter 1n A data bits. The range is 0x0002 to 0xFFFF. **NOTE:** Do not write "0x0000" in the TIMER1n_ADR register when PPG mode.

12.2.3 TIMER1n_BDR: timer/counter 1n B data register

TIMER1n_BDR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0, 1, 2, 3, 4, 5 and 6)

TIMER10_BDR=0x4000_2108, TIMER11_BDR =0x4000_2208
TIMER12_BDR =0x4000_2308, TIMER13_BDR =0x4000_2708
TIMER14_BDR=0x4000_2808, TIMER15_BDR =0x4000_2908
TIMER16_BDR =0x4000_2A08

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		BDATA	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	BDATA	Timer/Counter 1n B data bits. The range is 0x0000 to 0xFFFF.

12.2.4 TIMER1n_CAPDR: timer/counter 1n capture data register

TIMER1n_ CAPDR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0, 1, 2, 3, 4, 5 and 6)

TIMER10_CAPDR =0x4000_210C, TIMER11_CAPDR =0x4000_220C
TIMER12_CAPDR =0x4000_230C, TIMER13_CAPDR =0x4000_270C
TIMER14_CAPDR =0x4000_280C, TIMER15_CAPDR =0x4000_290C
TIMER16_CAPDR =0x4000_2A0C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CAPD	
-		0x0000	
-		RO	

Bits	Name	Function
15 0	CAPD	Timer/Counter 1n capture data bits.

12.2.5 TIMER1n_PREDR: timer/counter 1n prescaler data register

TIMER1n_PREDR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0, 1, 2, 3, 4, 5 and 6)

TIMER10_PREDR =0x4000_2110, TIMER11_PREDR =0x4000_2210
TIMER12_PREDR =0x4000_2310, TIMER13_PREDR =0x4000_2710
TIMER14_PREDR =0x4000_2810, TIMER15_PREDR =0x4000_2910
TIMER16_PREDR =0x4000_2A10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		PRED	
-		0xFFF	
-		RW	

Bits	Name	Function
11 0	PRED	Timer/Counter 1n prescaler data bits.

12.2.6 TIMER1n_CNT: timer/counter n counter register

TIMER1n_CNT is a 32-bit register, and able to do 32/16/8-bit access. (n = 0, 1, 2, and 3)

TIMER10_CNT =0x4000_2114, TIMER11_CNT =0x4000_2214
TIMER12_CNT =0x4000_2314, TIMER13_CNT =0x4000_2714
TIMER14_CNT =0x4000_2814, TIMER15_CNT =0x4000_2914
TIMER16_CNT =0x4000_2A14

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CNT	
-		0x0000	
-		RO	

Bits	Name	Function
15 0	CNT	Timer/Counter 1n Counter bits.

12.3 Functional description

12.3.1 Timer counter 10/11/12/13/14/15/16

Timer/counter 1n can be clocked by an internal or an external clock source (EC1n). Its clock source is selected by a clock selection logic which is controlled by clock selection bits (T1nCLK). (n = 0, 1, 2, 3, 4, 5 and 6)

- TIMER 1n clock source: {f_{CLK}/(TIMER1n_PREDR +1)}, EC1n

In capture mode, by T1nC channel, data is captured into input capture data register (TIMER1n_CAPDR). TIMER1n results the comparison between a counter and the data register through T1nO port in timer/counter mode. In addition, TIMER 1n outputs PWM wave form through T1nO port in the PPG mode.

Table 56 introduces various operating modes of TIMER 1n according to the value of timer/counter register.

Table 56. TIMER 1n Operating Modes

TIMER1n_CR <EN>	TIMER1n_CR <MS>	TIMER1n_PREDR	Description
1	00	0xXXX	16-bit Timer/Counter Mode
1	01	0xXXX	16-bit Capture Mode
1	10	0xXXX	16-bit PPG Mode(one-shot mode)
1	11	0xXXX	16-bit PPG Mode(repeat mode)

NOTE: n = 0, 1, 2, 3, 4, 5 and 6

12.3.2 16-bit timer/counter mode

16-bit timer/counter mode is selected by control register as shown in Figure 56. The 16-bit timer has a counter and data register. The counter register is increased by internal or external clock input. Timer n can use the clock input with 12-bit prescaler division rates (TIMER1n_PREDR) and external clock (EC1n). When each value of TIMER1n_CNT and TIMERn_ADR are identical in timer 1n, a match signal is generated and the interrupt of Timer 1n occurs. The TIMER1n_CNT values are automatically cleared by the match signal. It can be also cleared by software (T1nCLR).

Figure 56. TIMER 1n Block Diagram in Timer/Counter Mode (n = 0, 1, 2, 3, 4, 5 and 6)

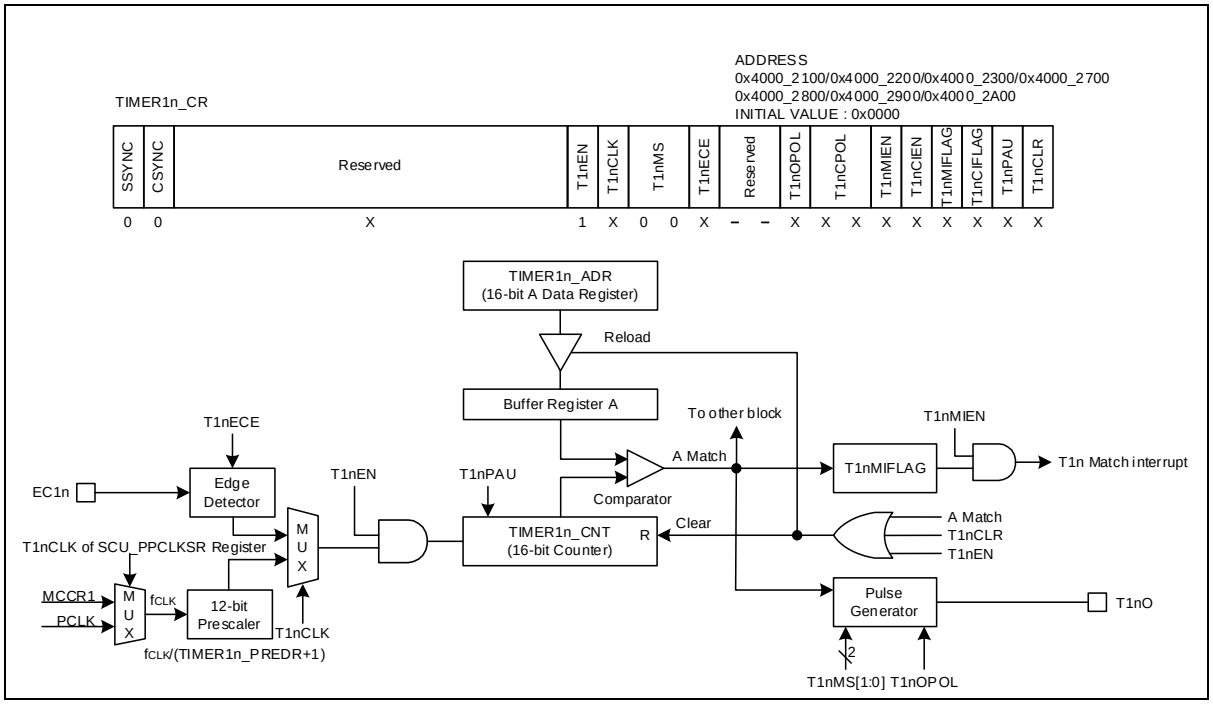


Figure 57. Timer/Counter Mode Timing Example of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)

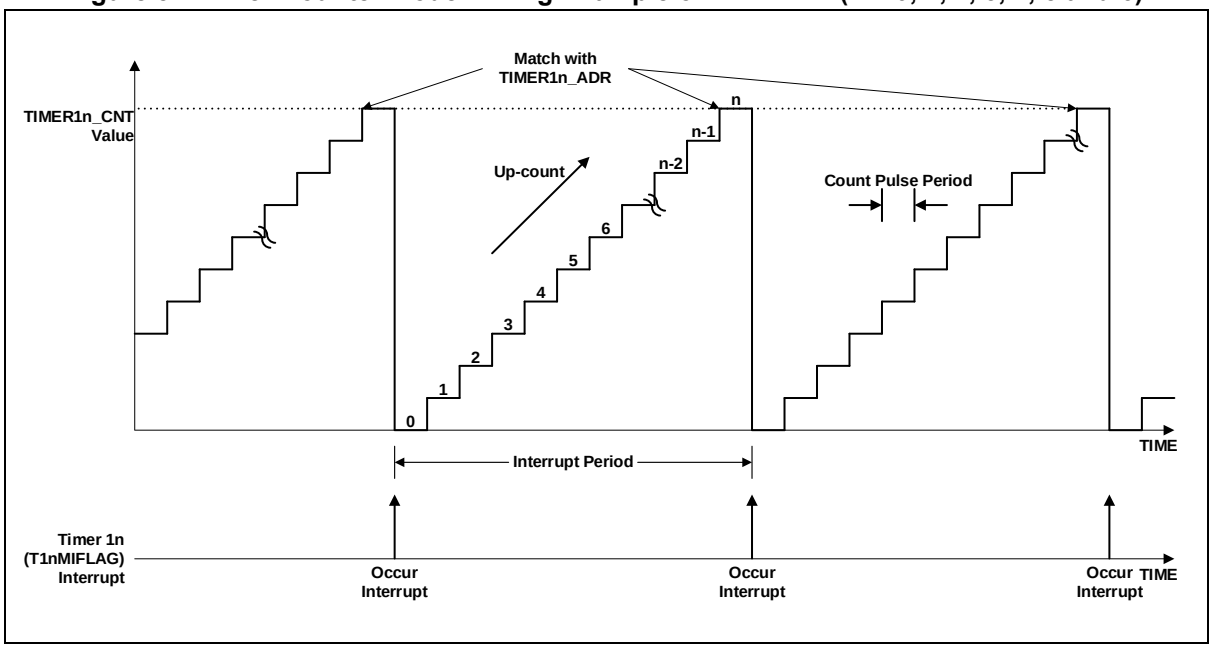
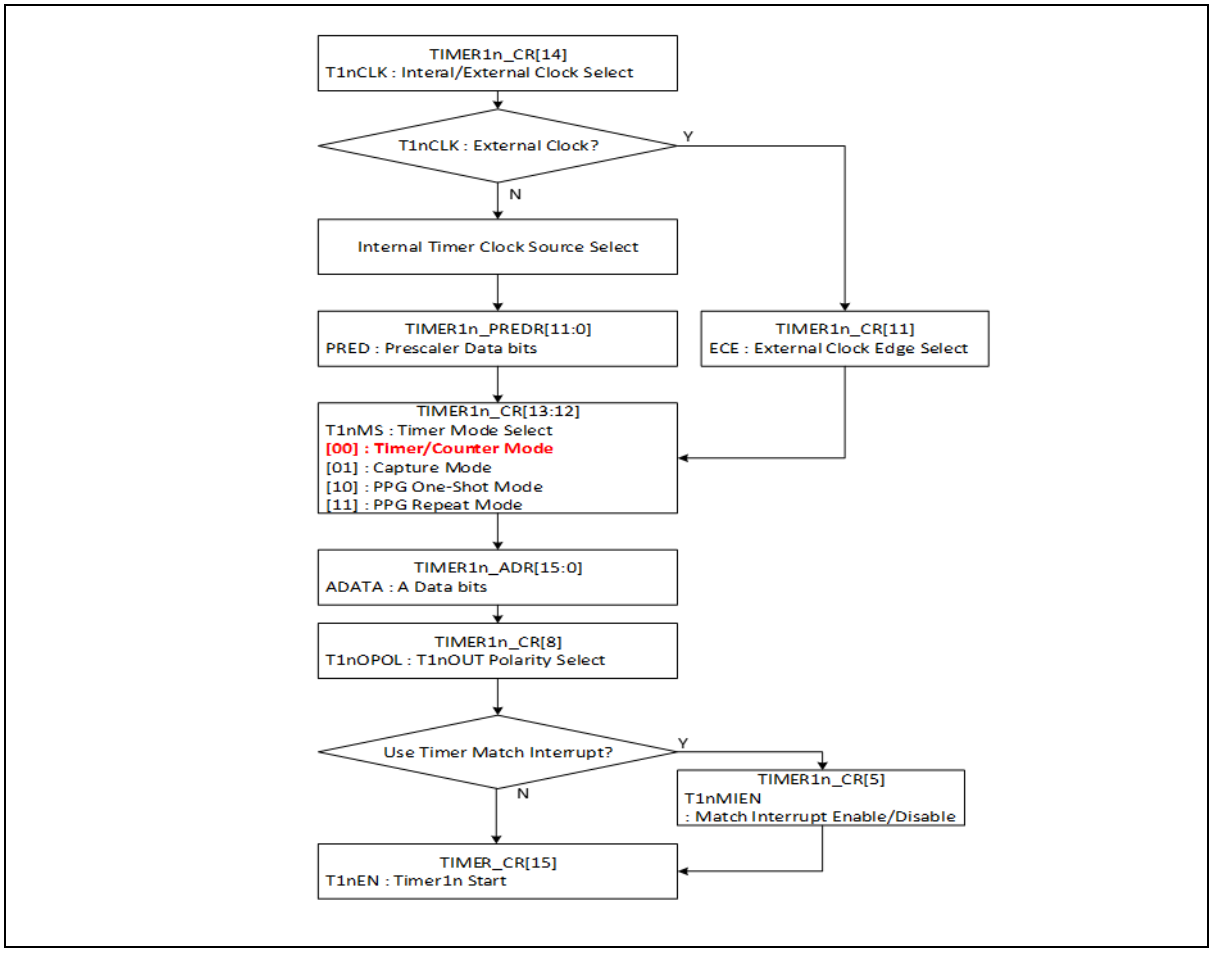


Figure 57 shows the timer/counter mode operation of timer/counter 1n. Refer to "Figure 13. Clock Change Procedure" for information of internal timer clock source and "section 5.2 Pin multiplexing" for information of Timer1n Output pin.

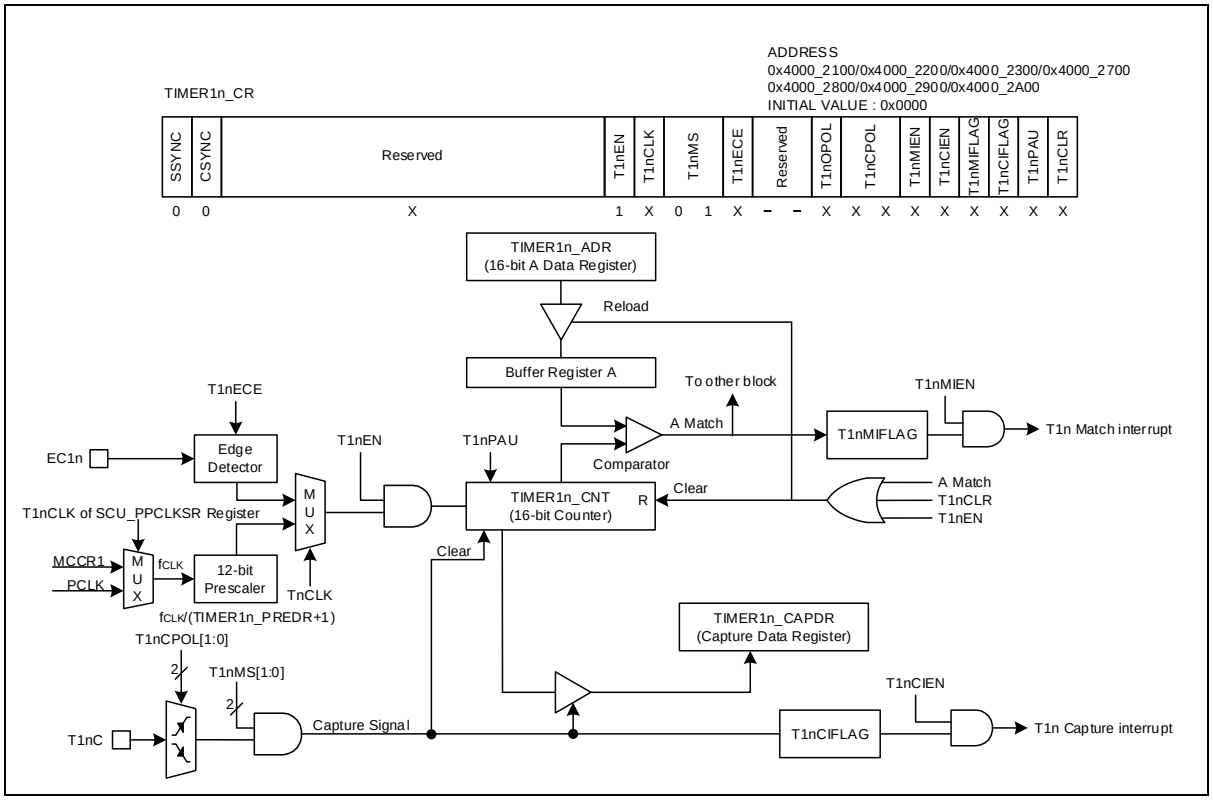
Figure 58. Timer/Counter Mode Operation Sequence of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)



12.3.3 16-bit capture mode

Timer n capture mode is evoked by setting T1nMS[1:0] as '01'. It can use an internal clock as a clock source. Basically, it has the same function as 16-bit timer/counter mode, and the interrupt occurs when TIMER1n_CNT is equal to TIMER1n_ADR. TIMER1n_CNT value can be automatically cleared by a match signal. It can be cleared by software too (TnCLR). A timer interrupt in capture mode is very useful when pulse width of captured signal is wider than the maximum period of the timer. The capture result is loaded into TIMER1n_CAPDR. In the timer n capture mode, timer 1n output (TnO) waveform is not available.

Figure 59. TIMER 1n Block Diagram in Capture Mode (n = 0, 1, 2, 3, 4, 5 and 6)



Capture mode of TIMER 1n can be configured by following the procedure introduced below:

1. Corresponding capture pin for T1nC (Refer to 5.2 Pin multiplexing)
2. Select a clock source for a TIMER 1n block by setting TIMER1n_CR[14] bit.
3. For internal clock: Set TIMER1n_PREDR on reference to Figure 13.
4. For external clock: Set TIMER1n_CR[11] with T1nECE bit in order to select an edge of the external clock.
5. Set T1nMS of TIMER1n_CR[13:12] to 1b' (capture mode) to enable timer capture mode.
6. CPOL in TIMER1n_CR register

Figure 60. Capture Mode Timing Example of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)

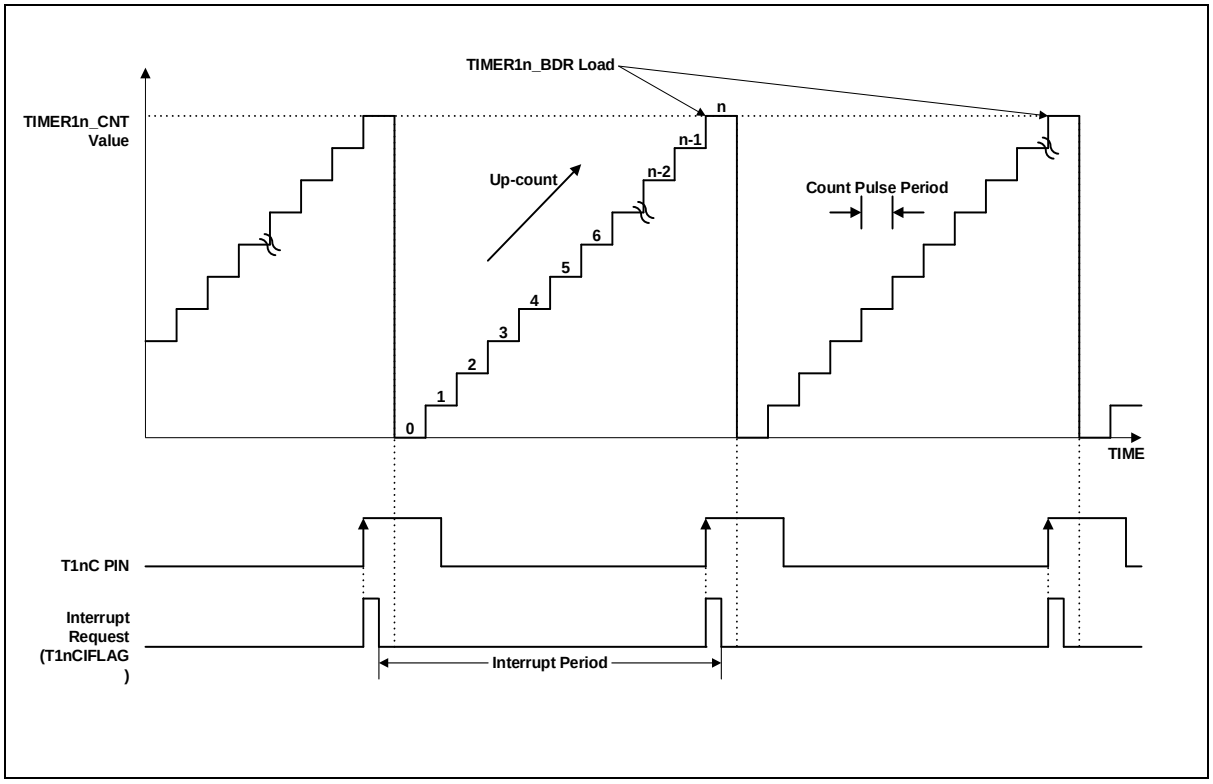


Figure 61. Express Timer Overflow in Capture Mode of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)

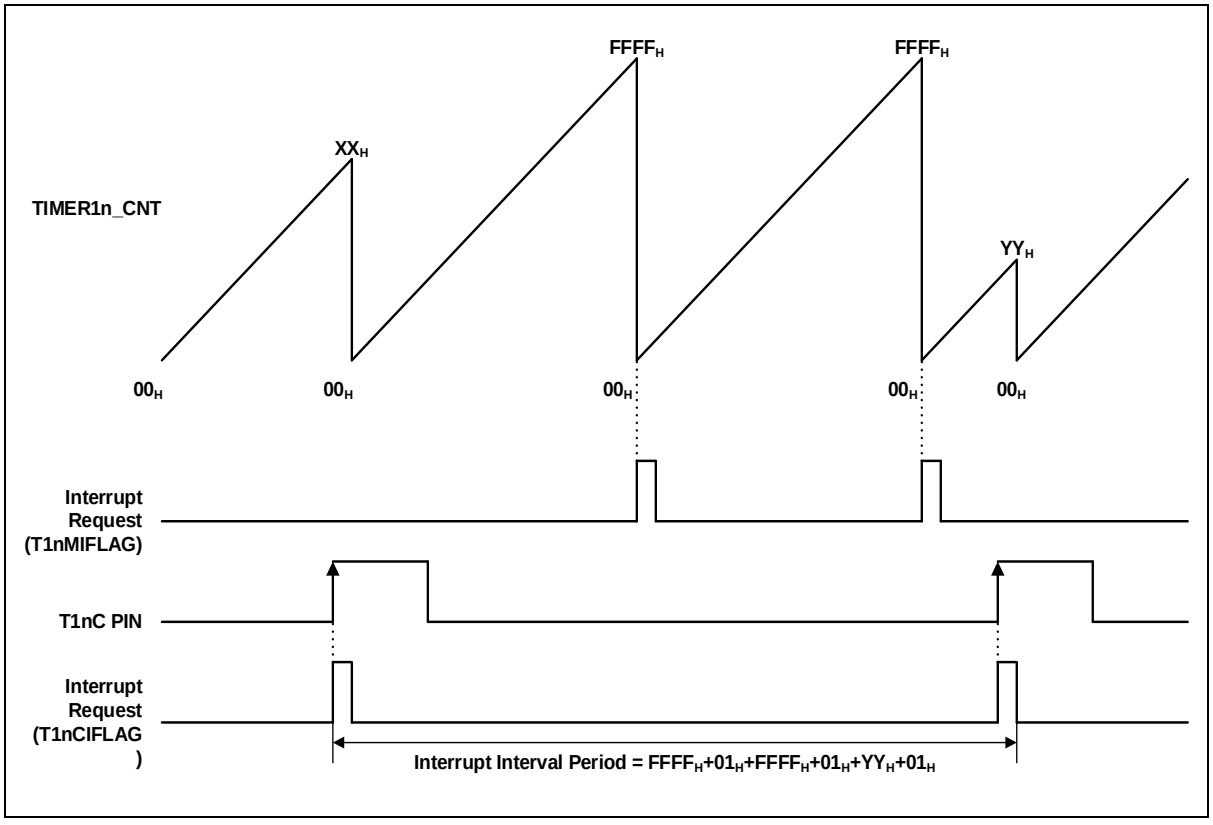
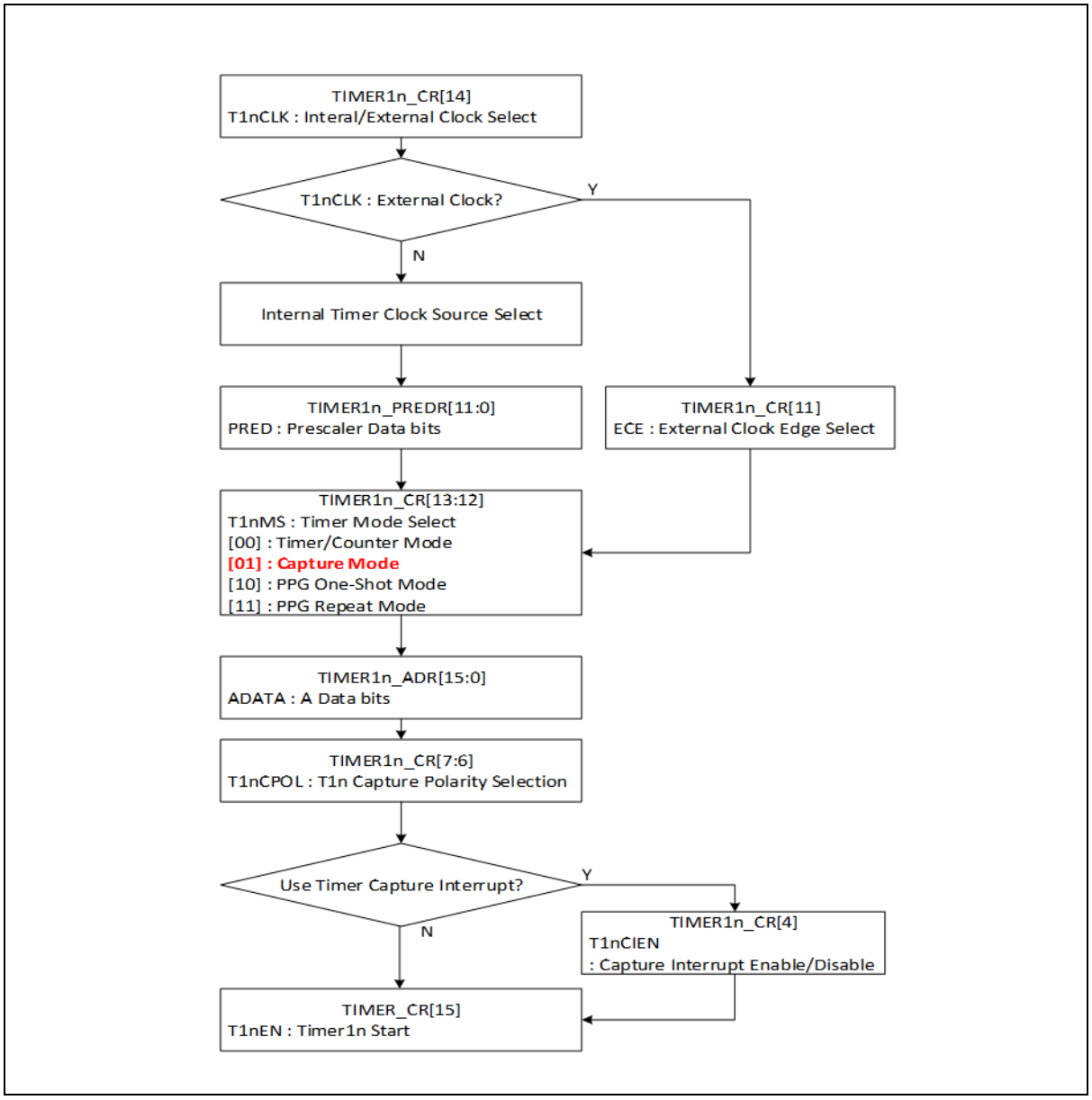


Figure 62 shows the capture mode operation of timer/counter 1n. Refer to Figure 13 for internal timer clock source and section 5.2 Pin multiplexing for Timer1n Output pin.

Figure 62. Capture Mode Operation Sequence of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)



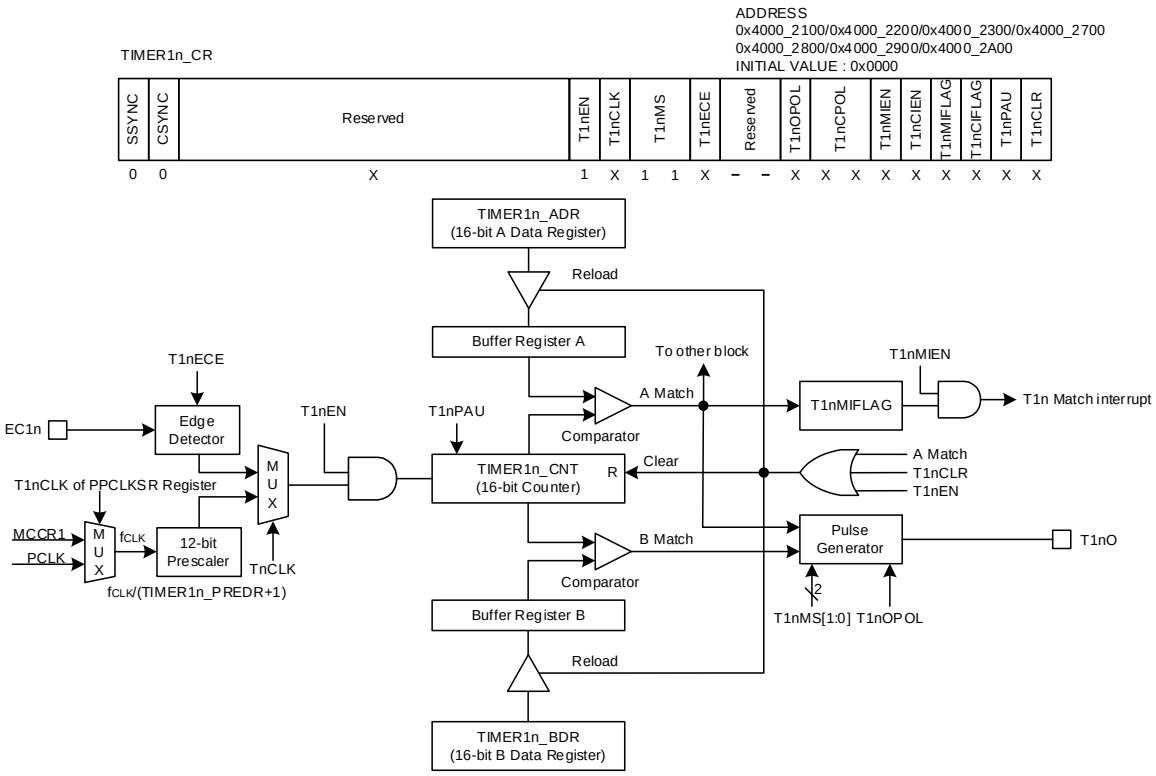
12.3.4 16-bit PPG mode

Timer 1n has a PPG (Programmable Pulse Generation) function. In PPG mode, the TnO pin outputs up to 16-bit resolution PWM output. This pin should be configured as a PWM output by setting corresponding PnAFSRx to 'AF6'. Period of the PWM output is determined by the TIMERn_ADR, and duty of the PWM output is determined by the TIMER1n_BDR.

IMPORTANT

1. External Clock (EC1n) is included in only Timer12.
2. Timer Capture pin (T1nC) and Timer Out pin (T1nO) are included in only Timer10, 11 and 12.

Figure 63. TIMER 1n Block Diagram in PPG Mode (n = 0, 1, 2, 3, 4, 5 and 6)



NOTE: The TnEN is automatically cleared to logic "0" after one pulse is generated at a PPG one-shot mode.

Figure 64. PPG Mode Timing Example of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)

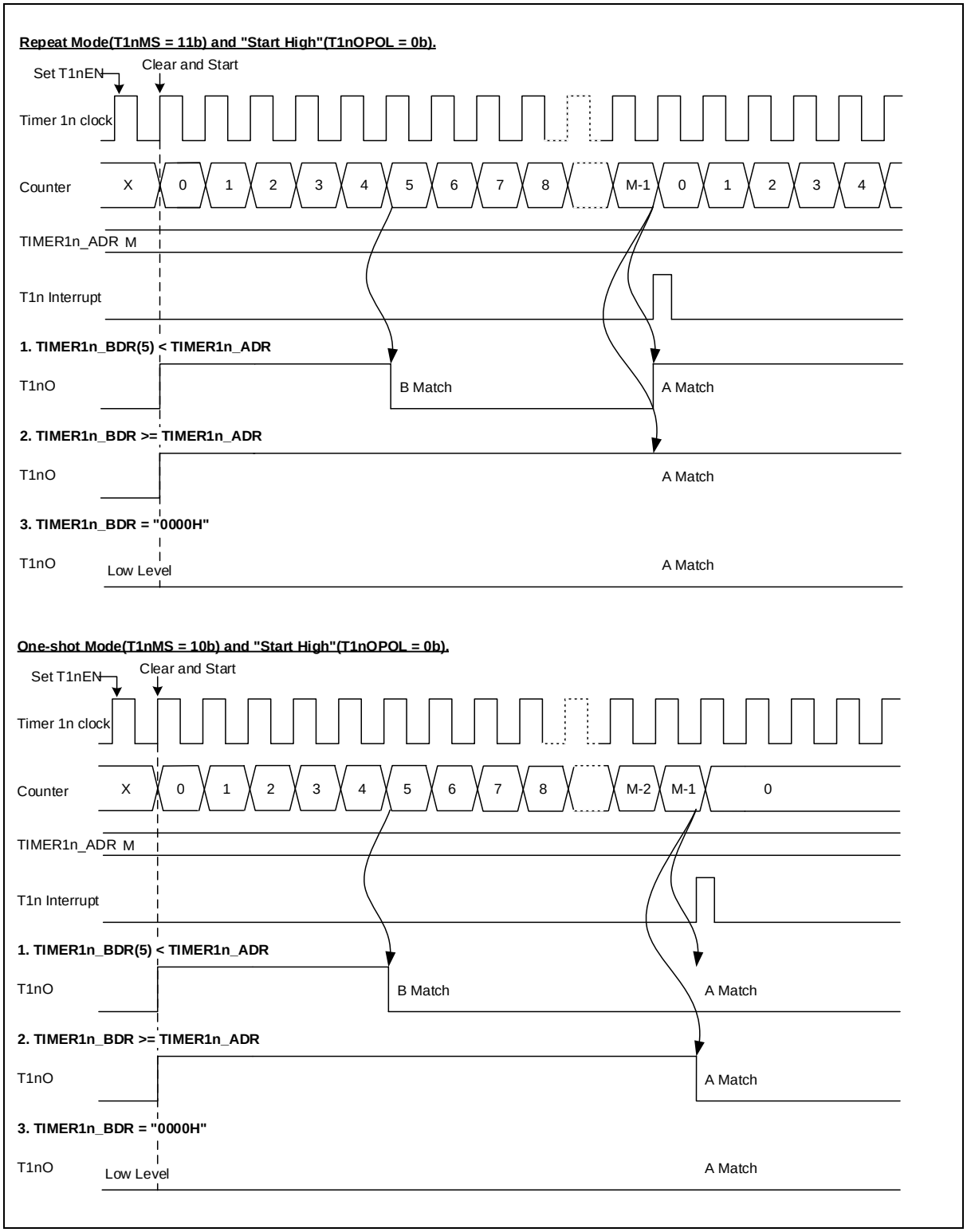
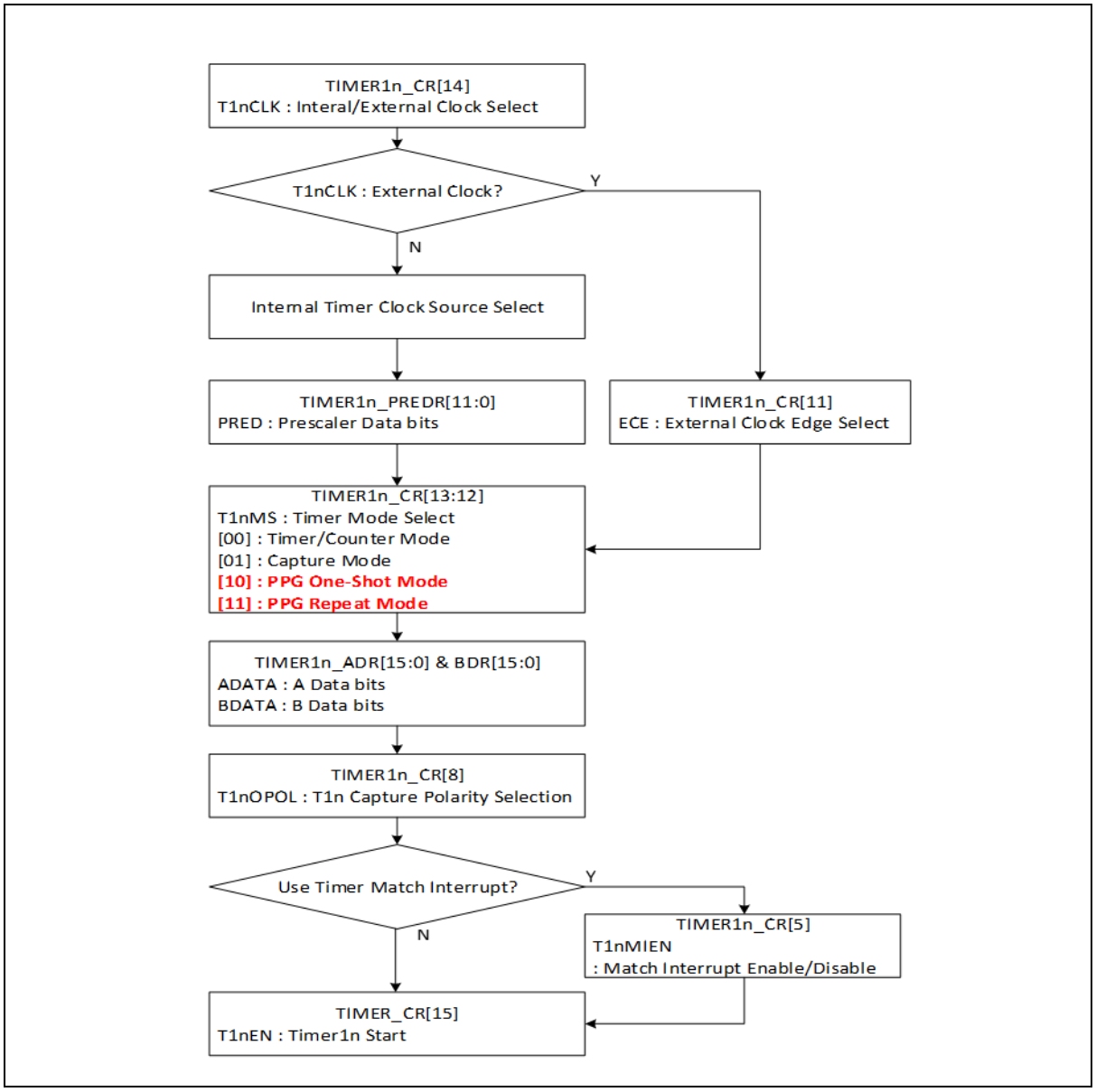


Figure 65 shows the PPG mode operation of timer/counter 1n. Refer to Figure 13 for internal timer clock source and a section 5.2 Pin multiplexing for Timer1n Output pin.

Figure 65. PPG Mode Operation Sequence of TIMER 1n (n = 0, 1, 2, 3, 4, 5 and 6)



12.3.5 The synchronization with TIMER30

A31G226 provides the timer synchronization that realizes requirements of the home appliances and motor application products. An ADC trigger source is set to work in user defined-period of TIMERn that is synchronized with TIMER30. In addition, the sampled data of the ADC is captured.

Table 57 describes Start and Stop operations of the TIMERn timer when the TIMERn_CR<SSYNC> is enabled. If TIMER1n timer is synchronized with TIMER30, it starts when both TIMER30 and TIMERn are set to Start, while TIMER1n timer stops when either TIMER30 or TIMERn is set to Stop.

Table 57. Start/ Stop Operation when TIMER1n is synchronized with TIMER30

TIMERn_CR <SSYNC>	TIMER30_CR <EN>	TIMER1n_CR <EN>	Description
1	1	1	TIMERn synchronized with TIMER30 starts to operate when TIMER30 starts.
1	1	0	TIMERn stops while TIMER30 continues to operate.
1	0	1	TIMERn stops when TIMER30 stops.
1	0	0	Both TIMER30 and TIMERn stop.

Figure 66. Start/ Stop Timing of TIMER1n synchronized with TIMER30

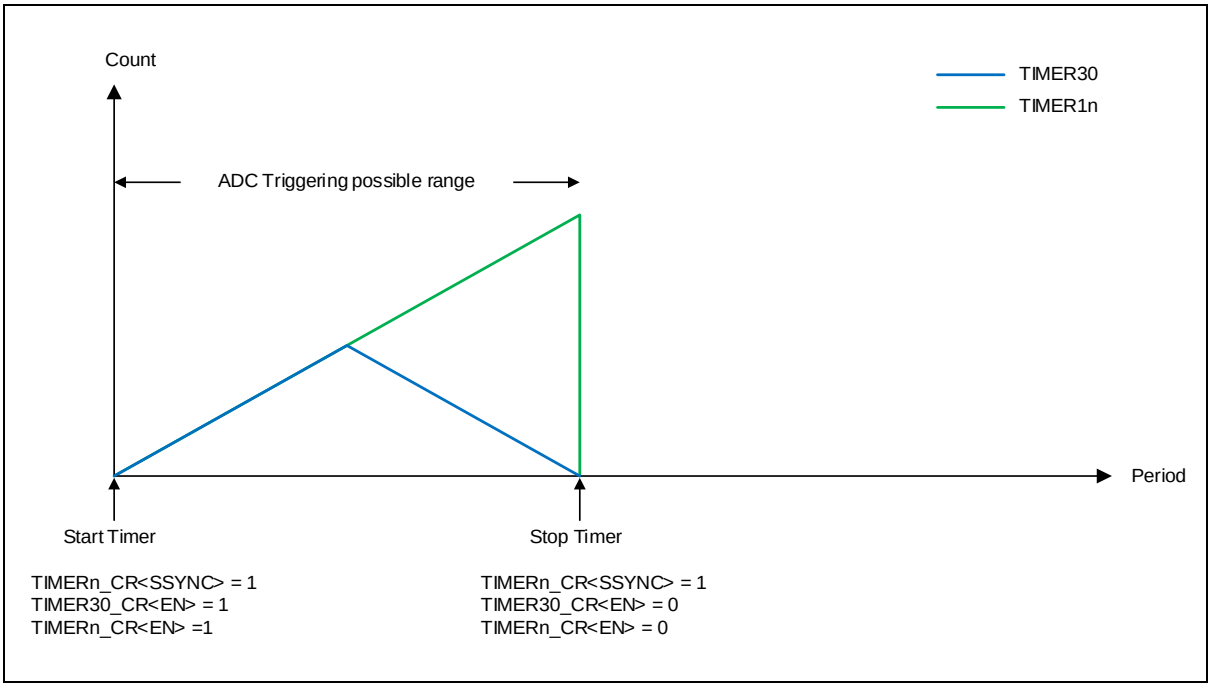
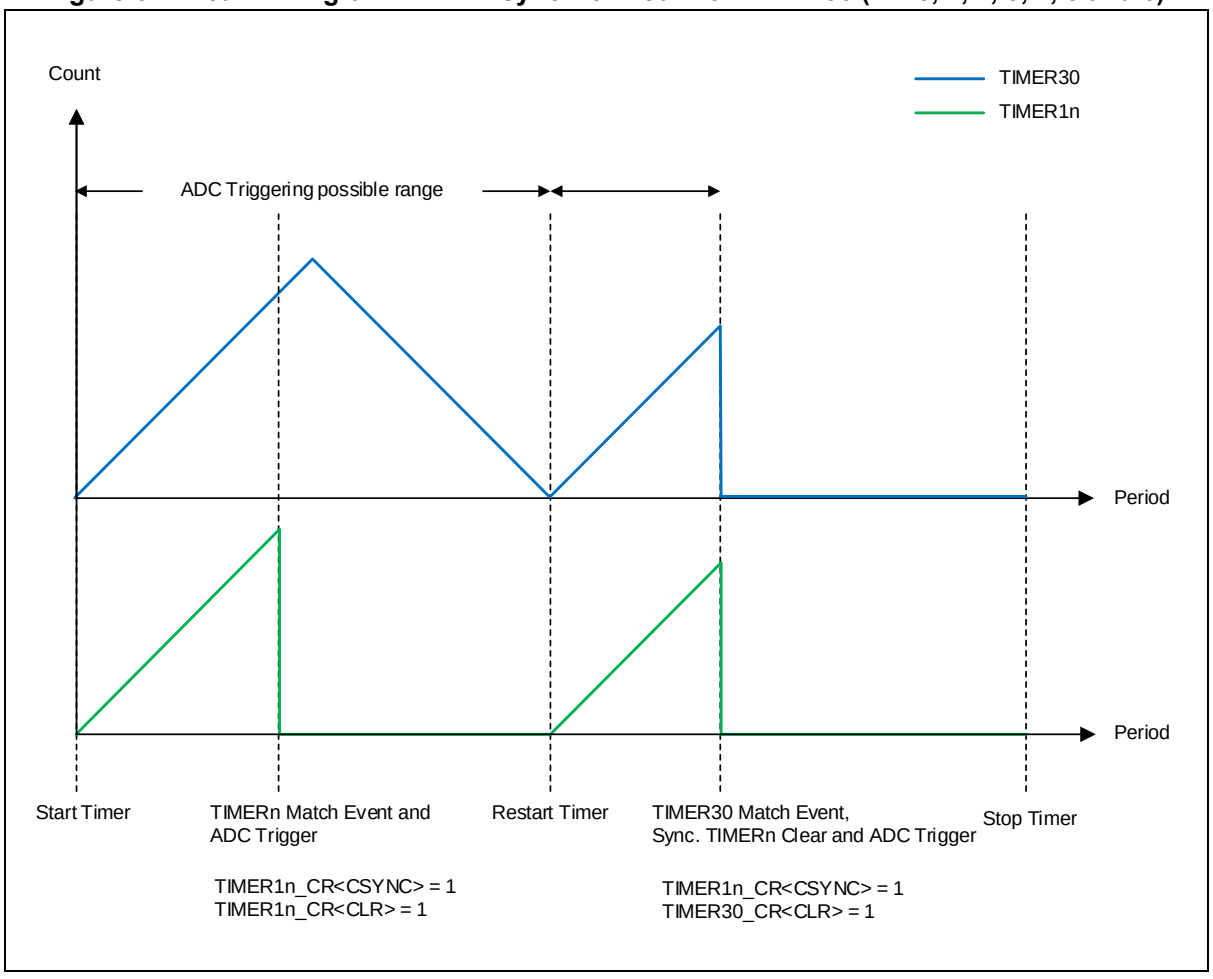


Table 58 describes Clear operations of TIMERn timer when TIMERn_CR<SSYNC> and TIMERn_CR<CSYNC> are enabled. If TIMERn timer is synchronized with TIMER30, the Clear operation is available when event of TIMERn Clear or TIMER30 Clear occurs.

Table 58. Clear Operations when TIMER1n is synchronized with TIMER30

TIMER1n_CR <SSYNC>	TIMER1n_CR <CSYNC>	TIMER30_CR <CLR>	TIMERn_CR <CLR>	Description
1	1	0	0	Both counters of TIMER1n and TIMER30 are not cleared.
1	1	1	0	TIMER30 is cleared, and then TIMER1n counter is cleared to start counting again.
1	1	0	1	Only TIMER1n counter is cleared to start counting again.

Figure 67. Clear Timing of TIMER1n synchronized with TIMER30 (n = 0, 1, 2, 3, 4, 5 and 6)



13 32-bit timer

32-bit timer block comprises 2 channels of 32-bit general purpose timers. TIMER2n channels have an independent 32-bit counter and a dedicated prescaler that feeds a counting clock. 32-bit timer supports periodic timer, PWM pulse, one-shot and capture mode. In addition, one more optional free-run timer is provided. Main purpose of 32-bit timer is a periodical tick timer.

32-bit timer of A31G22x series features the followings:

- 32-bit up-counter
- Periodic timer mode
- One-shot timer mode
- PWM pulse mode
- Capture mode
- 12-bit prescaler

Table 59 introduces pins assigned for 32-bit timer.

Table 59. Pin Assignment of 32-bit Timer: External Pins

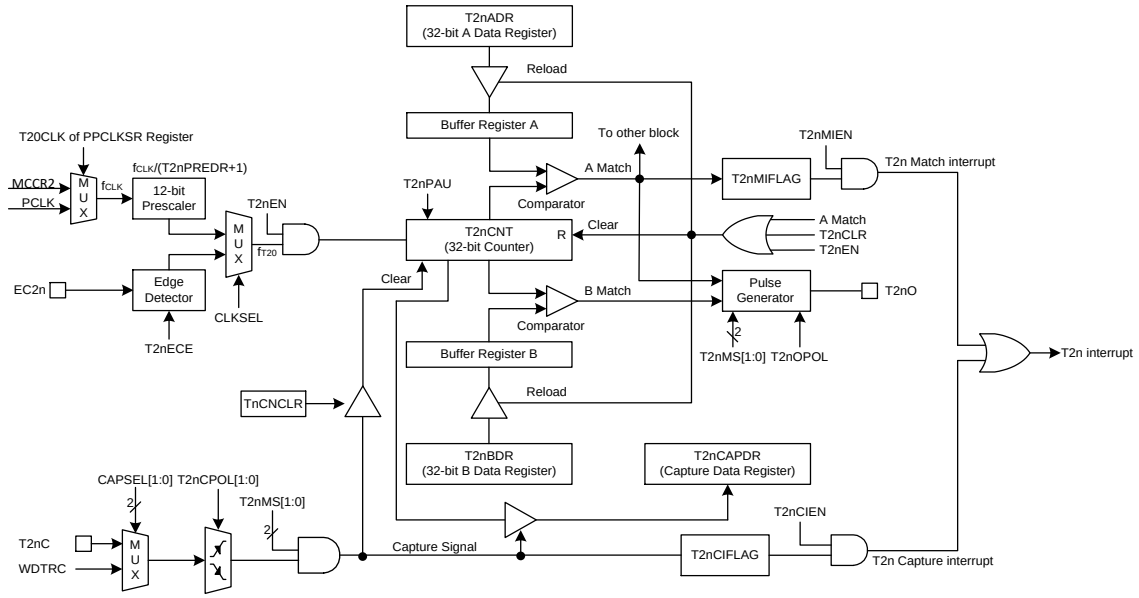
Pin name	Type	Description
EC2n	I	Timer 2n external clock input
T2nC	I	Timer 2n capture input
T2nO	O	Timer 2n Timer/PWM/one-shot output

NOTE: n = 0 and 1

13.1 32-bit timer block diagram

In this section, 32-bit timer is described in a block diagram in Figure 68.

Figure 68. 32-bit Timer Block Diagram



NOTES:

1. The clock source selection for PCLK or SCU_MCCR2 in the SCU_PPCLKSR register only supports TIMER20. (TIMER21 can only be used with PLCK.)
2. The T20EN is automatically cleared to logic "0" after one pulse is generated at a PPG one-shot mode.
3. After the T20EN bit is set, the counting of TIMER20_CNT may be delayed 2 clocks or more.

13.2 Registers

Base address of 32-bit timer is introduced in the followings:

Table 60. Base Address of 32-bit Timer

Name	Base address
TIMER20	0x4000_2500
TIMER21	0x4000_2600

Table 61. TIMER 2n Register Map

Name	Offset	Type	Description	Reset value	Ref.
TIMER2n_CR	0x0000	RW	Timer/Counter 2n Control Register	0x0000_0000	13.2.1
TIMER2n_ADR	0x0004	RW	Timer/Counter 2n A Data Register	0xFFFF_FFFF	13.2.2
TIMER2n_BDR	0x0008	RW	Timer/Counter 2n B Data Register	0xFFFF_FFFF	13.2.3
TIMER2n_CAPDR	0x000C	RO	Timer/Counter 2n Capture Data Register	0x0000_0000	13.2.4
TIMER2n_PREDR	0x0010	RW	Timer/Counter 2n Prescaler Data Register	0x0000_0FFF	13.2.5
TIMER2n_CNT	0x0014	RO	Timer/Counter 2n Counter Register	0x0000_0000	13.2.6

NOTE: n = 0 or 1

13.2.1 TIMER2n_CR: Timer/counter 2n control register

Timer module should be configured properly before running. When a target purpose is defined, the timer can be configured in the TIMER2n_CR register. After configuring TIMER2n_CR, a user can start or stop the timer function by using TIMER2n_CR.

TIMER2n_CR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0 or 1)

TIMER20_CR=0x4000_2500, TIMER21_CR=0x4000_2600

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		CNCLR	EN	CLK	MS	ECE	CAPSEL	OPOL		CPOL		MIEN	CIEN	MIFLAG	CIFLAG	PAU	CLR
-		0	0	0	0		0			0		0	0	0	0	0	0
-		RW	RW	RW	RW	RW	RW	RW		RW		RW	RW	RW	RW	RW	RW

Bits	Name	Function	
16	CNCLR	Timer 2n counter clear after Capture	
		0	Clear counter after capture
		1	Keep counter after capture
15	EN	Timer 2n Operation Enable bit.	
		0	Disable timer 2n operation.
		1	Enable timer 2n operation. (Counter clear and start)
14	CLK	Timer 2n Clock Selection bit.	
		0	Select an internal prescaler clock.
		1	Select an external clock. (EC2n)
NOTES: 1. This bit should be changed during T2nEN bit is '0'. 2. If you select an internal prescaler clock, you should set T2nCLK bit in the SCU_PPCLKSR register first.			
13 12	MS	Timer 2n Operation Mode Selection bits.	
		00	Timer/Counter mode. (T2nO: Toggle at A-match)
		01	Capture mode. (The A-match interrupt can occur)
		10	PPG one-shot mode. (T2nO: Programmable pulse output)
		11	PPG repeat mode. (T2nO: Programmable pulse output)
		NOTE: This bit should be changed during T2nEN bit is '0'.	
11	ECE	Timer 2n External Clock Edge Selection bit. (Input by EC2n pin)	
		0	Select falling edge of external clock.
		1	Select rising edge of external clock.
10 9	CAPSEL	Timer 2n Capture Signal Selection bits. (TIMER20 only)	
		00	Select an external capture signal.
		01	Not used
		10	Select the WDTRC (Watch-dog timer RC oscillator) signal.
		11	Not used

		NOTES: 1. This bit should be changed during T2nEN bit is '0'. 2. This field only supports for TIMER20.	
8	OPOL	Timer 2n Output Polarity Selection bit.	
		0	Start high. (T2nO is low level at disable)
		1	Start low. (T2nO is high level at disable)
7 6	CPOL	Timer 2n Capture Polarity Selection bits.	
		00	Capture on falling edge.
		01	Capture on rising edge.
		10	Capture on both of falling and rising edge.
		11	Reserved.
5	MIEN	Timer 2n Match Interrupt Enable bit.	
		0	Disable timer 2n match interrupt.
		1	Enable timer 2n match interrupt.
4	CIEN	Timer 2n Capture Interrupt Enable bit.	
		0	Disable timer 2n capture interrupt.
		1	Enable timer 2n capture interrupt.
3	MIFLAG	Timer 2n Match Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
2	CIFLAG	Timer 2n Capture Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
1	PAU	Timer 2n Counter Temporary Pause Control bit.	
		0	Continue counting.
		1	Temporary pause.
0	CLR	Timer 2n Counter and Prescaler Clear bit.	
		0	No effect.
		1	Clear timer 2n counter and prescaler. (Automatically cleared to '0' after operation)

13.2.2 TIMER2n_ADR: Timer/counter 2n A data register

TIMER2n_ADR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0 or 1)

TIMER20_ADR=0x4000_2504, TIMER21_ADR=0x4000_2604

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
ADATA			
0xFFFFFFFF			
RW			

Bits	Name	Function
31 0	ADATA	Timer/Counter 2n A Data bits. The range is 0x0002 to 0xFFFFFFFF.

NOTE: Do not write "0000H" in the TIMER20_ADR register when PPG mode.

13.2.3 TIMER2n_BDR: Timer/counter 2n B data register

TIMER2n_BDR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0 or 1)

TIMER20_BDR=0x4000_2508, TIMER21_BDR=0x4000_2608

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
BDATA			
0xFFFFFFFF			
RW			

Bits	Name	Function
31 0	BDATA	Timer/Counter 2n B Data bits. The range is 0x0000 to xFFFFFFFF.

13.2.4 TIMER2n_CAPDR: Timer/counter 2n capture data register

TIMER2n_ CAPDR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0 and 1)

TIMER20_CAPDR=0x4000_250C, TIMER21_CAPDR=0x4000_260C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
CAPD			
0x0000			
RO			

Bits	Name	Function
31 0	CAPD	Timer/Counter 2n Capture Data bits.

13.2.5 TIMER2n_PREDR: Timer/counter 2n prescaler data register

TIMER2n_PREDR is a 32-bit register, and able to do 32/16/8-bit access. (n = 0 and 1)

TIMER20_PREDR=0x4000_2510, TIMER21_PREDR=0x4000_2610

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			PRED
0x00000			0xFFF
–			RW

Bits	Name	Function
11 0	PRED	Timer/Counter 2n Prescaler Data bits. f_{CLK}/(TIMER2n_PREDR +1)

13.2.6 TIMER2n_CNT: Timer/counter 2n counter register

TIMER2n_CNT is a 32-bit register, and able to do 32/16/8-bit access.

TIMER20_CNT=0x4000_2514, TIMER21_CNT=0x4000_2514

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
CNT			
0x00000000			
RO			

Bits	Name	Function
31 0	CNT	Timer/Counter 2n Counter bits.

13.3 Functional description

13.3.1 Timer counter 20/21

Timer/counter 2n can be clocked by an internal or an external clock source (EC2n). Its clock source is selected by a clock selection logic which is controlled by clock selection bits (T2nCLK).

- TIMER 2n clock source: {f_{CLK}/(TIMER2n_PREDR +1), EC2n (n = 0, 1)

In capture mode, by T2nC, data is captured into input capture data register (TIMER2n_CAPDR). TIMER 2n results the comparison between a counter and the data register through T2nO port in timer/counter mode. In addition, TIMER 2n outputs PWM wave form through T2nO port in the PPG mode.

Table 62 introduces various operating modes of TIMER 2n according to the value of timer/counter register.

Table 62. TIMER 2n Operating Modes (n = 0 and 1)

TIMER2n_CR <EN>	TIMER2n_CR <MS>	TIMER2n_PREDR	Description
1	00	0xXXX	32-bit Timer/Counter Mode
1	01	0xXXX	32-bit Capture Mode
1	10	0xXXX	32-bit PPG Mode(one-shot mode)
1	11	0xXXX	32-bit PPG Mode(repeat mode)

13.3.2 32-bit timer/counter mode

32-bit timer/counter mode is selected by control register as shown in Figure 69. The 32-bit timer has a counter and data register.

The counter register is increased by internal or external clock input. TIMER 2n can use the clock input with 12-bit prescaler division rates (TIMER2n_PREDR) and external clock (EC2n). When each value of TIMER20_CNT and TIMER2n_ADR are identical in TIMER2n, a match signal is generated and the interrupt of Timer 2n occurs. The TIMER2n_CNT values are automatically cleared by the match signal. It can be also cleared by software (T2nCLR).

Figure 69. TIMER 2n Block Diagram in Timer/Counter Mode (n= 0 and 1)

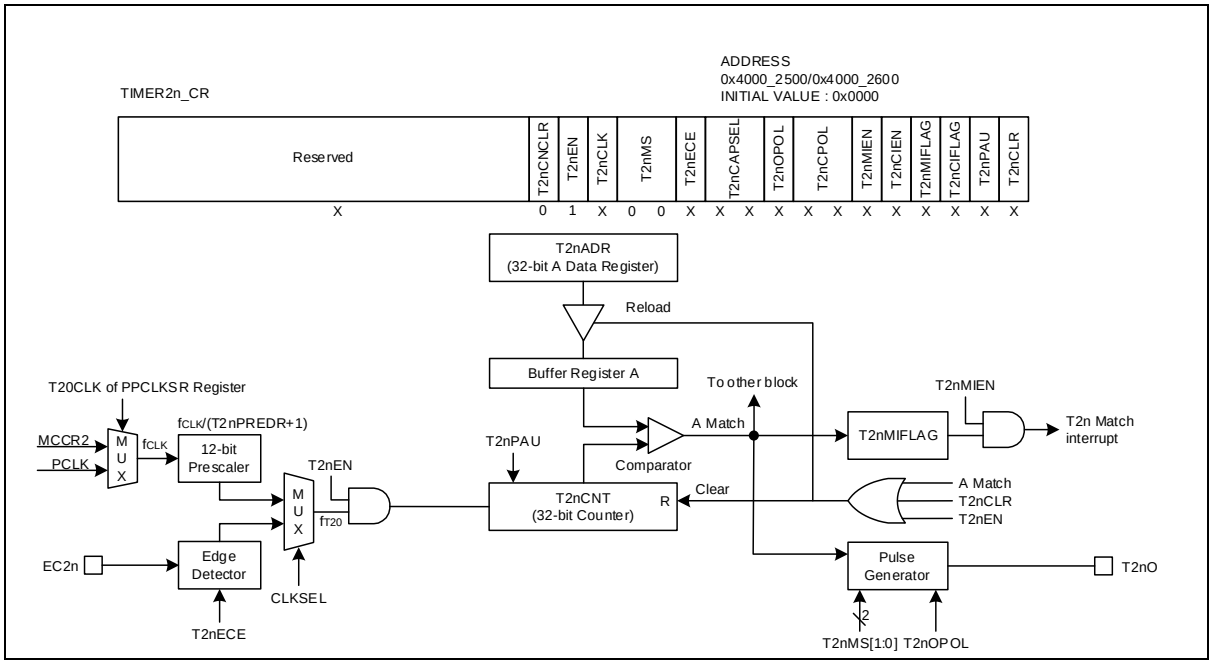


Figure 70. Timer/Counter Mode Timing Example of TIMER 2n (n= 0 and 1)

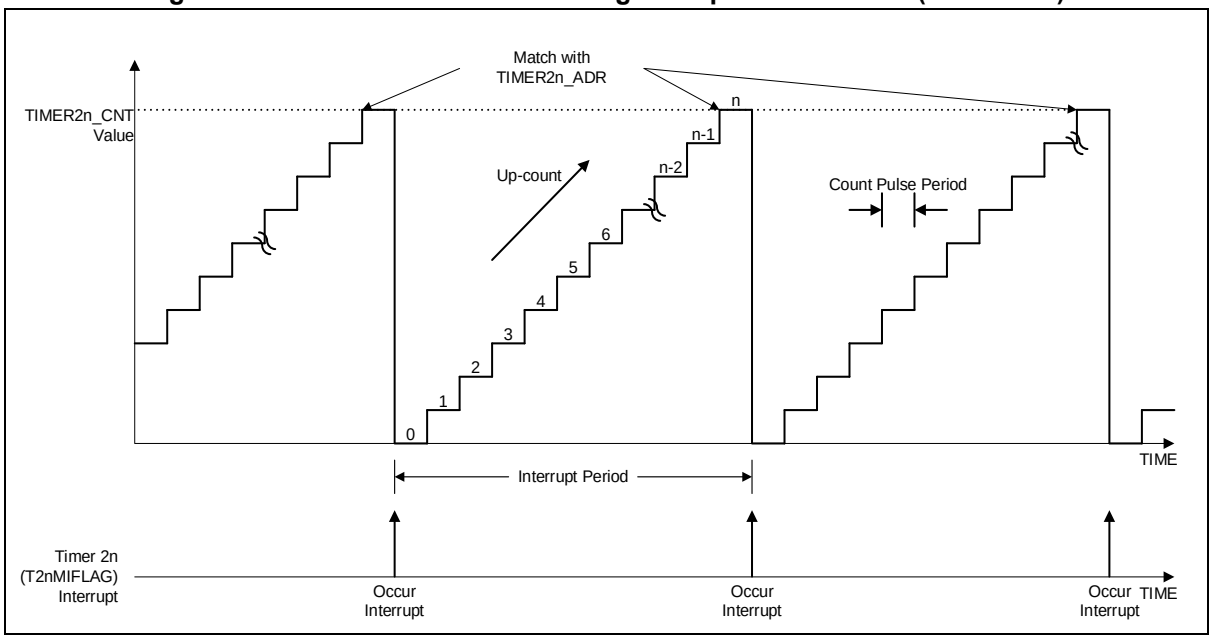
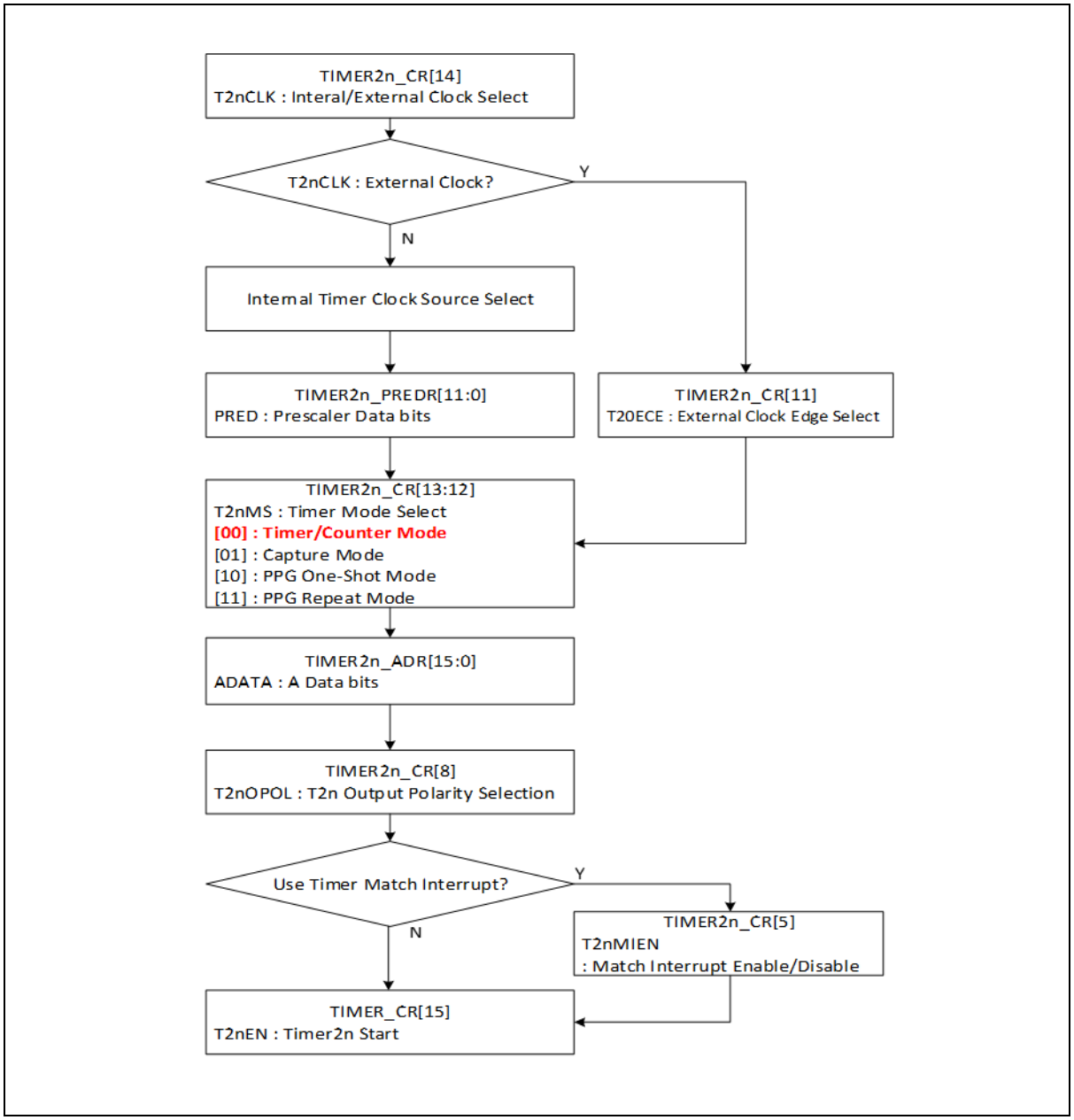


Figure 71 shows the timer/counter mode operation of timer/counter 2n. Refer to Figure 13 for internal timer clock source and 5.2 for Timer2n Output pin. (n = 0 and 1)

Figure 71. Timer/Counter Mode Operation Sequence of TIMER 2n (n = 0 or 1)



13.3.3 32-bit capture mode

Timer 2n capture mode is evoked by setting T2nMS[1:0] as '01'. It can use an internal clock as a clock source. Basically, it has the same function as 32-bit timer/counter mode, and the interrupt occurs when TIMER2n_CNT is equal to TIMER2n_ADR. TIMER2n_CNT value can be cleared by software (T2nCLR).

A timer interrupt in capture mode is very useful when pulse width of captured signal is wider than the maximum period of the timer. The capture result is loaded into TIMER2n_BDR. In the TIMER 2n capture mode, TIMER 2n output (T2nO) waveform is not available. (n = 0 or 1).

Figure 72. TIMER 2n Block Diagram in Capture Mode

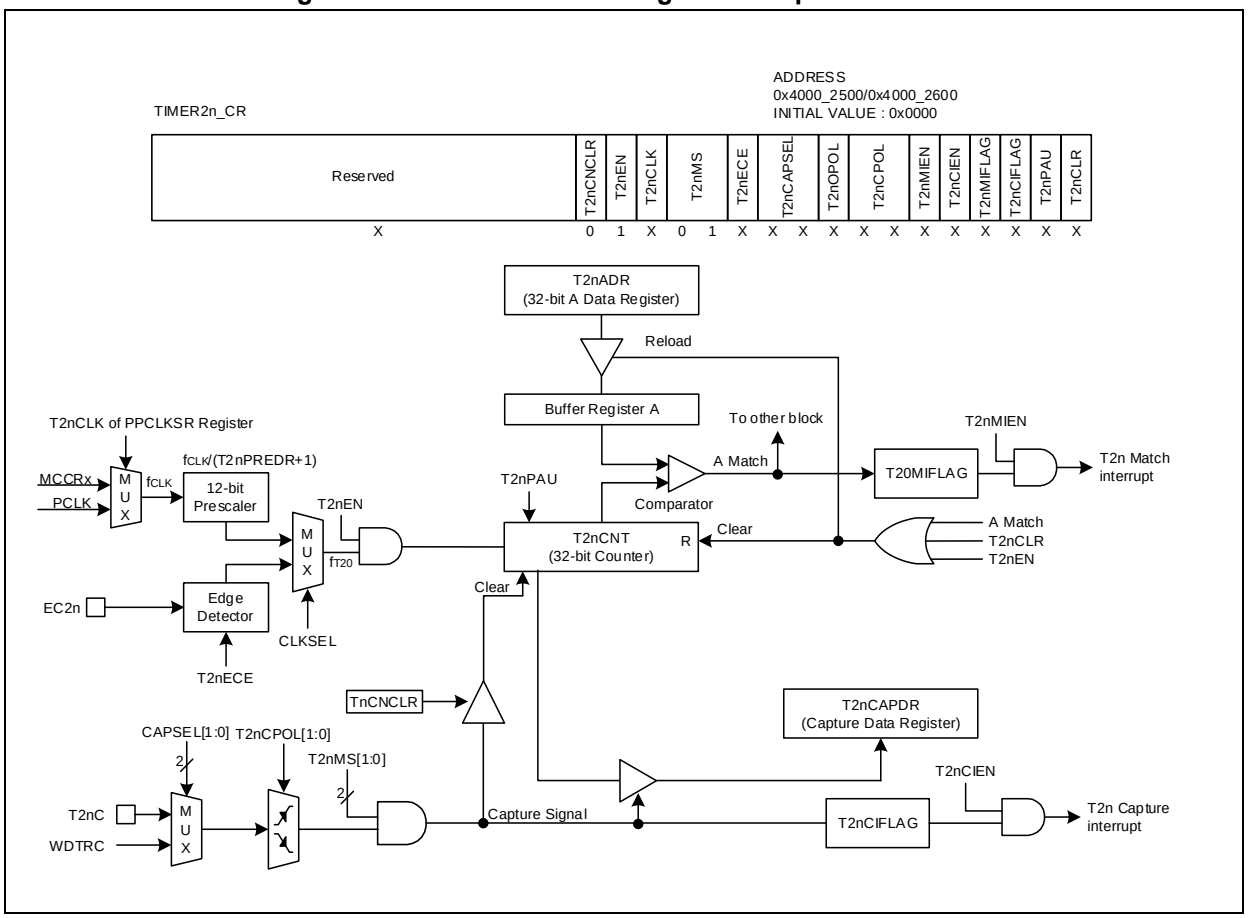


Figure 73. Capture Mode Timing Example of TIMER 2n (n = 0 and 1)

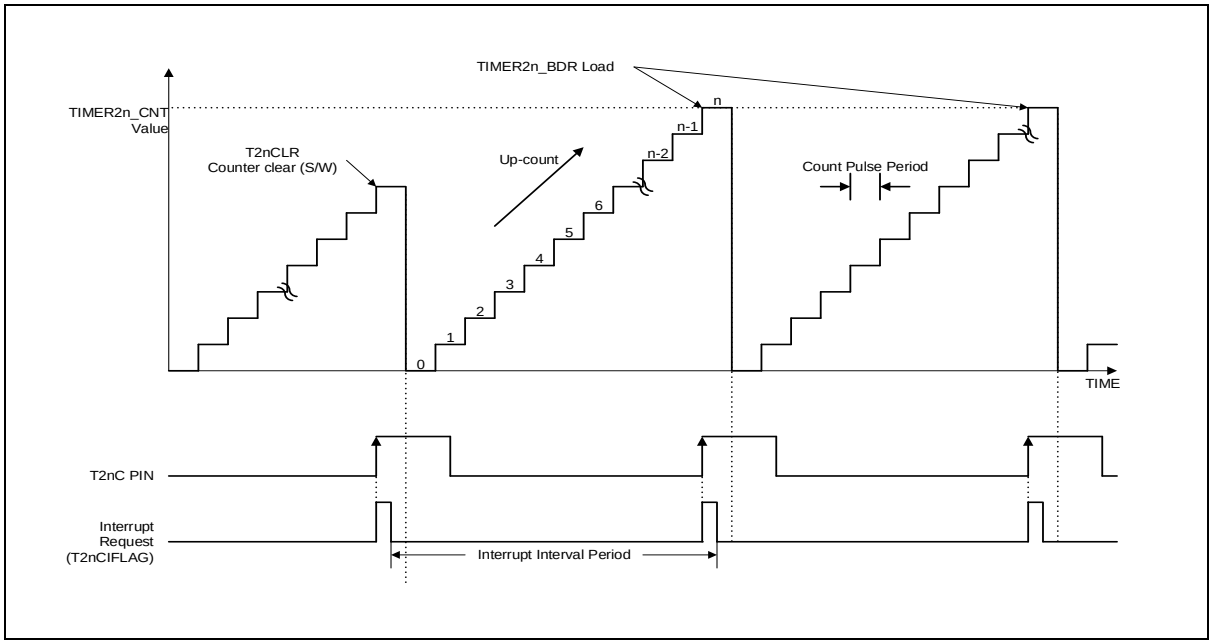


Figure 74. Express Timer Overflow in Capture Mode of TIMER 2n (T2nCNCLR = 1'b1)

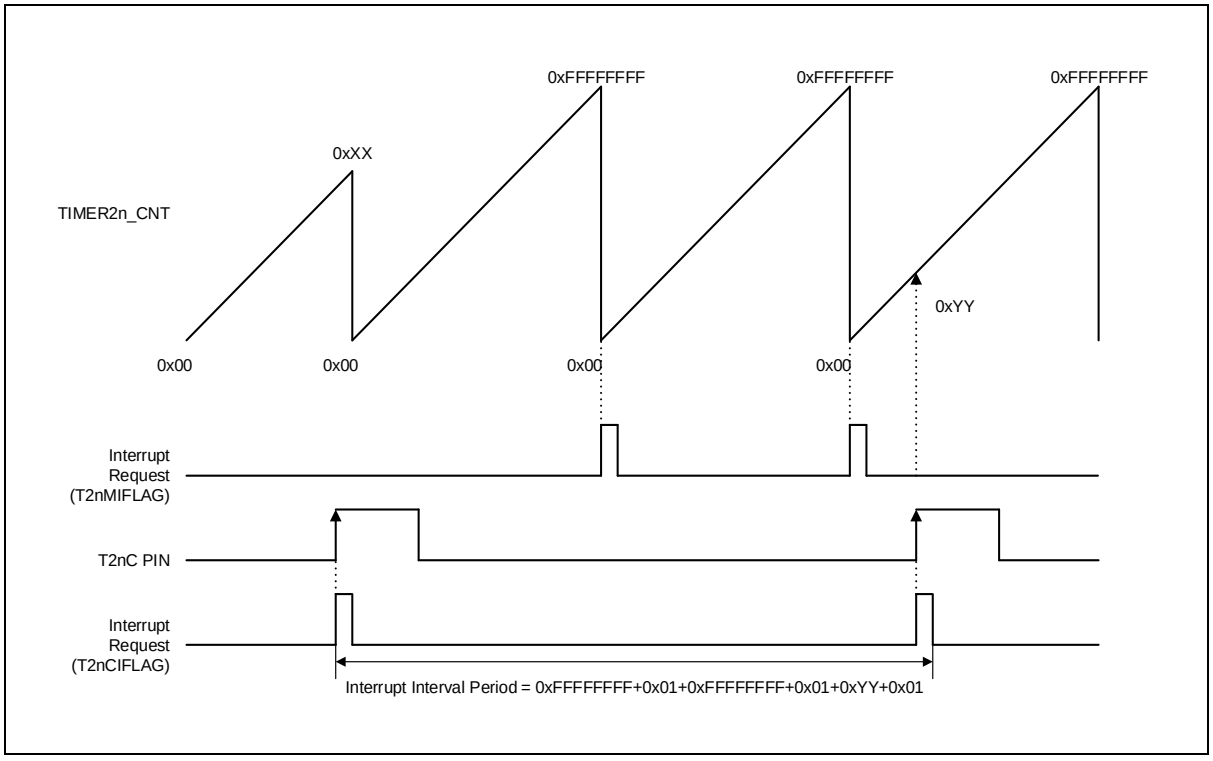


Figure 75. Express Timer Overflow in Capture Mode of TIMER 2n (T2nCNCLR = 1'b0)

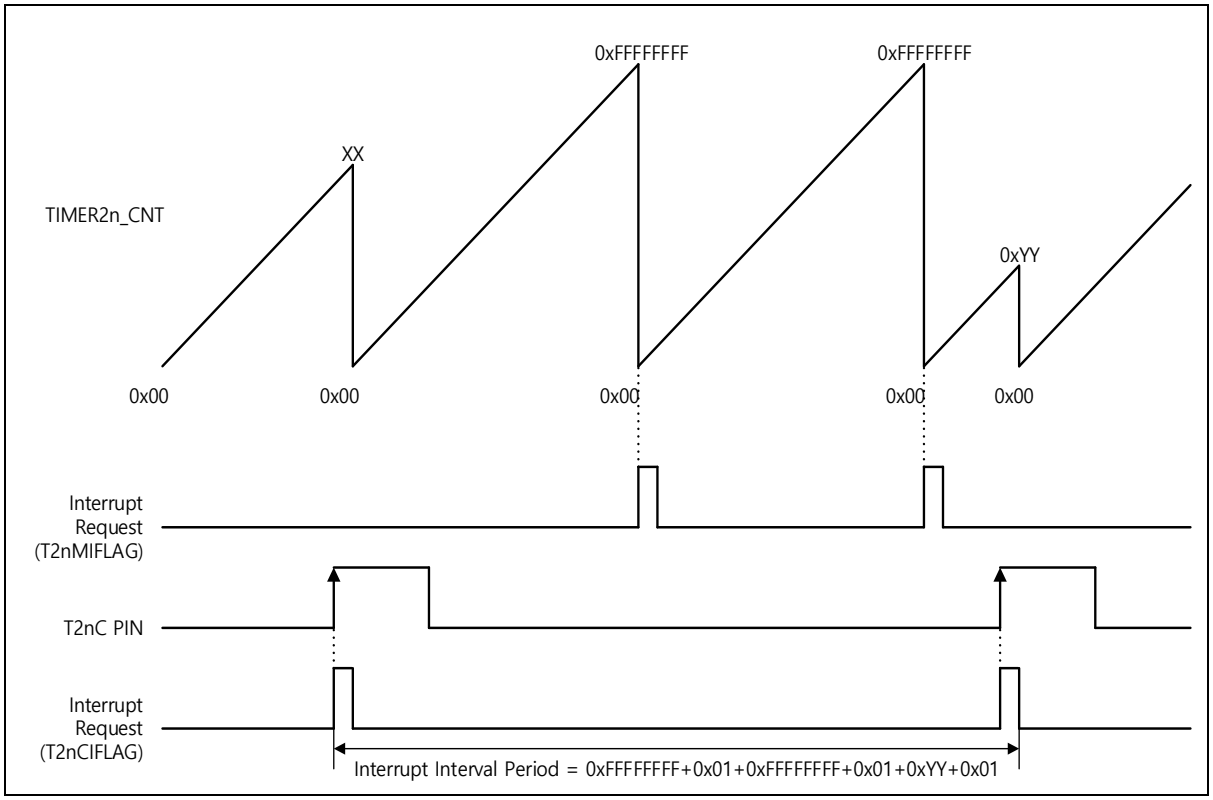
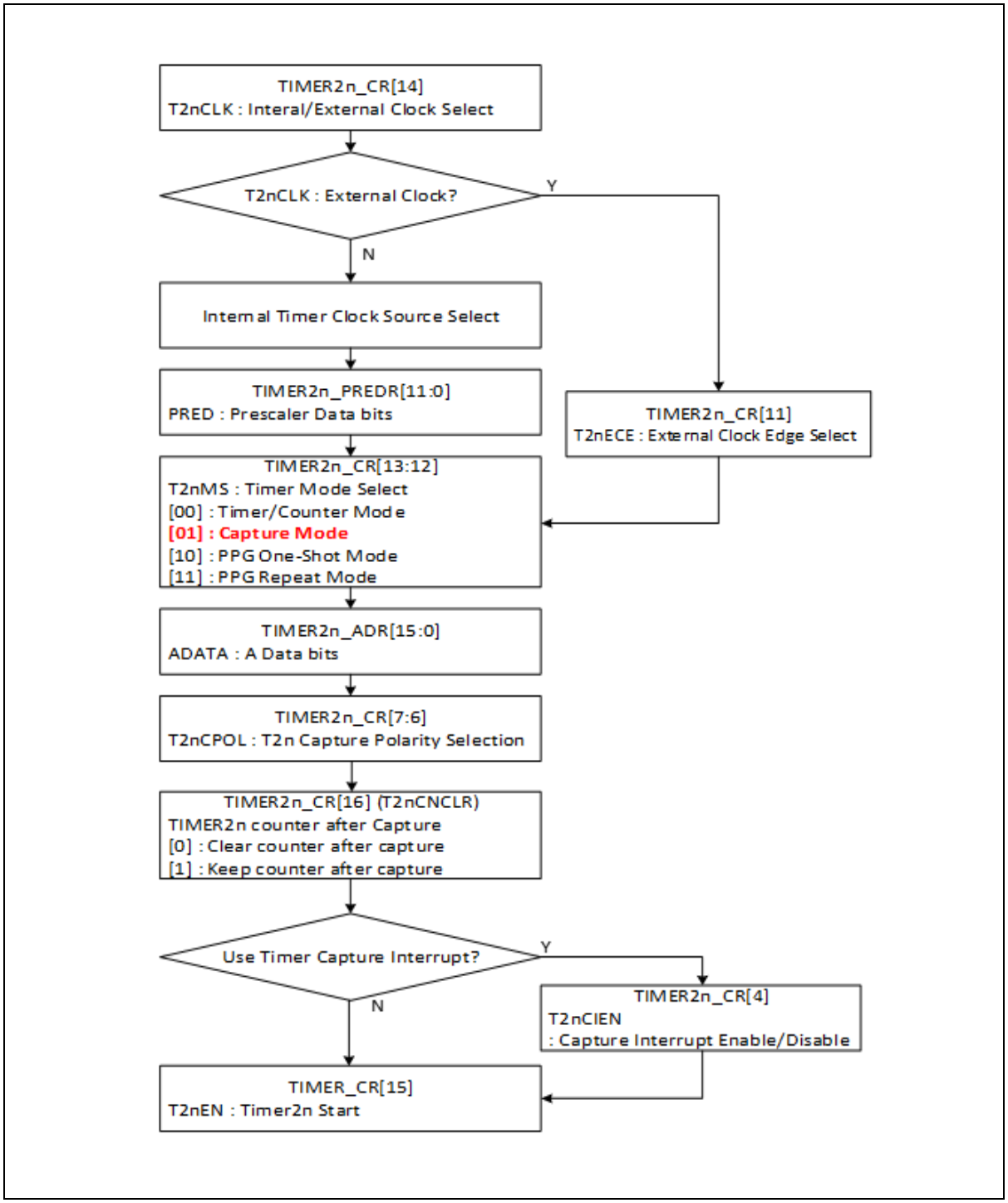


Figure 76 shows the capture mode operation of timer/counter 2n. Refer to Figure 13 for internal timer clock source and 5.2 for Timer2n Output pin. (n = 0 and 1)

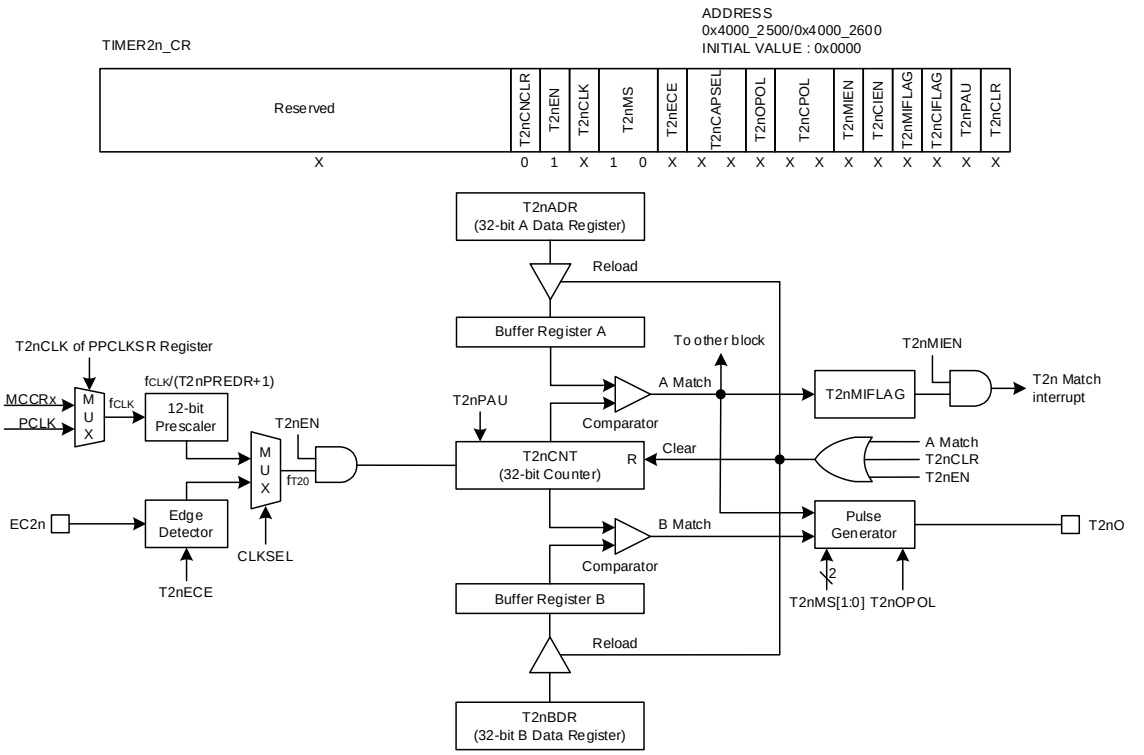
Figure 76. Capture Mode Operation Sequence of TIMER 2n (n = 0 or 1)



13.3.4 32-bit PPG mode

Timer 2n has a PPG (Programmable Pulse Generation) function. In PPG mode, the T2nO pin outputs up to 32-bit resolution PWM output. This pin should be configured as a PWM output by setting corresponding PnAFSRx to 'AF6' or 'AF7'. Period of the PWM output is determined by the TIMER2n_ADR, and duty of the PWM output is determined by the TIMER2n_BDR.

Figure 77. TIMER 2n Block Diagram in PPG Mode



NOTE: The T20EN is automatically cleared to logic "0" after one pulse is generated at a PPG one-shot mode.

Figure 78. PPG Mode Timing Example of TIMER 2n (n = 0 or 1)

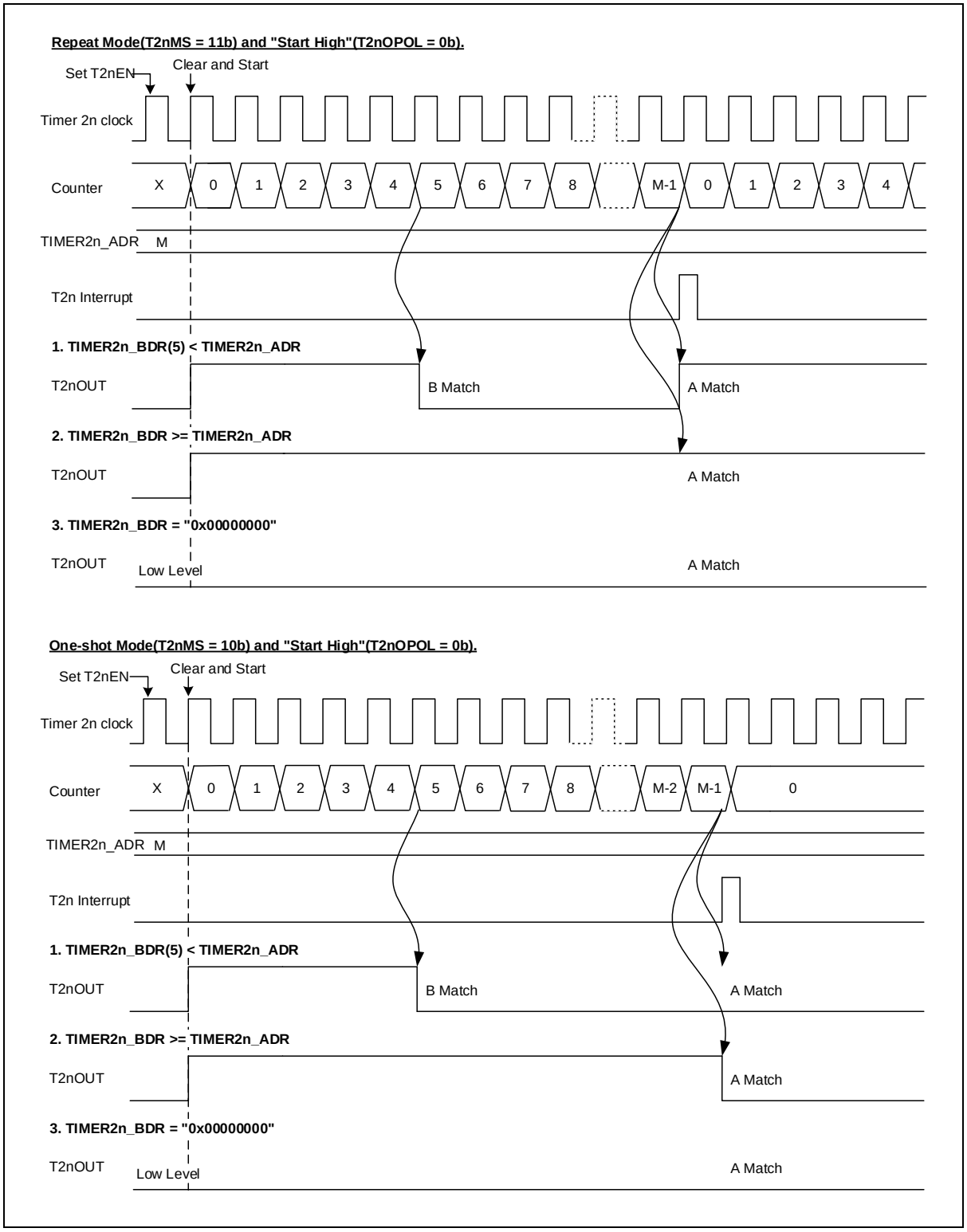
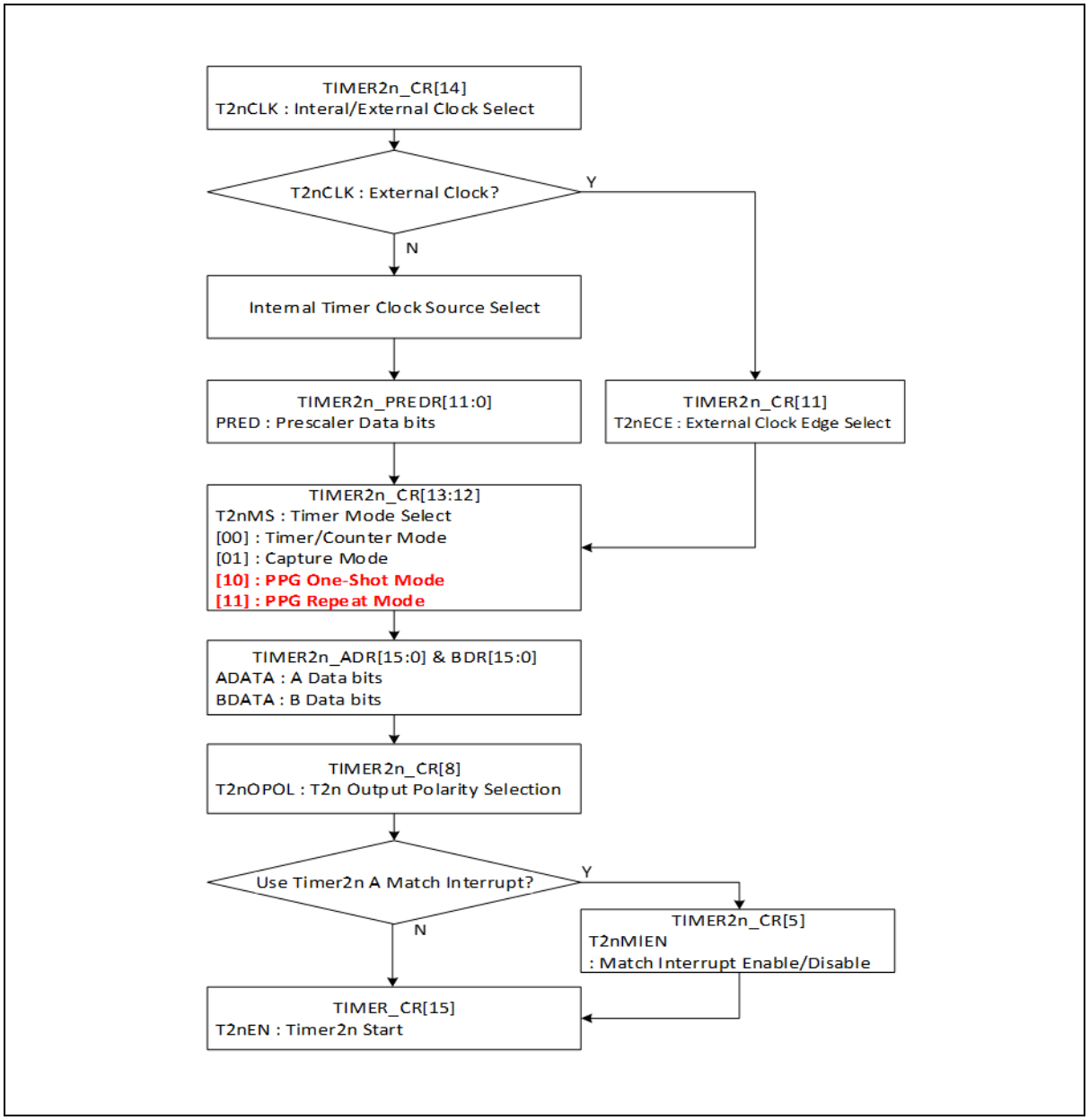


Figure 79 shows the PPG mode operation of timer/counter 2n. Refer to Figure 13 for internal timer clock source and 5.2 for Timer2n Output pin. (n = 0 and 1)

Figure 79. PPG Mode Operation Sequence of TIMER 2n (n = 0 or 1)



14 Timer counter 30

Timer counter 30 consists of a multiplexer, a comparator, 16-bit sized timer data registers A/B/C, and many other registers used for its operation. Main features are listed in the followings:

- 3-phase complementary PWM Outputs
- 16-bit up/down-counter
- Periodic timer mode
- Back-to-Back mode
- Capture mode
- 12-bit prescaler

Table 63 introduces pins assigned for the timer counter 30.

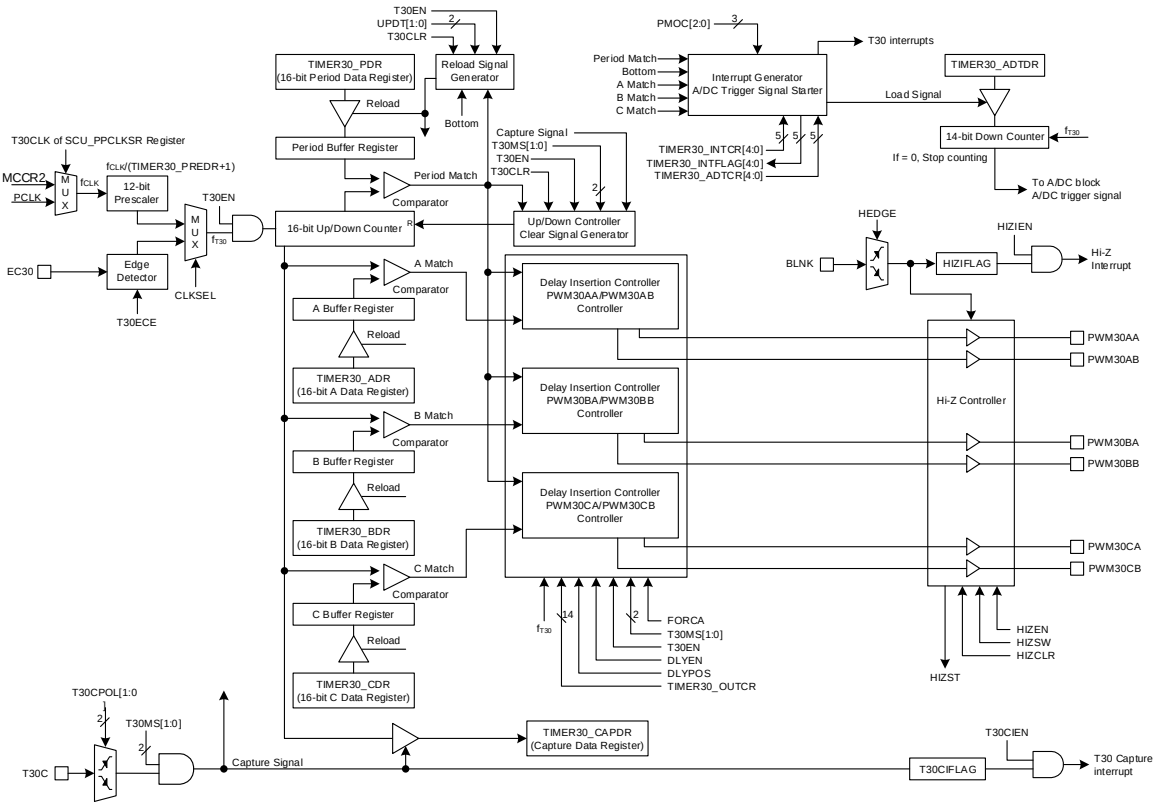
Table 63. Pin Assignment of Timer Counter 30: External Pins

Pin name	Type	Description
EC30	I	External clock input of TIMER30
T30C	I	Capture input of TIMER30
BLNK	I	External sync signal input of TIMER30
PWM30AA	O	PWM output
PWM30AB	O	PWM output
PWM30BA	O	PWM output
PWM30BB	O	PWM output
PWM30CA	O	PWM output
PWM30CB	O	PWM output

14.1 Timer counter 30 block diagram

In this section, timer counter 30 is introduced in a block diagram.

Figure 80. Timer Counter 30 Block Diagram



NOTES:

1. The Period/A/B/C match interrupts except for Bottom can occur in the interval mode.
2. The Period match interrupt only can occur in the capture mode.
   A. The 16-bit up/down counter in the capture mode is cleared to "0000H" after the counter value is loaded to the TIMER30_CAPDR register at valid edge.
   B. The HIZSW (high-impedance output software setting) bit can be used for debugging mode. That is, All TIMER30 PWM outputs can be high-impedance state with setting the HIZSW bit to '1' by the UI of debugger.
   C. After the EN bit is set, the counting of TIMER30_CNT may be delayed 2 clocks or more.

14.2 Registers

Base address of 3-phase PWM timer 30 is introduced in the followings:

Table 64. Base Address of Timer Counter 30

Name	Base address
TIMER30	0x4000_2400

Table 65. Timer Counter 30 Register Map

Name	Offset	Type	Description	Reset value	Ref.
TIMER30_CR	0x0000	RW	Timer/Counter 30 Control Register	0x0000_0000	14.2.1
TIMER30_PDR	0x0004	RW	Timer/Counter 30 Period Data Register	0x0000_FFFF	14.2.2
TIMER30_ADR	0x0008	RW	Timer/Counter 30 A Data Register	0x0000_FFFF	14.2.3
TIMER30_BDR	0x000C	RW	Timer/Counter 30 B Data Register	0x0000_FFFF	14.2.4
TIMER30_CDR	0x0010	RW	Timer/Counter 30 C Data Register	0x0000_FFFF	14.2.5
TIMER30_CAPDR	0x0014	RO	Timer/Counter 30 Capture Data Register	0x0000_0000	14.2.6
TIMER30_PREDR	0x0018	RW	Timer/Counter 30 Prescaler Data Register	0x0000_0FFF	14.2.7
TIMER30_CNT	0x001C	RO	Timer/Counter 30 Counter Register	0x0000_0000	14.2.8
TIMER30_OUTCR	0x0020	RW	Timer/Counter 30 Output Control Register	0x0000_0000	14.2.9
TIMER30_DLY	0x0024	RW	Timer/Counter 30 PWM Output Delay Data Register	0x0000_0000	14.2.10
TIMER30_INTCR	0x0028	RW	Timer/Counter 30 Interrupt Control Register	0x0000_0000	14.2.11
TIMER30_INTFLAG	0x002C	RW	Timer/Counter 30 Interrupt Flag Register	0x0000_0000	14.2.12
TIMER30_HIZCR	0x0030	RW	Timer/Counter 30 High-Impedance Control Register	0x0000_0000	14.2.13

Table 63. Timer Counter 30 Register Map (continued)

Name	Offset	Type	Description	Reset value	Ref.
TIMER30_ADTCR	0x0034	RW	Timer/Counter 30 A/DC Trigger Control Register	0x0000_0000	14.2.14
TIMER30_ADTDR	0x0038	RW	Timer/Counter 30 A/DC Trigger Generator Data Register	0x0000_0000	14.2.15

14.2.1 TIMER30_CR: timer/counter 30 control register

Timer module should be configured properly before running. When target purpose is defined, the timer can be configured in this register. After configuring this register, you can start or stop the timer function by TIMER30_CR register.

TIMER30_CR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_CR=0x4000_2400

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		EN	CLK	MS		ECE	FORCA	DLYEN		DLYPOS	CPOL		UPDT	PMOC			CLR
-		0	0	00		0	0	0		0	00		00	000			0
-		RW	RW	RW		RW	RW	RW		RW	RW		RW	RW			RW

Bits	Name	Function	
15	EN	Timer 30 Operation Enable bit.	
		0	Disable timer 30 operation.
		1	Enable timer 30 operation. (Counter clear and start)
14	CLK	Timer 30 clock selection bit.	
		0	Select an internal prescaler clock.
		1	Select an external clock. (EC30)
NOTES: 1. This bit should be changed during EN bit in TIMER30_CR is "0b". 2. If you select an internal prescaler clock, you should set CLK in TIMER30_CR bit in the SCU_PPCLKSR register first.			
13 12	MS	Timer 30 operation mode selection bits.	
		00	Interval mode. (all match interrupts can occur)
		01	Capture mode. (the period-match interrupt can occur)
		10	Back-to-back mode. (all interrupts can occur)
		11	Not used.
		NOTE: This bit should be changed during EN in TIMER30_CR bit is "0b".	
11	ECE	Timer 30 external clock edge selection bit. (Input by EC3n pin)	
		0	Select falling edge of external clock.
		1	Select rising edge of external clock.
10	FORCA	Timer 30 Output Mode Selection bit. This bit should be changed when the EN in TIMER30_CR is "0b".	
		0	6-channel mode (The PWM30xA/PWM30xB pins are output according to the TIMER30_xDR registers, respectively)
		1	Force A-channel mode (The all PWM30xA/PWM30xB pins are output according to the only TIMER30_ADR registers)
9	DLYEN	Delay time insertion enable bit.	
		0	Disable to insert delay time to the PWM30xA/PWM30xB.
		1	Enable to insert delay time to the PWM30xA/PWM30xB.
8	DLYPOS	Delay time insertion position.	

		0	Insert at front of PWM30xA and at back of PWM30xB pins.
		1	Insert at back of PWM30xA and at front of PWM30xB pins.
7 6	CPOL	Timer 30 capture polarity selection bits.	
		00	Capture on falling edge.
		01	Capture on rising edge.
		10	Capture on both of falling and rising edge.
		11	Reserved
6 4	UPDT	Data reload time selection bits.	
		00	Update data to buffer at the time of writing.
		01	Update data to buffer at period match.
		10	Update data to buffer at bottom.
		11	Not used.
3 1	PMOC	Period match interrupt occurrence selection.	
		000	Once every period match.
		001	Once every 2 period match.
		010	Once every 3 period match.
		011	Once every 4 period match.
		100	Once every 5 period match.
		101	Once every 6 period match.
		110	Once every 7 period match.
		111	Once every 8 period match.
NOTES: 1. A period match counter is cleared as 0x00 when the CLR in TIMER30_CR bit is set. 2. When changing the PMOC value, must clear the period match counter with CLR in TIMER30_CR. Otherwise, malfunction may occur.			
0	CLR	Timer 30 counter and prescaler clear bit.	
		0	No effect.
		1	Clear timer 30 counter and prescaler (Automatically cleared to "0b" after operation)

14.2.2 TIMER30_PDR: timer/counter 30 period data register

TIMER30_PDR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_PDR=0x4000_2404

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		PDATA	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	PDATA	Timer/Counter 30 Period Data bits. The range is 0x0002 to 0xFFFF.

NOTE: Do not write "0x0000" in the TIMER30_PDR register when PPG mode.

14.2.3 TIMER30_ADR: timer/counter 30 A data register

TIMER30_ADR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_ADR=0x4000_2408

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		ADATA	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	ADATA	Timer/Counter 30 A data bits. The range is 0x0000 to 0xFFFF.

14.2.4 TIMER30_BDR: timer/counter 30 B data register

TIMER30_BDR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_BDR=0x4000_240C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		BDATA	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	BDATA	Timer/Counter 30 B data bits. The range is 0x0000 to 0xFFFF.

14.2.5 TIMER30_CDR: timer/counter 30 C data register

TIMER30_CDR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_CDR=0x4000_2410

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CDATA	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	CDATA	Timer/Counter 30 C data bits. The range is 0x0000 to 0xFFFF.

14.2.6 TIMER30_CAPDR: timer/counter 30 capture data register

TIMER30_CAPDR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_CAPDR=0x4000_2414

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CAPD	
-		0x0000	
-		RO	

Bits	Name	Function
15 0	CAPD	Timer/Counter 30 capture data bits.

14.2.7 TIMER30_PREDR: timer/counter 30 prescaler data register

TIMER30_PREDR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_PREDR=0x4000_2418

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		PRED	
-		0xFFF	
-		RW	

Bits	Name	Function
11 0	PRED	Timer/Counter 30 prescaler data bits.

14.2.8 TIMER30_CNT: timer/counter 30 counter register

TIMER30_CNT is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_CNT=0x4000_241C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CNT	
-		0x0000	
-		RO	

Bits	Name	Function
15 0	CNT	Timer/Counter 30 counter bits.

14.2.9 TIMER30_OUTCR: timer/counter 30 output control register

TIMER30_OUTCR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_OUTCR=0x4000_2420

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WTIDKY		POLB	POLA	PABOE	PBBOE	PCBOE	PAAOE	PBAOE	PCAOE	Reserved	LVLAB	LVLBB	LVLCB	Reserved	LVLAA	LVLBA	LVLCA
0x0000		0	0	0	0	0	0	0	0	-	0	0	0	-	0	0	0
RW		RW	RW	RW	RW	RW	RW	RW	RW	-	RW	RW	RW	-	RW	RW	RW

Bits	Name	Function	
31 16	WTIDKY	Write Identification Key. On writes, write 0xE06C to these bits, otherwise the write is ignored.	
15	POLB	PWM30xB output polarity selection bit. (x : A, B and C)	
		0	Low level start. (The PWM30xB pins are started with low level after counting)
		1	High level start. (The PWM30xB pins are started with high level after counting)
14	POLA	PWM30xA Output Polarity Selection bit. (x : A, B and C)	
		0	Low level start. (The PWM30xA pins are started with low level after counting)
		1	High level start. (The PWM30xA pins are started with high level after counting)
13	PABOE	PWM30AB output enable bit.	
		0	Disable output.
		1	Enable output.
12	PBBOE	PWM30BB output enable bit.	
		0	Disable output.
		1	Enable output.
11	PCBOE	PWM30CB output enable bit.	
		0	Disable output.
		1	Enable output.
10	PAAOE	PWM30AA output enable bit.	
		0	Disable output.
		1	Enable output.
9	PBAOE	PWM30BA output enable bit.	
		0	Disable output.
		1	Enable output.
8	PCAOE	PWM30CA output enable bit.	
		0	Disable output.
		1	Enable output.

6	LVLAB	Configure PWM30AB output when disable.	
		0	Low level
		1	High level
5	LVLBB	Configure PWM30BB output when disable.	
		0	Low level
		1	High level
4	LVLCB	Configure PWM30CB output when disable.	
		0	Low level
		1	High level
2	LVLAA	Configure PWM30AA output when disable.	
		0	Low level
		1	High level
1	LVLBA	Configure PWM30BA output when disable.	
		0	Low level
		1	High level
0	LVLCA	Configure PWM30CA output when disable.	
		0	Low level
		1	High level

14.2.10 TIMER30_DLY: timer/counter 30 PWM output delay data register

TIMER30_DLY is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_DLY=0x4000_2424

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			DLY
-			0x000
-			RW

Bits	Name	Function
9 0	DLY	Timer/Counter 30 PWM delay data bits. Delay time: (DLY[9:0]+1)÷$f_{TIMER30}$

14.2.11 TIMER30_INTCR: timer/counter 30 interrupt control register

TIMER30_INTCR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_INTCR=0x4000_2428

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved				HIZIEN	CIEN	BTIEN	PMIEN	AMIEN	BMIEN	CMIEN
-				0	0	0	0	0	0	0
-				RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
6	HIZIEN	Timer 30 output high-impedance interrupt enable bit.
	0	Disable timer 30 output high-impedance interrupt.
	1	Enable timer 30 output high-impedance interrupt.
5	CIEN	Timer 30 capture interrupt enable bit.
	0	Disable timer 30 capture interrupt.
	1	Enable timer 30 capture interrupt.
4	BTIEN	Timer 30 bottom interrupt enable bit.
	0	Disable timer 30 period interrupt.
	1	Enable timer 30 period interrupt.
3	PMIEN	Timer 30 period match interrupt enable bit.
	0	Disable timer 30 period interrupt.
	1	Enable timer 30 period interrupt.
2	AMIEN	Timer 30 A-ch match interrupt enable bit.
	0	Disable timer 30 A-ch match interrupt.
	1	Enable timer 30 A-ch match interrupt.
1	BMIEN	Timer 30 B-ch match interrupt enable bit.
	0	Disable timer 30 B-ch match interrupt.
	1	Enable timer 30 B-ch match interrupt.
0	CMIEN	Timer 30 C-ch match interrupt enable bit.
	0	Disable timer 30 C-ch match interrupt.
	1	Enable timer 30 C-ch match interrupt.

14.2.12 TIMER30_INTFLAG: timer/counter 30 interrupt flag register

TIMER30_INTFLAG is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_INTFLAG=0x4000_242C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved				HIZIFLAG	CIFLAG	BTIFLAG	PMIFLAG	AMIFLAG	BMIFLAG	CMIFLAG
-				0	0	0	0	0	0	0
-				RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function	
6	HIZIFLAG	Timer 30 Output High-Impedance Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
5	CIFLAG	Timer 30 Capture Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
4	BTIFLAG	Timer 30 Bottom Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
3	PMIFLAG	Timer 30 Period Match Flag Enable bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
2	AMIFLAG	Timer 30 A-ch Match Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
1	BMIFLAG	Timer 30 B-ch Match Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.
0	CMIFLAG	Timer 30 C-ch Match Interrupt Flag bit.	
		0	No request occurred.
		1	Request occurred, This bit is cleared to '0' when write '1'.

14.2.13 TIMER30_HIZCR: timer/counter 30 high-impedance control register

TIMER30_HIZCR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_HIZCR=0x4000_2430

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			HIZEN	Reserved		HIZSW	Reserved	HEDGE	HIZSTA	HIZCLR
-			0	-		0	-	0	0	0
-			RW	-		RW	-	RW	RO	RW

Bits	Name	Function	
7	HIZEN	PWM30xA/PWM30xB output high-impedance enable bit.	
		0	Disable to control the output high-impedance.
		1	Enable to control the output high-impedance.
4	HIZSW	High-impedance output software setting.	
		0	No effect.
		1	PWM30xA/PWM30xB pins go into high impedance. (Automatically cleared to "0b" after operation)
2	HEDGE	High-impedance edge selection.	
		0	Falling edge of the BLNK pin.
		1	Rising edge of the BLNK pin.
1	HIZSTA	High-impedance status.	
		0	Indicates that the pins are not under a Hi-Z state.
		1	Indicates that the pins are under a Hi-Z state.
0	HIZCLR	High-impedance output clear bit.	
		0	No effect.
		1	Clear high-impedance output. (The PWM30xA/PWM30xB pins are back to output and this bit is automatically cleared to "0b" after operation)

NOTE: Where x = A, B, and C.

14.2.14 TIMER30_ADTCR: timer/counter 30 A/DC trigger control register

TIMER30_ADTCR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_ADTCR=0x4000_2434

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5	4	3	2	1	0
Reserved				BTTG	PMTG	AMTG	BMTG	CMTG
-				0	0	0	0	0
-				RW	RW	RW	RW	RW

Bits	Name	Function	
4	BTTG	Select timer 30 bottom for ADC trigger signal generator.	
		0	Disable ADC trigger signal generator by bottom.
		1	Enable ADC trigger signal generator by bottom.
3	PMTG	Select timer 30 period match for ADC trigger signal generator.	
		0	Disable ADC trigger signal generator by period match.
		1	Enable ADC trigger signal generator by period match.
2	AMTG	Select timer 30 a-ch match for ADC trigger signal generator.	
		0	Disable ADC trigger signal generator by a-ch match.
		1	Enable ADC trigger signal generator by a-ch match.
1	BMTG	Select timer 30 b-ch match for ADC trigger signal generator.	
		0	Disable ADC trigger signal generator by b-ch match.
		1	Enable ADC trigger signal generator by b-ch match.
0	CMTG	Select timer 30 c-ch match for ADC trigger signal generator.	
		0	Disable ADC trigger signal generator by c-ch match.
		1	Enable ADC trigger signal generator by c-ch match.

NOTES:

1. A trigger signal generation is not related with the PMOC[2:0] bits of TIMER30_CR register.
2. If several source for trigger is selected, a signal can be lost in case of the trigger generation counter is reloaded by another signal.

14.2.15 TIMER30_ADTDR: timer/counter 30 A/DC trigger generator data register

TIMER30_ADTDR is a 32-bit register, and able to do 32/16/8-bit access.

TIMER30_ADTDR=0x4000_2438

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		ADTDATA	
-		0x0000	
-		RW	

Bits	Name	Function
13 0	CNT	Timer/Counter 30 A/DC Trigger Generation Data bits.

NOTES:

1. ADTDR of Timer30 uses the adcnt timer as tclk (timer30 clock).
2. The adcnt timer counter is a 14-bit down counter.
3. It count down from the value written in ADTDR to 0. (Delay role).

14.3 Functional description

14.3.1 Timer counter 30

The timer/counter 30 can be clocked by an internal or an external clock source (EC30). The clock source is selected by a clock selection logic which is controlled by the clock selection bits (CLK in TIMER30_CR).

- TIMER 30 clock source: {PCLK/(TIMER30_PREDR +1), EC30

In capture mode, by CAPD, data is captured into input capture data register (TIMER30_CAPDR).

The PWM wave form to PWM30AA, PWM30AB, PWM30BA, PWM30BB, PWM30CA, PWM3CB Port (6-channel).

Table 66. Timer 30 Operating Modes

TIMER30_CR <EN>	TIMER30_CR <MS>	TIMER30_PREDR	Description
1	00	0xXXX	16-bit Interval Mode
1	01	0xXXX	16-bit Capture Mode
1	10	0xXXX	16-bit back-to-back Mode

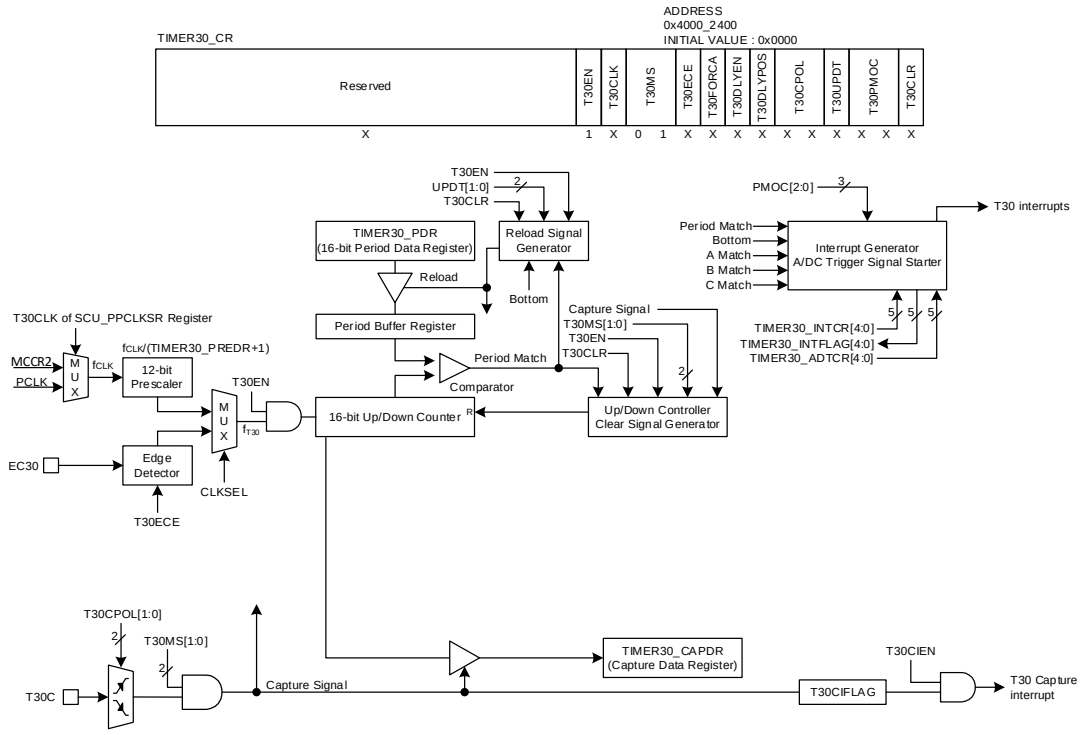
14.3.2 Timer 30 capture mode

16-bit timer 30 capture mode is set by configuring MS[1:0] in TIMER30_CR as '01'. An internal clock input or an external clock input can be used as a clock source. Basically, the 16-bit timer 30 capture mode has the same function as the 16-bit interval mode has. Interrupts occur when value of TIMER30's 16-bit up/down counter equals to the one of TIMER30_PDR. The 16-bit up/down counter values are automatically cleared by a match signal. It can be cleared by software (CLR in TIMER30_CR) too.

The 16-bit timer 30's interrupt in capture mode is very useful when the pulse width of captured signal is wider than the maximum period of the timer. The capture result is loaded into TIMER30_CAPDR.

Figure 81 shows 16-bit capture mode of the timer 30.

Figure 81. 16-bit Capture Mode of Timer 30

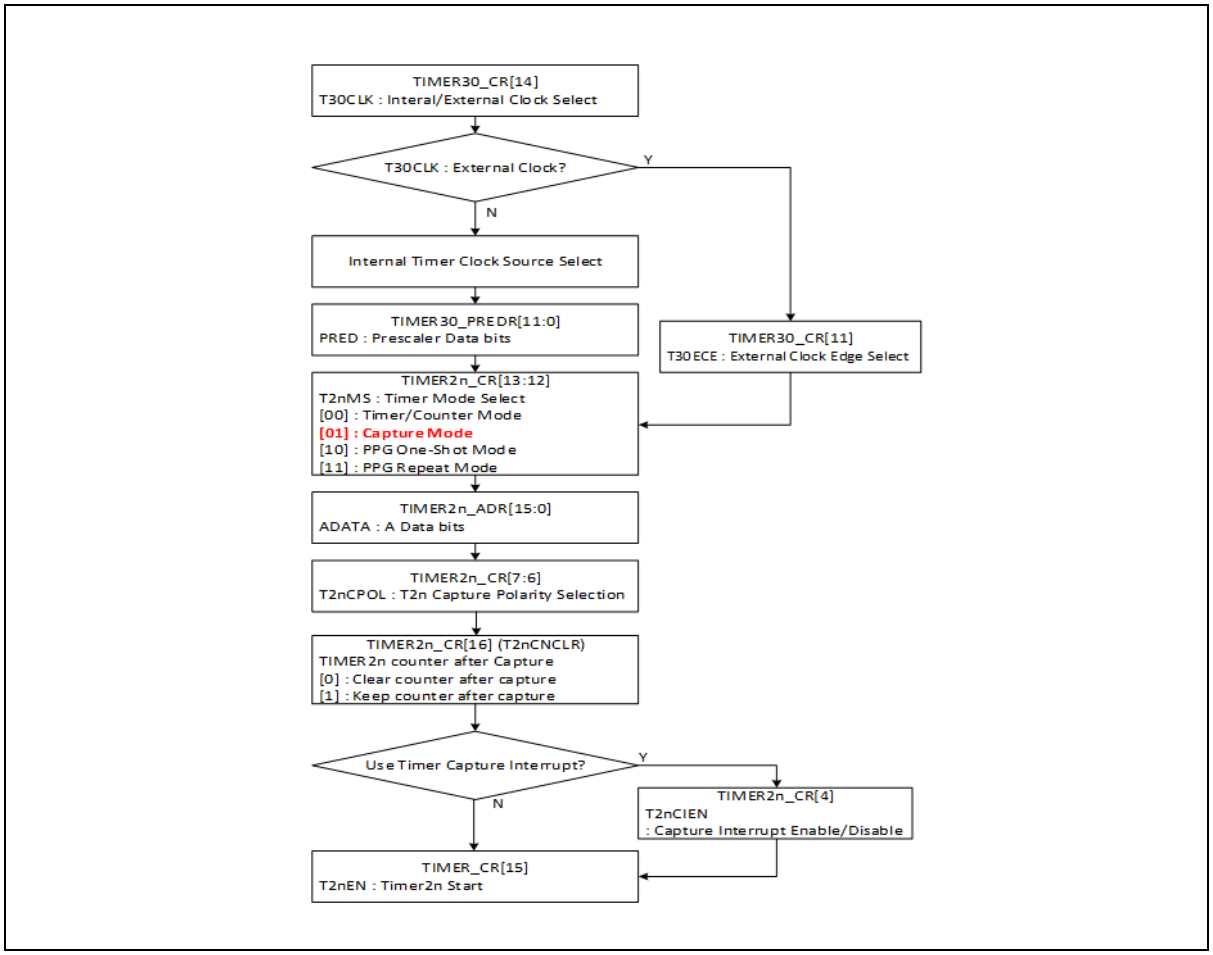


NOTES:

1. The period match only can occur in the capture mode.
2. The 16-bit up/down counter in the capture mode is cleared to "0x0000" after the counter value is loaded to the TIMER30_CAPDR register at valid edge.

Figure 82 shows the capture mode operation of 32-bit Timer30. Refer to Figure 13. Clock Change Procedure for internal timer clock source and section 5.2 Pin multiplexing for Timer 30 capture pin.

Figure 82. Capture Mode Operation Sequence of TIMER 30



14.3.3 Timer 30 interval mode

Timer 30 interval mode is set by configuring MS[1:0] in TIMER30_CR as '00'. The timer 30 has a counter and data registers. The 16-bit up/down counter is increased by an internal or an external clock input. The timer 30 can use the input clock with 12-bit prescaler division rates (TIMER30_PREDR[11:0]). When the value of TIMER30 16-bit up/down counter and the value of TIMER30_PDR are identical in timer 30, a match signal is generated and the period match interrupt of timer 30 is occurred. The period match interrupt can be occurred which once every 1, 2, 3, 4, 5, 6, 7, or 8 period match (PMOC[2:0]). The 16-bit up/down counter value is automatically cleared by match signal. It can be cleared by software (CLR in TIMER30_CR) too.

The timer 30 Interval mode can be operated for BLDC motor control. It has 6-channel pins output up to 16-bit resolution PWM output. When the value of 16-bit up/down counter and TIMER30_PDR are identical in timer 30, a period match signal is generated and the period match interrupt of timer 30 is occurred.

The timer 30 A, B, and C match signals are generated and the A, B, and C match interrupts of timer 30 are occurred, when the 16-bit counter value are identical to the value of TIMER30_xDR. The period and duty of the PWM output is determined by the TIMER30_PDR (PWM period register), and T3xDR (each channel PWM duty register).

- PWM Period = [TIMER30_PDR] X Source Clock
- PWM Duty(A-ch) = [TIMER30_ADR] X Source Clock
- PWM Duty(B-ch) = [TIMER30_BDR] X Source Clock
- PWM Duty(C-ch) = [TIMER30_CDR] X Source Clock

The POLA/POLB bit of TIMER30_OUTCR register decides the polarity of PWM output. If the POLA/POLB bit is set to '1', the PWM30xA/PWM30xB output is high level start, respectively. And if the POLA/POLB bit is cleared to '0b', the PWM30xA/PWM30xB output is low level start, respectively.

Table 67. PWM Channel Polarity

PxAOE	PxBOE	POLxA	POLxB	PWM3xA Pin put	PWM3xB Pin Output
1	1	0	0	Low level start	Low level start
		0	1	Low level start	High level start
		1	0	High level start	Low level start
		1	1	High level start	High level start

NOTE: Where x = A, B, and C.

Figure 83. Example of PWM at 4MHZ

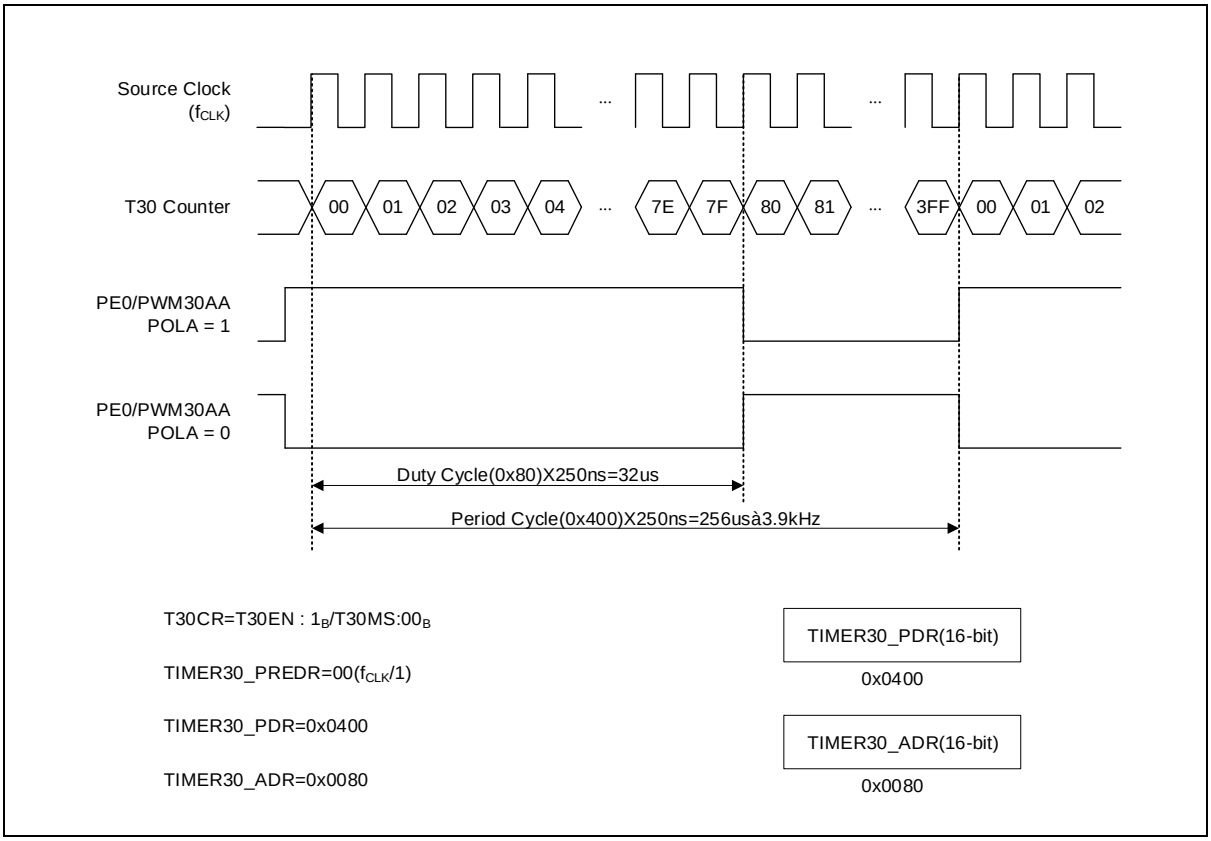
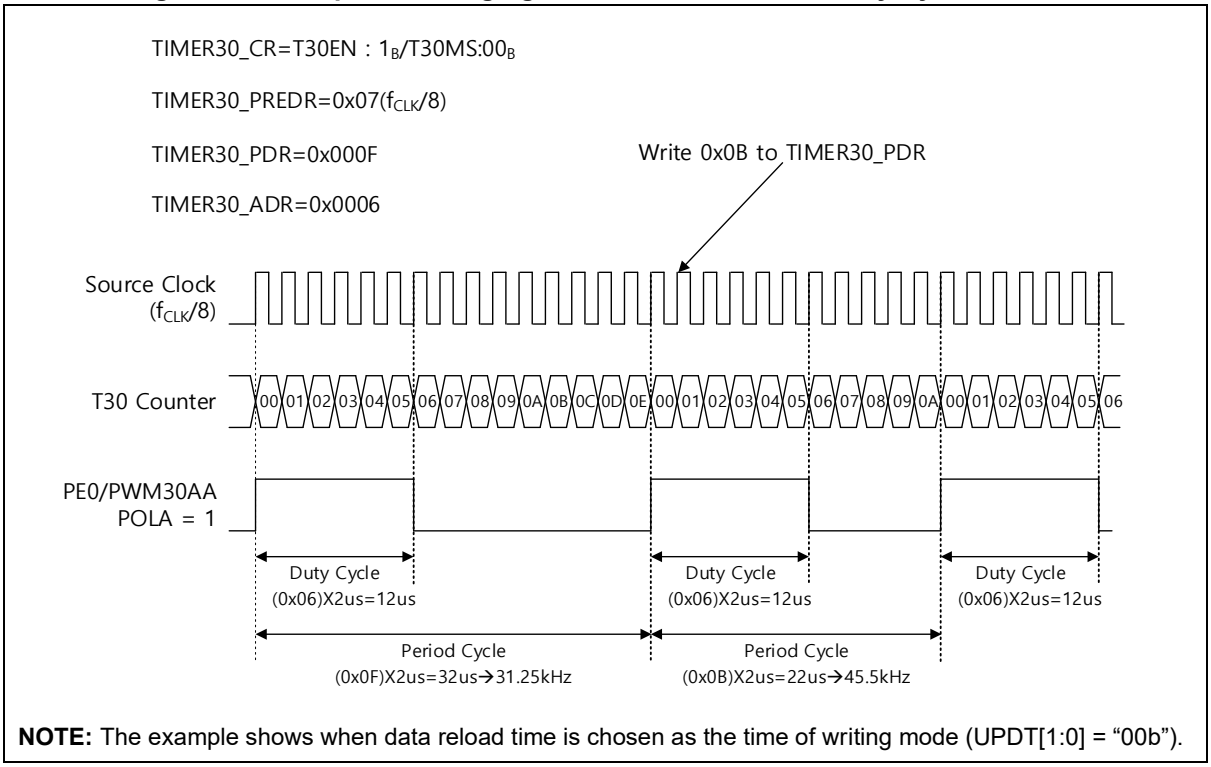


Figure 84. Example of Changing the Period in Absolute Duty Cycle at 4MHz



NOTE: The example shows when data reload time is chosen as the time of writing mode (UPDT[1:0] = "00b").

Data reload time selection

The data reload time can choose among "update data to buffer at the time of writing", "update data to buffer at period match", or "update data to buffer at bottom".

PWM output delay

If using the DLYEN bit, DLYPOS bit, and TIMER30_DLY register, it can delay the PWM output. The DLYPOS setting to '0', the delay inserts at front of PWM30xA and at back of PWM30xB pins. The DLYPOS setting to '1', the delay inserts at back of PWM30xA and at front of PWM30xB pins. The settings of DLYEN bit, DLYPOS bit, and TIMER30_DLY register are applied equally to all PWM channels.

Figure 85. Interval Mode Timing Chart With "DLYPOS = 0"

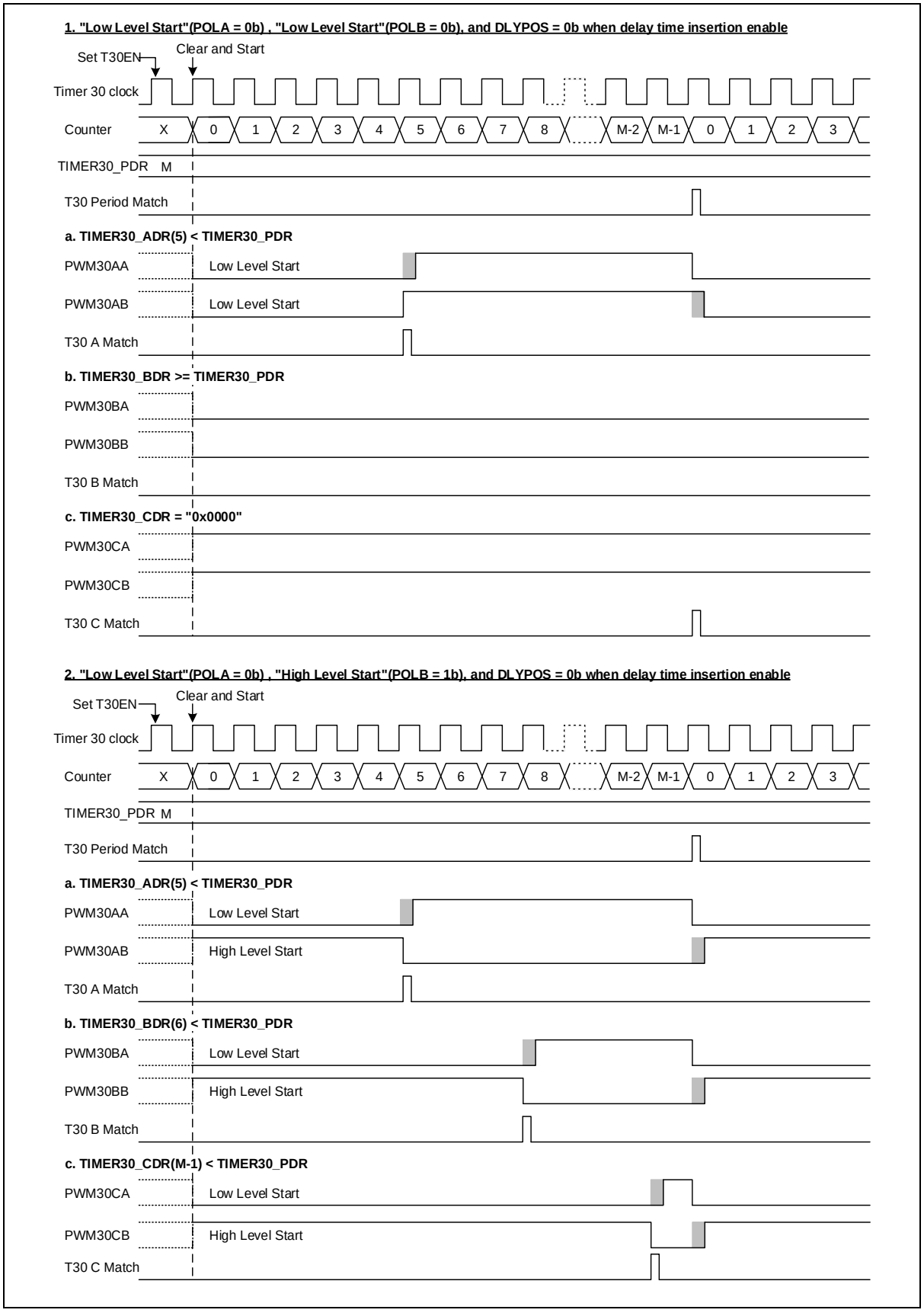
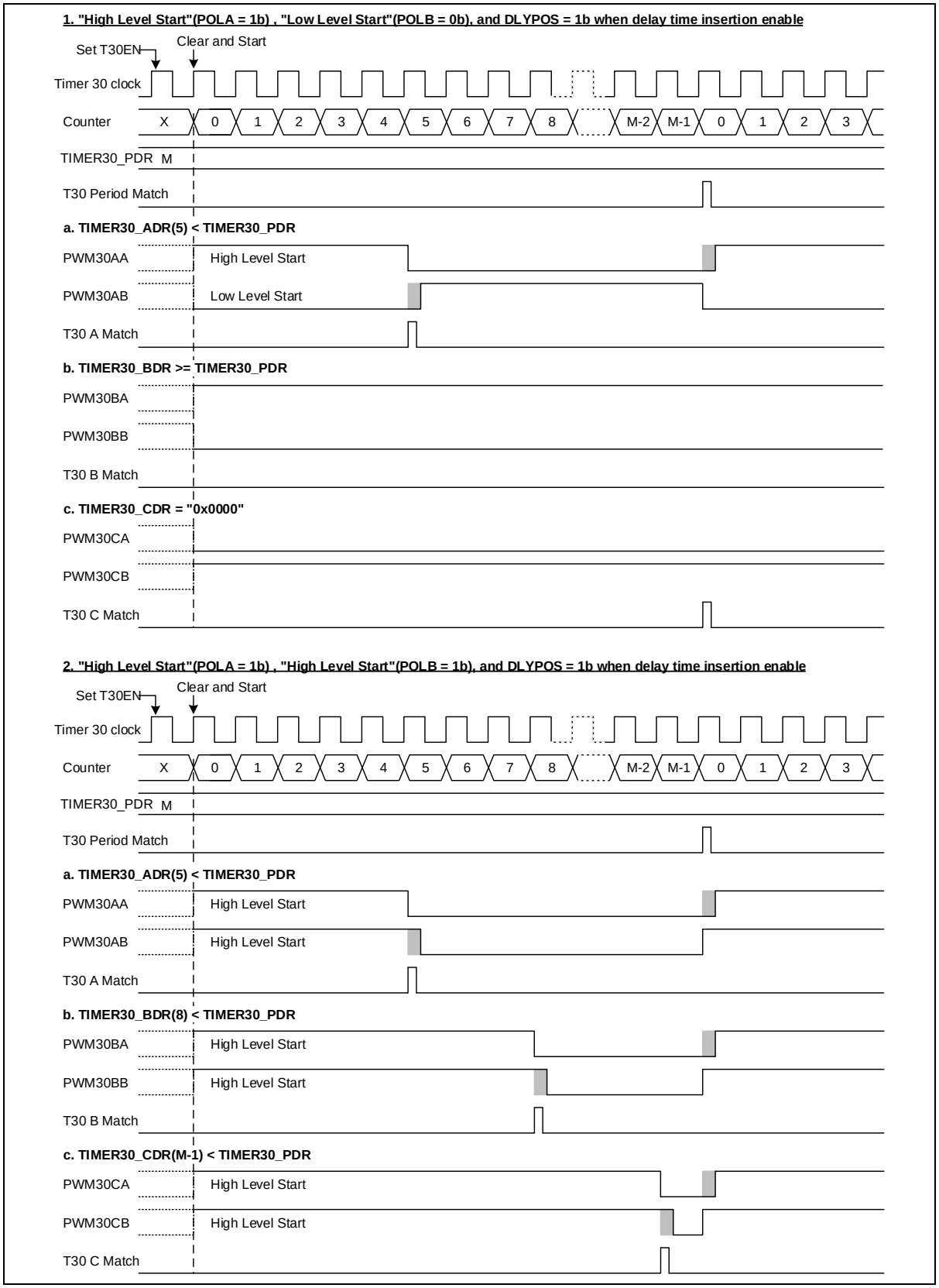


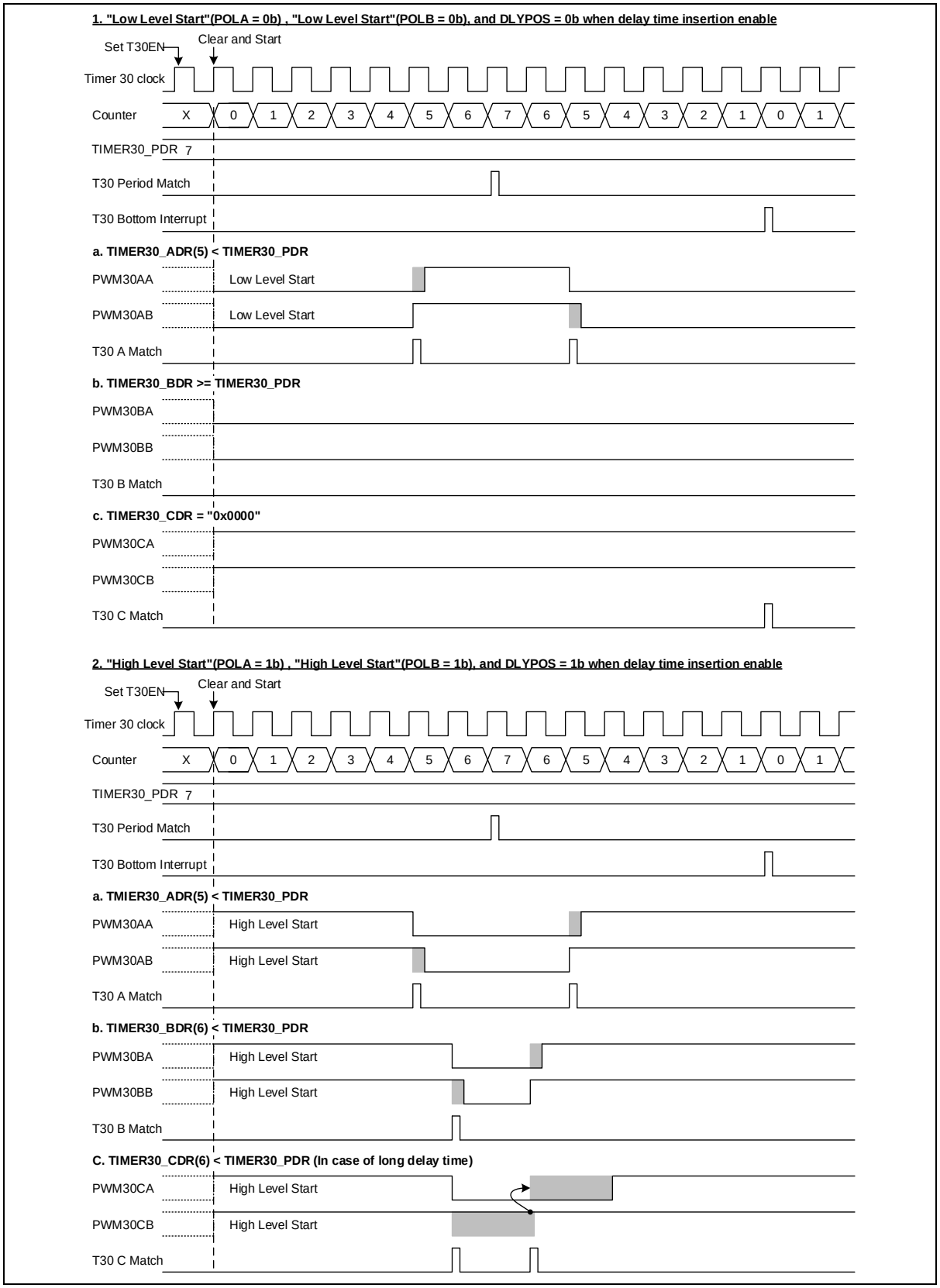
Figure 86. Interval Mode Timing Chart With "DLYPOS = 1



Back-to-back mode

The back-to-back mode is set by MS[1:0] in TIMER30_CR as '10'. In the back-to-back mode, the 16-bit up/down counter repeats up/down count. In fact, the effective duty and period becomes twofold of the register set values. If the TIMER30_PDR's data value is set to "0x3210, 16-bit up/down counter will increase until it reaches 0x3210. At this point, a period match signal is generated and the period match interrupt occurs. And then the 16-bit up/down counter will decrease until it reaches 0x0000. At this point, the bottom interrupt occurs. It is repeated in this way.

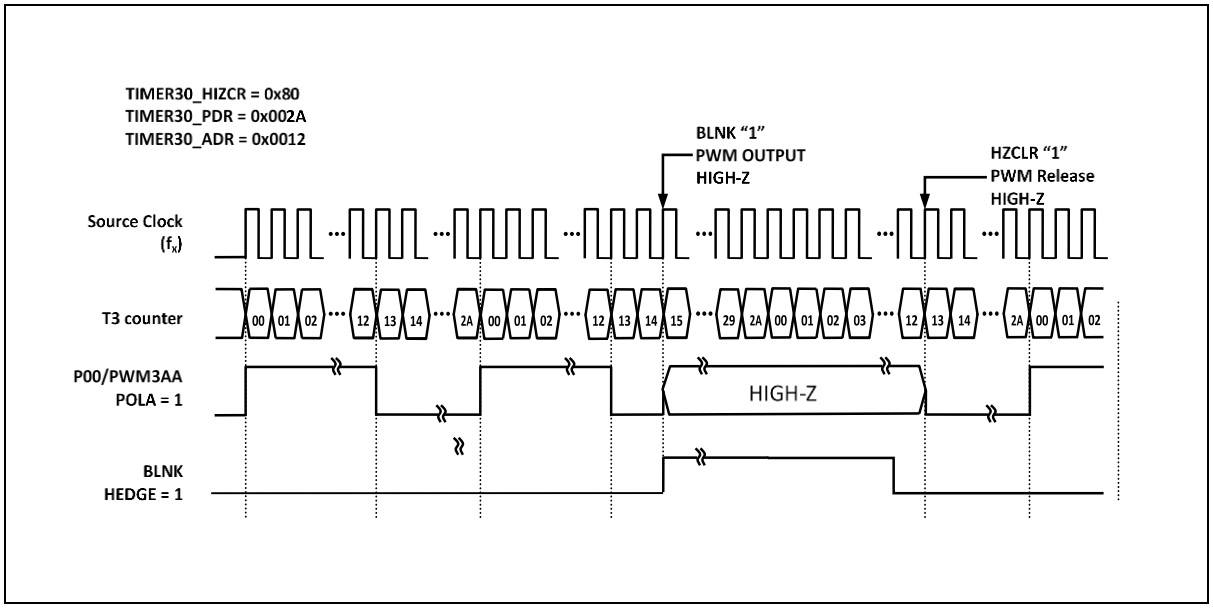
Figure 87. Back-to-Back Mode Timing Chart



Emergency protective function

This protective function is used for emergency stop, when the PWM30xA/PWM30xB output high-impedance enable bit, HIZEN is enabled. When the signal on the external BLNK input pin or internal comparator 3 output goes active (falling or rising edge triggered), the PWM30xA/PWM30xB ports are immediately disabled high-impedance against output and a high-impedance interrupt is occurred. The TIMER30_HIZCR register is used for high-impedance control. The high-impedance source is the external BLNK input pin. The high-impedance edge can be selected by HEDGE bit as falling or rising edge. If the HIZST read value is '1', it indicates that the pins are under a high-impedance state. To return from the high-impedance state, the HIZCLR bit set to '1'. If HIZSW bit is set to '1', PWM30xA/PWM30xB pins go into high impedance by software. It can be used for debugging. (x: A, B and C).

Figure 88. Example of PWM External Synchronization with BLNK Input (x: A, B and C)



Force A-channel mode

If FORCA bit is set to '1', it is possible to enable or disable all PWM output pins through PWM outputs which occur from A-ch duty counter. It is noted that the inversion outputs of A, B, C channel have the same A-ch output waveform.

Figure 89. Example of Force A-Channel Mode

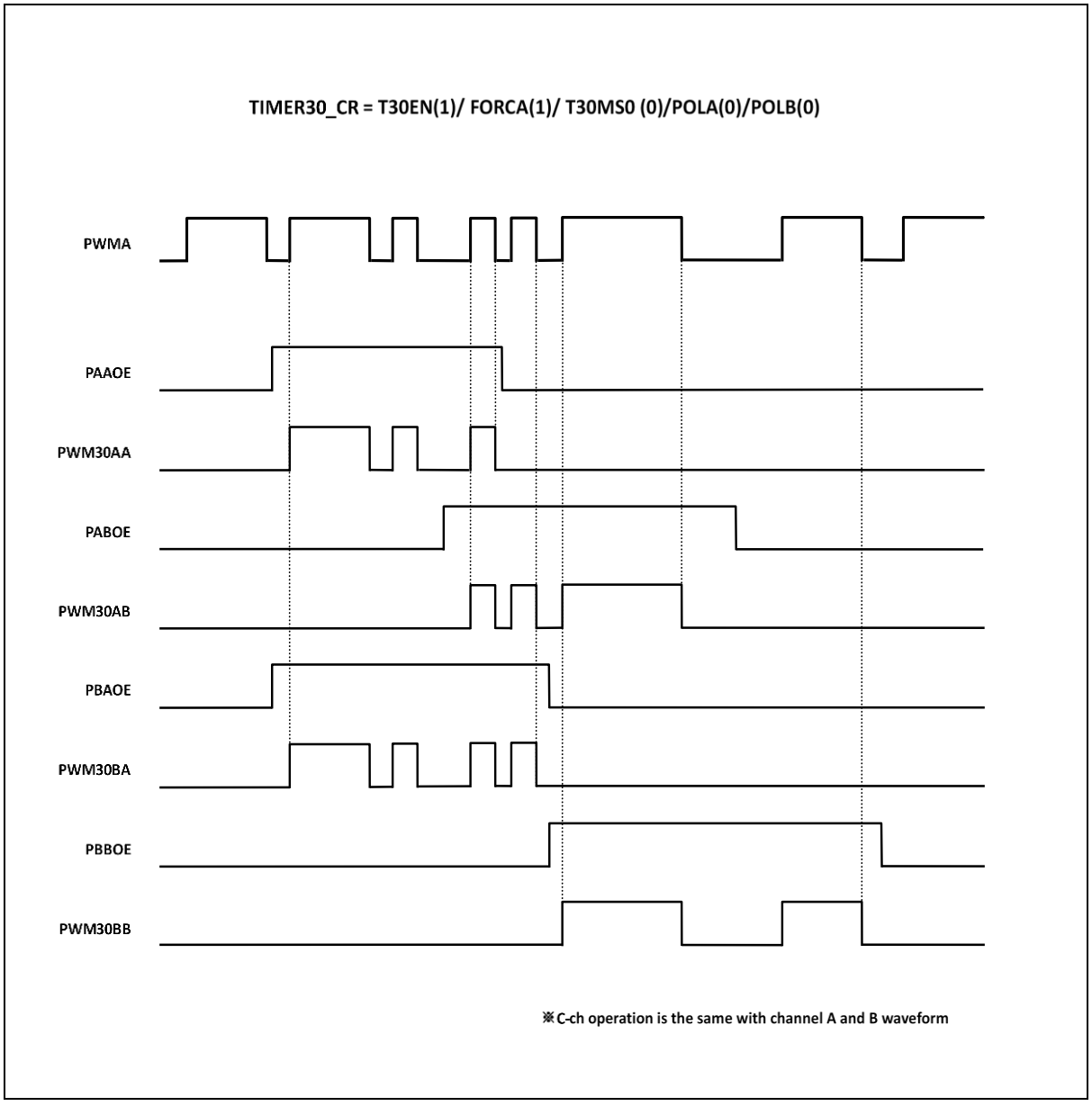
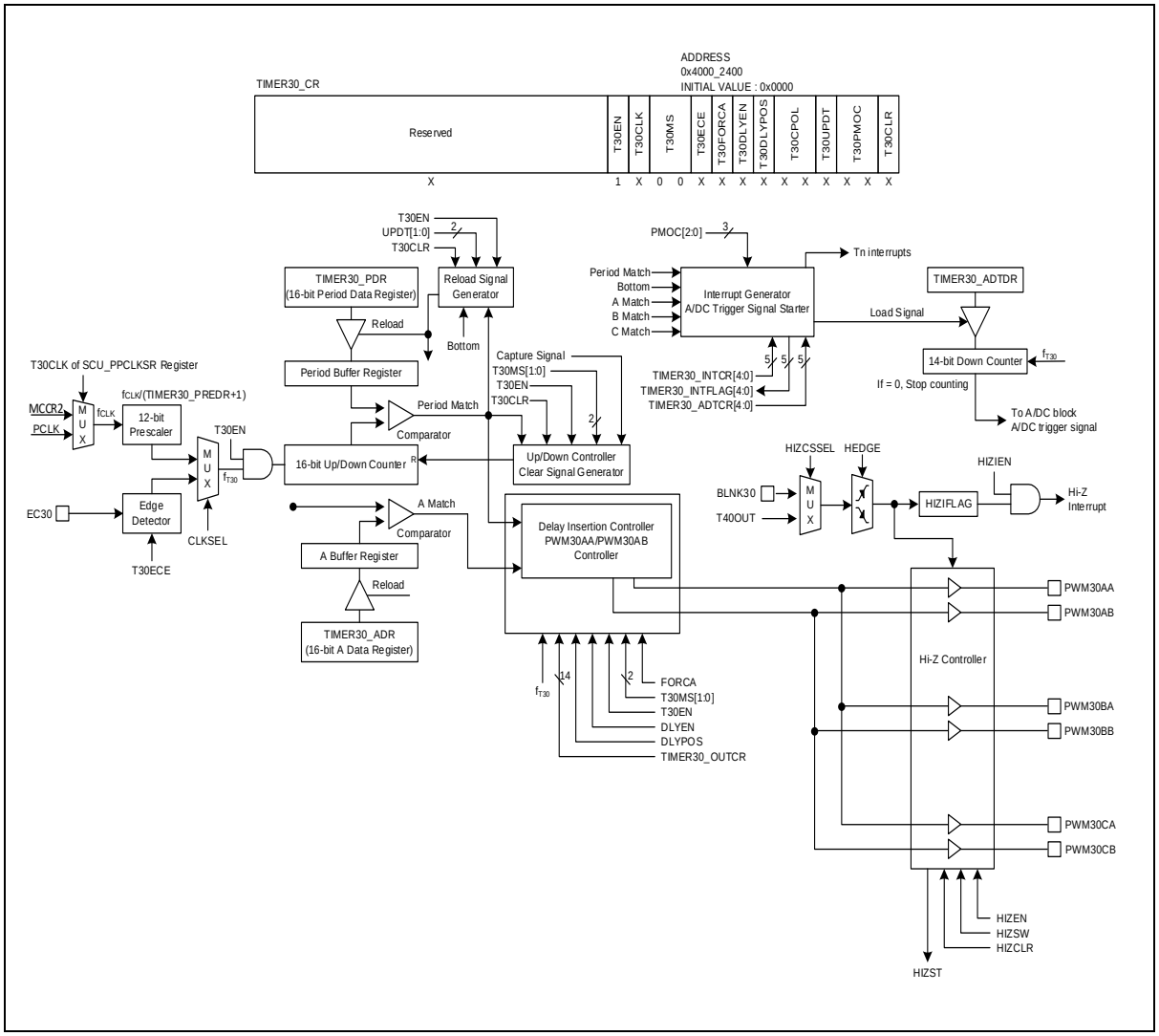


Figure 90. Force A-Channel Mode Block Diagram



6-channel mode

If FORCA bit sets to '0', it is possible to enable or disable PWM output pin and inversion output pin generated through the duty counter of each channel. The inversion output is the reverse phase of the PWM output. A AA/AB output of the A-channel duty register, a BA/BB output of the B-channel duty register, a CA/CB output of the C-channel duty register are controlled respectively.

Figure 91. Example of 6-Channel Mode

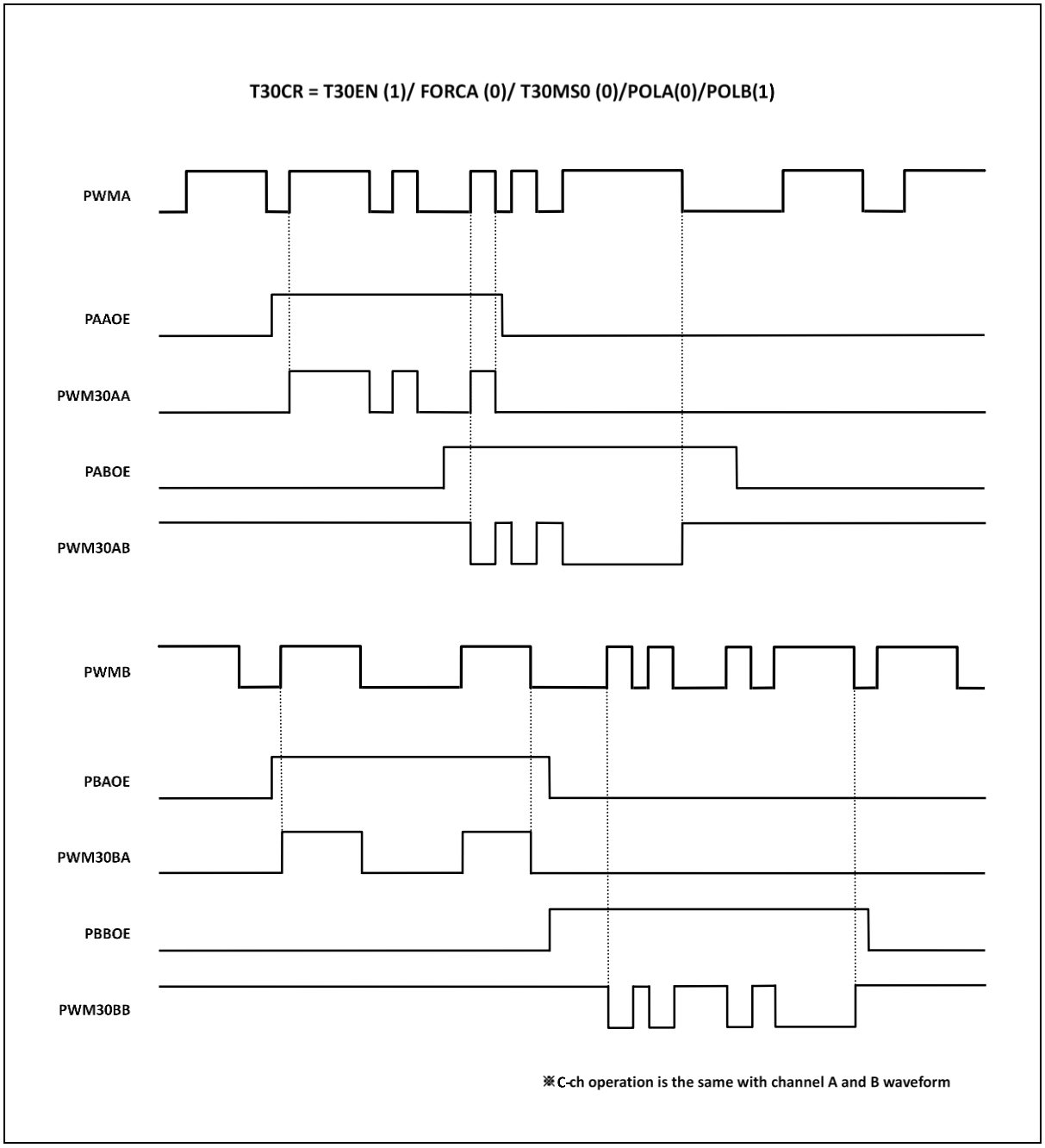
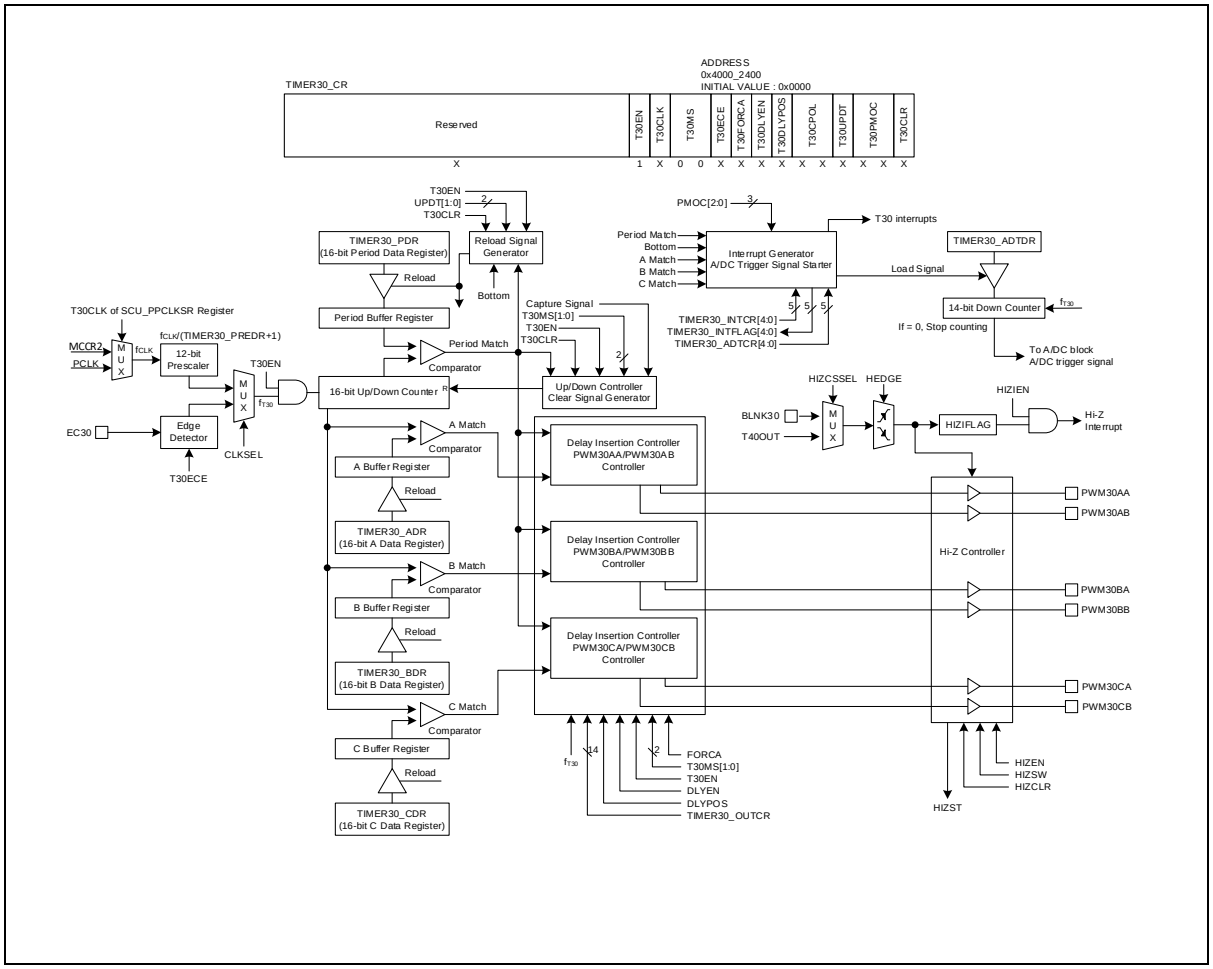


Figure 92. 6-Channel Mode Block Diagram



15 Universal Synchronous/Asynchronous Receiver/Transmitter (USART)

Universal Synchronous and Asynchronous serial Receiver and Transmitter (USART) is a highly flexible serial communication device. The main features are listed below:

- Full Duplex Operation. (Independent Serial Receive and Transmit Registers)
- Asynchronous or Synchronous Operation.
- Baud Rate Generator.
- Supports Serial Frames with 5,6,7,8, or 9 Data bits and 1 or 2 Stop bits.
- Odd or Even Parity Generation and Parity Check are Supported by Hardware.
- Data Over-Run Detection.
- Framing Error Detection.
- Three Separate Interrupts on TX Completion, TX Data Register Empty and RX Completion.
- Double Speed Asynchronous communication mode.

Additional features are:

- 0% Error Baud Rate by floating point count register.
- Supports receive time out interrupt.
- Supports direct memory access and interrupt.

Table 68 introduces pins assigned for the USART.

Table 68. Pin Assignment of USART: External Pins

Pin name	Type	Description
TXDn	O	UART Channel n transmit output
RXDn	I	UART Channel n receive input
SSn	I/O	SPIn Slave select input / output
SCKn	I/O	SPIn Serial clock input / output
MOSIn	I/O	SPIn Serial data (Master output, Slave input)
MISOn	I/O	SPIn Serial data (Master input, Slave output)

NOTE: n = 10, 11, 12 and 13

15.1 USART block diagram

In this section, USART and SPIN are introduced in block diagrams.

Figure 93. USART Block Diagram (n = 10, 11, 12, and 13)

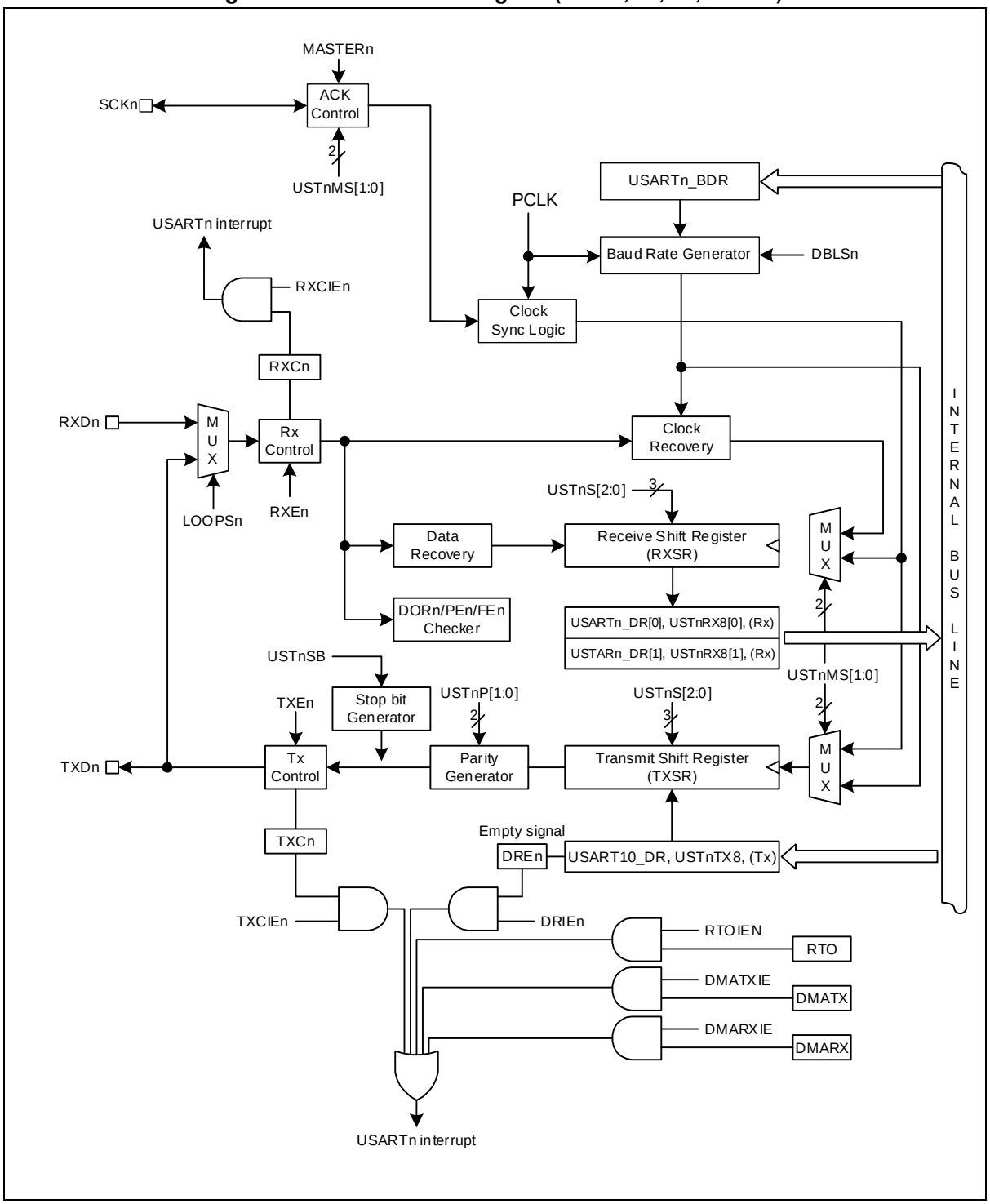
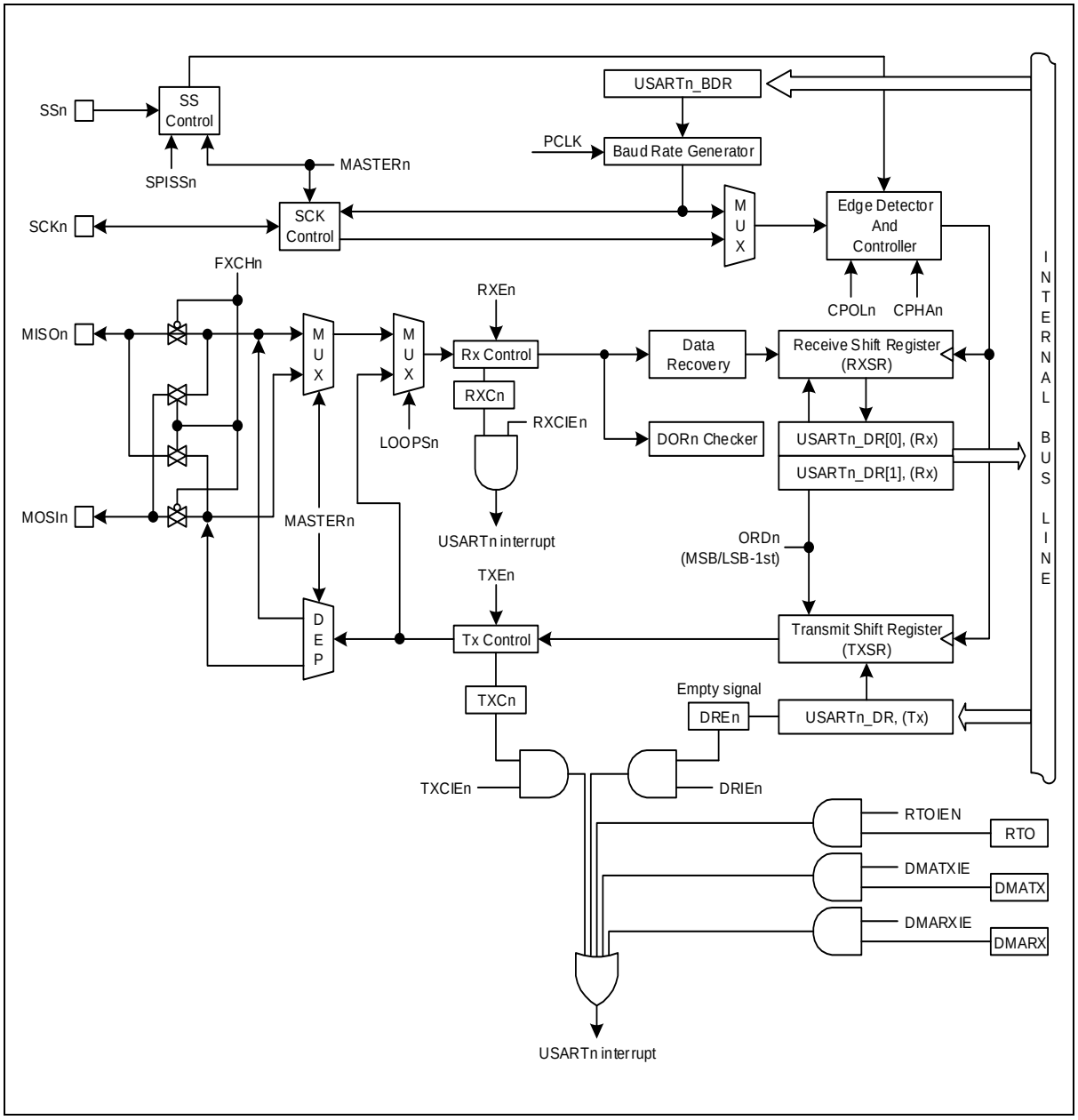


Figure 94. SPIN Block Diagram (n = 10, 11, 12, and 13)



15.2 Registers

Base address of USART is introduced in the followings:

Table 69. Base Address of USART

Name	Base address
USART 10	0x4000_3800
USART 11	0x4000_3900
USART 12	0x4000_3A00
USART 13	0x4000_3B00

Table 70. USART Register Map

Name	Offset	Type	Description	Reset value	Ref.
USARTn_CR1	0x00	RW	USARTn Control Register 1	0x0000_0000	15.2.1
USARTn_CR2	0x04	RW	USARTn Control Register 2	0x0000_0000	15.2.2
USARTn_ST	0x0C	RW	USARTn Status Register	0x0000_0080	15.2.3
USARTn_BDR	0x10	RW	USARTn Baud Rate Generation Register	0x0000_0FFF	15.2.4
USARTn_DR	0x14	RW	USARTn Data Register	0x0000_0000	15.2.5
USARTn_BFR	0x18	RW	USARTn Baud-Rate Fraction Counter Register	0x0000_0000	15.2.6
USARTn_RTO	0x1C	RW	USARTn Receive Time Out Register	0x00FF_FFFF	15.2.7

NOTE: n = 10, 11, 12 and 13

15.2.1 USARTn_CR1: USARTn control register 1

USART module should be configured properly before running. USARTn_CR1 is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_CR1=0x4000_3800, USART11_CR1=0x4000_3900
USART12_CR1=0x4000_3A00, USART13_CR1=0x4000_3B00

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		MS		PG		DLEN			ORD	CPOL	CPHA	DRIE	TXCIE	RXCIE	Reserved	TXE	RXE
-		00		00		000			0	0	0	0	0	0		0	0
-		RW		RW		RW			RW	RW	RW	RW	RW	RW		RW	RW

Bits	Name	Function			
15 14	MS	USARTn Operation Mode Selection bits.			
		00	Asynchronous Mode. (UART)		
		01	Synchronous Mode.		
		10	Reserved.		
		11	SPI Mode		
13 12	PG	Selects Parity Generation and Check method. (only UART mode)			
		00	No parity.		
		01	Reserved.		
		10	Even parity.		
		11	Odd parity.		
11 9	DLEN	Selects the length of data bit in a frame when Asynchronous or Synchronous mode.			
		000	5 bit.		
		001	6 bit.		
		010	7 bit.		
		011	8 bit.		
		111	9 bit.		
		Others	Reserved.		
8	ORD	Selects the first data bit to be transmitted. (only SPI mode)			
		0	LSB-first.		
		1	MSB-first.		
7	CPOL	Selects the clock polarity of ACK in synchronous or SPI mode.			
		0	TXD Change @Rising Edge, RXD Change @Falling Edge.		
		1	TXD Change @Falling Edge, RXD Change @Rising Edge.		
6	CPHA	The CPOL and this bit determine if data are sampled on the leading or trailing edge of SCK. (only SPI mode)			
		CPOL	CPHA	Leading edge	Trailing edge
		0	0	Sample (Rising)	Setup (Falling)

		0	1	Setup (Rising)	Sample (Falling)
		1	0	Sample (Falling)	Setup (Rising)
		1	1	Setup (Falling)	Sample (Rising)
NOTE: When the USARTn_CR1 register is set to CPOL = 0, CPHA = 1 / CPOL = 1, CPHA = 1 in Master mode, if data is continuously transmitted by more than 2 bytes, the communication is not available because the SCKn waveform is output abnormally. In this case, DMA function is also unavailable.					
5	DRIE	Transmit Data Register Empty Interrupt Enable bit.			
		0	Disable the transmit data empty interrupt.		
		1	Enable the transmit data empty interrupt.		
4	TXCIE	Transmit Complete Interrupt Enable bit.			
		0	Disable transmit complete interrupt.		
		1	Enable transmit complete interrupt.		
3	RXCIE	Receive Complete Interrupt Enable bit.			
		0	Disable receive complete interrupt.		
		1	Enable receive complete interrupt.		
1	TXE	Enables the Transmitter unit.			
		0	Transmitter is disabled.		
		1	Transmitter is enabled.		
0	RXE	Enables the Receiver unit.			
		0	Receiver is disabled.		
		1	Receiver is enabled.		
NOTE: The CPOL and CPHA bits should be changed during the TXE and RXE bits are "0b"					

15.2.2 USARTn_CR2: USARTn control register 2

USART module should be configured properly before running. USARTn_CR2 is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_CR2=0x4000_3804, USART11_CR2=0x4000_3904
USART12_CR2=0x4000_3A04, USART13_CR2=0x4000_3B04

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		DMATXIE	DMARXIE	RTOIE	RTOEN	BFREN	EN	DBLS	MASTER	LOOPS	DISSCK	SSEN	FXCH	SB	TX8	RX8	
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
-		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

Bits	Name	Function
14	DMATXIE	DMA TX Interrupt bit
		0 Disable DMA TX Interrupt
		1 Enable DMA TX Interrupt
13	DMARXIE	DMA RX Interrupt bit
		0 Disable DMA RX Interrupt
		1 Enable DMA RX Interrupt
12	RTOIE	RTO Interrupt bit
		0 Disable RTO Interrupt
		1 Enable RTO Interrupt
11	RTOEN	Activate RTO Block by supplying.
		0 Disable RTO
		1 Enable RTO
10	BFREN	Activate Baud Rate Fraction Counter Register
		0 Disable Fraction Counter Register
		1 Enable Fraction Counter Register.
9	EN	Activate USARTn Block by supplying.
		0 Disable USARTn block.
		1 Enable USARTn block.
8	DBLS	Selects receiver sampling rate. (only UART mode)
		0 Normal asynchronous operation.
		1 Double speed asynchronous operation.
7	MASTER	Selects master or slave in SPIn or synchronous mode and controls the direction of SCKn pin.
		0 Slave operation. (External clock for SCKn)
		1 Master operation. (Internal clock for SCKn)
6	LOOPS	Control the Loop Back mode of USARTn for test mode.
		0 Normal operation.
		1 Loop Back mode.

5	DISSCK	In synchronous mode operation, selects the waveform of SCKn output.	
		0	SCKn is free-running while USARTn is enabled in synchronous master mode.
		1	SCKn is active while any frame is on transferring.
4	SSEN	This bit controls the SSn pin operation. (only SPI mode)	
		0	Disable.
		1	Enable.
3	FXCH	SPIn port function exchange control bit. (only SPI mode)	
		0	No effect.
		1	Exchange MOSIn and MISOn function.
2	SB	Selects the length of stop bit in asynchronous or synchronous mode.	
		0	1 Stop bit.
		1	2 Stop bit.
1	TX8	The ninth bit of data frame in asynchronous or synchronous mode of operation. Write this bit first before loading the USARTn_DR register.	
		0	MSB (9^{th} bit) to be transmitter is '0'.
		1	MSB (9^{th} bit) to be transmitter is '1'.
0	RX8	The ninth bit of data frame in asynchronous or synchronous mode of operation. Read this bit first before reading the receive buffer (only UART mode)	
		0	MSB (9^{th} bit) to be received is '0'.
		1	MSB (9^{th} bit) to be received is '1'.

15.2.3 USARTn_ST: USARTn status register

USARTn_ST is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_ST =0x4000_380C, USART11_ST =0x4000_390C
USART12_ST =0x4000_3A0C, USART13_ST =0x4000_3B0C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10	9	8	7	6	5	4	3	2	1	0
Reserved			DMATXF	DMARXF	DRE	TXC	RXC	Reserved	RTOF	DOR	FE	PE
-			0	0	1	0	0	-	0	0	0	0
-			RW	RW	RW	RW	RO	-	RW	RO	RW	RW

Bits	Name	Function	
9	DMATXF	DMA Transmit Operation Complete flag. (DMA to USART)	
		0	DMA Transmit Operation is working or is disabled
		1	DMA Transmit Operation is done
8	DMARXF	DMA Receive Operation Complete flag. (USART to DMA)	
		0	DMA Receive Operation is working or is disabled
		1	DMA Transmit Op is done
7	DRE	Transmit Data Register Empty Interrupt Flag. The DRE flag indicates if the transmit data register (USARTn_DR) is ready to receive new data. If DRE is '1', the data register is empty and ready to be written.	
		0	Transmit buffer is not empty.
		1	Transmit buffer is empty. This bit is cleared to '0' when write '1'.
6	TXC	Transmit Complete Interrupt Flag. This flag is set when the entire frame in the transmit shift register has been shifted out and there is no new data currently present in the transmit buffer.	
		0	No request occurred.
		1	Transmit buffer is empty and the data in transmit shift register are shifted out completely. This bit is cleared to '0' when write '1'.
5	RXC	Receive Complete Interrupt Flag. This flag is set when there are unread data in the receive buffer and cleared when all the data in the receive buffer are read.	
		0	There is no data unread in the receive buffer.
		1	There are more than 1 data in the receive buffer.
3	RTOF	Receive Time Out Interrupt flag. This bit is cleared to '0' when write '1'.	
		0	Receive time out is not generated
		1	Receive time out is generated
2	DOR	This bit is set if data Over-Run occurs. While this bit is set, the incoming data frame is ignored. This flag is valid until the receive buffer is read.	
		0	No Data Over-Run.
		1	Data Over-Run detected.
1	FE	This bit is set if the first stop bit of next character in the receive buffer is detected as '0'. This bit is valid until the receive buffer is read. (only UART mode)	
		0	No Frame Error.
		1	Frame Error detected.

0	PE	This bit is set if the next character in the receive buffer has a Parity Error while parity is checked. This bit is valid until the receive buffer is read. (only UART mode)	
		0	No Parity Error.
		1	Parity Error detected.

15.2.4 USARTn_BDR: USARTn baud rate generation register

USARTn_BDR is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_BDR =0x4000_3810, USART11_BDR =0x4000_3910
USART12_BDR =0x4000_3A10, USART13_BDR =0x4000_3B10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			BDATA
-			0xFFF
-			RW

Bits	Name	Function
11 0	BDATA	The value in this register is used to generate internal baud rate in UART mode or to generate SCK clock in SPI mode. To prevent malfunction, do not write '0' in UART mode and do not write '0' or '1' in synchronous or SPI mode.

15.2.5 USARTn_DR: USARTn data register

USARTn_DR is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_DR =0x4000_3814, USART11_DR =0x4000_3914
USART12_DR =0x4000_3A14, USART13_DR =0x4000_3B14

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			DATA
-			0x00
-			RW

Bits	Name	Function
7 0	DATA	The USART Transmit buffer and Receive buffer share the same I/O address with this DATA register. Tx: The Transmit Data Buffer is the destination for data written to the USARTn_DR register. Rx: Reading the USARTn_DR register returns the contents of the Receive Buffer. Write to this register only when the DRE flag is set. In SPI master mode, the SCK clock is generated when data are written to this register.

NOTE: This byte won't be written when the block is disabled or the both of TXE and RXE bits in USARTn_CR1 are "0b".

15.2.6 USARTn_BFR: USARTn baud rate fraction count register

USARTn_BFR is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_FPCR =0x4000_3818, USART11_FPCR =0x4000_3918
USART12_FPCR =0x4000_3A18, USART13_FPCR =0x4000_3B18

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			BFC
-			0xFFF
-			RW

Bits	Name	Function
7 0	BFC	USARTn baud rate fraction counter 8-bit floating point counter

15.2.7 USARTn_RTO: USARTn RTO register

USARTn_RTO is a 32-bit register, and able to do 32/16/8-bit access (n = 10, 11, 12, and 13).

USART10_RTO =0x4000_381C, USART11_RTO =0x4000_391C
USART12_RTO =0x4000_3A1C, USART13_RTO =0x4000_3B1C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	RTO		
-	0xFFFFFF		
-	RW		

Bits	Name	Function
23 0	RTO	USART receive time out register

15.3 Functional description

USART comprises clock generator, transmitter and receiver.

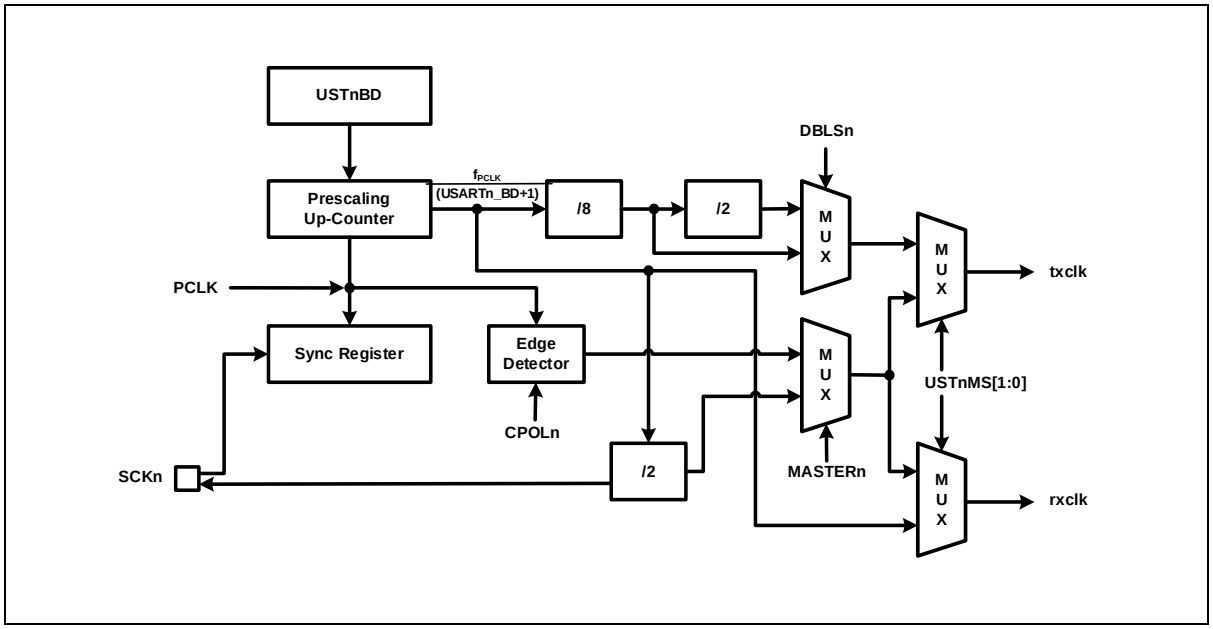
The clock generation logic consists of synchronization logic for external clock input used by synchronizing or SPI slave operation, and the baud rate generator for asynchronous or master (synchronous or SPI) operation.

The transmitter consists of a single write buffer, a serial shift register, parity generator and control logic for handling different serial frame formats. The write buffer allows continuous transfer of data without any delay between frames.

The receiver is the most complex part of the UART module due to its clock and data recovery units. The recovery unit is used for asynchronous data reception. In addition to the recovery unit, the receiver includes a parity checker, a shift register, a two-level receive FIFO (USARTn_DR) and control logic. The receiver supports the same frame formats as the transmitter and can detect frame error, data overrun and parity errors (n = 10, 11, 12 and 13).

15.3.1 USART clock generation

Figure 95. Clock Generation Block Diagram (USARTn, n = 10, 11, 12, and 13)



The clock generation logic generates the base clock for the transmitter and receiver. The USART supports four modes of clock operation and those are normal asynchronous, double speed asynchronous, master synchronous and slave synchronous modes.

The clock generation scheme for master SPI and slave SPI mode is the same as master synchronous and slave synchronous operation mode.

MS[1:0] bits in USARTn_CR1 register selects asynchronous or synchronous operation. Asynchronous double speed mode is controlled by the DBLS[8] bit in the USARTn_CR2 register.

MASTER[7] bit in USARTn_CR2 register controls whether the clock source is internal (master mode, output pin) or external (slave mode, input pin). The SCKn pin is active only when the USART operates in synchronous or SPI mode.

Table 71 shows the equations for calculating baud rate (in bps).

Table 71. Equations for Calculating USART Baud Rate Register Settings

Operating Mode	Equation for Calculating Baud Rate
Asynchronous Normal Mode (DBLS[8]=0)	Baud Rate= PCLK/(16(USARTn_BDR+1))
Asynchronous Double Speed Mode (DBLS[8]=1)	Baud Rate= PCLK/(8(USARTn_BDR+1))
Synchronous or SPI Master Mode	Baud Rate= PCLK/(2(USARTn_BDR+1))

NOTE: n = 10, 11, 12, and 13

15.3.2 External clock (SCKn)

External clocking is used in the synchronous or SPI slave mode of operation.

External clock input from the SCKn pin is sampled by a synchronization logic to remove meta-stability. Output from the synchronization logic must be passed through an edge detector before it is used by the transmitter and receiver. This process introduces two CPU clock period delay. The maximum frequency of the external SCKn pin is limited by 1MHz.

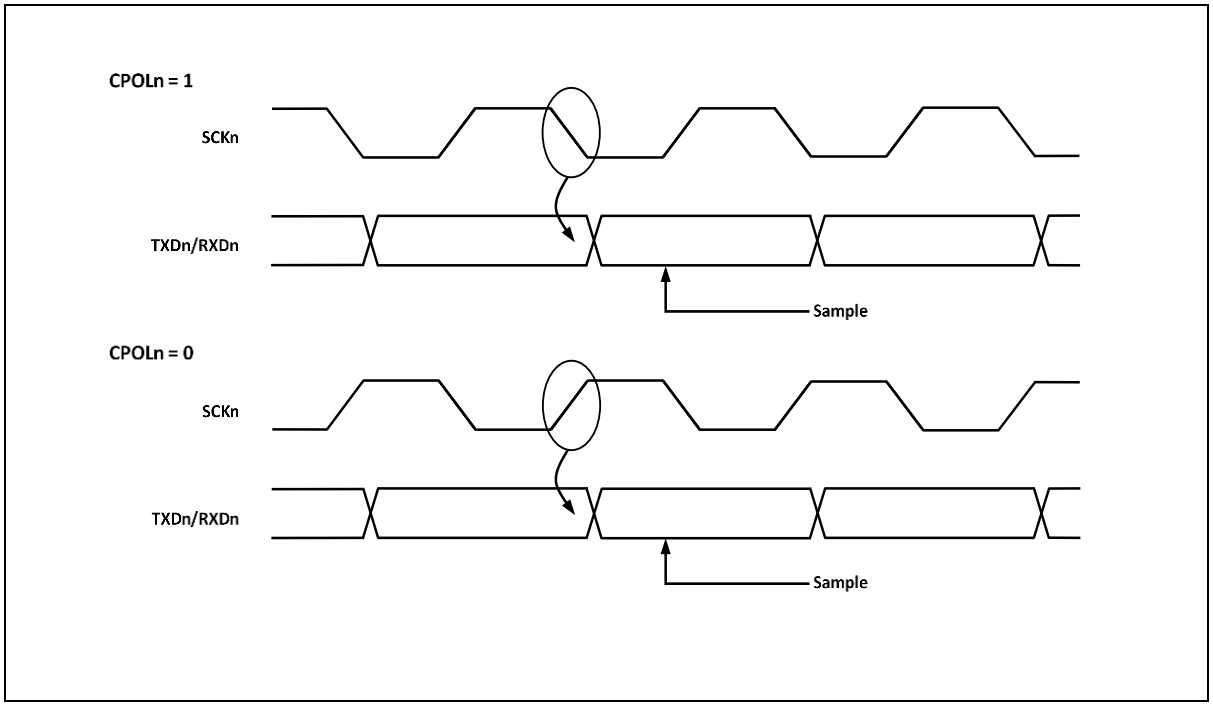
15.3.3 Synchronous mode operation

External clocking is used in the synchronous or SPI slave mode of operation.

When synchronous or SPI mode is used, the SCKn pin will be used as either clock input (slave) or clock output (master). Data sampling and transmitter are issued on the different edge of SCKn clock respectively. For example, if data input on RXDn (MISOn in SPI mode) pin is sampled on the rising edge of SCKn clock, data output on TXDn (MOSIn in SPI mode) pin is altered on the falling edge.

CPOL[7] bit in USARTn_CR1 register selects which SCKn clock edge is used for data sampling and which is used for data change. As shown in Figure 96, when CPOL[7] is zero, the data will be changed at rising edge of SCKn and sampled at falling edge of SCKn.

Figure 96. Synchronous Mode SCKn Timing (USARTn, n = 10, 11, 12, and 13)



15.3.4 UART data format

A serial frame is defined to be one character of data bits with synchronization bits (start and stop bits), and optionally a parity bit for error detection. USART supports all 30 combinations of the following as valid frame formats.

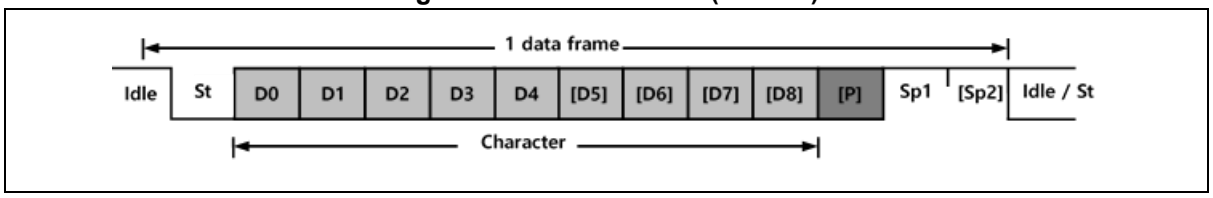
- 1 start bit
- 5, 6, 7, 8 or 9 data bits
- None, even or odd parity bit.
- 1 or 2 stop bits.

A frame starts with a start bit followed by the least significant data bit (LSB). Next data bits, up to nine, are succeeding, ending with the most significant bit (MSB). If parity function is enabled, the parity bit is inserted between the last data bit and the stop bit.

A high-to-low transition on data pin is considered as start bit. When a complete frame is transmitted, it can be directly followed by a new frame, or the communication line can be set to an idle state. The idle means high state of data pin.

Figure 97 shows possible combinations of the frame formats. Bits inside brackets are optional.

Figure 97. Frame Format (USART)



Single data frame consists of the following bits:

- Idle: No communication on communication line (TXDn/RXDn)
- St: Start bit (Low)
- Dm: Data bits (0~8)
- Parity bit: Even parity, Odd parity, No parity
- Stop bit(s): 1 bit or 2 bits

Frame format of UART is set by configuring DLEN[11:9], PE[13:12] bits in USARTn_CR1 register and SB[2] bit in USARTn_CR2 register. Transmitter and receiver use the same figures (n = 10, 11, 12 and 13).

15.3.5 UART parity bit

Parity bit is calculated by doing an exclusive-OR of all data bits. If odd parity is used, the result of the exclusive-OR is inverted. Parity bit is located between the MSB and first stop bit of a serial frame.

- Peven = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 0
- Podd = Dm-1 ^ … ^ D3 ^ D2 ^ D1 ^ D0 ^ 1
- Peven: Parity bit using even parity
- Podd: Parity bit using odd parity
- Dm: Data bit n of the character

15.3.6 UART transmitter

UART transmitter is enabled by configuring TXE[1] bit in USARTn_CR1 register. When the transmitter is enabled, TXDn pin should be set to TXDn function for the serial output pin of UART by the PB_AFSR1/PD_AFSR1 and PB_MOD/PD_MOD. Baud rate, operation mode and frame format must be set up once before starting any transmission.

In synchronous operation mode, SCKn pin is used as a transmission clock, so it should be selected to function SCKn by PB_AFSR1/PD_AFSR1 and PB_MOD/PD_MOD (n = 10, 11, 12 and 13).

UART sending TX data

A data transmission is initiated by loading the transmit buffer (USARTn_DR register I/O location) with the data to be transmitted. The data be written in transmit buffer is moved to the shift register when the shift register is ready to send a new frame. The shift register is loaded with the new data if it is in idle state or immediately after the last stop bit of the previous frame is transmitted. When the shift register is loaded with new data, it will transfer one complete frame according to the settings of control registers. If the 9-bit characters are used in asynchronous or synchronous operation mode, the ninth bit must be written to the TX8[1] bit in USARTn_CR2 register before it is loaded to the transmit buffer USARTn_DR register (n = 10, 11, 12 and 13).

UART transmitter flag and interrupt

The USART transmitter has two flags which indicate its state. One is USART data register empty flag (DRE[7]) and the other is transmit completion flag (TXC[6]). Both flags can be interrupt sources.

DRE[7] flag indicates whether the transmit buffer is ready to receive new data. This bit is set when the transmit buffer is empty and cleared when the transmit buffer contains data to be transmitted but has not yet been moved into the shift register. And also this flag can be cleared by writing '0' to this bit position. Writing '1' to this bit position is prevented.

When the data register empty interrupt enable (DRIE[5]) bit in USARTn_CR1 register is set and the global interrupt is enabled, USARTn_ST status register empty interrupt is generated while DRE[7] flag is set.

The transmit complete (TXC[6]) flag bit is set when the entire frame in the transmit shift register has been shifted out and there is no more data in the transmit buffer. The TXC[6] flag is automatically cleared when the transmit complete interrupt serve routine is executed, or it can be cleared by writing '0' to TXC[6] bit in USARTn_ST register.

When the transmit complete interrupt enable (TXCIE[4]) bit in USARTn_CR1 register is set and the global interrupt is enabled, UART transmit complete interrupt is generated while TXC[6] flag in USARTn_ST register is set (n = 10, 11, 12 and 13).

UART parity generator

The parity generator calculates the parity bit for the serial frame data to be sent. When parity bit is enabled by setting PG[13:12] bit value to '10' (even parity) or '11' (odd parity) in USARTn_CR1 register, the transmitter control logic inserts the parity bit between the MSB and the first stop bit of the frame to be sent.

UART disabling transmitter

Disabling the transmitter by clearing the TXE[1] bit in USARTn_CR1 will not become effective until ongoing transmission is completed. When the transmitter is disabled, the TXDn pin can be used as a normal general purpose I/O (n = 10, 11, 12 and 13).

15.3.7 UART receiver

USART receiver is enabled by setting the RXE[0] bit in the USARTn_CR1 register. When the receiver is enabled, the RXDn pin should be set to RXDn function for the serial input pin of UART by PB_AFSR1/PD_AFSR1 and PB_MOD/PD_MOD. The baud-rate, mode of operation and frame format must be set before serial reception. In synchronous or SPI operation mode the SCKn pin is used as transfer clock input, so it should be selected to do SCKn function by PB_AFSR1/PD_AFSR1 and PB_MOD/PD_MOD. In SPI operation mode the SSn input pin in slave mode can be configured as SSn output pin in master mode. This can be done by setting SSEN[4] bit in USARTn_CR2 register (n = 10, 11, 12 and 13).

UART receiving RX data

When UART is in synchronous or asynchronous operation mode, the receiver starts data reception when it detects a valid start bit (LOW) on RXDn pin. Each bit after start bit is sampled at pre-defined baud-rate (asynchronous) or sampling edge of SCKn (synchronous), and shifted into the receive shift register until the first stop bit of a frame is received. Even if there's the second stop bit in the frame, the second stop bit is ignored by the receiver. That is, receiving the first stop bit means that a complete serial frame is presented in the receiver shift register and contents of the shift register are to be moved into the receive buffer. The receive buffer is read by reading the USARTn_DR register.

If 9-bit characters are used (DLEN[2:0] = "111"), the ninth bit is stored in the RX8[0] bit position in the USARTn_CR2 register. The ninth bit must be read from the RX8[0] bit before reading the low 8 bits from the USARTn_TDR register. Likewise, the error flags Fen, DOR[2], PE[0] bit must be read before reading the data from USARTn_DR register. It's because the error flags are stored in the same FIFO position of the receive buffer (n = 10, 11, 12 and 13).

UART receiver flag and interrupt

The UART receiver has one flag that indicates the receiver state.

A receive complete (RXC[5]) flag indicates whether there are unread data in the receive buffer. This flag is set when there is unread data in the receive buffer and cleared when the receive buffer is empty. If the receiver is disabled (RXE[0] = 0), the receiver buffer is flushed and the RXC[5] flag is cleared.

When a receive complete interrupt enable (RXCIE[3]) bit in the USARTn_CR1 register is set and global interrupt is enabled, the UART receiver complete interrupt is generated while RXC[5] flag is set.

The UART receiver has three error flags which are frame error (FE[2]), data overrun (DOR[2]) and parity error (PE[0]). These error flags can be read from the USARTn_ST register. As received data is stored in the 2-level receive buffer, these error flags are also stored in the same position of receive buffer. So, before reading received data from USARTn_DR register, read the USARTn_ST register first which contains error flags.

The frame error (FE[1]) flag indicates the state of the first stop bit. The Fen flag is "0" when the stop bit was correctly detected as "1", and the Fen flag is "1" when the stop bit was incorrect, i.e. detected as "0". This flag can be used for detecting out-of-sync conditions between data frames.

The data overrun (DOR[2]) flag indicates data loss due to a receive buffer full condition. DOR[2] occurs when the receive buffer is full, and another new data is presented in the receive shift register which are to be stored into the receive buffer. After the DOR[2] flag is set, all the incoming data are lost. To avoid data loss or clear this flag, read the receive buffer.

The parity error (PE[0]) flag indicates that the frame in the receive buffer had a parity error when received. If parity check function is not enabled (PG[13:12] = 0 in USARTn_CR1 register), the PE[0] bit in USARTn_ST register is always read "0" (n = 10, 11, 12 and 13).

UART parity checker

If parity bit is enabled (PG[13:12]=1 in USARTn_CR1 register), the Parity Checker calculates the parity of the data bits in incoming frame and compares the result with the parity bit from the received serial frame (n = 10, 11, 12 and 13).

UART disabling receiver

In contrast to transmitter, disabling the Receiver by clearing RXE[0] bit makes the Receiver inactive immediately.

When the receiver is disabled, the receiver flushes the receive buffer, the remaining data in the buffer is all reset, and the RXDn pin can be used as a normal general purpose I/O (n = 10, 11, 12 and 13).

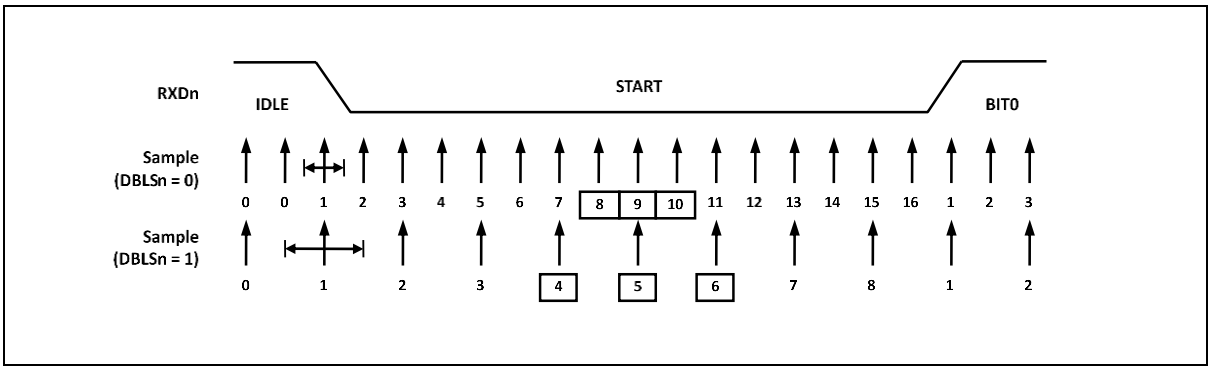
Asynchronous data reception

To receive asynchronous data frame, the USART includes a clock and data recovery unit. The clock recovery logic is used for synchronizing the internally generated baud-rate clock to the incoming asynchronous serial frame on the RXDn pin.

The data recovery logic does sampling and low pass filtering the incoming bits, and removing the noise of RXDn pin.

Figure 98 illustrates the sampling process of the start bit of an incoming frame. The sampling rate is 16 times of the baud-rate in normal mode and 8 times the baud-rate for double speed mode (DBLS[8] = 1 in USARTn_CR2 register). The horizontal arrows show the synchronization variation due to the asynchronous sampling process. Note that larger time variation is shown when using the double speed mode (n = 10, 11, 12 and 13).

Figure 98. Asynchronous Start Bit Sampling (n = 10, 11, 12 and 13)

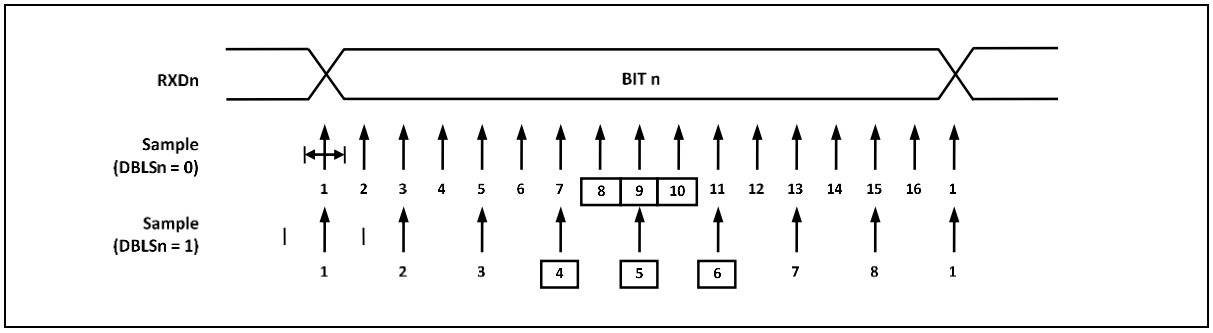


When the receiver is enabled (RXE[0] = 1), the clock recovery logic tries to find a high-to-low transition on the RXDn line, the start bit condition. After detecting high to low transition on RXDn line, the clock recovery logic uses samples 8, 9 and 10 for normal mode to decide if a valid start bit is received. If more than 2 samples have logical low level, it is considered that a valid start bit is detected and the internally generated clock is synchronized to the incoming data frame. And the data recovery can begin. The synchronization process is repeated for each start bit.

As described above, when the receiver clock is synchronized to the start bit, the data recovery can begin. Data recovery process is almost same to the clock recovery process. The data recovery logic samples 16 times for each incoming bits for normal mode and 8 times for double speed mode, and uses sample 8, 9 and 10 to decide data value. If more than 2 samples have low levels, the received bit is considered to a logic '0' and if more than 2 samples have high levels, the received bit is considered to a logic '1'.

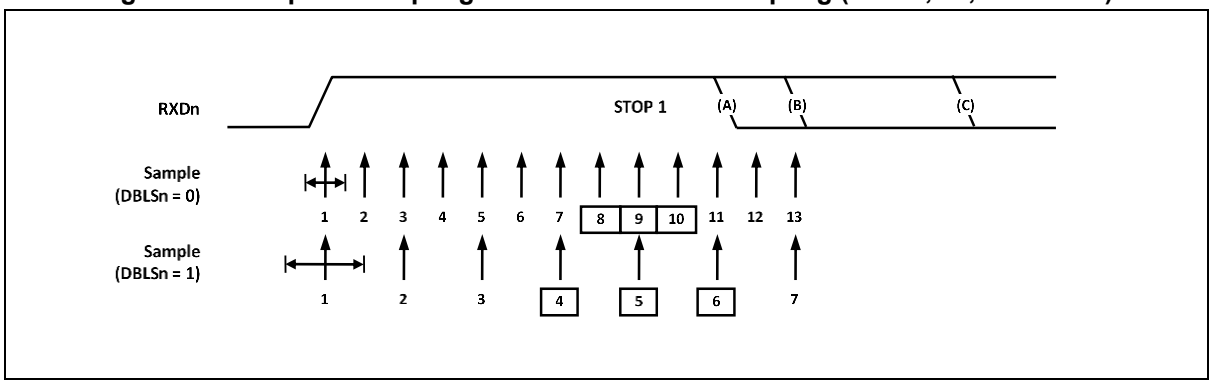
The data recovery process is then repeated until a complete frame is received including the first stop bit. The decided bit value is stored in the receive shift register in order. Note that the Receiver only uses the first stop bit of a frame. Internally, after receiving the first stop bit, the Receiver is in idle state and waiting to find start bit.

Figure 99. Asynchronous Data and Parity Bit Sampling (n = 10, 11, 12 and 13)



The process for detecting stop bit is same as clock and data recovery process. That is, if 2 or more samples of 3 center values have high level, correct stop bit is detected, else a frame error (FE[1]) flag is set. After deciding whether the first stop bit is valid or not, the Receiver goes to idle state and monitors the RXDn line to check a valid high to low transition is detected (start bit detection). (n = 10, 11, 12 and 13).

Figure 100. Stop Bit Sampling and Next Start Bit Sampling (n = 10, 11, 12 and 13)



15.3.8 SPI mode

USART can be set to operate in industrial standard SPI compliant mode. The SPI mode features the followings:

- Full Duplex, Three-wire synchronous data transfer.
- Mater and Slave Operation.
- Supports all four SPI modes of operation (mode 0, and 1).
- Selectable LSB first or MSB first data transfer.
- Double buffered transmit and receive.
- Programmable transmit bit rate.

When the SPI mode is enabled by configuring MS[15:14] as "11", the slave select (SSn) pin becomes active LOW input in slave mode operation, or can be output in master mode operation if USARTn_CR2<SSEN[4]> bit is set to '0'.

Note that during SPI mode of operation, the pin RXDn is renamed as MISOn, and TXDn is renamed as MOSIn for compatibility to other SPI devices (n = 10, 11, 12, and 13).

15.3.9 SPI clock formats and timing

To accommodate a wide variety if synchronous serial peripherals from different manufacturers, the USART has a clock polarity bit (CPOL[7]) and a clock phase control bit (CPHA[6]) to select one of four clock formats for data transfers. CPOL[7] selectively insert an inverter in series with the clock. CPHA[6] chooses between two different clock phase relationships between the clock and data. Note that CPHA[6] and CPOL[7] bits in USARTn_CR1 register have different meanings according to the USARTn_CR1<MS[15:14]> bits which decides the operating mode of USART(MS[15:14] = '01').

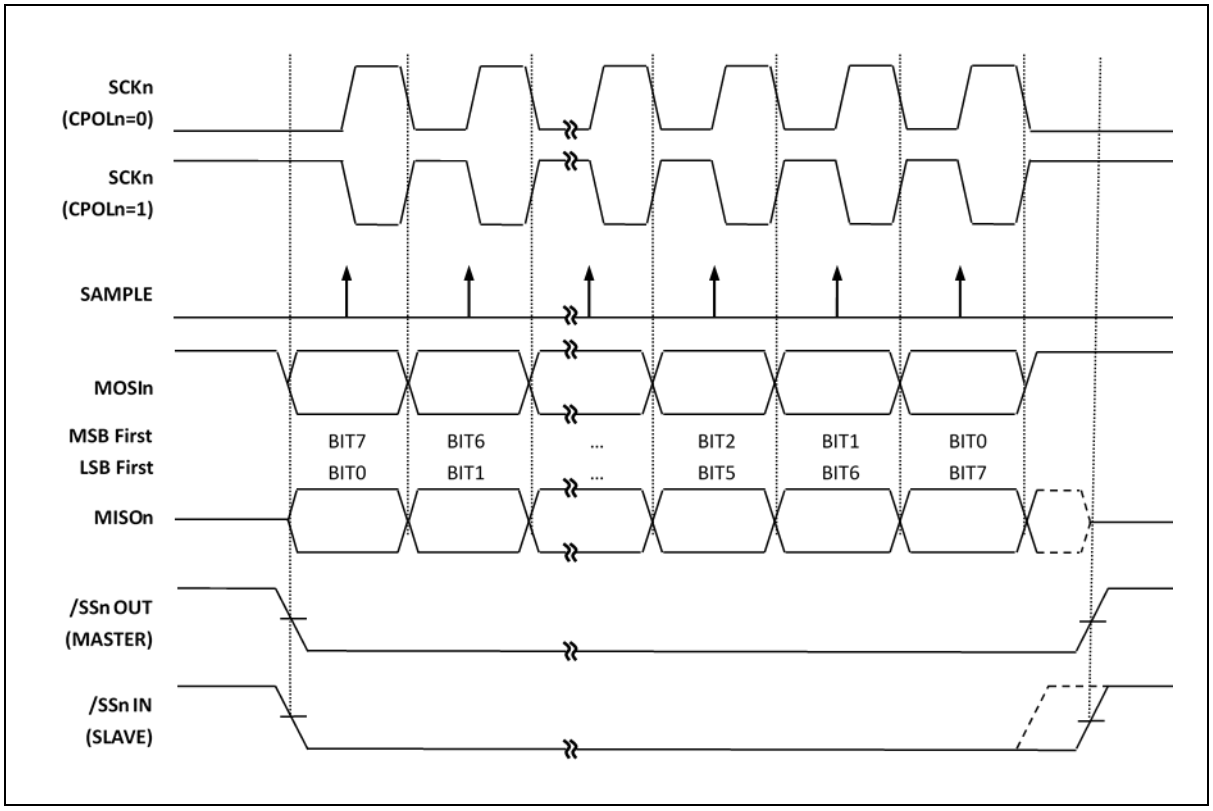
Table 72 shows four combinations of CPOL and CPHA for SPI modes 10, 11, 12 and 13 (n = 10, 11, 12, and 13).

Table 72. CPOL Functionality

SPIn Mode	CPOL	CPHA	Leading Edge	Trailing Edge
0	0	0	Sample (Rising)	Setup (Falling)
1	0	1	Setup (Rising)	Sample (Falling)
2	1	0	Sample (Falling)	Setup (Rising)
3	1	1	Setup (Falling)	Sample (Rising)

NOTE: n = 10, 11, 12 and 13

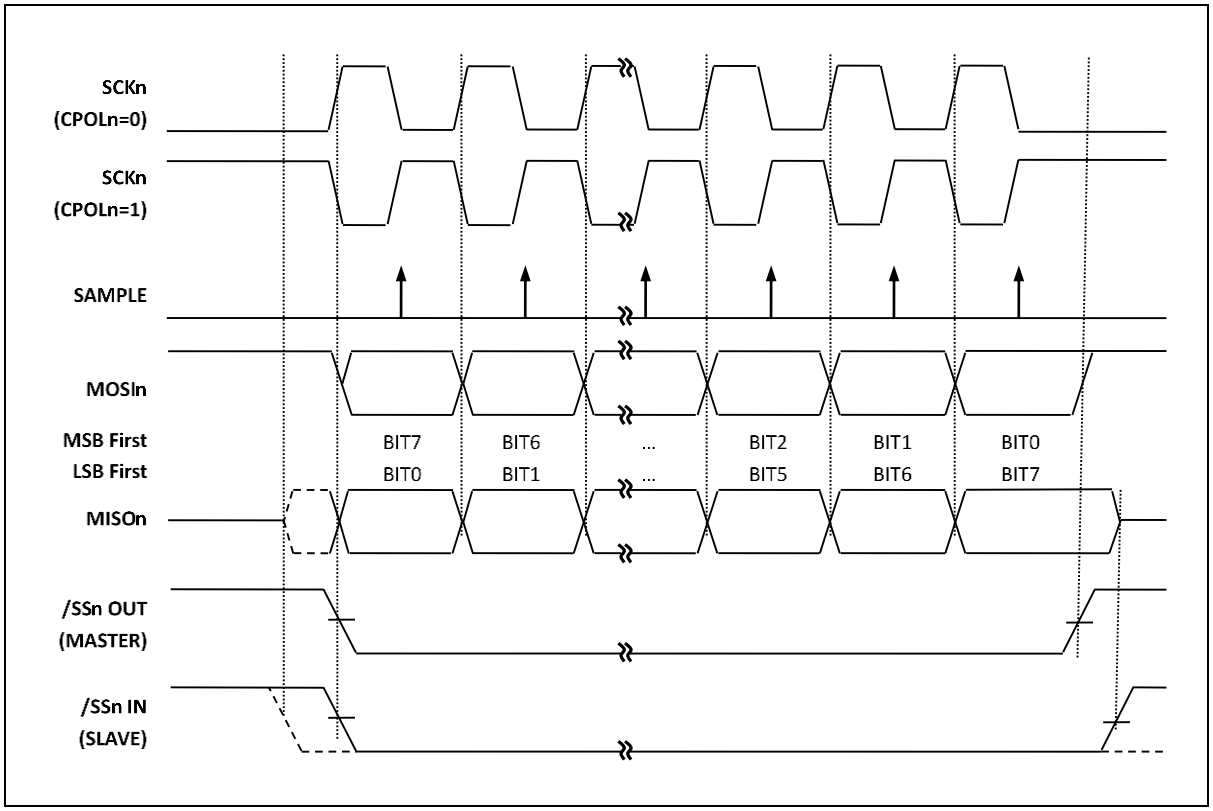
Figure 101. USART SPIn Clock Formats when CPHAn = 0 (n = 10, 11, 12 and 13)



When CPHA[6]=0, the slave begins to drive its MISOn output with the first data bit value when SSn goes to active low. The first SCKn edge causes both the master and the slave to sample the data bit value on their MISOn and MOSIn inputs, respectively.

At the second SCKn edge, the USART shifts the second data bit value out to the MOSIn and MISOn outputs of the master and slave, respectively. Unlike the case of CPHA[6] = 1, when CPHA[6] = 0, the slave's SSn input must go to its inactive high level between transfers. This is because the slave can prepare the first data bit when it detects falling edge of SSn input (n = 10, 11, 12 and 13).

Figure 102. USART SPIn Clock Formats when CPHAn=1 (n = 10, 11, 12 and 13)



When CPHA[6] = 1, the slave begins to drive its MISOn output when SSn goes active low, but the data is not defined until the first SCKn edge. The first SCKn edge shifts the first bit of data from the shifter onto the MOSIn output of the master and the MISOn output of the slave.

The next SCKn edge causes both the master and slave to sample the data bit value on their MISOn and MOSIn inputs, respectively. At the third SCKn edge, the USART shifts the second data bit value out to the MOSIn and MISOn output of the master and slave respectively. When CPHA[6] = 1, the slave's SSn input is not required to go to its inactive high level between transfers.

Because the SPIn logic reuses the USART resources, SPIn mode of operation is similar to that of synchronous or asynchronous operation. SPIn transfer is initiated by checking for the USART Data Register Empty flag (DRE[7] = 1) in USARTn_ST register and then writing a byte of data to the USARTn_DR register. In master mode of operation, even if transmission is not enabled (TXE[0] = 0), writing data to the USARTn_DR register is necessary because the clock SCKn is generated from transmitter block.

15.3.10 Baud rate settings (example)

Table 73. Baud Rate Settings Example

Baud Rate	fOSC=1.00MHz				fOSC=1.8432MHz				fOSC=2.00MHz			
	DBLSn=0		DBLSn=1		DBLSn=0		DBLSn=1		DBLSn=0		DBLSn=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	25	0.2%	51	0.2%	47	0.0%	95	0.0%	51	0.2%	103	0.2%
4800	12	0.2%	25	0.2%	23	0.0%	47	0.0%	25	0.2%	51	0.2%
9600	6	-7.0%	12	0.2%	11	0.0%	23	0.0%	12	0.2%	25	0.2%
14.4K	3	8.5%	8	-3.5%	7	0.0%	15	0.0%	8	-3.5%	16	2.1%
19.2K	2	8.5%	6	-7.0%	5	0.0%	11	0.0%	6	-7.0%	12	0.2%
28.8K	1	8.5%	3	8.5%	3	0.0%	7	0.0%	3	8.5%	8	-3.5%
38.4K	1	-18.6%	2	8.5%	2	0.0%	5	0.0%	2	8.5%	6	-7.0%
57.6K	-	-	1	8.5%	1	-25.0%	3	0.0%	1	8.5%	3	8.5%
76.8K	-	-	1	-18.6%	1	0.0%	2	0.0%	1	-18.6%	2	8.5%
115.2K	-	-	-	-	-	-	1	0.0%	-	-	1	8.5%
230.4K	-	-	-	-	-	-	-	-	-	-	-	-

Baud Rate	fOSC=3.6864MHz				fOSC=4.00MHz				fOSC=7.3728MHz			
	DBLSn=0		DBLSn=1		DBLSn=0		DBLSn=1		DBLSn=0		DBLSn=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	95	0.0%	191	0.0%	103	0.2%	207	0.2%	191	0.0%	-	-
4800	47	0.0%	95	0.0%	51	0.2%	103	0.2%	95	0.0%	191	0.0%
9600	23	0.0%	47	0.0%	25	0.2%	51	0.2%	47	0.0%	95	0.0%
14.4K	15	0.0%	31	0.0%	16	2.1%	34	-0.8%	31	0.0%	63	0.0%
19.2K	11	0.0%	23	0.0%	12	0.2%	25	0.2%	23	0.0%	47	0.0%
28.8K	7	0.0%	15	0.0%	8	-3.5%	16	2.1%	15	0.0%	31	0.0%
38.4K	5	0.0%	11	0.0%	6	-7.0%	12	0.2%	11	0.0%	23	0.0%
57.6K	3	0.0%	7	0.0%	3	8.5%	8	-3.5%	7	0.0%	15	0.0%
76.8K	2	0.0%	5	0.0%	2	8.5%	6	-7.0%	5	0.0%	11	0.0%
115.2K	1	0.0%	3	0.0%	1	8.5%	3	8.5%	3	0.0%	7	0.0%
230.4K	-	-	1	0.0%	-	-	1	8.5%	1	0.0%	3	0.0%
250K	-	-	1	-7.8%	-	-	1	0.0%	1	-7.8%	3	-7.8%
0.5M	-	-	-	-	-	-	-	-	-	-	1	-7.8%

Baud Rate	fOSC=8.00MHz				fOSC=11.0592MHz				fOSC=14.7456MHz			
	DBLSn=0		DBLSn=1		DBLSn=0		DBLSn=1		DBLSn=0		DBLSn=1	
	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR	UBAUD	ERROR
2400	207	0.2%	-	-	-	-	-	-	-	-	-	-
4800	103	0.2%	207	0.2%	143	0.0%	-	-	191	0.0%	-	-
9600	51	0.2%	103	0.2%	71	0.0%	143	0.0%	95	0.0%	191	0.0%
14.4K	34	-0.8%	68	0.6%	47	0.0%	95	0.0%	63	0.0%	127	0.0%
19.2K	25	0.2%	51	0.2%	35	0.0%	71	0.0%	47	0.0%	95	0.0%
28.8K	16	2.1%	34	-0.8%	23	0.0%	47	0.0%	31	0.0%	63	0.0%
38.4K	12	0.2%	25	0.2%	17	0.0%	35	0.0%	23	0.0%	47	0.0%
57.6K	8	-3.5%	16	2.1%	11	0.0%	23	0.0%	15	0.0%	31	0.0%
76.8K	6	-7.0%	12	0.2%	8	0.0%	17	0.0%	11	0.0%	23	0.0%
115.2K	3	8.5%	8	-3.5%	5	0.0%	11	0.0%	7	0.0%	15	0.0%
230.4K	1	8.5%	3	8.5%	2	0.0%	5	0.0%	3	0.0%	7	0.0%
250K	1	0.0%	3	0.0%	2	-7.8%	5	-7.8%	3	-7.8%	6	5.3%
0.5M	-	-	1	0.0%	-	-	2	-7.8%	1	-7.8%	3	-7.8%
1M	-	-	-	-	-	-	-	-	-	-	1	-7.8%

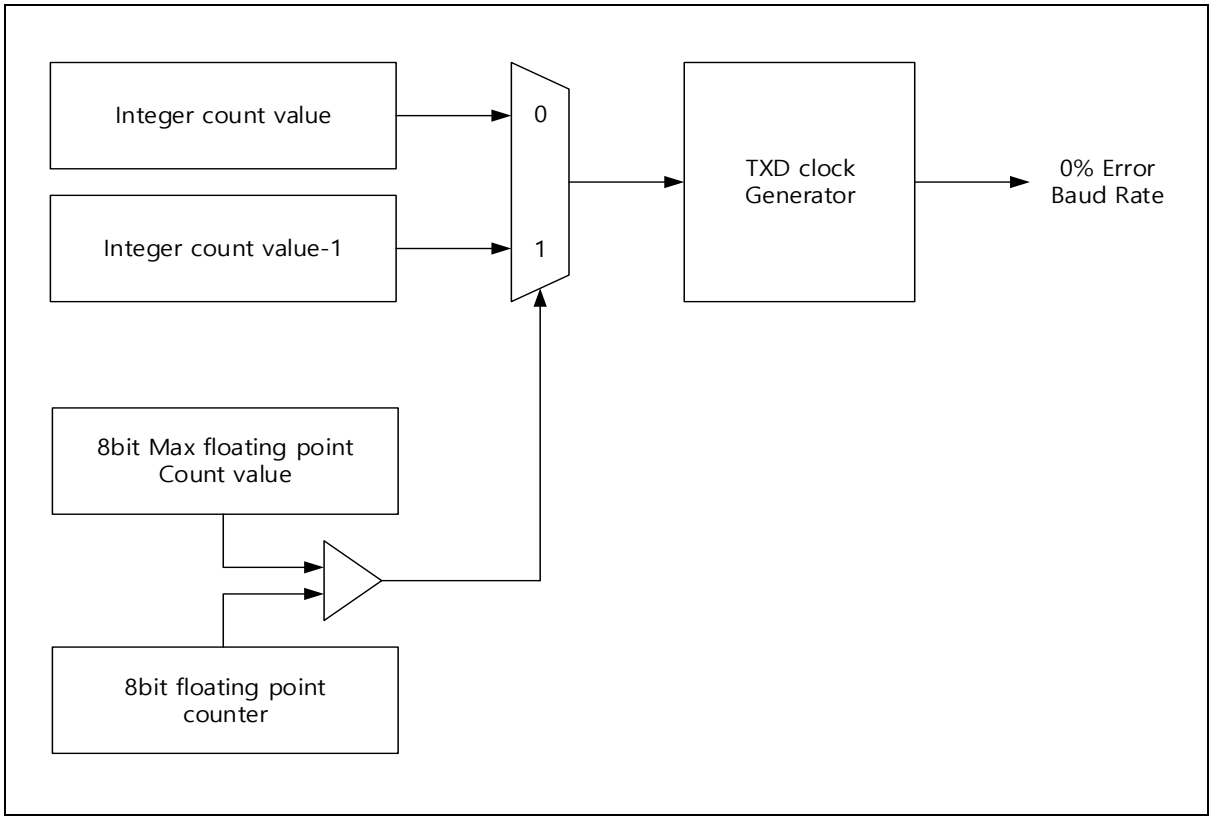
15.3.11 0% error baud rate

This USART system supports the fraction counter logic for the 0% error of baud rate. By using the 8 bits floating point counter logic, the cumulative error to below the decimal point can be removed.

The fraction counter value is defined by baud rate error. In the baud rate formula, USARTn_BDR is presented the integer count value. For example, If you want to use the 57600 baud rate (f_{XIN} = 16MHz), a calculated integer count value must be 17.36 value (USARTn_BDR+1 = 16000000/(16×57600) = 17.36).

Here, USARTn_BDR which can be set is the nearest big integer number 17. To realize 0% error of baud rate, fraction counter value must be 92 ((0.36) x 256 ≒ 92). Namely you have to write 92 (decimal number) to USARTn_BFR and 17 (decimal number) to USART_BDR register.

Figure 103. 0% Error Baud Rate Diagram



15.3.12 Receive Time Out (RTO)

This USART system supports the receive time out (RTO) interrupt. The RTO counter uses the system clock and continues counting while the RXD input is not present. If the RTO counter matches USARTn_RTO, RTOF becomes 1. RTOEN is set to 1 to enable this operation, and RTOEN is automatically set to 0 when an RTO match occurs. You can use RTO interrupts with RTOIE.

16 Universal Asynchronous Receiver/Transmitter (UART)

2-channel UART (Universal Asynchronous Receiver/Transmitter) modules are provided. UART operation status including error status can be read from status register. The prescaler which generates proper baud rate, is exist for each UART channel. The prescaler can divide the UART clock source which is PCLK, from 1 to 65535. And baud rate generation is by clock which internally divided by 16 of the prescaled clock and 8-bit precision clock tuning function.

Programmable interrupt generation function will help to control the communication via UART channel

- Standard asynchronous control bit (start, stop, and parity) configurable
- Programmable serial communication
- 5, 6, 7 or 8 data bit transfer
- Even, odd, or no-parity bit insertion and detection
- 1, 1.5 or 2 stop bit-insertion and detection
- 16-bit baud rate generation with 8-bit fraction control
- Hardware inter-frame delay function
- Stop bit error detection
- Detail status register

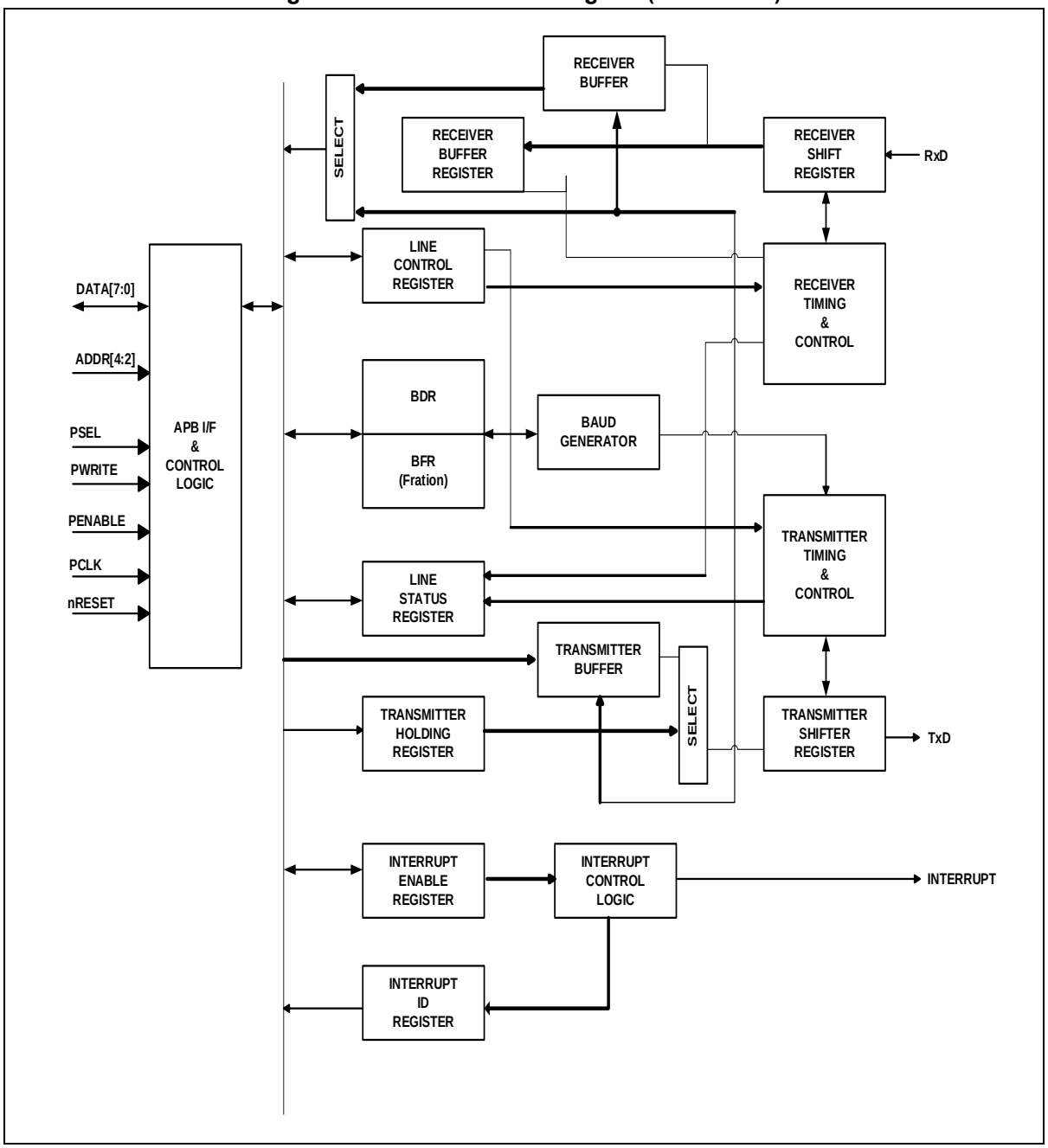
Table 74 introduces pins assigned for the UART channels.

Table 74. Pin Assignment of UART: External Pins

Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
TXD0	O	UART Channel 0 transmit output	O	O	O
RXD1	I	UART Channel 0 receive input	O	O	O
TXD1	O	UART Channel 1 transmit output	O	O	O
RXD1	I	UART Channel 1 receive input	O	O	O

16.1 UART block diagram

Figure 104. UARTn Block Diagram (n = 0 and 1)



16.2 Registers

Base address of USART is introduced in the followings:

Table 75. Base Address of USART

Name	Base address
UART 0	0x4000_4000
UART 1	0x4000_4100

Table 76. UART Register Map

Name	Offset	Type	Description	Reset value	Ref.
UARTn_RBR	0x00	RO	Receive data buffer register	0x0000_0000	0
UARTn_THR	0x00	WO	Transmit data hold register	0x0000_0000	16.2.1
UARTn_IER	0x04	RW	Interrupt enable register	0x0000_0000	16.2.2
UARTn_IIR	0x08	RO	Interrupt ID register	0x0000_0001	16.2.3
UARTn_LCR	0x0C	RW	Line control register	0x0000_0000	16.2.4
UARTn_DCR	0x10	RW	Data Control Register	0x0000_0000	16.2.5
UARTn_LSR	0x14	RO	Line status register	0x0000_0060	16.2.6
UARTn_BDR	0x20	RW	Baud rate Divisor Latch Register	0x0000_0000	16.2.7
UARTn_BFR	0x24	RW	Baud rate Fractional Counter Value	0x0000_0000	16.2.8
UARTn_IDTR	0x30	RW	Inter-frame Delay Time Register	0x0000_0000	16.2.9

NOTE: n = 0 and 1

UARTn_RBR: UARTn receive data buffer register

UARTn Receive Data Buffer Register is 32-bit register. Received data will be read out from this register. Maximum length of data is 8 bits. Last data received will be maintained in this register until a new byte is received. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_RBR=0x4000_4000, UART1_RBR=0x4000_4100

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			RBR
-			0x00
-			RO

Bits	Name	Function
7 0	RBR	UARTn receive data buffer bits

16.2.1 UARTn_THR: UARTn transmit data hold register

UARTn Transmit Data Hold Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_THR=0x4000_4000, UART1_THR=0x4000_4100

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			THR
-			0x00
-			WO

Bits	Name	Function
7 0	THR	UARTn transmit data hold bits

16.2.2 UARTn_IER: UARTn interrupt enable register

UARTn Interrupt Enable Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_IER=0x4000_4004, UART1_IER=0x4000_4104

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6	5	4	3	2	1	0
Reserved				DTXIEN	DRXIEN	TXEIE	RLSIE	THREIE	DRIE
-				0	0	0	0	0	0
-				RW	RW	RW	RW	RW	RW

Bits	Name	Function
5	DTXIEN	DMA Transmit done Interrupt Enable
		0 Disable DMA transmit done interrupt
		1 Enable DMA transmit done interrupt
4	DRXIEN	DMA Receive done Interrupt Enable
		0 Disable DMA receive done interrupt
		1 Enable DMA receive done interrupt
3	TXEIE	Transmit Register Empty Interrupt Enable.
		0 Disable transmit register empty interrupt.
		1 Enable transmit register empty interrupt.
2	RLSIE	Receiver Line Status Interrupt Enable bit.
		0 Disable receiver line status interrupt.
		1 Enable receiver line status interrupt.
1	THREIE	Transmit Holding Register Empty Interrupt Enable bit.
		0 Disable transmit holding register empty interrupt.
		1 Enable transmit holding register empty interrupt.
0	DRIE	Data Receive Interrupt Enable bit.
		0 Disable data receive interrupt.
		1 Enable data receive interrupt.

16.2.3 UARTn_IIR: UARTn interrupt ID register

UARTn Interrupt ID Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_IIR=0x4000_4008, UART1_IIR=0x4000_4108

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3	2 1	0
Reserved				TXE	DMAF	IID	IPEN
-				0	0	0	0
-				RW	RW	RW	RW

Bits	Name	Function
4	TXE	Transmit complete (Refer to the table of interrupt source IDs below)
		0 the transmit data hold register (THR) is empty
		1 transmit data hold register (THR) is currently empty and new data can be written to the register.
3	DMAF	DMA Tx or Rx Operation Complete flag.
		0 DMA Operation is working or is disabled
		1 DMA Operation is done
2 1	IID	UARTn Interrupt ID bits. **NOTE:** The UARTn supports 3-priority interrupt generation and the interrupt source ID register shows one interrupt source which has highest priority among pending interrupts. The priority is defined as below. A. Receive line status interrupt. B. Receive data ready interrupt and character timeout interrupt. C. Transmit hold register empty interrupt.
0	IPEN	Interrupt Pending bit.
		0 Interrupt is pending.
		1 No interrupt is pending.

Table 77. UART IIR Description

Priority	TXE	DMAF	IID		IPEN	Interrupt sources		
	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Interrupt Name	Interrupt condition	Interrupt clear
-	0	0	0	0	1	None	-	-
1	0	0	1	1	0	Receive line status	Overrun, parity, framing, break error, etc.	Read LSR
2	0	0	1	0	0	Receive data present	There is data received	Read the receive register or IIR

Table 77. UART IIR Description(continued)

Priority	TXE	DMAF	IID		IPEN	Interrupt sources		
	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Interrupt Name	Interrupt condition	Interrupt clear
3	0	0	0	1	0	Transmit data	The transmit buffer is empty	Write to the transmit data hold register or IIR
4	1	X	X	X	X	Transmit register empty	The transmit data hold register is empty	Write to the transmit data hold register or read IIR
5	0	1	1	0	0	DMA Rx complete	DMA Rx Complete	Read IIR
6	0	1	0	1	0	DMA Tx complete	DMA Tx Complete	Read IIR
7	1	1	X	X	X	Transmit register empty and DMA Complete	The transmit register is empty and DMA Tx has been completed	Read IIR

NOTE: After checking the above bits, Read data buffer to avoid losing interrupt source.

16.2.4 UARTn_LCR: UARTn line control register

UARTn Line Control Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_LCR=0x4000_400C, UART1_LCR=0x4000_410C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1 0
Reserved				BREAK	STICKP	PARITY	PEN	STOPBIT	DLEN
-				0	0	0	0	0	0
-				RW	RW	RW	RW	RW	RW

Bits	Name	Function	
6	BREAK	Transfer Break Control bit. The TXDn pin will be driven at low state in order to notice the alert to the receiver.	
		0	Normal transfer mode.
		1	Break transmit mode.
5	STICKP	Force Parity bit. This bit is effective when the PEN bit is set to '1'.	
		0	Disable parity stuck.
		1	Enable parity stuck. A parity is always the value of the PARITY bit.
4	PARITY	Parity Mode and Parity Stuck Selection bit.	
		0	Odd parity mode.
		1	Even parity mode.
3	PEN	Parity Bit Transfer Enable bit.	
		0	Disable parity transfer.
		1	Enable parity transfer.
2	STOPBIT	Stop Bit Length Selection bit.	
		0	1 stop bit.
		1	1.5 or 2 stop bit. 1.5 stop bit in case of 5-bit data length 2 stop bit in case of 6-bit/7-bit/8-bit data length.
1 0	DLEN	Data Length Selection bits	
		00	5-bit data length
		01	6-bit data length
		10	7-bit data length
		11	8-bit data length

Parity bit will be generated according to bit 3, 4, 5 of UARTn_LCR register. Table 78 shows the variation of parity bit generation.

Table 78. Parity Bit

STICKP[5]	PARITY[4]	PEN[3]	Parity Mode
X	X	0	No Parity
0	0	1	Odd Parity
0	1	1	Even Parity
1	0	1	Force parity as "1"
1	1	1	Force parity as "0"

16.2.5 UARTn_DCR: UARTn data control register

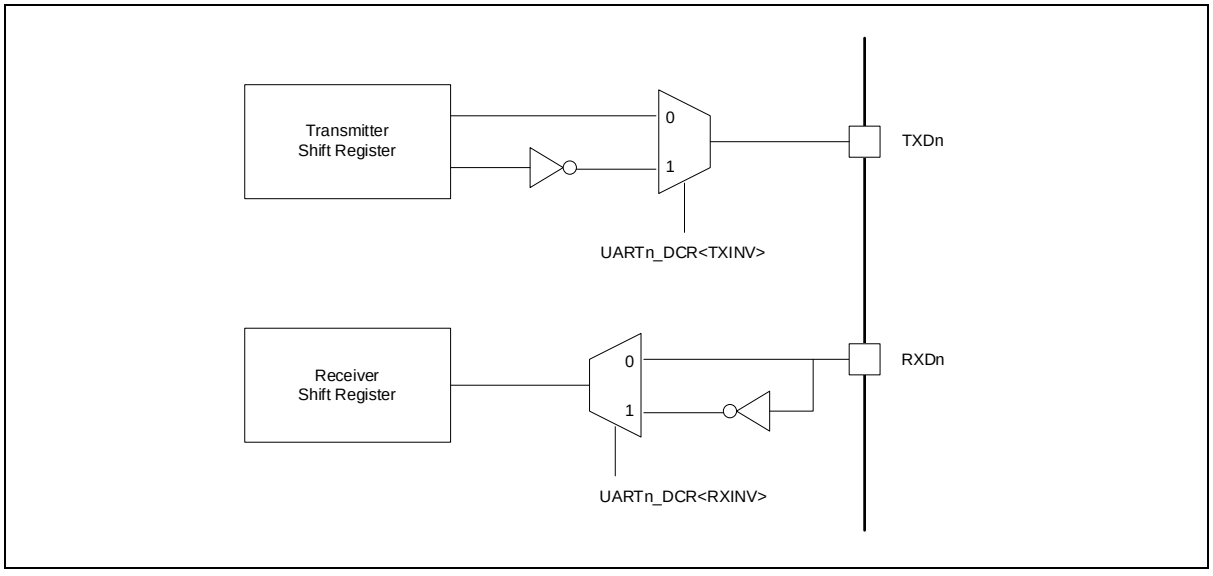
UARTn data control register supports TXDn and RXDn output signal inversion and loopback mode function to connect TXDn signal internally to RXDn. (n = 0 or 1)

UART0_DCR=0x4000_4010, UART1_DCR=0x4000_4110

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5	4	3	2	1 0
Reserved				LBON	RXINV	TXINV	Reserved
-				0	0	0	-
-				RW	RW	RW	-

Bits	Name	Function	
4	LBON	Local Loopback Test Mode Enable bit.	
		0	Normal mode.
		1	Local loopback mode. TXDn connected to RXDn internally.
3	RXINV	Receive Data Inversion Selection bit.	
		0	Normal receive data input.
		1	Inverted receive data input.
2	TXINV	Transmit Data Inversion Selection bit.	
		0	Normal transmit output.
		1	Inverted transmit output.

Figure 105. Data inversion control diagram (n = 0 and 1)



16.2.6 UARTn_LSR: UARTn line status register

UARTn Line Status Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_LSR=0x4000_4014, UART1_LSR=0x4000_4114

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved				TEMT	THRE	BI	FE	PE	OE	DR
-				1	1	0	0	0	0	0
-				RO	RO	RO	RO	RO	RO	RO

Bits	Name	Function	
6	TEMT	Transmit Empty bit.	
		0	Transmit register has data or is transferring.
		1	Transmit register is empty.
5	THRE	Transmit Holding Register Empty bit. **NOTE:** This bit will be set to '1' when it starts transmit.	
		0	Transmit holding register is not empty.
		1	Transmit holding register is empty
4	BI	Break condition Indication bit.	
		0	Normal status.
		1	Break condition is detected.
3	FE	Frame Error indicator bit.	
		0	No frame error.
		1	Frame error occurs. The receive character did not have a valid stop bit.
2	PE	Parity Error Indicator bit.	
		0	No parity error.
		1	Parity error occurs. The receive character have not correct parity information.
1	OE	Overrun Error indicator bit.	
		0	No overrun error.
		1	Overrun error occurs. Additional data arrive while the RHR is full.
0	DR	Data Receive indicator bit.	
		0	No data in receive holding register.
		1	Data has been received and is saved in the receive holding register.

This register provides the status of data transfers between transmitter and receiver. User can get the line status information from this register and can handle the next process. Bit 1,2,3,4 will arise the line status interrupt when RLSIE bit in UARTn_IER register is set. Other bits can generate its interrupt when its interrupt enable bit in UARTn_IER register is set.

16.2.7 UARTn_BDR: UARTn baud rate divisor latch register

UARTn Baud Rate Divisor Latch Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_BDR=0x4000_4020, UART1_BDR=0x4000_4120

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		BDR	
-		0x0000	
-		WO	

Bits	Name	Function
15 0	BDR	Baud rate Divider latch value Baud rate = $f_{UARTnCLK}$/(16 x BDR[15:0] x 2). **NOTE:** The UART block won't work if the BDR[15:0] ≤ 0x0003

To establish the communication with UART channel, the baud rate should be set properly. The programmable baud rate generate is provided to give from 1 to 65535 divider number. The 16 bit divider register (UARTn_BDR) should be written for expected baud rate. $UART_{clock}$ is PCLK.

Baud rate calculation formula is below.

$$BDR = \frac{UART_{clock}}{16 \times BaudRate \times 2}$$

In case of 48 MHz UARTclock speed, the divider value and error rate is described in Table 79

Table 79. Example of baud rate calculation on $UART_{CLOCK}$ = 48MHz system (without BFR)

Baud rate	Divider	Error (%)
1200	1250	0.0%
2400	625	0.0%
4800	312	0.16%
9600	156	0.16%
19200	78	0.16%
38400	39	0.16%
57600	26	0.16%
115200	13	0.16%

16.2.8 UARTn_BFR: UARTn baud rate fraction counter register

UARTn Transmit Data Hold Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_THR=0x4000_4000, UART1_THR=0x4000_4100

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			BFR
-			0x00
-			WO

Bits	Name	Function	
5	BFR	Baud rate Fraction counter value. **NOTE:** 8-bit fractional counter will count up by FCNT value every (baud rate)/16 period and whenever fractional counter overflow is happen, the divisor value will increment by 1. So this period will be compensated. Then next period, the divisor value will return to original set value.	
		0	Disable fraction counter.
		N	Fraction compensation mode is operating. Fraction counter is incremented by FCNT.

$$FCNT = Floating\ Point\ Value * 256$$

FCNT value can calculated above equation. For example, the target baud rate is 4800 bps and $UART_{CLOCK}$ is 48MHz case, the BDR value is 312.5. The integer number 520 should be the BDR value and the floating number 0.5 will make the FNCT value as below.

FCNT = 0.5 * 256 = 128, so the FCNT value is 128.

8-bit fractional counter will count up by FCNT value every (baud rate)/16 periods and whenever fractional counter overflow is happen, the divisor value will increment by 1. So this period will be compensated. Then next period, the divisor value will return to original set value.

Table 80. Example of Baud Rate Calculation on $UART_{CLOCK}$ = 48MHz System (without BFR)

Baud rate	Divider	FCNT	Error (%)
1200	1250	0	0.00%
2400	625	0	0.00%
4800	312	128	0.00%
9600	156	64	0.00%
19200	78	32	0.00%
38400	39	16	0.00%
57600	26	10	0.01%
115200	13	5	0.01%

16.2.9 UARTn_IDTR: UARTn inter frame delay time register

UARTn Inter-Frame Delay Time Register is 32-bit register. This Register is able to 32/16/8-bit access. (n = 0 and 1)

UART0_LSR=0x4000_4014, UART1_LSR=0x4000_4114

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5 4 3	2 1 0
Reserved			SMS	DMS	Reserved	WAITVAL
-			0	0	-	000
-			RW	RW	-	RW

Bits	Name	Function
7	SMS	Start bit Multi Sampling enable bit.
		0 Multi sampling is disable for start bit, Single sample will be done at 8/16 baud rate for the start bit.
		1 Multi sampling is enabled for start bit. Sampling is done 3 times at 7/16, 8/16 and 9/16 baud rate. Dominant value of 3 samples will be selected for the start bit.
6	DMS	Data bit Multi Sampling enable bit
		0 Multi sampling is disable for data bit, Single sample will be done at 8/16 baud rate for the data bit.
		1 Multi sampling is enabled for data bit. Sampling is done 3 times at 7/16, 8/16 and 9/16 baud rate. Dominant value of 3 samples will be selected for the data bit.
2 0	WAITVAL	Wait Time Value. Dummy delay can be inserted between 2 continuous Transmits. $Wait\ Time = \frac{WAITVAL[2:0]}{Baud\ Rate}$

16.3 Functional description

The UART module is compatible with 16450 UART. Additionally fractional baud rate compensation logic is provided. It doesn't have internal FIFO block. So data transfer will establish interactively support.

16.3.1 Receiver start bit and data bit sampling

Receiver Sampling Strategy

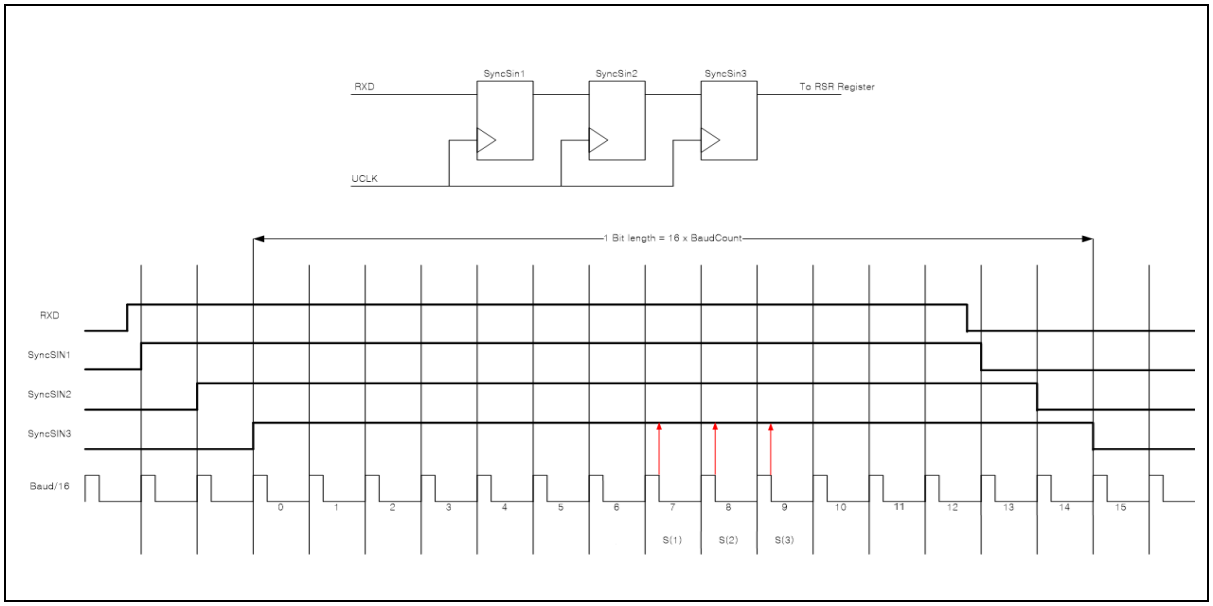
In the receiver block, there are synchronous logics inside to prevent abnormal noise on the RXD input signal line. The start bit and data bit sampling type can be configured to any mode of single sampling or multi sampling. The start bit detection strategy can be selected by the SMS bit on UARTn_IDTR register, the data bit detection strategy can be selected by the DMS bit on UARTn_IDTR register.

The Figure 106 shows the timing diagram of a data sampling.

If the DMS bit was selected, the multi-sampling feature is activated and a bit value is to be determined by most occurrence value of the time of S(1), S(2), S(3) from the 9th clock in a bit period.

If the DMS bit was not selected, the single-sampling feature is activated and a bit value is to be determined only by the value at S(2) timing. The DMS bit does not affect start bit strategy, and the SMS has the same features to the DMS. It only affects the start bit detection.

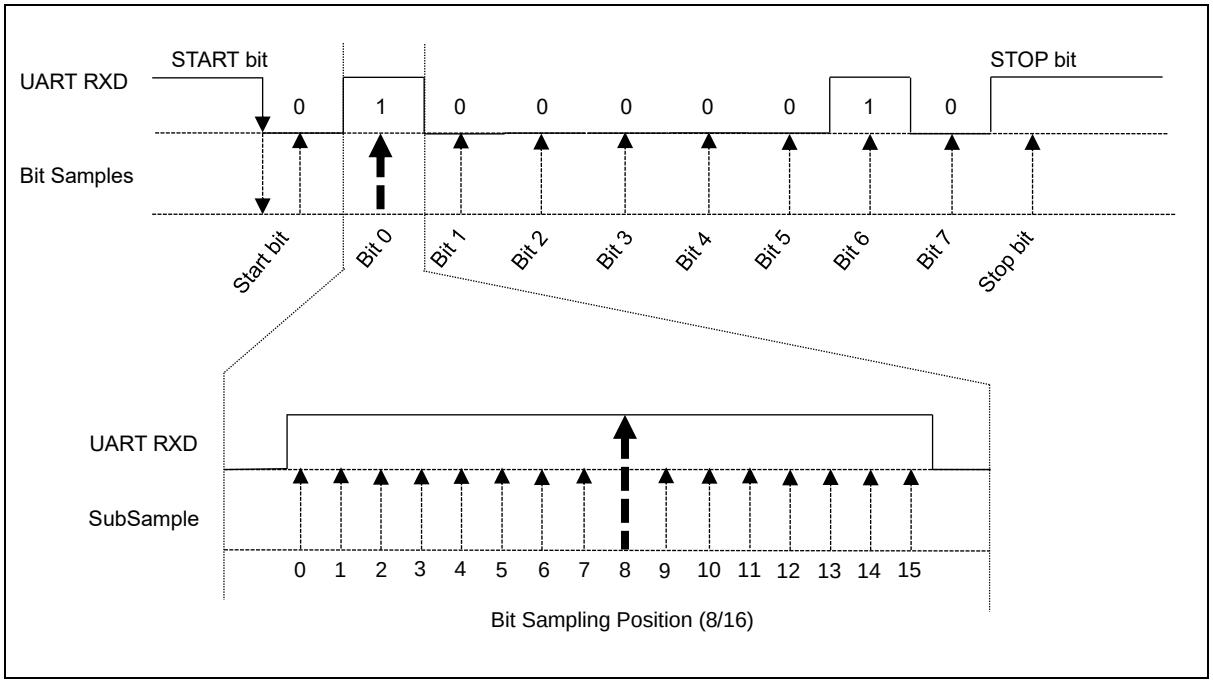
Figure 106. The timing diagram of data sampling



Data Bit Single Sampling

The UARTs operates as following timing. If the falling edge on the receive line, UART judges as the start bit. From the start timing, UART oversamples 16 times of 1-bit and detect the bit value at the 8th sample of 16 samples.

Figure 107. The Sampling Timing of UART Receiver



It is recommended to enable de-bounce settings in the PCU block to reinforce the immunity of external glitch noise.

Data bit Multi Sampling

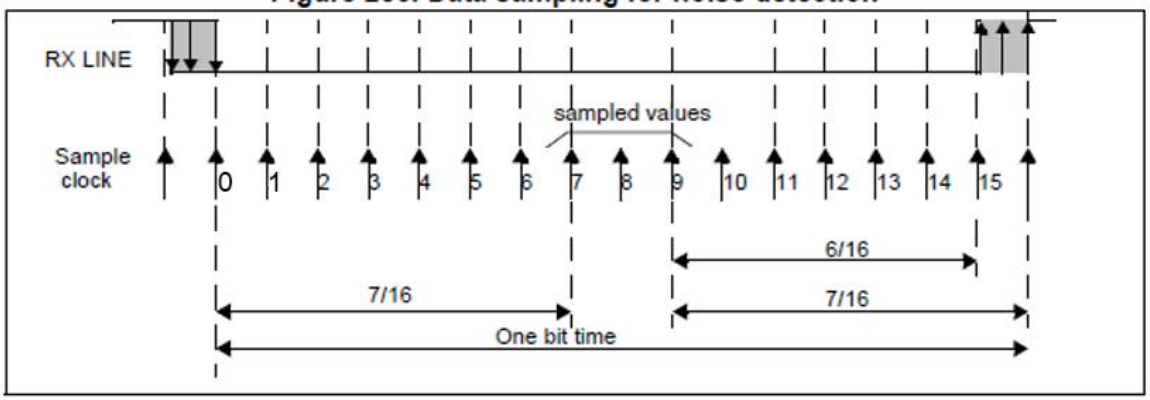
If the SMS or DTS is '1', most occurrence value of signals on 7th, 8th, 9th clocks will be the bit value.

Figure 108. The multi-sampling timing of start bit and Rx data bit

Noise error

Over-sampling techniques are used (except in synchronous mode) for data recovery by discriminating between valid incoming data and noise.

Figure 283. Data sampling for noise detection



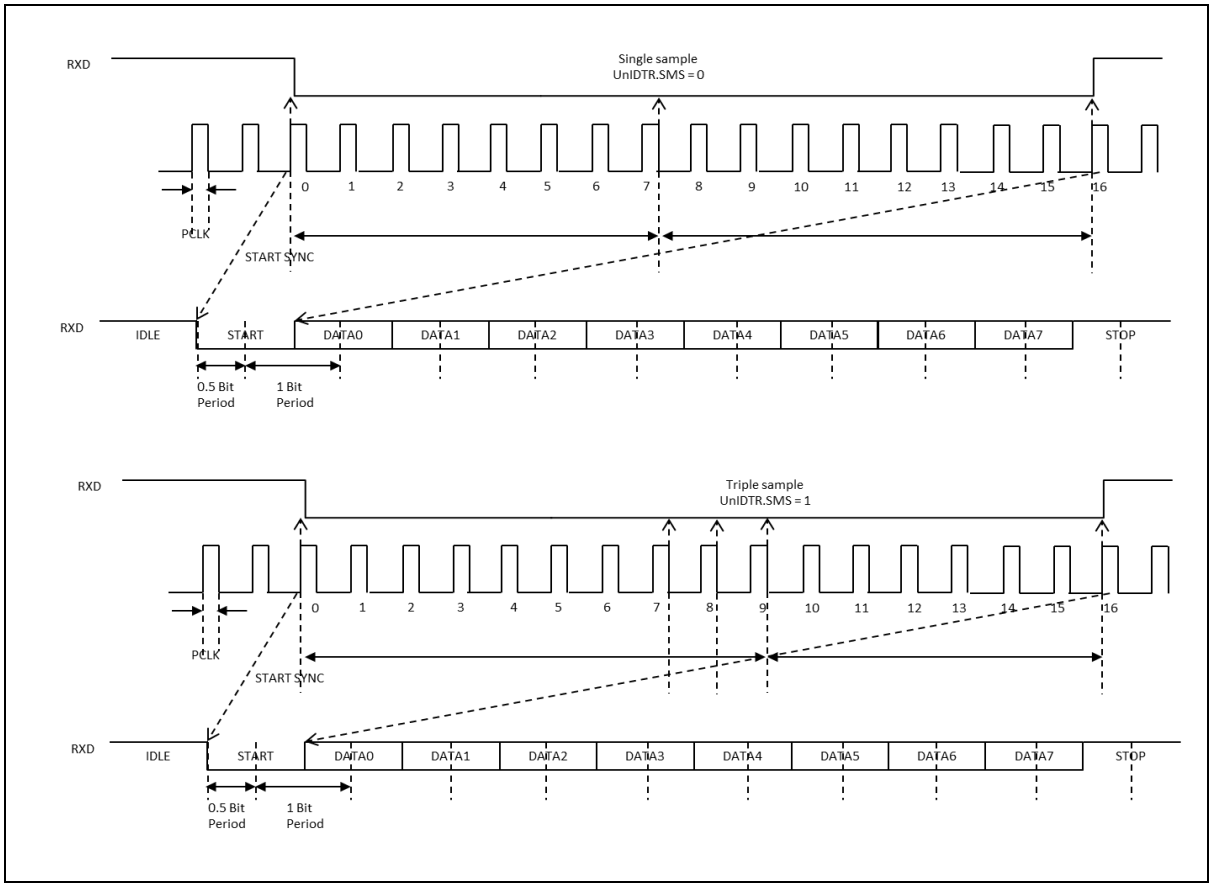
Receiver Start Bit Detection

In the UART receiver block, there are two types of determination strategy of START bit. One is by single-clock sampling and the other is by multi-clock sampling.

The single-clock sampling feature is the general sampling method, this feature acquire start bit value in the center of the bit period.

The multi-clock sampling feature has noise-rejection function for accurate communication against line noise. The timing diagrams of the single sampling and multi sampling features are shown as followings.

Figure 109. Timing of Start Bit Detection in Single/Multi-Sampling Modes



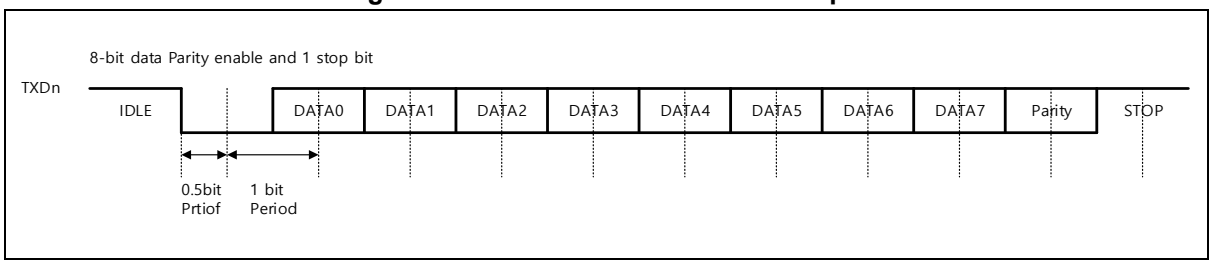
16.3.2 Transmitter

The transmitter has data transmit function. The start bit, data bits, optional parity bit and stop bit are serially shifted, least significant bit first.

The number of data bit is selected in the DLAN[1:0] filed in UARTn_LCR register.

The parity bit is set according to the PARITY and PEN bit filed in UARTn_LCR register. If the parity type is even then the parity bit depends on the one bit sum of all data bits. For odd parity, the parity bit is the inverted sum of all data bits. The number of stop bits is selected in the STOPBIT filed in UARTn_LCR register. The example of transmit data format is below Figure 110.

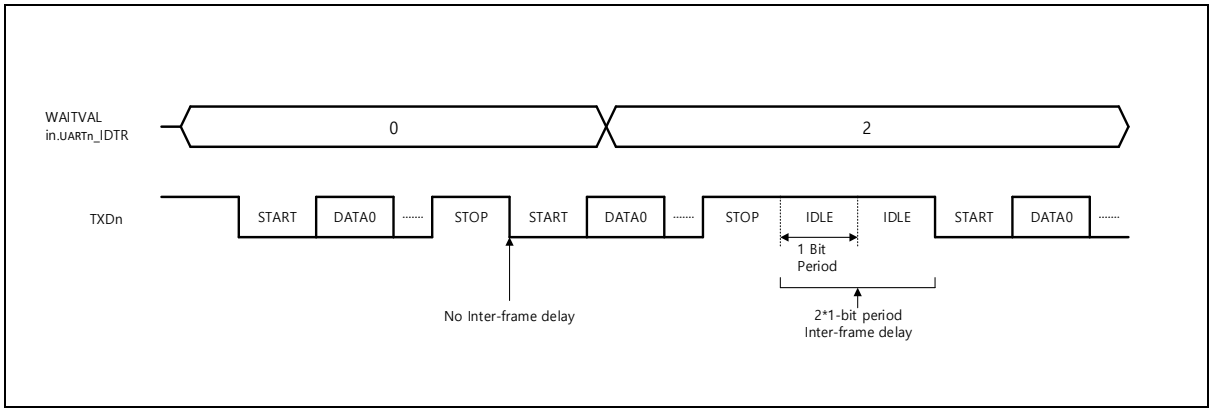
Figure 110. Transmit data format example



16.3.3 Inter-frame delay transmission

The inter-frame delay function allows the transmitter to insert an idle state on the TXD line between 2 characters. The width of the idle state is defined in WAITVAL field in UARTn_IDTR register. When this field is set 0, no time-delay is generated. Otherwise, the transmitter holds a high level on TXD after each transmitted character during the number of bit periods defined in WATIVAL field.

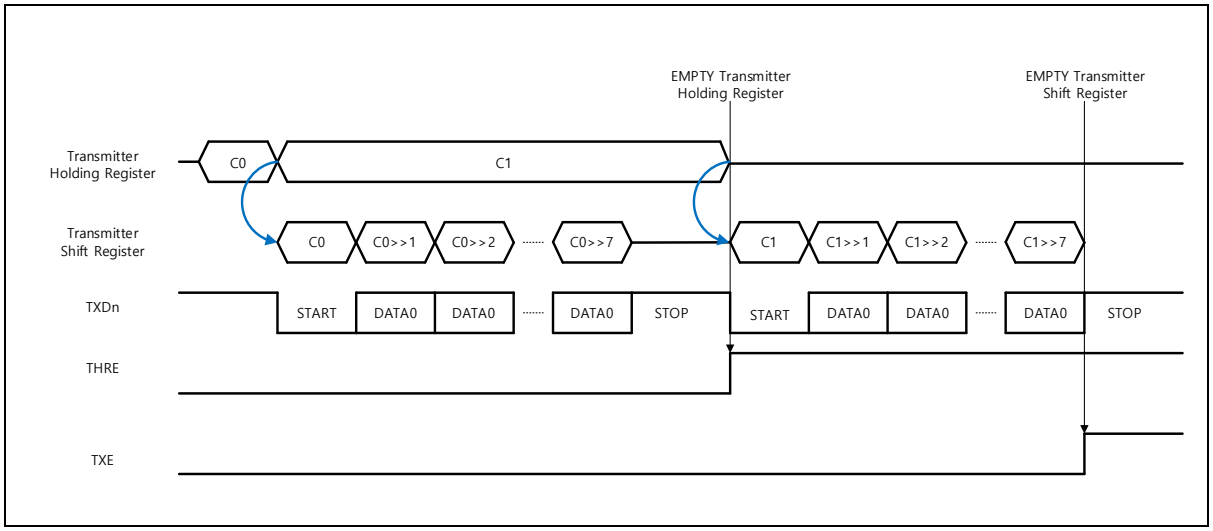
Figure 111. Inter-frame delay timing diagram



16.3.4 Transmit interrupt

The transmit operation makes some kind of interrupt flags. When transmitter holding register is empty, the THRE interrupt flag will be set. When transmitter shifter register is empty, the TXE interrupt flag will be set. User can select which interrupt timing is best for the application.

Figure 112. Transmit interrupt timing diagram



16.3.5 DMA transfers

UART supports DMA interface function which optionally depends on the device. Start memory address and length of the transfer data are programmed in the registers of DMA block. The end of transfer is notified by the transfer done flag.

The completion of data transfer is related to a notification by the transfer complete flag. Once the entire transmit data is written to the transmit data hold register (THR), the DMA flag bit in UARTn_IIR register that indicates DMA complete is flagged along with the ID of the complete interrupt written to the register.

Once the entire receive data is written to the target DMA memory, the UARTn_IIR register's DMA complete flag is set along with the ID of the complete interrupt written to the IIR register. Therefore, the UART RxD signal is already idle when the DMA complete interrupt occurred.

17 Serial Peripheral Interface (SPI)

A channel serial interface is provided for synchronous serial communications with external peripherals. SPI block support both of master and slave mode. Four signals will be used for SPI communication such as SS, SCK, MOSI, and MISO.

SPI of A31G22x series features the followings:

- Master or Slave operation.
- Programmable clock polarity and phase.
- 8, 9, 16, 17-bit wide transmit/receive register.
- 8, 9, 16, 17-bit wide data frame.
- Loop-back mode.
- Programmable start, burst, and stop delay time.
- DMA transfer operation.

Table 81 introduces pins assigned for SPI.

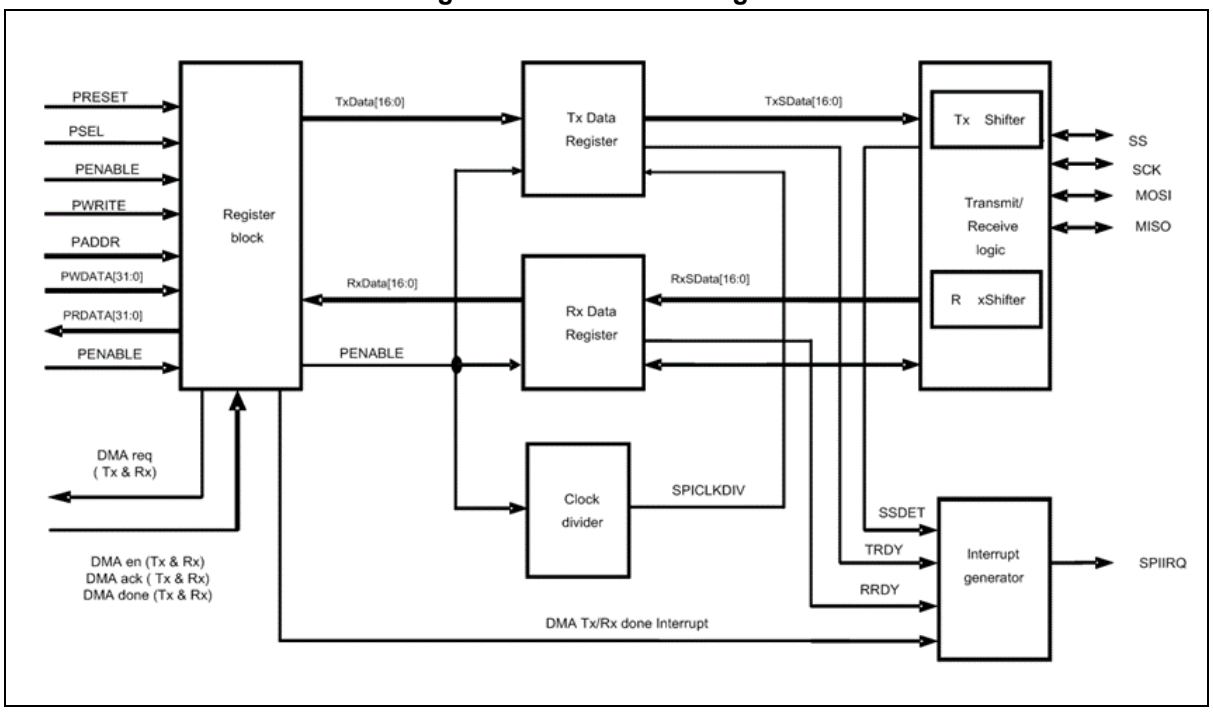
Table 81. Pin Assignment of SPI: External Pins

Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
SS20	I/O	Slave Select signal of SPI0	O	O	O
SCK20	I/O	Serial Clock signal of SPI0	O	O	O
MOSI20	I/O	Master-Out Slave-In Data signal of SPI0	O	O	O
MISO20	I/O	Master-In Slave-Out Data signal of SPI0	O	O	O
SS21	I/O	Slave Select signal of SPI1	O	O	O
SCK21	I/O	Serial Clock signal of SPI1	O	O	O
MOSI21	I/O	Master-Out Slave-In Data signal of SPI1	O	O	O
MISO21	I/O	Master-In Slave-Out Data signal of SPI1	O	O	O

17.1 SPI block diagram

In this section, SPI is described in a block diagram in Figure 113.

Figure 113. SPI Block Diagram



17.2 Registers

Base address of SPI is introduced in the followings:

Table 82. Base Address of SPI

Name	Base address
SPI20	0x4000_4C00
SPI21	0x4000_4D00

Table 83. SPI Register Map

Name	Offset	Type	Description	Reset value	Ref.
SPIn_TDR	0x00	WO	SPIn Transmit Data Register	0x0000_0000	17.2.1
SPIn_RDR	0x00	RO	SPIn Receive Data Register	0x0000_0000	17.2.2
SPIn_CR	0x04	RW	SPIn Control Register	0x0000_1020	17.2.3
SPIn_SR	0x08	RW	SPIn Status Register	0x0000_0006	17.2.4
SPIn_BR	0x0C	RW	SPIn Baud rate Register	0x0000_FFFF	17.2.5
SPIn_EN	0x10	RW	SPIn Enable register	0x0000_0000	17.2.6
SPIn_LR	0x14	RW	SPIn delay Length Register	0x0001_0101	17.2.7

NOTE: n = 20 and 21

17.2.1 SPIn_TDR: SPI transmit data register

SPIn_TDR is a 17-bit sized read/write register. It contains serial transmit data. This register is used to transmit a data to the other SPI channel a devices. (n = 20 and 21)

SPI20_TDR=0x4000_4C00, SPI21_TDR=0x4000_4D00

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		TDR	
-		0x00000	
-		RW	

Bits	Name	Function
16 0	TDR	Transmit Data Register

17.2.2 SPIn_RDR: SPI receive data register

SPIn_RDR is a 17-bit sized read/write register. It contains serial receive data. This register is used to receive data from the other SPI channel the device. (n = 20 and 21)

SPI20_RDR=0x4000_4C00, SPI21_RDR=0x4000_4D00

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		RDR	
-		0x00000	
-		RW	

Bits	Name	Function
16 0	RDR	Receive Data Register

17.2.3 SPIn_CR: SPI control register

SPIn_CR is a 20-bit sized read/write register and can be set to configure SPI operation mode. (n = 20 and 21)

SPI20_CR=0x4000_4C04, SPI21_CR=0x4000_4D04

31 30 29 28 27 26 25 24	23 22 21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		TXBC	RXBC	DTXIE	DRXIE	SSCIE	TXIE	RXIE	SSMOD	SSOUT	LBE	SSMARK	SSMO	SSPOL	Reserved		MS	MSBF	CPHA	CPOL	BITSZ	
-		0	0	0	0	0	0	0	0	0	1	0	0	0	-		1	0	0	0	00	
-		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	-		RW	RW	RW	RW	RW	

Bits	Name	Function
20	TXBC	Tx buffer clear bit.
		0 No action
		1 Clear Tx buffer
19	RXBC	Rx buffer clear bit
		0 No action
		1 Clear Rx buffer
18	TXDIE	DMA Tx done interrupt enable bit.
		0 DMA Tx done interrupt is disabled.
		1 DMA Tx done interrupt is enabled.
17	RXDIE	DMA Rx done interrupt enable bit.
		0 DMA Rx done interrupt is disabled.
		1 DMA Rx done interrupt is enabled.
16	SSCIE	SS edge change interrupt enable bit.
		0 nSS interrupt is disabled.
		1 nSS interrupt is enabled for both edges (L→H, H→L)
15	TXIE	Transmit interrupt enable bit.
		0 Transmit interrupt is disabled.
		1 Transmit interrupt is enabled.
14	RXIE	Receive interrupt enable bit.
		0 Receive interrupt is disabled.
		1 Receive interrupt is enabled.
13	SSMOD	SS auto/manual output select bit. **NOTE:** When performing high-speed SPI data communication faster than 1 MHz, it is recommended to manually control the SS signal by setting SSMOD = 1.
		0 SS output is not set by SSOUT (SPIn_CR[12]). SS signal is in normal operation mode.
		1 SS output signal is set by SSOUT.
12	SSOUT	SS output signal select bit.
		0 SS output is 'L.'

		1	SS output is 'H'.
11	LBE	Loop-back mode select bit in master mode.	
		0	Loop-back mode is disabled.
		1	Loop-back mode is enabled.
10	SSMASK	SS signal masking bit in slave mode.	
		0	SS signal masking is disabled. Receive data when SS signal is active.
		1	SS signal masking is enabled. Receive data at SCLK edges. SS signal is ignored.
9	SSMO	SS output signal select bit.	
		0	SS output signal is disabled.
		1	SS output signal is enabled.
8	SSPOL	SS signal Polarity select bit.	
		0	SS signal is ACTIVE-LOW.
		1	SS signal is ACTIVE-HIGH.
5	MS	Master/Slave select bit.	
		0	SPI is in SLAVE mode.
		1	SPI is in MASTER mode.
4	MSBF	MSB/LSB transmit select bit.	
		0	LSB is transferred first.
		1	MSB is transferred first.
3	CPHA	SPI clock phase bit.	
		0	Sampling of data occurs at odd edges (1, 3, 5,…, 15).
		1	Sampling of data occurs at even edges (2, 4, 6,…, 16).
2	CPOL	SPI clock polarity bit.	
		0	Active-high clocks selected.
		1	Active-low clocks selected.
1 0	BITSZ	Transmit/Receive Data Bits select bit.	
		00	8 bits
		01	9 bits
		10	16 bits
		11	17 bits

NOTES:

1. CPOL=0, CPHA=0 : data sampling at rising edge, data changing at falling edge
2. CPOL=0, CPHA=1 : data sampling at falling edge, data changing at rising edge
3. CPOL=1, CPHA=0 : data sampling at falling edge, data changing at rising edge
4. CPOL=1, CPHA=1 : data sampling at rising edge, data changing at falling edge

17.2.4 SPIn_SR: SPI status register

SPIn_SR is a 10-bit sized read/write register. It contains the status of SPI interface. (n = 20 and 21)

SPI20_SR=0x4000_4C08, SPI21_SR=0x4000_4D08

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10	9	8	7	6	5	4	3	2	1	0
Reserved			TXDMAF	RXDMAF	Reserved	SSDET	SSON	OVRF	UDRF	TXIDLE	TRDY	RRDY
-			0	0	-	0	0	0	0	1	1	0
-			RC	RC	-	RC	RC	RC	RC	RO	RC	RC

Bits	Name	Function	
9	TXDMAF	DMA transmit operation complete flag. (DMA to SPI)	
		0	DMA transmit operation is working or is disabled.
		1	DMA transmit operation is done. This value is cleared by writing '1' to this field.
8	RXDMAF	DMA receive operation Complete flag. (SPI to DMA)	
		0	DMA receive operation is working or is disabled.
		1	DMA receive operation is done. This value is cleared by writing '1' to this field.
6	SSDET	The rising or falling edge of SS signal detect flag.	
		0	SS edge is not detected.
		1	SS edge is detected. The bit is cleared when it is written as "0". This value is cleared by writing '1' to this field.
5	SSON	SS signal status flag.	
		0	SS signal is inactive.
		1	SS signal is active. This value is cleared by writing '1' to this field.
4	OVRF	Receive overrun error flag.	
		0	Receive overrun error is not detected.
		1	Receive overrun error is detected. This bit is cleared by writing or reading SPIn_RDR.
3	UDRF	Transmit underrun error flag.	
		0	Transmit underrun is not occurred.
		1	Transmit underrun is occurred. This bit is cleared by writing or reading SPIn_TDR.
2	TXIDLE	Transmit/Receive operation flag.	
		0	SPI is transmitting data
		1	SPI is in IDLE state.
1	TRDY	Transmit buffer empty flag.	
		0	Transmit buffer is busy.
		1	Transmit buffer is ready. This bit is cleared by writing data to SPIn_TDR.
0	RRDY	Receive buffer ready flag.	

0	Receive buffer has no data.
1	Receive buffer has data. This bit is cleared by writing data to SPIn_RDR.

17.2.5 SPIn_BR: SPI baud rate register

SPIn_BR is a 16-bit sized read/write register. Baud rate can be set by writing the register.

SPI20_BR=0x4000_4C0C, SPI21_BR=0x4000_4D0C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		BR	
-		0xFFFF	
-		RW	

Bits	Name	Function
15 0	BR	Baud rate setting bits Baud Rate = PCLK / (BR + 1)

NOTES:

1. BR[15:0] must be set 2 or greater. (BR[15:0] ≥ 2)
2. For SPI speed, it is recommended to set the BR value to 2 or higher so that the SPI input clock is divided by at least 3.
   e.g., PCLK = 24 MHz, BR = 2, SPI Freq. = 24 MHz / (2 + 1) = 8 MHz

17.2.6 SPIn_EN: SPI enable register

SPIn_EN is a bit sized read/write register. It contains SPI enable bit. (n = 20 and 21)

SPI20_EN=0x4000_4C10, SPI21_EN=0x4000_4D10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			ENABLE
-			0
-			RW

Bits	Name	Function	
0	ENABLE	SPI Enable bit	
		0	SPI is disabled. SPIn_SR register value is initialized by writing "0" to this bit but other registers aren't initialized.
		1	SPI is enabled. When this bit is written as "1", the dummy data of transmit buffer will be shifted. To prevent this, write data to SPIn_TDR before this bit is active.

17.2.7 SPIn_LR: SPI delay length register

SPIn_LR is a 24-bit sized read/write register. It contains values regarding each length of Start, Burst, and Stop respectively (n = 20 and 21). Settings of SPIn_LR is affected by each value of Start, Burst, and Stop Delay respectively when SS output function is enabled. However, only Burst Delay is affected when SS output function is disabled and a user controls it using GPIO directly.

Data Read Access Timing during SPI communication includes 2 more cycles than the Flash Access Timing. To achieve the optimized SPI communication, a user should consider the Flash Access Timing which is set in the system.

For example, Data Read Access Timing of SPI communication is totally 7 cycles, if code flash wait of the system is set to 4, because of 5 cycles (4 waits) + 2 cycles = 7 cycles.

To optimize the communication, a user must set each value of Start, Burst, and Stop Delay respectively by considering this operation.

SPI20_LR=0x4000_4C14, SPI21_LR=0x4000_4D14

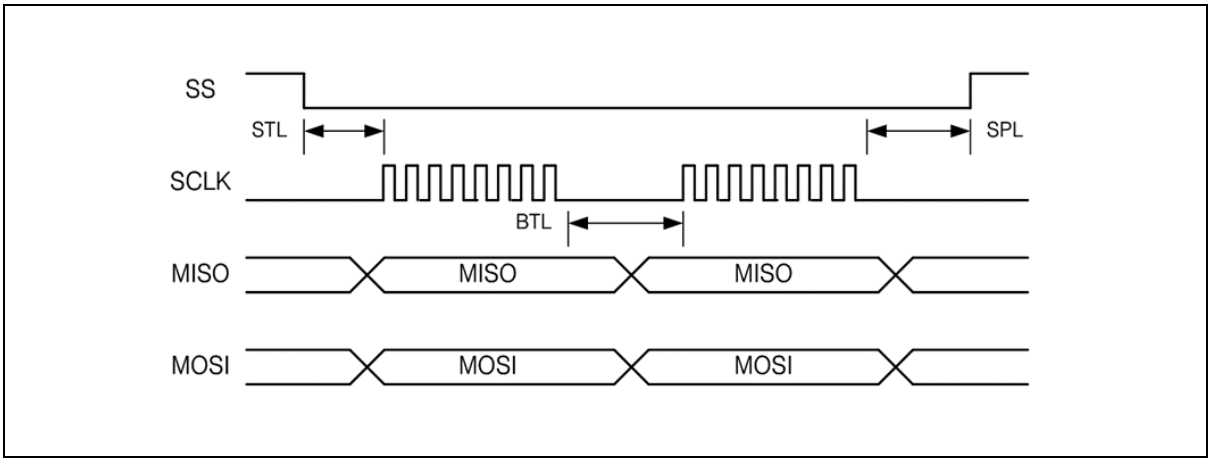
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	SPL	BTL	STL
-	0x01	0x01	0x01
-	RW	RW	RW

Bits	Name	Function
23 16	SPL	Stop length value 0x01 to 0xFF: 1 to 255 SCLKs. (SPL ≥ 1)
15 8	BTL	Burst length value 0x01 to 0xFF: 1 to 255 SCLKs. (BTL ≥ 1)
7 0	STL	Start length value 0x01 to 0xFF: 1 to 255 SCLKs. (STL ≥ 1)

NOTE: Due to the nature of code Flash, the access time is slower than RAM, causing flash waits. Since the reception of data during SPI high speed communication performed in code flash may be faster than the data access time, so we recommend using a delay between the sending SPI data. By running code in SRAM and using DMA, you can use it without delay in high speed communication.

$$\mathbf{BTL\ (Burst\ Delay)} \geq \frac{(\boldsymbol{Access\ wait\ of\ Code\ Flash} + \mathbf{1}) * \boldsymbol{SCLK} * (\boldsymbol{xbit} + \mathbf{2})}{\boldsymbol{HCLK}} + \frac{\mathbf{4}}{(\boldsymbol{BR} + \mathbf{1})}$$

Figure 114. SPI wave form (STL, BTL and SPL)



17.3 Functional description

SPI Transmit block and Receive block share Clock Gen Block but they are independent each other. Transmit block and Receive block have double buffers and SPI is available for back to back transfer operation.

17.3.1 SPI timing

SPI has four modes of operation. These modes essentially control the way data is clocked in or out of an SPI device. The configuration is done by two bits in the SPI control register (SPIn_CR). The clock polarity is specified by the CPOL control bit, which selects an active high or active low clock. The clock phase (CPHA) control bit selects one of the two fundamentally different transfer formats.

To ensure a proper communication between master and slave both devices have to run in the same mode. This can require a reconfiguration of the master to match the requirements of different peripheral slaves.

The clock polarity has no significant effect on the transfer format. Switching this bit causes the clock signal to be inverted (active high becomes active low and idle low becomes idle high). The settings of the clock phase, however, selects one of the two different transfer timings, which are described closer in the next two chapters. Since the MOSI and MISO lines of the master and the slave are directly connected to each other, the diagrams show the timing of both device, master and slave.

The nSS line is the slave select input of the slave. The nSS pin of the master is not shown in the diagrams. It has to be inactive by a high level on this pin (if configured as input pin) or by configuring it as an output pin.

The timing of a SPI transfer where CPHA is zero is shown in Figure 115 and Figure 116. Two wave forms are shown for the SCK signal -one for CPOL equals zero and another for CPOL equals one.

When the SPI is configured as a slave, the transmission starts with the falling edge of the /SS line. This activates the SPI of the slave and the MSB of the byte stored in its data register (SPIn_TDR) is output on the MISO line. The actual transfer is started by a software write to the SPIn_TDR of the master. This causes the clock signal to be generated. In cases where the CPHA equals zero, the SCLK signal remains zero for the first half of the first SCLK cycle. This ensures that the data is stable on the input lines of both the master and the slave.

The data on the input lines is read with the edge of the SCLK line from its inactive to its active. The edge of the SCLK line from its active to its inactive state (falling edge if CPOL equals zero and rising edge if CPOL equals one) causes the data to be shifted one bit further so that the next bit is output on the MOSI and MISO lines.

Figure 115. SPI Transfer Timing 1/4 (CPHA=0, CPOL=0, MSBF=0)

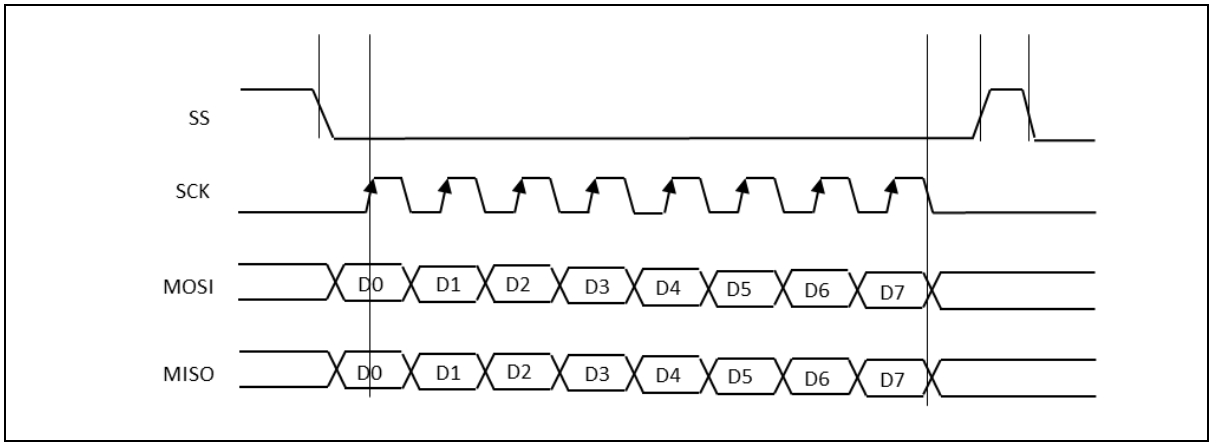
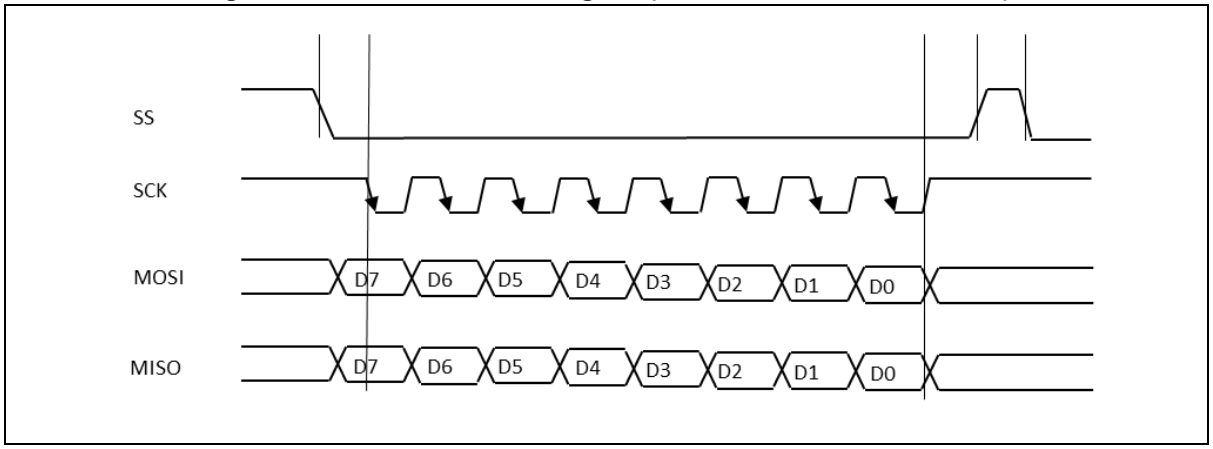


Figure 116. SPI Transfer Timing 2/4 (CPHA=0, CPOL=1, MSBF=1)



The timing of a SPI transfer where CPHA is one is shown in Figure 117 and Figure 118. Two wave forms are shown for the SCLK signal -one for CPOL equals zero and another for CPOL equals one.

Like in the previous cases the falling edge of the nSS lines selects and activates the slave. Compared to the previous cases, where CPHA equals zero, the transmission is not started and the MSB is not output by the slave at this stage. The actual transfer is started by a software write to the SPIn_TDR of the master what causes the clock signal to be generated. The first edge of the SCLK signal from its inactive to its active state (rising edge if CPOL equals zero and falling edge if CPOL equals one) causes both the master and the slave to output the MSB of the byte in the SPIn_TDR.

As shown in Figure 117 and Figure 118, there is no delay of half a SCLK-cycle. The SCLK line changes its level immediately at the beginning of the first SCLK-cycle. The data on the input lines is read with the edge of the SCLK line from its active to its inactive state (falling edge if CPOL equals zero and rising edge if CPOL equals one). After eight clock pulses the transmission is completed.

Figure 117. SPI Transfer Timing 3/4 (CPHA=1, CPOL=0, MSBF=0)

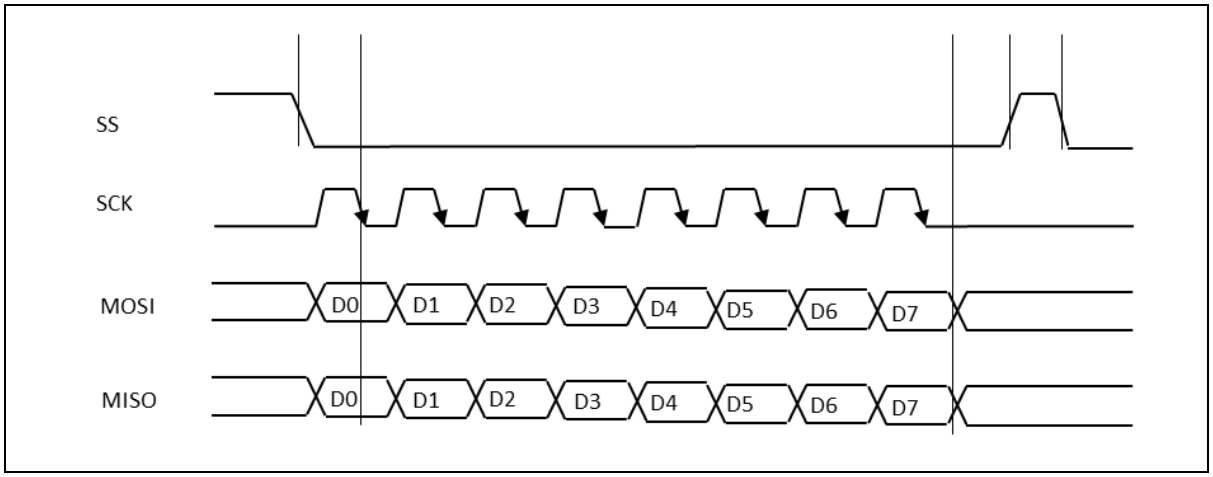
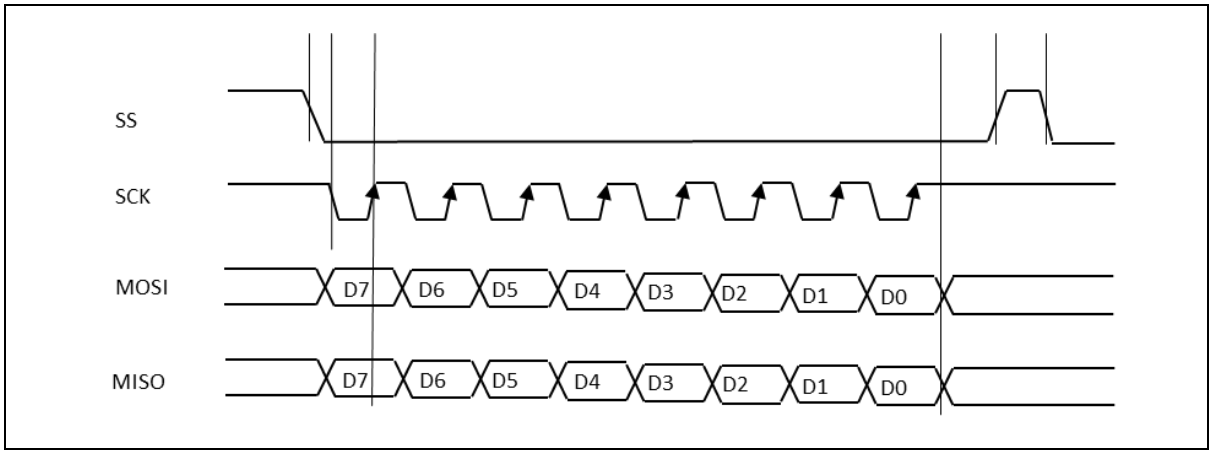


Figure 118. SPI Transfer Timing 4/4 (CPHA=1, CPOL=1, MSBF=1)



17.3.2 DMA handshake

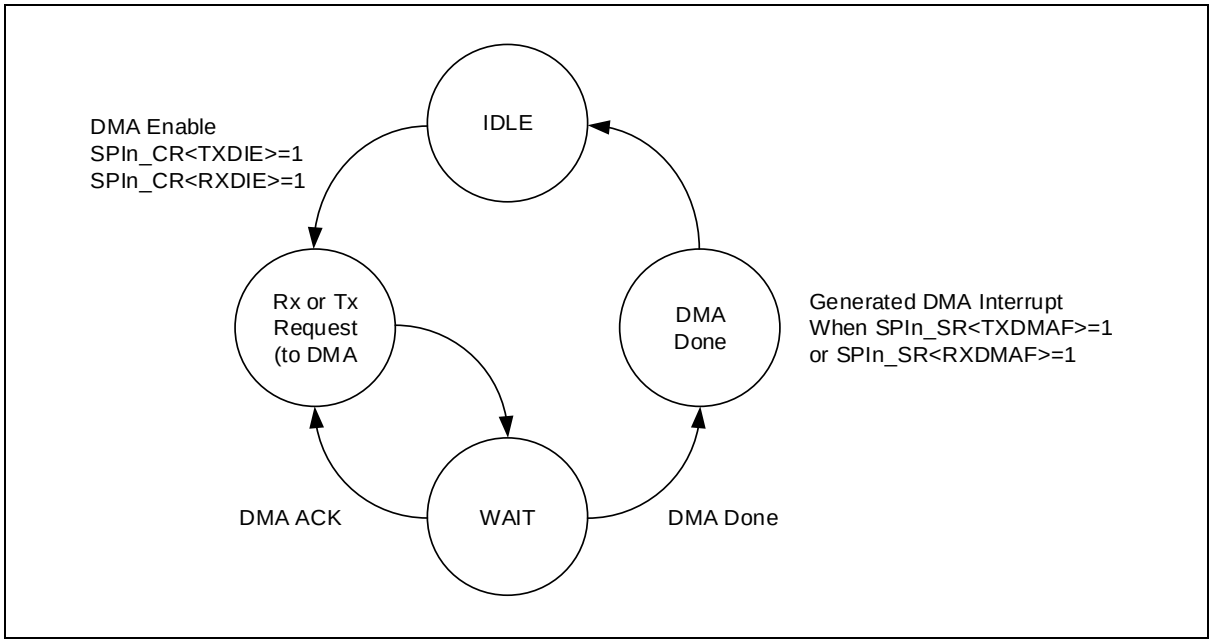
SPI supports DMA handshaking operation. In order to operate DMA handshake, DMA registers should be set first (see Chapter 9 Direct Memory Access Controller (DMAC)). As Transmitter and receiver are independent each other, SPI can operate the two channels at the same time.

After DMA channel for receiver is enabled and receive buffer is filled, SPI sends Rx request to DMA to empty the buffer and waits ACK signal from DMA. If Receive buffer is filled again after ACK signal, SPI sends Rx request. If DMA Rx DONE becomes high, RXDMAF (SPIn_SR[8]) goes "1" and an interrupt is serviced when RXDIE (SPIn_CR[17]) is set.

Likewise, if transmit buffer is empty after DMA channel for transmitter is enabled, SPI sends Tx request to DMA to fill the buffer and waits ACK signal from DMA. If transmit buffer is empty again after ACK signal, SPI sends Tx request. If DMA Tx DONE becomes high, TXDMAF (SPIn_SR[9]) goes "1" and an interrupt is serviced when TXDIE (SPIn_CR[18]) is set.

Slave transmitter sends dummy data at the first transfer (8 to 17 SCLKs) in DMA handshake mode.

Figure 119. DMA Handshake Flow Chart



18 I2C interface

I2C is one of industrial standard serial communication protocols, and which uses 2 bus lines such as Serial Data Line (SDAn) and Serial Clock Line (SCLn) to exchange data. Because both SDAn and SCLn lines are open-drain output, each line needs pull-up resistor respectively.

I2C features the followings (n = 0, 1 and 2):

- Compatible with I2C bus standard.
- Multi-master operation.
- Up to 400kHz data transfer read speed.
- 7-bit address.
- Support two slave address.
- Both master and slave operation.
- Bus busy detection

Introduces pins assigned for I2C interface.

Table 84. Pin Assignment of I2C: External Pins

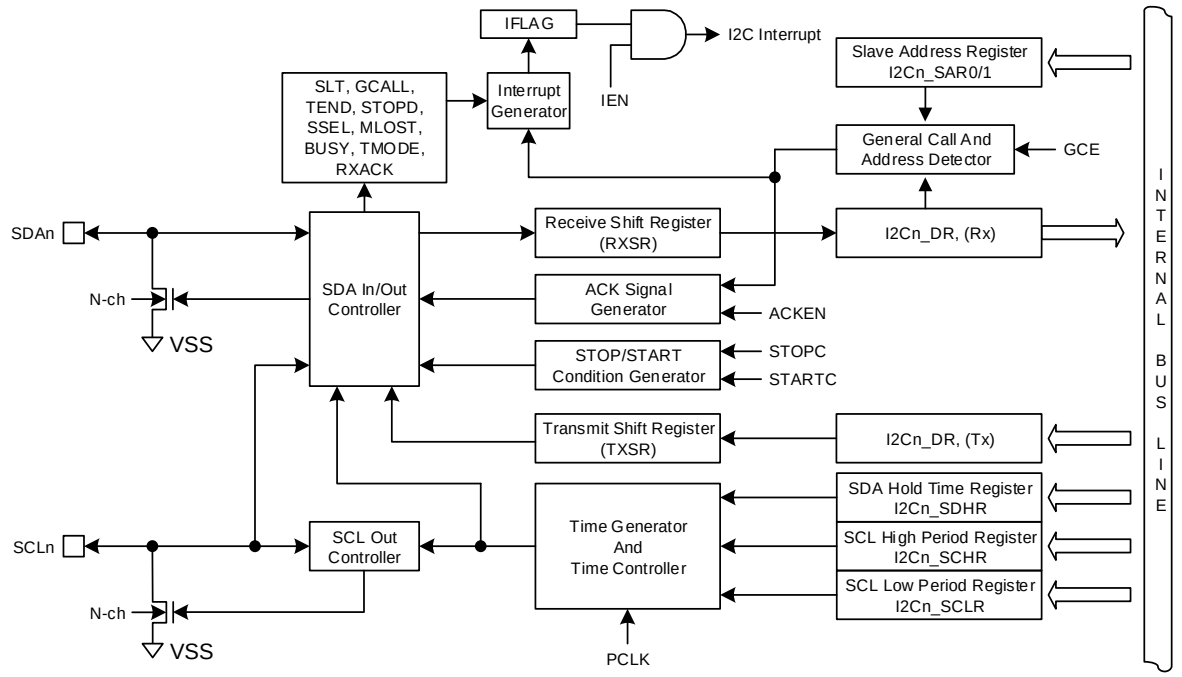
Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
SCL0	I/O	I2C channel 0 Serial clock bus line (open-drain)	O	O	O
SDA0	I/O	I2C channel 0 Serial data bus line (open-drain)	O	O	O
SCL1	I/O	I2C channel 1 Serial clock bus line (open-drain)	O	O	O
SDA1	I/O	I2C channel 1 Serial data bus line (open-drain)	O	O	O
SCL2	I/O	I2C channel 2 Serial clock bus line (open-drain)	O	O	-
SDA2	I/O	I2C channel 2 Serial data bus line (open-drain)	O	O	-

NOTE: n = 0, 1 and 2

18.1 I2C block diagram

In this section, I2C interface block is described in a block diagram.

Figure 120. I2C Block Diagram



NOTE: When the corresponding port of I2C is a sub-function for SCL/SDA pin, the SCL/SDA pins are automatically set to the N-channel open-drain outputs and the input latch is read in the case of reading the pins. The corresponding pull-up resistor is determined by the control register.

18.2 Registers

Base address of I2C is introduced in the followings:

Table 85. Base Address of I2C Interface

Name	Base address
I2C0	0x4000_4800
I2C1	0x4000_4900
I2C2	0x4000_4A00

Table 86. I2C Register Map

Name	Offset	Type	Description	Reset value	Ref.
I2Cn_CR	0x00	RW	I2Cn Control Register	0x0000_0000	18.2.1
I2Cn_ST	0x04	RW	I2Cn Status Register	0x0000_0000	18.2.2
I2Cn_SAR1	0x08	RW	I2Cn Slave Address Register 1	0x0000_0000	18.2.3
I2Cn_SAR2	0x0C	RW	I2Cn Slave Address Register 2	0x0000_0000	18.2.4
I2Cn_DR	0x10	RW	I2Cn Data Register	0x0000_0000	18.2.5
I2Cn_SDHR	0x14	RW	I2Cn SDA Hold Time Register	0x0000_0001	18.2.6
I2Cn_SCLR	0x18	RW	I2Cn SCL Low Period Register	0x0000_003F	18.2.7
I2Cn_SCHR	0x1C	RW	I2Cn SCL High Period Register	0x0000_003F	18.2.8
I2Cn_SLTCR	0x20	RW	I2Cn SCL low timeout control register	0x0000_0000	18.2.9
I2Cn_SLTPDR	0x24	RW	I2Cn SCL Low Timeout Period Data Register	0x00FF_FFFF	18.2.10
I2C_MBCR	0x28	RW	I2Cn Manual Bus Control Register	0x0000_030C	18.2.11

NOTE: n = 0, 1 and 2

18.2.1 I2Cn_CR: I2Cn control register

The register can be set to configure I2C operation mode and simultaneously allowed for I2C transactions to be kicked off. I2Cn_CR is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_CR=0x4000_4800, I2C1_CR=0x4000_4900, I2C2_CR=0x4000_4A00

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
Reserved			EN	TXDLYENB	IEN	IFLAG	ACKEN	IMASTER	STOPC	STARTC
-			0	0	0	0	0	0	0	0
-			RW	RW	RW	RO	RW	RO	RW	RW

Bits	Name	Function
7	EN	Activate I2Cn Block by supplying.
		0 Disable I2Cn block.
		1 Enable I2Cn block.
6	TXDLYENB	I2Cn_SDHR Register Control bit.
		0 Enable I2Cn_SDHR register.
		1 Disable I2Cn_SDHR register.
5	IEN	I2Cn Interrupt Enable bit.
		0 Disable I2Cn interrupt.
		1 Enable I2Cn interrupt.
4	IFLAG	I2Cn Interrupt Flag bit. This bit is cleared when all interrupt source bits in the I2Cn_ST register are cleared to "0b".
		0 No request occurred.
		1 Request occurred.
3	ACKEN	Controls ACK signal generation at ninth SCL period.
		0 No ACK signal is generated. (SDA = 1)
		1 ACK signal is generated. (SDA = 0)
		NOTE: AC K signal is output (SDA = 0) for the following 3 cases. Where x = 0 and 1. 1. When received address packet equals to SLAx[6:0] bits in I2Cn_SARx register. 2. When received address packet equals to value 0x00 with GCALL enabled. 3. When I2Cn operates as a receiver (master or slave)
2	IMASTER	Represent Operation Mode of I2Cn. This bit is cleared to "0b" on STOP condition.
		0 I2Cn is in slave mode.
		1 I2Cn is in master mode.
1	STOPC	STOP Condition Generation when I2Cn is master.
		0 No effect.

		1	STOP condition is to be generated.
0	STARTC	START condition generation when I2Cn is master.	
		0	No effect.
		1	START or Repeated START condition is to be generated.

18.2.2 I2Cn_ST: I2Cn status register

I2Cn_ST is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_ST=0x4000_4804, I2C1_ST=0x4000_4904, I2C2_ST=0x4000_4A04

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6	5	4	3	2	1	0
SLT	Reserved		GCALL	TEND	STOPD	SSEL	MLOST	BUSY	TMODE	RXACK
0	-		0	0	0	0	0	0	0	0
RW	-		RW	RW	RW	RW	RW	RW	RO	RW

Bits	Name	Function
31	SLT	This bit shows SCL low timeout status
		0 SCL low timeout has not occurred.
		1 SCL low timeout has occurred.
7	GCALL	This bit has different meaning depending on whether I2C is master or slave. When I2C is a master, this bit represents whether it received AACK (address ACK) from slave.
		0 No AACK is received. (Master mode)
		1 AACK is received (Master mode). It may be set to '1' after address transmission.
		When I2C is a slave, this bit is used to indicate general call.
		0 General call address is not detected. (Slave mode)
		1 General call address is detected. (Slave mode)
6	TEND	This bit is set when 1-byte of data is transferred completely.
		0 1 byte of data is not completely transferred.
		1 1 byte of data is completely transferred.
5	STOPD	This bit is set when a STOP condition is detected.
		0 A STOP condition is not detected.
		1 A STOP condition is detected.
4	SSEL	This bit is set when I2C is addressed by other master.
		0 I2C is not selected as a slave.
		1 I2C is addressed by other master and acts as a slave.
3	MLOST	This bit represents the result of bus arbitration in master mode.
		0 I2C maintains bus mastership.
		1 I2C has lost bus mastership during arbitration process.
2	BUSY	This bit reflects bus status.
		0 I2C bus is idle, so a master can issue a START condition.
		1 I2C bus is busy.
1	TMODE	This bit is used to indicate whether I2C is transmitter or receiver.
		0 I2C is a receiver.
		1 I2C is a transmitter.

0	RXACK	This bit shows the state of ACK signal.	
		0	No ACK is received.
		1	ACK is received at ninth SCL period.

NOTES:

1. The SLT, GCALL, TEND, STOPD, SSEL, and MLOST bits can be source of interrupt.
2. When an I2C interrupt occurs except for DEEP-SLEEP mode, the SCL line is held low. To release SCL, Clear to "0b" all interrupt source bits in I2Cn_ST register.
3. The SLT, GCALL, TEND, STOPD, SSEL, MLOST, and RXACK bits are cleared when '1' is written to the corresponding bit.

18.2.3 I2Cn_SAR1: I2Cn slave address register 1

I2Cn_SAR1 is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_SAR1=0x4000_4808, I2C1_SAR1=0x4000_4908, I2C2_SAR2=0x4000_4A08

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1	0
Reserved			SLA	GCALLEN
–			0000000	0
–			RW	RW

Bits	Name	Function	
7 1	SLA	These bits configure the slave address 0 in slave mode.	
0	GCALLEN	This bit decides whether I2Cn allows general call address 0 or not in I2Cn slave mode.	
		0	Ignore general call address 0.
		1	Allow general call address 0.

18.2.4 I2Cn_SAR2: I2Cn slave address register 2

I2Cn_SAR2 is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_SAR2=0x4000_480C, I2C1_SAR2=0x4000_490C, I2C2_SAR2=0x4000_4A0C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1	0
Reserved			SLA	GCALLEN
-			0000000	0
-			RW	RW

Bits	Name	Function	
7 1	SLA	These bits configure the slave address 1 in slave mode.	
0	GCALLEN	This bit decides whether I2Cn allows general call address 1 or not in I2Cn slave mode.	
		0	Ignore general call address 1.
		1	Allow general call address 1.

18.2.5 I2Cn_DR: I2Cn data register

I2Cn_DR is a 32-bit register and able to do 32/16/8-bit access. This register is used to receive data transmitted by other I2C channel by reading DATA field, or to transfer data to another I2C channel by being written in DATA field. (n = 0, 1 and 2)

I2C0_DR=0x4000_4810, I2C1_DR=0x4000_4910, I2C2_DR=0x4000_4A10

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			DATA
			0x00
			RW

Bits	Name	Function
7 0	DATA	The I2Cn_DR transmit buffer and receive buffer share the same I/O address with this DATA register. The transmit data buffer is the destination for data written to the I2Cn_DR register. Reading the I2Cn_DR register returns the contents of the receive buffer.

18.2.6 I2Cn_SDHR: I2Cn SDA hold time register

I2Cn_ SDHR is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_SDHR=0x4000_4814, I2C1_SDHR=0x4000_4914, I2C2_SDHR=0x4000_4A14

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			HLDT
-			0x001
-			RW

Bits	Name	Function
11 0	HLDT	This register is used to control SDA output timing from the falling edge of SCL. Note that SDA is changed after tPCLK X (I2Cn_SDHR+2). In master mode, load half the value of I2Cn_SCLR to this register to make SDA change in the middle of SCL. In slave mode, configure this register regarding the frequency of SCL from master. The SDA is changed after tPCLK X (I2Cn_SDHR+2) in master mode. So, to insure operation in slave mode, the value tPCLK X (I2Cn_SDHR + 2) must be smaller than the period of SCL.

18.2.7 I2Cn_SCLR: I2Cn SCL low period register

I2Cn_ SCLR is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_SCLR=0x4000_4818, I2C1_SCLR=0x4000_4918, I2C2_SCLR=0x4000_4A18

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			SCLL
-			0x03F
-			RW

Bits	Name	Function
11 0	SCLL	This register defines the low period of SCL in master mode. The base clock is PCLK and the period is calculated by the formula: tPCLK X (4 X I2Cn_SCLR + 2) where tPCLK is the period of PCLK.

18.2.8 I2Cn_SCHR: I2Cn SCL high period register

I2Cn_ SCHR is a 32-bit register and able to do 32/16/8-bit access. (n = 0, 1 and 2)

I2C0_SCHR=0x4000_481C, I2C1_SCHR=0x4000_491C, I2C2_SCLR=0x4000_4A1C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			SCLH
-			0x03F
-			RW

Bits	Name	Function
11 0	SCLH	This register defines the high period of SCL in master mode. The base clock is PCLK and the period is calculated by the formula: tPCLK X (4 X I2Cn_SCHR + 2) where tPCLK is the period of PCLK.

18.2.9 I2Cn_SLTCR: I2Cn SCL low timeout control register

I2Cn SCL Low Timeout Control Register is 32-bit register. This register supports to configure SCL low timeout and interrupt function. (n = 0, 1 and 2)

I2C0_SLTCR=0x4000_4820, I2C1_ SLTCR=0x4000_4920, I2C2_ SLTCR=0x4000_4A20

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0	
Reserved			SLTINT	SLTEN
			0	0
			RW	RW

Bits	Name	Function	
1	SLTINT	Selection of SCL low timeout Interrupt.	
		0	Disable
		1	Enable
0	SLTEN	SCL low timeout enable bit.	
		0	Disable timeout
		1	Enable timeout

18.2.10 I2Cn_SLTPDR: I2Cn SCL low timeout period data register

I2Cn SCL Low Timeout Period Data Register is 32-bit register. When the SCL low timeout function is activated, it runs from the time the SCL signal goes low to the duration set by the user. (n = 0, 1 and 2)

I2C0_SLTPDR=0x4000_4824, I2C1_SLTPDR=0x4000_4924, I2C2_SLTPDR=0x4000_4A24

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved	PDATA		
-	0xFFFFFF		
-	RW		

Bits	Name	Function
23 0	PDATA	This register defines the period of SCL low timeout. The base clock is PCLK and the period is calculated by the formula: t_{PCLK} X 4 X (PDATA + 1) where t_{PCLK} is the period of PCLK.

18.2.11 I2Cn_MBCR: I2Cn manual bus control register

I2Cn Manual Bus Control Register is 32-bit register. The I2Cn_MBCR register supports manual control of SCL and SDA signals on I2Cn channel. (n = 0, 1 and 2)

I2C0_MBCR=0x4000_4828, I2C1_MBCR=0x4000_4928, I2C2_MBCR=0x4000_4A28

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10	9	8	7 6 5 4	3	2	1	0
Reserved			SCLS	SDAS	Reserved	SCLO	SDAO	SCLMCE	SDAMCE
-			1	1	-	1	1	0	0
-			RO	RO	-	RW	RW	RW	RW

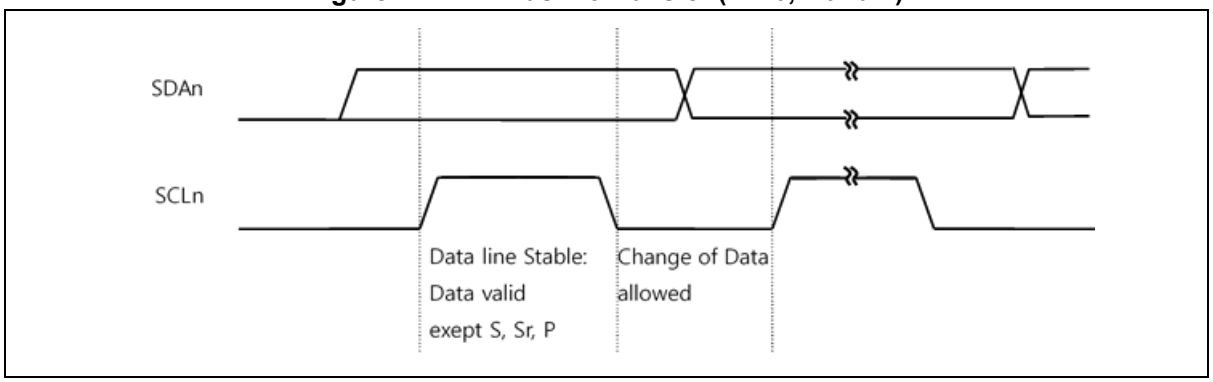
Bits	Name	Function
9	SCLS	SCL status bit.
		0 Low
		1 High
8	SDAS	SDA status bit.
		0 Low
		1 High
3	SCLO	SCL output data bit. (This value is valid when SCLMCE = 1)
		0 Low
		1 High (Open-drain)
2	SDAO	SDA output data bit. (This value is valid when SDAMCE = 1)
		0 Low
		1 High (Open-drain)
1	SCLMCE	SCL manual control enable bit.
		0 Disable manual control
		1 Enable manual control
0	SDAMCE	SDA manual control enable bit.
		0 Disable manual control
		1 Enable manual control

18.3 Functional description

18.3.1 I2C bit transfer

Data on the SDAn line must be stable during HIGH period of the clock, SCLn. HIGH or LOW state of the data line can only change when the clock signal on the SCLn line is LOW. The exceptions are START(S), repeated START(Sr) and STOP(P) condition where data line changes when clock line is high.

Figure 121. I2C Bus Bit Transfer (n = 0, 1 and 2)



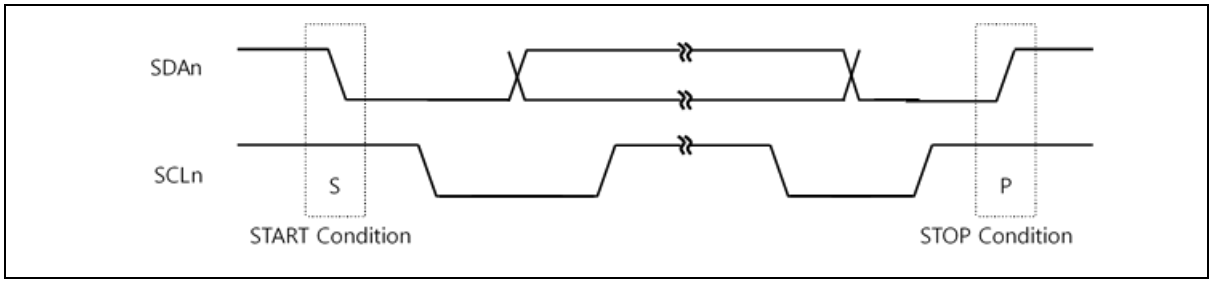
18.3.2 START/repeated START/STOP

One master can issue a START (S) condition to notice other devices connected to the SCLn, SDAn lines that it will use the bus. A STOP (P) condition is generated by the master to release the bus lines so that other devices can use it.

- A high to low transition on the SDAn line while SCLn is high defines a START (S) condition.
- A low to high transition on the SDAn line while SCLn is high defines a STOP (P) condition.

START and STOP conditions are always generated by the master. The bus is considered to be busy after START condition. The bus is considered to be free again after STOP condition, i.e., the bus is busy between START and STOP condition. If a repeated START condition (Sr) is generated instead of STOP condition, the bus stays busy. So, the START and repeated START conditions are functionally identical.

Figure 122. START and STOP Condition (n = 0, 1 and 2)

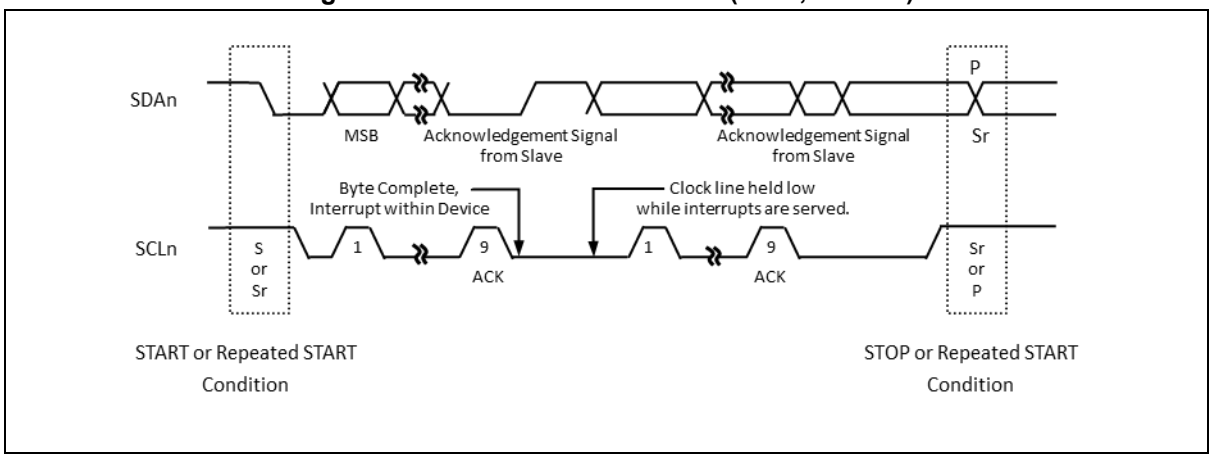


18.3.3 Data transfer

Every byte put on the SDAn line must be 8-bits long. The number of bytes that can be transmitted per transfer is unlimited. Each byte has to be followed by an acknowledge bit. Data is transferred with the most significant bit (MSB) first.

If a slave can't receive or transmit another complete byte of data until it has performed some other function, it can hold the clock line SCLn LOW to force the master into a wait state. Data transfer then continues when the slave is ready for another byte of data and releases clock line SCLn.

Figure 123. I2C Bus Data Transfer (n = 0, 1 and 2)



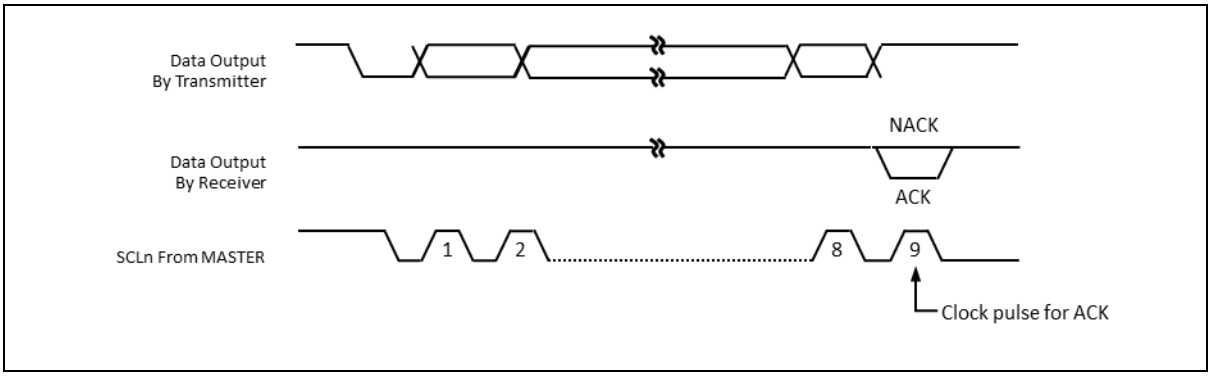
18.3.4 Acknowledge

The acknowledge related clock pulse is generated by the master. The transmitter releases the SDAn line (HIGH) during the acknowledge clock pulse. The receiver must pull down the SDAn line during the acknowledge clock pulse so that it remains stable LOW during the HIGH period of this clock pulse. When a slave is addressed by a master (Address Packet), and if it is unable to receive or transmit because it's performing some real time function, the data line must be left HIGH by the slave.

And also, when a slave addressed by a master is unable to receive more data bits, the slave receiver must release the SDAn line (Data Packet). The master can then generate either a STOP condition to abort the transfer, or a repeated START condition to start a new transfer.

If a master receiver is involved in a transfer, it must signal the end of data to the slave transmitter by not generating any acknowledge on the last byte that was clocked out of the slave. The slave transmitter must release the data line to allow the master to generate a STOP or repeated START condition.

Figure 124. I2C Bus Acknowledge (n = 0, 1 and 2)



18.3.5 Synchronization/arbitration

Clock synchronization is performed using the wired-AND connection of I2C interfaces to the SCLn line. This means that a HIGH to LOW transition on the SCLn line will cause the devices concerned to start counting off their LOW period and it will hold the SCLn line in that state until the clock HIGH state is reached. However the LOW to HIGH transition of this clock may not change the state of the SCLn line if another clock is still within its LOW period. In this way, a synchronized SCLn clock is generated with its LOW period determined by the device with the longest clock LOW period, and its HIGH period determined by the one with the shortest clock HIGH period.

A master may start a transfer only if the bus is free. Two or more masters may generate a START condition. Arbitration takes place on the SDAn line, while the SCLn line is at the HIGH level, in such a way that the master which transmits a HIGH level, while another master is transmitting a LOW level will switch off its DATA output state because the level on the bus doesn't correspond to its own level. Arbitration continues for many bits until a winning master gets the ownership of I2C bus. Its first stage is comparison of the address bits.

Figure 125. Clock Synchronization during the Arbitration Procedure (n = 0, 1 and 2)

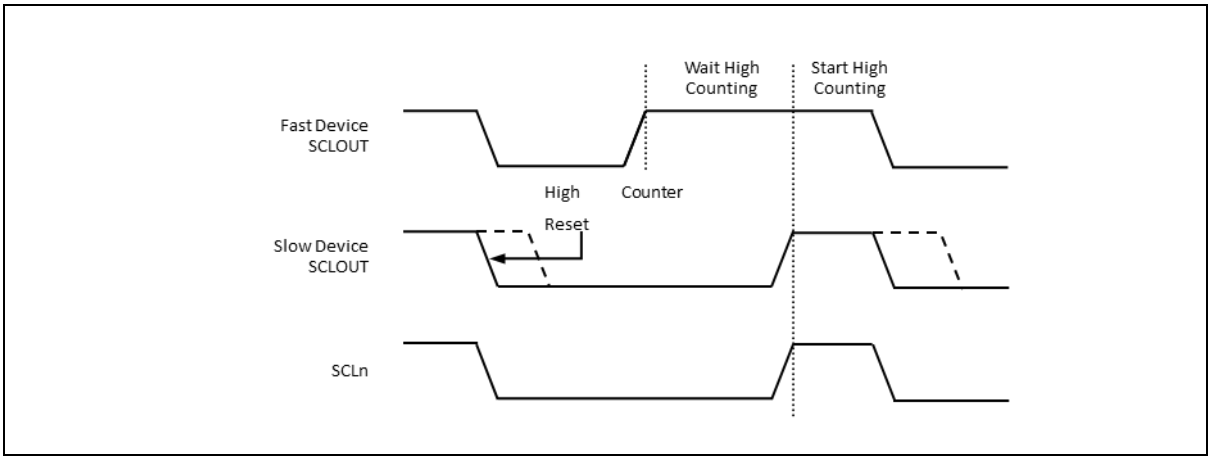
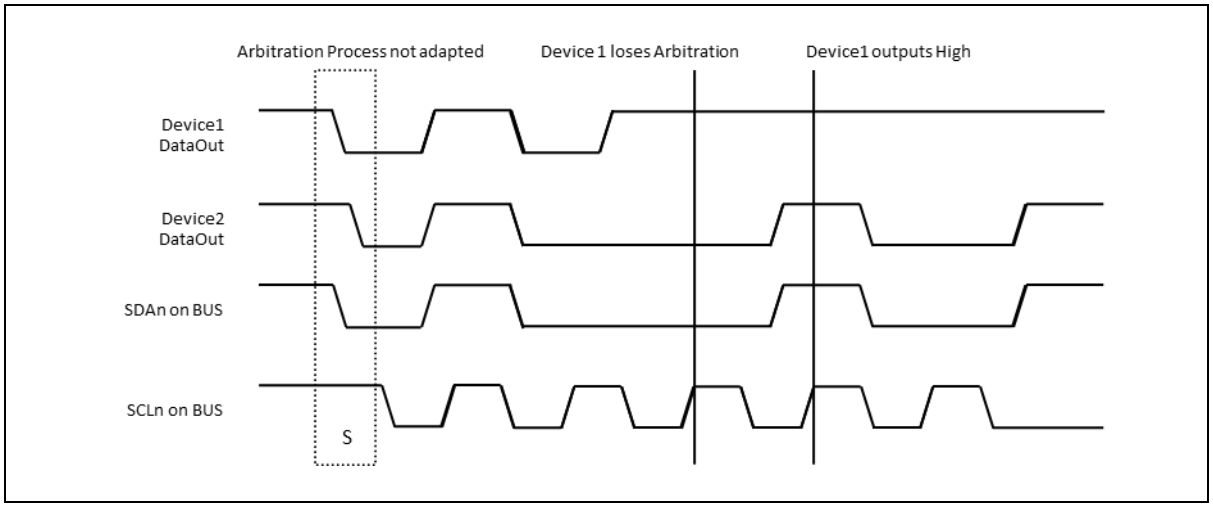


Figure 126. Arbitration Procedure between Two Masters (n = 0, 1 and 2)



18.3.6 I2C operation

The I2C is byte-oriented and interrupt based. Interrupts are issued after all bus events except for a transmission of a START condition. Because the I2C is interrupt based, the application software is free to carry on other operations during an I2C byte transfer.

Note that when an I2C interrupt is generated, IFLAG flag in IEN register is set, it is cleared when all interrupt source bits in the I2Cn_ST register are cleared to "0b". When I2C interrupt occurs, the SCLn line is hold LOW until clearing "0b" all interrupt source bits in I2Cn_ST register. When the I2CnIFLAG flag is set, the I2Cn_ST contains a value indicating the current state of the I2C bus. According to the value in I2Cn_ST, software can decide what to do next.

I2C can operate in 4 modes by configuring master/slave, transmitter/receiver. The operating mode is configured by a winning master. A more detailed explanation follows below. (n = 0, 1 and 2)

Master transmitter

To operate I2C in master transmitter, follow the recommended steps below:

1. Enable I2C by setting I2Cn_EN bit in I2Cn_CR. This provides main clock to the peripheral.
2. Load SLA+W into the I2Cn_DR where SLA is address of slave device and W is transfer direction from the viewpoint of the master. For master transmitter, W is '0'. Note that I2Cn_DR is used for both address and data.
3. Configure baud rate by writing desired value to both I2Cn_SCLR and I2Cn_SCHR for the Low and High period of SCLn line.
4. Configure the I2Cn_SDHR to decide when SDAn changes value from falling edge of SCLn. If SDA should change in the middle of SCLn LOW period, load half the value of I2Cn_SCLR to the I2Cn_SDHR.
5. Set the STARTC bit in I2Cn_CR. This transmits a START condition. And also configure how to handle interrupt and ACK signal. When the STARTC bit is set, 8-bit data in I2Cn_DR is transmitted out according to the baud-rate.
6. This is ACK signal processing stage for address packet transmitted by master. When 7-bit address and 1-bit transfer direction is transmitted to target slave device, the master can know whether the slave acknowledged or not in the 9th high period of SCLn. If the master gains bus mastership, I2C generates GCALL interrupt regardless of the reception of ACK from the slave device. When I2C loses bus mastership during arbitration process, the MLOST bit in I2Cn_ST is set, and I2C waits in idle state or can be operate as an addressed slave. To operate as a slave when the MLOST bit in I2Cn_ST is set, the ACKEN bit in I2Cn_CR must be set and the received 7-bit address must equal to the SLAn bits in I2Cn_SAR1/2.
   In this case I2C operates as a slave transmitter or a slave receiver (go to appropriate section).
   In this stage, I2C holds the SCLn LOW. This is because to decide whether I2C continues serial transfer or stops communication. The following steps continue assuming that I2C does not lose mastership during first data transfer.

   I2C (master) can choose one of the following cases regardless of the reception of ACK signal from slave.

   Case 1. Master receives ACK signal from slave, so continues data transfer because slave can receive more data from master. In this case, load data to transmit to I2Cn_DR.

   Case 2. Master stops data transfer even if it receives ACK signal from slave. In this case, set the STOPC bit in I2Cn_CR.

Case 3. Master transmits repeated START condition with not checking ACK signal. In this case, load SLA+R/W into the I2Cn_DR and set STARTC bit in I2Cn_CR.

After doing one of the actions above, clear all interrupt source bits in I2Cn_ST to "0b" to release SCLn line.

For the Case 1, move to step 7. For the Case 2, move to step 9 to handle STOP interrupt. For the Case 3, move back to step 6 after transmitting the data in I2Cn_DR and if transfer direction bit is '1' go to master receiver section.

7. 1-Byte of data is being transmitted. During data transfer, bus arbitration continues.

8. This is ACK signal processing stage for data packet transmitted by master. I2C holds the SCLn LOW. When I2C loses bus mastership while transmitting data arbitrating other masters, the MLOST bit in I2Cn_ST is set. If then, I2C waits in idle state. When the data in I2Cn_DR is transmitted completely, I2C generates TEND interrupt.

   I2C can choose one of the following cases regardless of the reception of ACK signal from slave.

   Case A. Master receives ACK signal from slave, so continues data transfer because slave can receive more data from master. In this case, load data to transmit to I2Cn_DR.

   Case B. Master stops data transfer even if it receives ACK signal from slave. In this case, set the STOPC bit in I2CnCR.

   Case C. Master transmits repeated START condition with not checking ACK signal. In this case, load SLA+R/W into the I2Cn_DR and set the STARTC bit in I2Cn_CR.

After doing one of the actions above, clear all interrupt source bits in I2Cn_ST to "0b" to release SCL line. For the Case A, move back to step 7. For the Case B, move to step 9 to handle STOP interrupt. For the Case C, move back to step 6 after transmitting the data in I2C_DR, and if transfer direction bit is '1' go to master receiver section.

9. This is the final step for master transmitter function of I2C, handling STOP interrupt. The STOP bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write "0" to I2CnST. After this, I2C enters in idle state.

Master receiver

To operate I2C in master receiver, follow the recommended steps below:

1. Enable I2C by setting I2Cn_EN bit in I2Cn_CR. This provides main clock to the peripheral.

2. Load SLA+R into the I2Cn_DR where SLA is address of slave device and R is transfer direction from the viewpoint of the master. For master receiver, R is '1'. Note that I2Cn_DR is used for both address and data.

3. Configure baud rate by writing desired value to both I2Cn_SCLR and I2Cn_SCHR for the Low and High period of SCLn line.

4. Configure the I2CnSDHR to decide when SDAn changes value from falling edge of SCLn. If SDAn should change in the middle of SCLn LOW period, load half the value of I2Cn_SCLR to the I2Cn_SDHR.

5. Set the STARTC bit in I2Cn_CR. This transmits a START condition. And also configure how to handle interrupt and ACK signal. When the STARTC bit is set, 8-bit data in I2Cn_DR is transmitted out according to the baud-rate.

6. This is ACK signal processing stage for address packet transmitted by master. When 7-bit address and 1-bit transfer direction is transmitted to target slave device, the master can know whether the slave acknowledged or not in the 9th high period of SCLn. If the master gains bus mastership, I2C generates GCALL interrupt regardless of the reception of ACK from the slave device. When I2C loses bus mastership during arbitration process, the MLOST bit in I2Cn_ST is set, and I2C waits in idle state or can be operate as an addressed slave. To operate as a slave when the MLOST bit in I2Cn_ST is set, the ACKEN bit in I2Cn_CR must be set and the received 7-bit address must equal to the SLA bits in I2Cn_SAR1/2. In this case I2C operates as a slave transmitter or a slave receiver (go to appropriate section). In this stage, I2C holds the SCLn LOW. This is because to decide whether I2C continues serial transfer or stops communication. The following steps continue assuming that I2C does not lose mastership during first data transfer.

   I2C (master) can choose one of the following cases according to the reception of ACK signal from slave.

   Case 1. Master receives ACK signal from slave, so continues data transfer because slave can prepare and transmit more data to master. Configure ACKEN bit in I2Cn_CR to decide whether I2C Acknowledges the next data to be received or not.

   Case 2. Master stops data transfer because it receives no ACK signal from slave. In this case, set the STOPC bit in I2Cn_CR.

   Case 3. Master transmits repeated START condition due to no ACK signal from slave. In this case, load SLA+R/W into the I2Cn_DR and set STARTC bit in I2Cn_CR.

After doing one of the actions above, clear all interrupt source bits in I2Cn_ST to "0b" to release SCLn line. For the Case 1, move to step 7. For the Case 2, move to step 9 to handle STOP interrupt. For the Case 3, move to step 6 after transmitting the data in I2Cn_DR and if transfer direction bit is '0' go to master transmitter section.

7. 1-Byte of data is being received.

8. This is ACK signal processing stage for data packet transmitted by slave. I2C holds the SCLn LOW. When 1-Byte of data is received completely, I2C generates TEND interrupt.

   I2C can choose one of the following cases according to the RXACK flag in I2Cn_ST.

   Case A. Master continues to receive data from slave. To do this, set ACKEN bit in I2Cn_CR to acknowledge the next data to be received.

   Case B. Master wants to terminate data transfer when it receives next data by not generating ACK signal. This can be done by clearing ACKEN bit in I2Cn_CR.

   Case C. Because no ACK signal is detected, master terminates data transfer. In this case, set the STOPC bit in I2Cn_CR.

   Case D. No ACK signal is detected, and master transmits repeated START condition. In this case, load SLA+R/W into the I2CnDR and set the STARTC bit in I2Cn_CR.

After doing one of the actions above, clear all interrupt source bits in I2Cn_ST to "0b" to release SCL line. For the Case A and B, move to step 7. For the Case C, move to step 9 to handle STOP interrupt. For the Case D, move to step 6 after transmitting the data in I2Cn_DR, and if transfer direction bit is '0' go to master transmitter section.

9. This is the final step for master receiver function of I2C, handling STOP interrupt. The STOP bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write "0" value to I2Cn_ST. After this, I2C enters idle state.

Slave transmitter

To operate I2C in slave transmitter, follow the recommended steps below:

1. If the main operating clock (SCLK) of the system is slower than that of SCLn, load value 0x00 into I2Cn_SDHR to make SDAn change within one system clock period from the falling edge of SCLn. Note that the hold time of SDAn is calculated by SDAH x period of SCLK where SDAH is multiple of number of SCLK coming from I2CnSDHR. When the hold time of SDAn is longer than the period of SCLK, I2C (slave) cannot transmit serial data properly.

2. Enable I2C by setting I2CnIEN bit and I2CnEN bit in I2Cn_CR. This provides main clock to the peripheral.

3. When a START condition is detected, I2C receives one byte of data and compares it with SLA bits in I2Cn_SAR1/2. If the GCALLEN bit in I2Cn_SAR1/2 is enabled, I2C compares the received data with value 0x00, the general call address.

4. If the received address does not equal to SLA bits in I2Cn_SAR, I2C enters idle state i.e., waits for another START condition. Else if the address equals to SLAn bits and the ACKEN bit is enabled, I2C generates SSEL interrupt and the SCLn line is held LOW. Note that even if the address equals to SLA bits, when the ACKEN bit is disabled, I2C enters idle state. When SSEL interrupt occurs, load transmit data to I2Cn_DR and clear to "0b" all interrupt source bits in I2Cn_ST to release SCLn line.

5. 1-Byte of data is being transmitted.

6. In this step, I2C generates TEND interrupt and holds the SCLn line LOW regardless of the reception of ACK signal from master. Slave can select one of the following cases:

   Case 1. No ACK signal is detected and I2C waits STOP or repeated START condition.

   Case 2. ACK signal from master is detected. Load data to transmit into I2Cn_DR.

After doing one of the actions above, clear all interrupt source bits in I2Cn_ST to "0b" to release SCLn line. For the Case 1, move to step 7 to terminate communication. For the Case 2, move back to step 5. In either case, a repeated START condition can be detected. For that case, move back to step 4.

7. This is the final step for slave transmitter function of I2C, handling STOP interrupt. The STOPC bit indicates that data transfer between master and slave is over. To clear I2Cn_ST, write "0" to I2Cn_ST. After this, I2C enters idle state.

Slave receiver

To operate I2C in slave receiver, follow the recommended steps below:

1. If the main operating clock (SCLK) of the system is slower than that of SCLn, load value 0x00 into I2Cn_SDHR to make SDAn change within one system clock period from the falling edge of SCLn. Note that the hold time of SDAn is calculated by SDAH x period of SCLK where SDAH is multiple of number of SCLK coming from I2Cn_SDHR. When the hold time of SDAn is longer than the period of SCLK, I2C (slave) cannot transmit serial data properly.

2. Enable I2C by setting I2CnIEN bit in I2CnCR. This provides main clock to the peripheral.

3. When a START condition is detected, I2C receives one byte of data and compares it with SLA bits in I2C_SAR. If the GCALLEN bit in I2Cn_SAR1/2 is enabled, I2C compares the received data with value 0x00, the general call address.

4. If the received address does not equal to SLAn bits in I2Cn_SAR1/2, I2C enters idle state i.e., waits for another START condition. Else if the address equals to SLA bits and the ACKEN bit is enabled, I2C generates SSEL interrupt and the SCLn line is held LOW. Note that even if the address equals to SLA bits, when the ACKEN bit is disabled, I2C enters idle state. When SSEL interrupt occurs and I2C is ready to receive data, clear to "0b" all interrupt source bits in I2Cn_ST to release SCLn line.

5. 1-Byte of data is being received.

6. In this step, I2C generates TEND interrupt and holds the SCLn line LOW regardless of the reception of ACK signal from master. Slave can select one of the following cases:

   Case 1. No ACK signal is detected (ACKEN=0) and I2C waits STOP or repeated START condition.

   Case 2. ACK signal is detected (ACKEN=1) and I2C can continue to receive data from master.

After doing one of the actions above, clear all interrupt source bits in I2CnST to "0b" to release SCLn line. For the Case 1, move to step 7 to terminate communication. For the Case 2, move back to step 5. In either case, a repeated START condition can be detected. For that case, move back to step 4.

7. This is the final step for slave receiver function of I2C, handling STOP interrupt. The STOPC bit indicates that data transfer between master and slave is over. To clear I2CnST, write "0" to I2CnST. After this, I2C enters idle state.

19 12-bit ADC

ADC block of A31G22x series consists of an independent ADC unit featuring the followings:

- Max. 18 Channels Analog Input with 12-bit Resolution.
  - 80-pin: 18 channels
  - 64-pin: 18 channels
  - 48-pin: 14 channels
- Single mode and continuous conversion mode
- Maximum 8 sequential conversion support
- Software trigger support
- Two internal trigger source (TIMER1n and TIMER30) Support
- Adjustable sample and hold time
- Maximum 1MHz conversion rate (Max. 1Msps)
- ADC Main Clock Frequency: Max. 16MHz
- Operating Power Supply: AVDD : 2.1V ~ 5.5V
- Analog Input Range: 0.0V ~ AVDD

Table 87 introduces pins assigned for ADC.

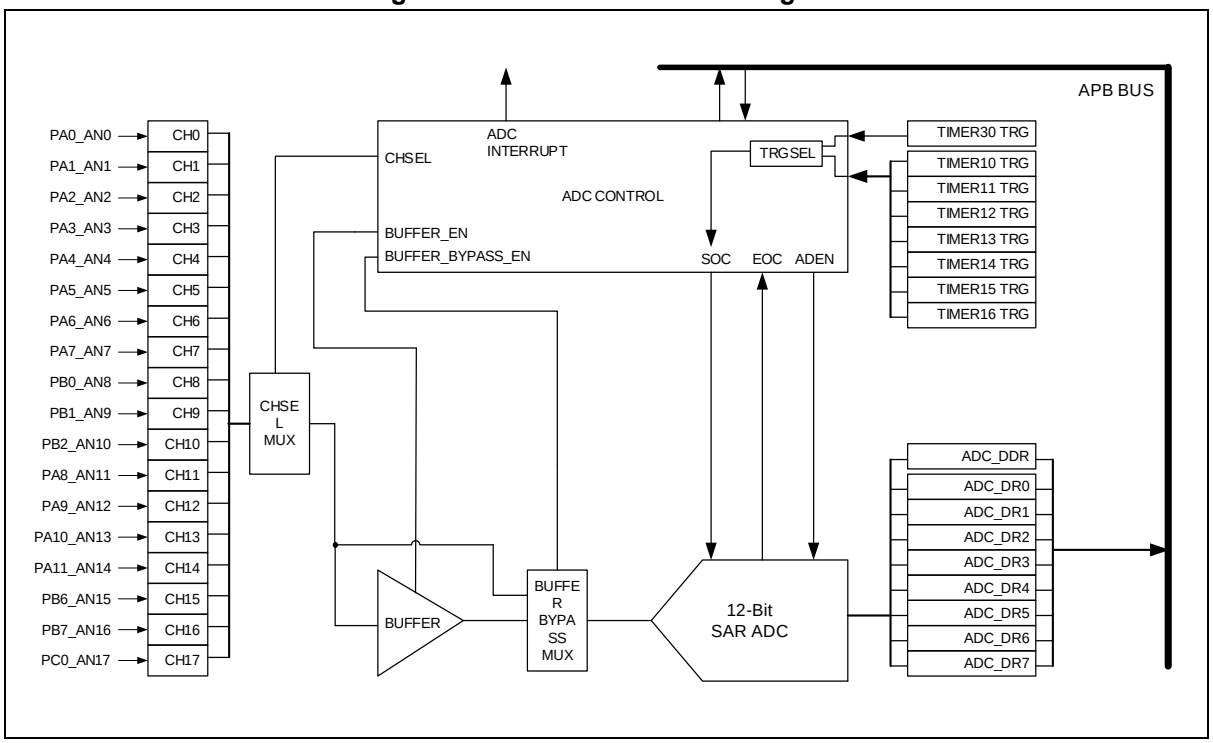
Table 87. Pin Assignment of ADC: External Signal

Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
VDD	P	Analog Power (Reference Voltage)	O	O	O
VSS	P	Analog GND	O	O	O
AN0	A	ADC Input 0	O	O	O
AN1	A	ADC Input 1	O	O	O
AN2	A	ADC Input 2	O	O	O
AN3	A	ADC Input 3	O	O	O
AN4	A	ADC Input 4	O	O	O
AN5	A	ADC Input 5	O	O	O
AN6	A	ADC Input 6	O	O	O
AN7	A	ADC Input 7	O	O	O
AN8	A	ADC Input 8	O	O	O
AN9	A	ADC Input 9	O	O	O
AN10	A	ADC Input 10	O	O	O
AN11	A	ADC Input 11	O	O	Not Used
AN12	A	ADC Input 12	O	O	Not Used
AN13	A	ADC Input 13	O	O	Not Used
AN14	A	ADC Input 14	O	O	Not Used
AN15	A	ADC Input 15	O	O	O
AN16	A	ADC Input 16	O	O	O
AN17	A	ADC Input 17	O	O	O

19.1 12-bit ADC block diagram

In this section, 12-bit ADC is described in a block diagram in Figure 127.

Figure 127. 12-bit ADC Block Diagram



19.2 Registers

Base address of ADC unit is introduced in the followings:

Table 88. Base Address of 12-bit ADC

Name	Base address
ADC	0x4000_3100

Table 89. 12-bit ADC Register Map

Name	Offset	Type	Description	Reset value	Ref.
ADC_MR	0x0000	RW	ADC Mode Register	0x0000_0000	19.2.1
ADC_CSCR	0x0004	RW	Current Sequence/Channel Register	0x0000_0000	19.2.2
ADC_CCR	0x0008	RW	ADC Clock Control Register	0x0000_8080	19.2.3
ADC_TRG	0x000C	RW	ADC Trigger Selection Register	0x0000_0000	19.2.4
ADC_SCSR1	0x0018	RW	ADC Channel Selection 1 Register	0x0000_0000	19.2.5
ADC_SCSR2	0x001C	RW	ADC Channel Selection 2 Register	0x0000_0000	19.2.6
ADC_CR	0x0020	RW	ADC Control Register	0x0000_0000	19.2.7
ADC_SR	0x0024	RC	ADC State Register	0x0000_0000	19.2.8
ADC_IER	0x0028	RW	ADC Interrupt Enable Register	0x0000_0000	19.2.9
ADC_DDR	0x002C	RO	ADC DMA Data Register	0x0000_0000	19.2.10
ADC_DR0	0x0030	RO	ADC Sequence 0 Data Register	0x0000_0000	19.2.11
ADC_DR1	0x0034	RO	ADC Sequence 1 Data Register	0x0000_0000	
ADC_DR2	0x0038	RO	ADC Sequence 2 Data Register	0x0000_0000	
ADC_DR3	0x003C	RO	ADC Sequence 3 Data Register	0x0000_0000	
ADC_DR4	0x0040	RO	ADC Sequence 4 Data Register	0x0000_0000	
ADC_DR5	0x0044	RO	ADC Sequence 5 Data Register	0x0000_0000	
ADC_DR6	0x0048	RO	ADC Sequence 6 Data Register	0x0000_0000	
ADC_DR7	0x004C	RO	ADC Sequence 7 Data Register	0x0000_0000	
ADC_CMPR	0x0070	RW	ADC Channel Comparator Register	0x0000_0000	19.2.12
ADC_BCR	0x0074	RW	ADC Buffer Control Register	0x0000_0001	19.2.13

19.2.1 ADC_MR: ADC mode register

ADC_MR is a mode setting register for the ADC Module. This register must be set first for the intended use of the ADC Module.

ADC_MR=0x4000_3100

31 30 29 28 27 26 25 24	23 22	21	20	19 18	17 16	15 14 13 12	11	10 9 8	7	6	5 4	3 2	1 0
Reserved		TRGINFO	CHINFO	Reserved	DMAEN	STSEL	Reserved	SEQCNT	ADEN	ARST	ADMOD	Reserved	TRGSEL
-		0	0	-	0	0x00	-	000	0	0	00	-	00
-		RW	RW	-	RW	RW	-	RW	RW	RW	RW	-	RW

Bits	Name	Function			
21	TRGINFO	Trigger information option (In external trigger mode)			
		0	Option disable		
		1	Trigger source information will be stored in ADCDR[31:24]		
20	CHINFO	Channel information option			
		0	Option disable		
		1	Converted channel information will be stored in ADCDR[20:16]		
17	DMAEN	DMA enable/disable bit (Must be set When ADEN =1)			
		0	Disable DMA		
		1	Enable DMA When the DMA function is enabled and the ADC receives a completion signal from the DMAC, a DMA request is generated at the interrupt request is generated or conversion is finished. (Include Burst Mode)		
16 12	STSEL	Sampling Time Selection			
		It is a value to set the time window section that can recognize the Next Trigger. The applied point is applied immediately after Trigger. ADC sampling time is (2+STSEL [4:0]) MCLK cycle. Minimum Sampling Time is 2 MCLK. When STSEL[4:0] = b'11111, Sampling channel always enable.			
10 8	SEQCNT	Number of conversion in a sequence If ADMOD[5:4] is 0 and SEQCNT[10:8] isn't 0, CSEQN increase to SEQCNT by trigger event. (SEQCNT apply only in Single/Sequential Mode)			
		000	Single Mode	100	5 Sequence ADC
		001	2 Sequence ADC	101	6 Sequence ADC
		010	3 Sequence ADC	110	7 Sequence ADC
		011	4 Sequence ADC	111	8 Sequence ADC
7	ADEN	ADC Enable bit			
		0	Disable		
		1	Enable		
6	ARST	After sequence finish, restart bit.			

		0	Stop after finish (ASTART must be set to 1 to restart)
		1	Restart after finish
5 4	ADMOD	ADC Mode Selection bit.	
		00	Single/Sequential Conversion Mode
		01	Burst Conversion Mode
		10	Multiple Conversion Mode
		11	No effect
1 0	TRGSEL	Trigger Selection bit.	
		00	Disables event triggers and enables the soft trigger only.
		01	Enables the TIMER1n event trigger
		10	Enables the TIMER30 event trigger
		11	Reserved

19.2.2 ADC_CSCR: ADC current sequence/channel register

ADC_CSCR consists of the Current Sequence Number and Current Active Channel value. CSEQN (Current Sequence Number) can set the Current Sequence Number immediately. The ADEN bit of the ADC_MR Register must be set before this register is set.

ADC_CSCR=0x4000_3104

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		CSEQN	Reserved	CACH
-		000	-	00000
-		RW	-	RO

Bits	Name	Function	
10 8	CSEQN	Current Sequence Number	
		000	Current Sequence is 0
		001	Current Sequence is 1
		010	Current Sequence is 2
		011	Current Sequence is 3
		100	Current Sequence is 4
		101	Current Sequence is 5
		110	Current Sequence is 6
		111	Current Sequence is 7
4 0	CACH	Current Active Channel	
		00000	ADC channel 0 is active
		00001	ADC channel 1 is active
		00010	ADC channel 2 is active
		00011	ADC channel 3 is active
		00100	ADC channel 4 is active
		00101	ADC channel 5 is active
		00110	ADC channel 6 is active
		00111	ADC channel 7 is active
		01000	ADC channel 8 is active
		01001	ADC channel 9 is active
		01010	ADC channel 10 is active
		01011	ADC channel 11 is active
		01100	ADC channel 12 is active
		01101	ADC channel 13 is active
		01110	ADC channel 14 is active
		01111	ADC channel 15 is active
		10000	ADC channel 16 is active

	10001	ADC channel 17 is active
	Others	Reserved

19.2.3 ADC_CCR: ADC clock control register

ADC_CCR is a clock control register of ADC Module.

ADC_CCR=0x4000_3108

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14 13 12 11 10 9 8	7	6	5	4 3 2 1 0
Reserved		ADCPDA	CLKDIV	ADCPD	EXTCLK	CLKINVT	Reserved
-		1	0000000	1	0	0	-
-		RW	RW	RW	RW	RW	-

Bits	Name	Function	
15	ADCPDA	ADC disablement for power saving	
		0	ADC normal mode
		1	ADC low power mode @ low speed operation
14 8	CLKDIV	ADC clock division value bit(When EXTCLK is 0, CLKDIV Enable) - CLKDIV = 0 → ADC Clock = ADC Input Clock (Bypass) - CLKDIV = 1 → ADC Clock = Clock Stop - CLKDIV ≥ 2 → ADC Clock = ADC Input Clock / CLKDIV - CLKDIV ≤ MCLK x 1/3 **NOTES:** 1. When ADC clock is divided by CLKDIV value is set to 2 or more. 2. ADC clock (ACLK) must be set not to exceed 1/3 of main system clock (MCLK) frequency in Sequential or Burst mode. For example, when using the Sequential Mode of the ADC in a system that uses 48MHz as the main system frequency, the ADC clock frequency is up to 16MHz.	
7	ADCPD	ADC DEEP-SLEEP	
		0	ADC Normal Mode
		1	ADC DEEP-SLEEP (Power-down) Mode
6	EXTCLK	ADC external clock configuration **NOTE:** In Sequential or Burst conversion mode the EXTCLK must be set to 0.	
		0	Internal clock (CLKDIV enablement)
		1	External clock (MCCR clock)
5	CLKINVT	Divide Clock Inversion (Option bit)	
		0	Duty ratio of divided clock is larger than 50%
		1	Duty ratio of divided clock is less than 50%

19.2.4 ADC_TRG: ADC trigger selection register

This Register is ADC Trigger Selection Register of ADC Module.

ADC_TRG=0x4000_310C

31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
SEQTRG7	SEQTRG6	SEQTRG5	SEQTRG4	SEQTRG3	SEQTRG2	SEQTRG1	SEQTRG0 BSTTRG
0000	0000	0000	0000	0000	0000	0000	0000
RW	RW	RW	RW	RW	RW	RW	RW

Bits	Name	Function
30 28	SEQTRG7	8th Sequence Trigger Source
27 24	SEQTRG6	7th Sequence Trigger Source
23 20	SEQTRG5	6th Sequence Trigger Source
19 16	SEQTRG4	5th Sequence Trigger Source
15 12	SEQTRG3	4th Sequence Trigger Source
11 8	SEQTRG2	3th Sequence Trigger Source
7 4	SEQTRG1	2ed Sequence Trigger Source
3 0	SEQTRG0 BSTTRG	1st Sequence Trigger Source Burst Conversion Trigger Source

NOTE: In multiple mode, the 1st sequence is given the highest priority, and the 15th sequence is given the lowest priority. Table 90 shows trigger sources represented by each value.

Table 90. ADC Trigger Source Table

Value	TIMER1n (n = 10 to 16) (TRGSEL[1:0] = 0x1)	TIMER30 (TRGSEL[1:0] = 0x2)
0	TIMER10	TIMER30
1	TIMER11	-
2	TIMER12	-
3	TIMER13	-
4	TIMER14	-
5	TIMER15	-
6	TIMER16	-
…	-	-
15	ASTART	ASTART

NOTES:

1. ASTART is a S/W trigger in the ADC_CR register.
2. In order to use TIMER30 as a trigger, TIMER30_ADTCR and TIMER30_ADTDR must be set.

19.2.5 ADC_SCSR1: ADC channel selection 1 register

ADC_SCSR1 is a 32-bit register. Each selected channel is associated with a trigger selection register. When ADEN of ADC_MR Register set, Can write in ADC_SCSR1.

ADC_SCSR1=0x4000_3118

31 30 29 28 27 26 25 24		23 22 21 20 19 18 17 16		15 14 13 12 11 10 9 8		7 6 5 4 3 2 1 0	
Reserved	SEQ3CH	Reserved	SEQ2CH	Reserved	SEQ1CH	Reserved	SEQ0CH
-	0x0	-	0x0	-	0x0	-	0x0
-	RW	-	RW	-	RW	-	RW

Bits	Name	Function
28 24	SEQ3CH	4th Conversion Sequence Channel Selection
20 16	SEQ2CH	3th Conversion Sequence Channel Selection
12 8	SEQ1CH	2th Conversion Sequence Channel Selection
4 0	SEQ0CH	1th Conversion Sequence Channel Selection

19.2.6 ADC_SCSR2: ADC channel selection 2 register

ADC_SCSR2 is a 32-bit register. Each selected channel is associated with a trigger selection register. When ADEN of ADC_MR Register set, Can write in ADC_SCSR2.

ADC_SCSR2=0x4000_311C

31 30 29 28 27 26 25 24		23 22 21 20 19 18 17 16		15 14 13 12 11 10 9 8		7 6 5 4 3 2 1 0	
Reserved	SEQ7CH	Reserved	SEQ6CH	Reserved	SEQ5CH	Reserved	SEQ4CH
-	0x0	-	0x0	-	0x0	-	0x0
-	RW	-	RW	-	RW	-	RW

Bits	Name	Function
28 24	SEQ7CH	8th Conversion Sequence Channel Selection
20 16	SEQ6CH	7th Conversion Sequence Channel Selection
12 8	SEQ5CH	6th Conversion Sequence Channel Selection
4 0	SEQ4CH	5th Conversion Sequence Channel Selection

19.2.7 ADC_CR: ADC control register

ADC_CR is a register for ADC Control.

ADC_CR=0x4000_3120

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6 5 4 3 2	1	0
Reserved			ASTOP	Reserved	TRGCLR	ASTART
-			0	-	0	0
-			WO	-	RW	RW

Bits	Name	Function	
7	ASTOP	ADC STOP bit	
		0	No Effect
		1	ADC conversion stop (will be clear next @ADC clock) If ASTOP set after conversion cycle start, present conversion would be completed.
1	TRGCLR	ADC all trigger flags cleared option	
		0	No clear
		1	Clear all trigger flags of previous ADC operation
0	ASTART	ADC START bit	
		0	No ADC Conversion
		1	ADC Conversion Start (will be clear next @ADC clock) ADEN should be "1" to start ADC. If ASTART is set as '1' when ARST is '0' in trigger event mode, ADC conversion will start once as SEQCNT set.

19.2.8 ADC_SR: ADC state register

ADC_SR is a register for indicating the ADC state.

ADC_SR=0x4000_3124

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6	5	4	3	2	1	0
Reserved			CMPIFLG	Reserved	DOVRUN	DMAF	TRGIF	EOSIF	Reserved	EOCIF
-			0	-	0	0	0	0	-	0
-			RC	-	RO	RO	RC	RC	-	RC

Bits	Name	Function	
8	CMPIFLG	Compare Interrupt Flag bit	
		0	No Interrupt occurred
		1	Interrupt occurred (Write '1' to Clear Flag)
5	DOVRUN	DMA Overrun Flag (Not Interrupt)	
		0	No Flag occurred
		1	Flag occurred
4	DMAF	DMA Done Received Flag (DMA transfer is completed)	
		0	No Flag occurred
		1	Flag occurred
3	TRGIF	ADC Trigger Interrupt Flag	
		0	No Flag occurred
		1	Flag occurred (Write '1' to Clear Flag)
2	EOSIF	Sequence End Interrupt Flag	
		0	No Flag occurred
		1	Flag occurred (Write '1' to Clear Flag)
0	EOCIF	Sequence Conversion End Interrupt Flag	
		0	No Flag occurred
		1	Flag occurred (Write '1' to Clear Flag)

NOTE: Flag check of polling method uses EOCIF bit.

Example code)

```
while ((ADC_SR & 0x01) == 1)    // EOCIF Flag Checking
{
ADCSR = 0x1;                    // EOCIF Flag Clear
}
```

19.2.9 ADC_IER: ADC interrupt enable register

ADC_IER is a register to enable ADC Interrupt.

ADC_IER=0x4000_3128

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5	4	3	2	1	0
Reserved				DMAIE	TRGIE	EOSIE	Reserved	EOCIE
-				0	0	0	-	0
-				RW	RW	RW	-	RW

Bits	Name	Function	
4	DMAIE	DMA Done Interrupt Enable	
		0	Disable
		1	Enable
3	TRGIE	ADC Trigger Conversion Interrupt Enable	
		0	Disable
		1	Enable
2	EOSIE	ADC Sequence Conversion Interrupt Enable	
		0	Disable
		1	Enable
0	EOCIE	ADC Single Conversion Interrupt Enable	
		0	Disable
		1	Enable

19.2.10 ADC_DDR: ADC DMA data register

ADC_DDR is a register of ADC DMA Conversion result.

ADC_DDR=0x4000_312C

31	30	29	28	27	26	25	24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
TRGINFO7	TRGINFO6	TRGINFO5	TRGINFO4	TRGINFO3	TRGINFO2	TRGINFO1	TRGINFO0	Reserved	ADMACH	ADC DMA Temporary Data	Reserved
0	0	0	0	0	0	0	0	-	0x00	0x0000	-
RO	RO	RO	RO	RO	RO	RO	RO	-	RO	RO	-

Bits	Name	Function
31 24	TRGINFOx (x=0~7)	ADC Trigger Information (ADC Trigger Information is Trigger source captured at EOC time.) *Must be enable TRGINFO function of ADC_MR Register. Multiple Mode) The lower the x of TRGINFOx, the higher the priority. Multiple TRGINFO bits can be read as 1 if Pending by another trigger. Single/Sequential Mode) It can reference the source Pending by another trigger. The currently processed trigger source must refer to CSEQN of CSCR register.
20 16	ADMACH	DMA ADC Channel Indicator *Enable the CHINFO function of ADC_MR register.
15 4	ADDMAR	DMA ADC Conversion Result Data (12-bit)

NOTE: Even when the DMA function is not used, data is temporarily stored in this buffer before the data is stored in the corresponding buffer, and channel information of the data can be also known.

19.2.11 ADC_DRx: ADC sequence x data register (x = 0 to 7)

ADC_DR indicates the result of the ADC Conversion. There are eight of these registers.

ADC_DR0=0x4000_3130, ADC_DR1=0x4000_3134, ADC_DR2=0x4000_3138, ADC_DR3=0x4000_313C
ADC_DR4=0x4000_3140, ADC_DR5=0x4000_3144, ADC_DR6=0x4000_3148, ADC_DR7=0x4000_314C

31	30	29	28	27	26	25	24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
TRGINFO7	TRGINFO6	TRGINFO5	TRGINFO4	TRGINFO3	TRGINFO2	TRGINFO1	TRGINFO0	Reserved	ACH	ADDATA	Reserved
0	0	0	0	0	0	0	0	-	0x00	0x000	-
RO	RO	RO	RO	RO	RO	RO	RO		RO	RO	-

Bits	Name	Function
31 24	TRGINFOx (x=0~7)	ADC Trigger Information (ADC Trigger Information is Trigger source captured at EOC time.) *Must be enable TRGINFO function of ADC_MR Register. Multiple Mode) The lower the x of TRGINFOx, the higher the priority. Multiple TRGINFO bits can be read as 1 if Pending by another trigger. Single/Sequential Mode) It can reference the source Pending by another trigger. The currently processed trigger source must refer to CSEQN of CSCR register.
20 16	ACH	ADC Channel Information * Enable the CHINFO function of ADC_MR register.
15 4	ADDATA	ADC Input data

19.2.12 ADC_CMPR: ADC channel compare register

ADC_CMPR is an ADC channel compare control register.

ADC_CMPR=0x4000_3170

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0

Reserved	CMPIEN	CMPEN	Reserved	LTE	CCH	CVAL	Reserved
-	0	0	-	0	00000	0x000	-
-	RW	RW	-	RW	RW	RW	-

Bits	Name	Function	
24	CMPIEN	Compare Interrupt Enable bit	
		0	Disable
		1	Enable
23	CMPEN	Compare Operation Enable bit	
		0	Disable
		1	Enable
21	LTE	AD Conversion Value Output Timing Setting	
		0	When ADC > CVAL, Output
		1	When ADC ≤ CVAL, Output
20 16	CCH	Compare Channel	
		00000	Compare Channel is ADC channel 0
		00001	Compare Channel is ADC channel 1
		00010	Compare Channel is ADC channel 2
		00011	Compare Channel is ADC channel 3
		00100	Compare Channel is ADC channel 4
		00101	Compare Channel is ADC channel 5
		00110	Compare Channel is ADC channel 6
		00111	Compare Channel is ADC channel 7
		01000	Compare Channel is ADC channel 8
		01001	Compare Channel is ADC channel 9
		01010	Compare Channel is ADC channel 10
		01011	Compare Channel is ADC channel 11
		01100	Compare Channel is ADC channel 12
		01101	Compare Channel is ADC channel 13
		01110	Compare Channel is ADC channel 14
		01111	Compare Channel is ADC channel 15
		10000	Compare Channel is ADC channel 16
		10001	Compare channel is ADC channel 17.
		Others	Reserved

15 4	CVAL	Compare Value (12-bit)

19.2.13 ADC_BCR: ADC buffer control register

ADC_BCR is an ADC buffer control register. When ADC buffer is enable, ADC conversion time will increase by being added ADC buffering timing clock.

ADC_BCR=0x4000_3174

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3 2	1	0
Reserved			TBUFSEL	Reserved	BUFEN	BYPSEL
-			000	-	0	1
-			RW	-	RW	RW

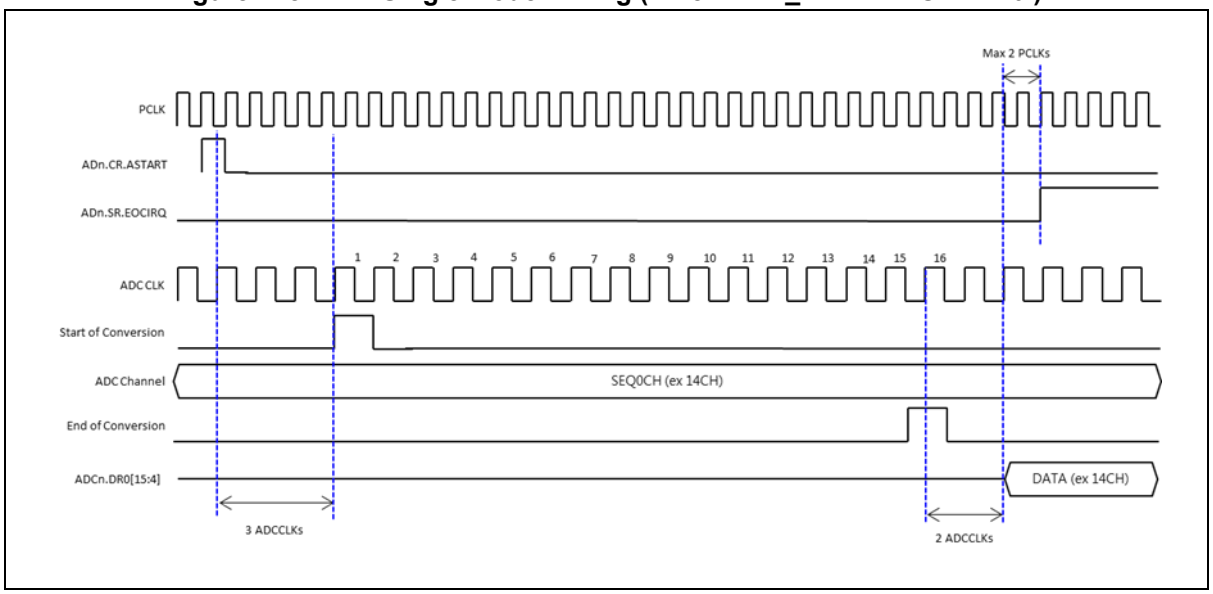
Bits	Name	Function	
6 4	TBUFSEL	Buffering off time selection bit	
		000	1*MCLK
		001	2*MCLK
		010	3*MCLK
		011	4*MCLK
		100	5*MCLK
		101	6*MCLK
		110	7*MCLK
		111	8*MCLK
1	BUFEN	ADC Input Buffer Enable bit	
		0	Input buffer power down period
		1	Input buffer operation period
0	BYPSEL	ADC Input Buffer Bypass Selection bit	
		0	Input buffer operation mode
		1	Input buffer bypass mode

19.3 Functional description

19.3.1 ADC single mode timing diagram

When ADC_MR<ADMOD> is 0x0 and ADC_MR<SEQCNT> is 0x0, The ADC conversion starts when the CR_ASTART bit is set to '1'. When ADC_CR<ASTART> is set, SOC (start of conversion) is active at 3 ADC clocks and ADC_SR<EOCIRQ> is set at 2 ADC clocks and 2 PCLKs after the end of conversion.

Figure 128. ADC Single Mode Timing (When ADC_MR<ADMOD> = '0')



19.3.2 ADC burst mode timing diagram

There are two source for creating SOC in Burst Mode. First, TRG event (timer and MPWM) and ASTART. If TRGSEL is set to Timer Event Trigger or MPWM Event Trigger, the SOC will be the trigger of ADC_TRG<BSTTRG>. For example, if ADC_TRG<BSTTRG> is set to TIMER3, ADC Conversion is started by the trigger of TIMER3. When a trigger event of BSTTRG occurs, the ADC will change the ADC channel to the value set by ADC_MR<SEQCNT>.

Figure 129. ADC Burst Mode Timing (When ADC_MR<ADMOD> = '1')

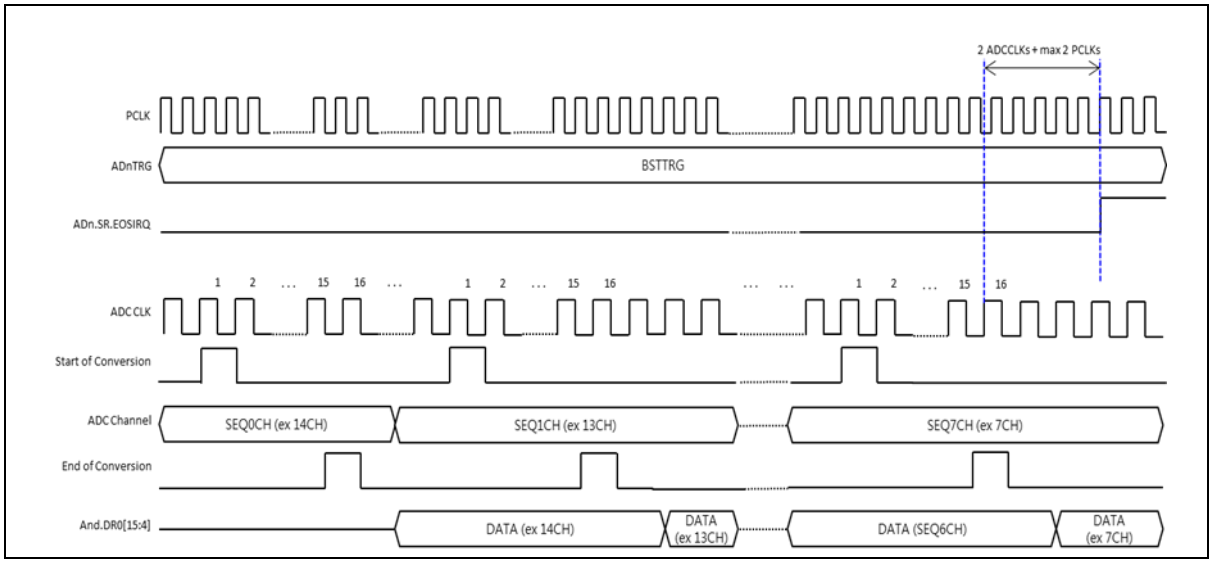
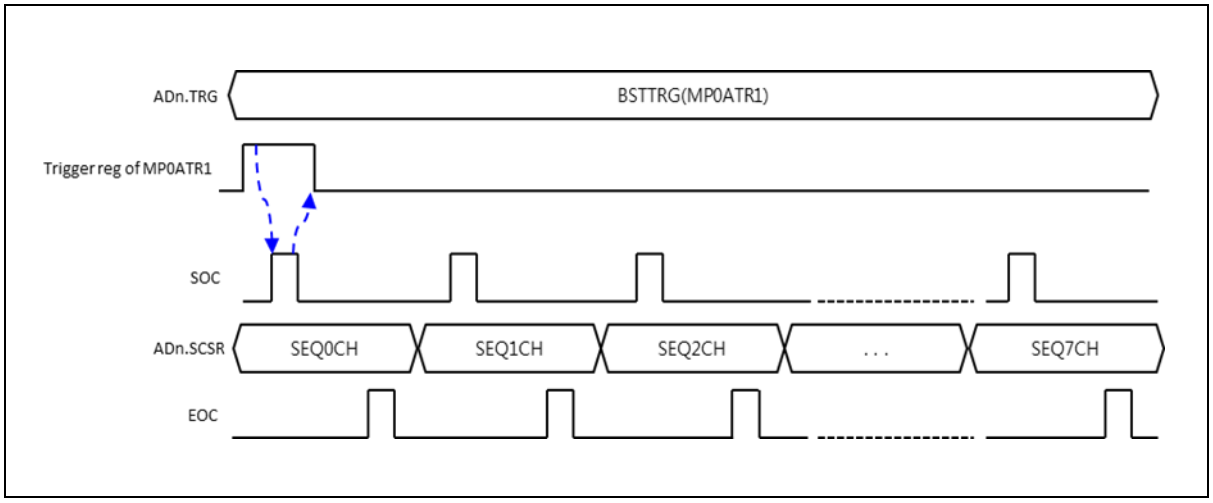


Figure 130. ADC Trigger Timing in Burst Mode (SEQCNT = 3'b111, 8 Sequential Conversion)



19.3.3 ADC sequential mode timing diagram

To set sequential mode, ADC_MR<ADMOD> is 2'b00 and ADC_MR<SEQCNT> must not be 3'b000. The operation of the sequential mode is the almost same as burst mode. Difference is the source of the SOC. Each SOC is created by a trigger of SEQTRGx with each SEQCNT.

Figure 131. ADC Sequential Mode Timing (When MR.AMOD = '0' and MR.SEQCNT ≠ '0')

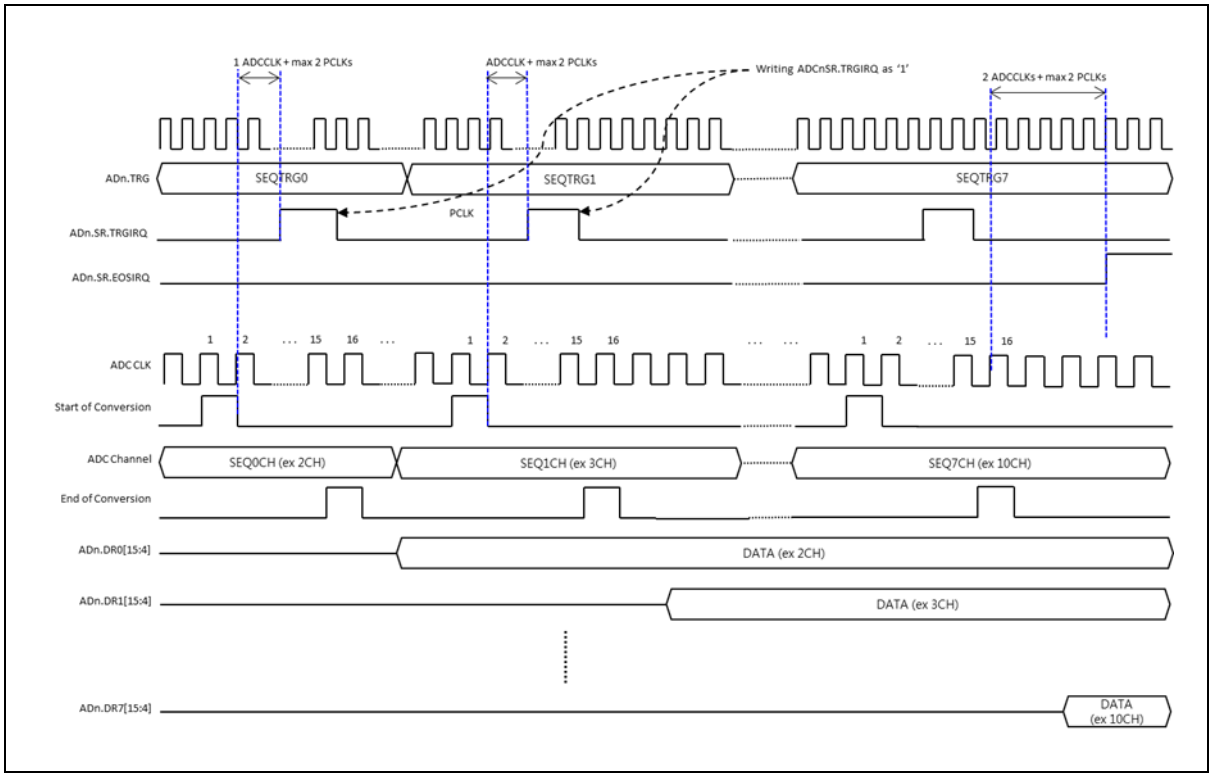
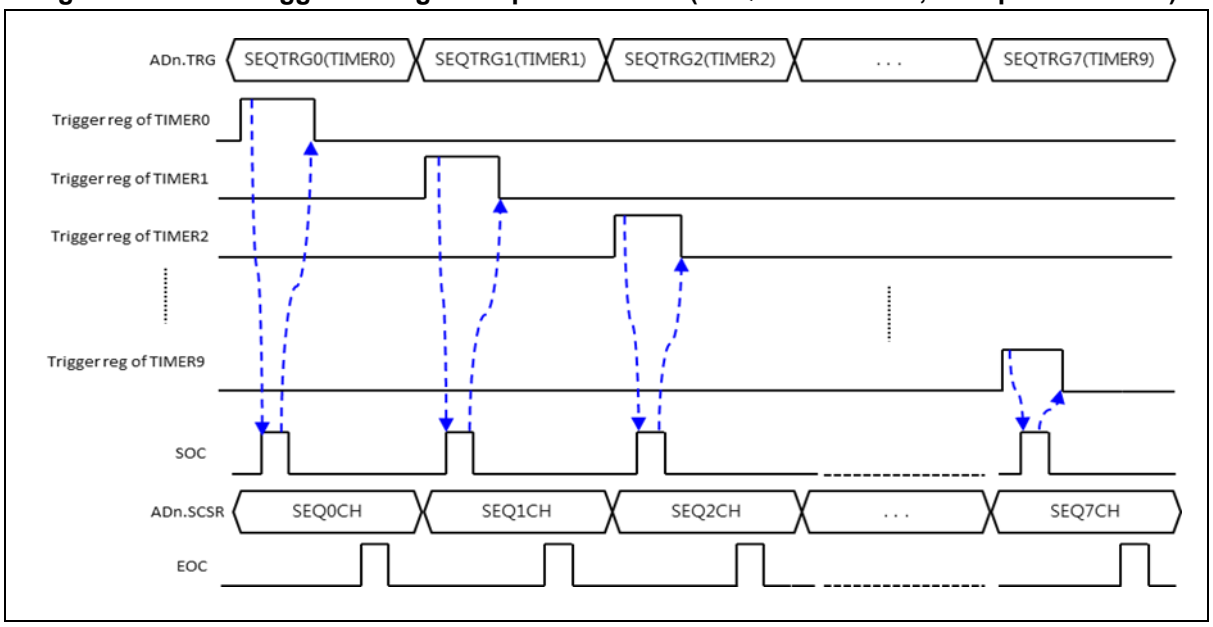


Figure 132. ADC Trigger Timing in Sequential Mode (SEQCNT = 3'b111, 8 Seq.-Conversion)



19.3.4 ADC multiple mode timing diagram

In multiple mode operation, if the desired trigger source is set to ADC_TRG<SEQTRG>, conversion begins when the corresponding trigger occurs regardless of the order of SEQTRG. For example, if ADC_MR<SEQCNT> is 3'b011, if one of the four trigger sources (TIMER, MPWM) is selected and ADC_CR<ASTART> is set, conversion will be processed by the first occurred trigger source, regardless of sequence. It is different from sequential mode and burst mode.

Figure 133. ADC Multiple Mode Timing (ADC_MR<ADMOD> = '2' and ADC_MR<SEQCNT> ≠ '0')

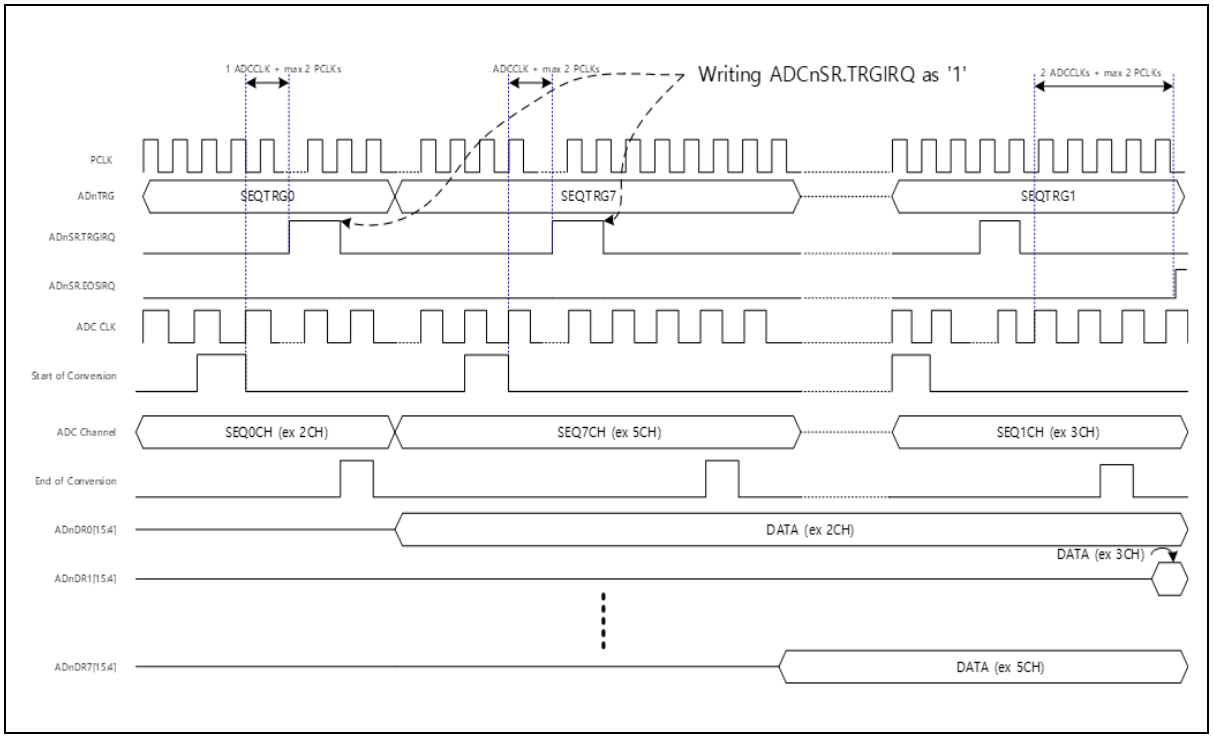
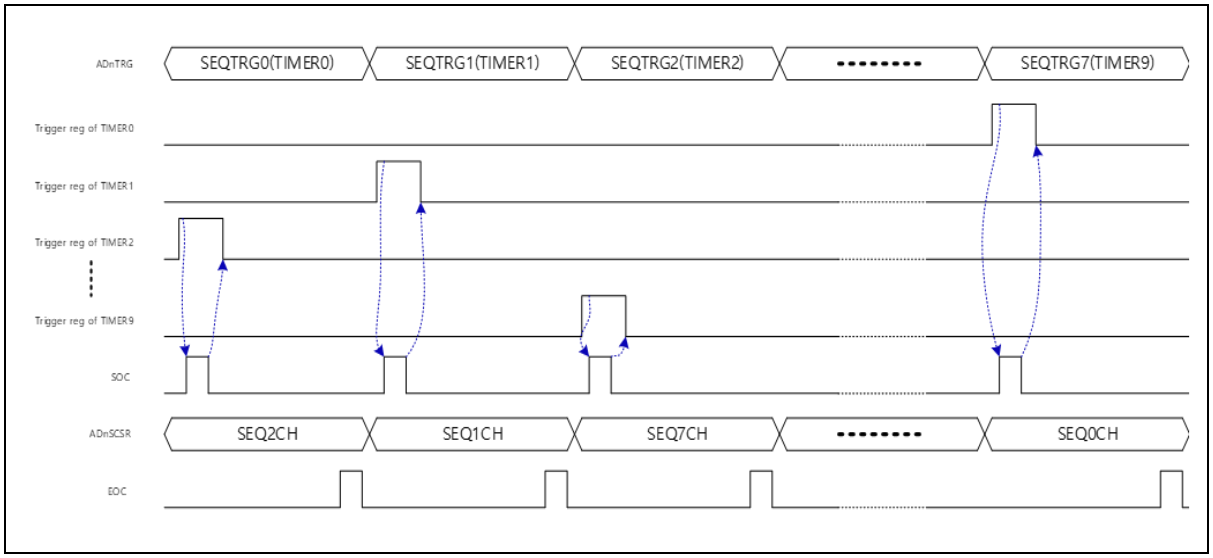


Figure 134. ADC Trigger Timing in Multiple Mode (SEQCNT = 3'b111, 8 Multi.-Conversion)



20 12-bit DAC

Digital-to-analog (D/A) converter uses successive approximation logic to convert 12-bit digital value to an analog output level.

DAC module has six registers which are the DAC control register (DACCR), DAC data high register (DACDRH), DAC data low register (DACDRL), DAC buffer high register (DACBRH), DAC buffer low register (DACBRL) and programmable gain selection register (PGSR).

Table 91 introduces pins assigned for DAC.

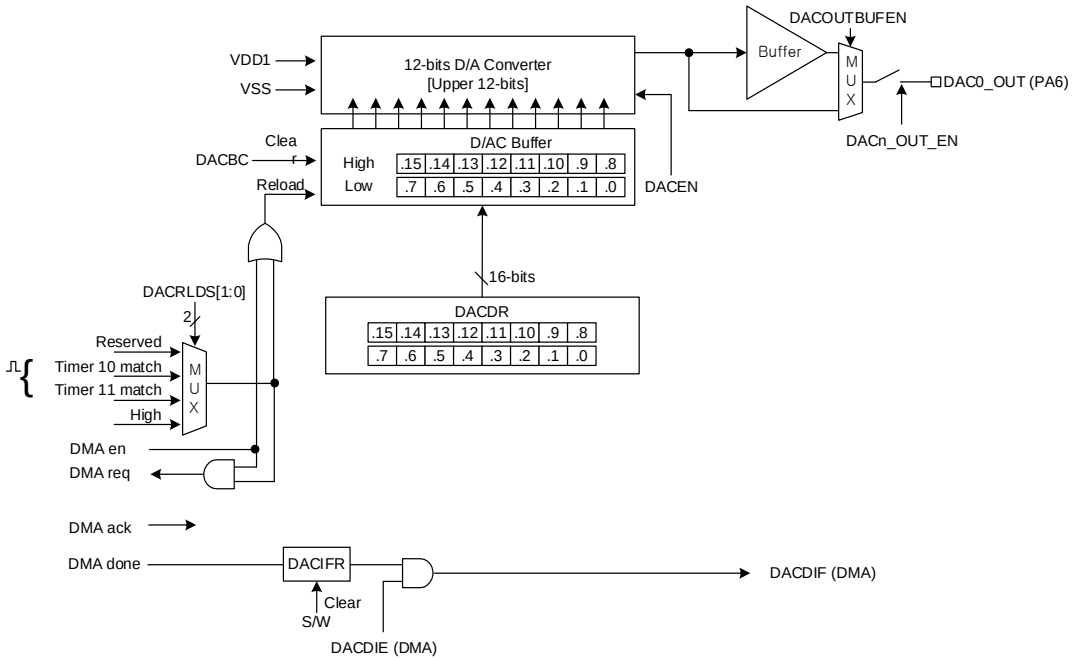
Table 91. Pin Assignment of DAC: External Signal

Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
DAO	A	DAC output (Referenced VDD)	O	O	O

20.1 12-bit DAC block diagram and analog power pin

In Figure 135, 12-bit DAC is described in a block diagram.

Figure 135. 12-bit DAC Block Diagram



NOTES:

1. Gain controller converts data in the sequence listed below:
   A. Get the DAC_DR[15:0] value.
   B. Adjust the data by selection gain.
   C. Write the data to the D/AC buffer.
2. DAC interface block doesn't work when DACRLDS bits of the DAC_CR register are set to "00b".

20.2 Registers

Base address of DAC unit is introduced in the followings:

Table 92. Base Address of DAC

Name	Base address
DAC	0x4000_3450

Table 93. 12-bit DAC Register Map

Name	Offset	Type	Description	Reset value	Ref.
DAC_DR	0x0000	RW	DAC data register	0x0000_0000	20.2.1
DAC_BR	0x0004	RO	DAC buffer register	0x0000_0000	20.2.2
DAC_CR	0x0008	RW	DAC control register	0x0000_0000	20.2.3
DAC_ICR	0x0014	RW	DAC Interrupt Control Register	0x0000_0000	20.2.4

20.2.1 DAC_DR: DAC data register

DAC_DR register converts input digital value into an analog signal output. DAC output value range can be input from GND to VDD based on LSB unit divided by reference voltage by 12-bit resolution.

DAC_DR=0x4000_3450

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		DACDR	
-		0x0000	
-		RW	

Bits	Name	Function
15 0	DACDR	DAC converted data (16-bit) The DACDR[15:0] is a binary format.

20.2.2 DAC_BR: DAC buffer register

DAC_BR is a read only register which is used for a DAC buffer. The DAC_BR register outputs upper 12bits of DAC_DR register as a DAC value, or Gain of DAC_PGSR and Offset of DAC_OFSCR in 16-bit.

To initialize this register, it needs to be cleared by setting DACBC bit of DAC_CR register to '1'. If DAC_DR register is reloaded while clearing this register, the updates will be read.

DAC_BR=0x4000_3454

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		DACBR	
-		0000	
-		RO	

Bits	Name	Function
15 0	DACBR	DAC Buffer Data (16-bit) The DACBR[15:0] is a binary format.

20.2.3 DAC_CR: DAC control register

DAC_CR register can configure the DAC operation, output channels and DAC update timing.

DAC_CR=0x4000_3458

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11	10	9 8	7 6 5	4	3	2 1	0
Reserved			DAC_OUT_E	DAC_BUF_E	Reserved	DACBC	Reserved	DACRLDS	DACEN
-			0	0	-	0	-	0	0
-			RW	RW	-	RW	-	RW	RW

Bits	Name	Function	
11	DAC_OUT_EN	Select DAC_OUT channel for DAO(PA6) pin output DAC_OUT is connected DAO pin in analog output mode. To use DAO pin to output DAC signal, PA6 port must be set to alternative function mode (AF3).	
		0	Disable
		1	Enable
10	DAC_BUF_EN	DAC Buffer Selection: If this bit is set to '1', it is allowed to add loads to the DAC output. When both conditions (to enable this bit and add loads to the DAC output) are met, output current will increase.	
		0	Disable (buffer bypass)
		1	Enable
4	DACBC	DAC Buffer Clear	
		0	No effect.
		1	Clear the D/AC buffer (When write, automatically cleared to '0' after being cleared)
2 1	DACRLDS	DAC Reload Selection. These bits select a reload signal to load data from D/AC data register to buffer.	
		00	Always
		01	Reserved
		10	Timer 10 match signal
		11	Timer 11 match signal
0	DACEN	DAC Enable Bit	
		0	Stop D/AC operation (Low level output)
		1	Start D/AC operation

20.2.4 DAC_ICR: DAC interrupt control register

DAC_ICR register controls DAC interrupts. To make available the interrupts, DMA register needs to be set.

According to the activation of each interrupt, the operation of corresponding flag can be checked. DMC DMA receive done interrupt is generated when DAC data receive is finished, and the corresponding flag is set to '1' to clear. DAC DMA under-run register generates Under-Run interrupt if DMA request occurs during DMA operation, and the corresponding flag is set to '1' to clear.

DAC_OFSCR=0x4000_3464

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6	5	4	3 2	1	0
Reserved				DUDRUNF	DMAIF	Reserved	DUDRUNE	DAMIE
-				0	0	-	0	0
-				RW	RW	-	RW	RW

Bits	Name	Function	
5	DUDRUNF	DMA under-run interrupt flag	
		0	No DMA request occurred.
		1	If a DMA request occurs during DMA transfer, DAC underrun error flag is set.
4	DMAIF	DMA done received interrupt flag (DMA transfer is completed)	
		0	Either a DAC DMA is in progress or is not being used.
		1	Completed DAC DMA
1	DUDRUNE	DMA under-run interrupt enable.	
		0	Disable the DAC DMA under-run interrupt
		1	Enable the DAC DMA under-run interrupt
0	DAMIE	DMA done received interrupt enable(DMA transfer is completed)	
		0	Disables the DAC DMA complete interrupt
		1	Enables the DAC DMA complete interrupt

20.3 Functional description

20.3.1 DAC data buffer register

DAC data and buffer registers are 16-bits, respectively. But only the upper 12-bits of the DAC buffer register specifies to generate DAC output signal. The reset value of the data and buffer is "0x0000". The DAC output value, VDAC, is calculated by the following formula:

VDAC = VDD X (n ÷ 4096), (n = 0, 1, 2,,,,, 4095. That is DACBR[15:4] value)

20.3.2 DAC interrupt with DMA

For the use of DAC interrupts, DMA should be enabled and DMA interrupt vector be used.

21 Comparator

Comparator of A31G22x series compares one analogue voltage level with external reference voltage, or internal reference voltage output voltage.

The comparator features the followings:

- 2 Comparators
- Internal BGR reference for comparator
- Comparator output de-bounce function
- Level and edge interrupt mode support for comparator

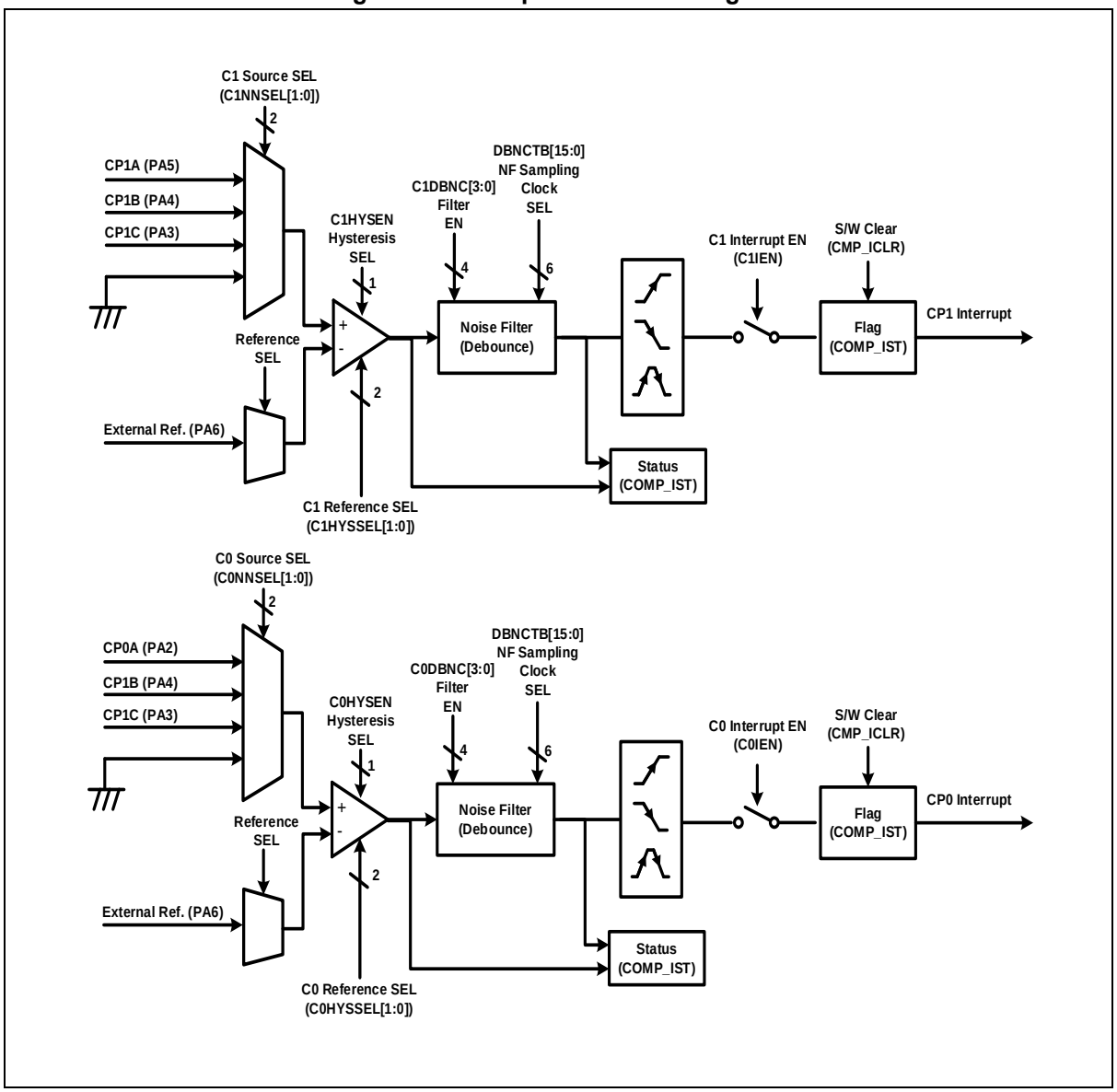
Table 94 introduces pins assigned for Comparator.

Table 94. Pin Assignment of Comparator: External Signal

Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
CP0A	A	Comparator 0 input A	O	O	O
CP1A	A	Comparator 1 input A	O	O	O
CP1B	A	Comparator 0,1 input B	O	O	O
CP1C	A	Comparator 0,1 input C	O	O	O
CREF0	A	Comparator 0 reference input	O	O	O
CREF1	A	Comparator 1 reference input	O	O	O

21.1 Comparator block diagram

Figure 136. Comparator Block Diagram



21.2 Registers

Base address of the comparator is introduced in the followings:

Table 95. Base Address of Comparator

Name	Base address
CMP	0x4000_3420

Table 96. Comparator Register Map

Name	Offset	Type	Description	Reset value	Ref.
CMP_CMP0CR	0x0000	RW	Comparator 0 Control Register	0x0000_0000	21.2.1
CMP_CMP1CR	0x0004	RW	Comparator 1 Control Register	0x0000_0000	21.2.2
CMP_DBNC	0x0010	RW	Comparator Debounce Register	0x0000_0000	21.2.3
CMP_ICON	0x0014	RW	Comparator Interrupt Control Register	0x0000_0000	21.2.4
CMP_IEN	0x0018	RW	Comparator Interrupt Enable Register	0x0000_0000	21.2.5
CMP_IST	0x001C	RO	Comparator Interrupt Status Register	0x0000_0000	21.2.6
CMP_ICLR	0x0020	RW	Comparator Interrupt Clear Register	0x0000_0000	21.2.7

21.2.1 CMP_CMP0CR: comparator 0 control register

CMP_CMP0CR=0x4000_3420

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16							15 14 13 12 11 10 9 8								7 6 5 4 3 2 1 0							
Reserved				C0HYSEN	Reserved		HYSSEL	Reserved				C0EN	Reserved					C0INNSEL		Reserved		C0INPSEL	
-				0	-		0	-				0	-					00		-		00	
-				RW	-		RW	-				RW	-					RW		-		RW	

Bits	Name	Function	
20	C0HYSEN	Comparator 0 Hysteresis Enable	
		0	Disable Hysteresis
		1	Enable Hysteresis
16	C0HYSSEL	Comparator 0 Hysteresis Select	
		0	5mV Hysteresis
		1	20mV Hysteresis
12	C0EN	Comparator 0 Enable bits	
		0	Disable Comparator
		1	Enable Comparator
5 4	C0INNSEL	Comparator 0 Reference (input -) Selection bit	
		00	CREF0 (PA7)
		01	Reserved
		10	Reserved
		11	Reserved
1 0	C0INPSEL	Comparator 0 Source (input +) Selection bit	
		00	CP0A (PA2)
		01	CP1B (PA4)
		10	CP1C (PA3)
		11	GND

21.2.2 CMP_CMP1CR: comparator 1 control register

CMP_CMP1CR=0x4000_3424

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16				15 14 13 12 11 10 9 8			7 6 5 4 3 2 1 0		
Reserved	C1HYSEN	Reserved	HYSSEL	Reserved	C1EN	Reserved		C1INNSEL	Reserved	C1INPSEL
-	0	-	0	-	0	-		00	-	00
-	RW	-	RW	-	RW	-		RW	-	RW

Bits	Name	Function	
20	C1HYSEN	Comparator 1 Hysteresis Enable	
		0	Disable Hysteresis
		1	Enable Hysteresis
16	C1HYSSEL	Comparator 1 Hysteresis Select	
		0	5mV Hysteresis
		1	20mV Hysteresis
12	C1EN	Comparator 1 Enable bits	
		0	Disable Comparator
		1	Enable Comparator
5 4	C1INNSEL	Comparator 1 Reference (input -) Selection bit	
		00	CREF1 (PA6)
		01	Reserved
		10	Reserved
		11	Reserved
1 0	C1INPSEL	Comparator 1 Source (input +) Selection bit	
		00	CP1A (PA5)
		01	CP1B (PA4)
		10	CP1C (PA3)
		11	GND

21.2.3 CMP_DBNC: comparator debounce register

CMP_DBNC=0x4000_3430

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4	3 2 1 0
DBNCTB		Reserved	C1DBNC	C0DBNC
0		-	0	0
RW		-	RW	RW

Bits	Name	Function
31 16	DBNCTB	Debounce time base counter System clock/(DBNCTB *2) becomes shift clock of debounce logic When DBNCTB is 0, system clock would be debounce clock.
7 4	C1DBNC	Debounce shift Selection When it is 0x0, debounce function is disable Shift number of debounce logic is (C1DBNC + 1) when C1DBNC is more than 1
3 0	C0DBNC	Debounce shift Selection When it is 0x0, debounce function is disable Shift number of debounce logic is (C0DBNC + 1) when C0DBNC is more than 1

21.2.4 CMP_ICON: comparator Interrupt control register

CMP_ICON=0x4000_3434

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13	12	11 10	9	8	7 6 5 4	3 2	1 0
Reserved		C1TPOL	C0TPOL	Reserved	C1IPOL	C0IPOL	Reserved	C1IMODE	C0IMODE
-		0	0	-	0	0	-	0	0
-		RW	RW	-	RW	RW	-	RW	RW

Bits	Name	Function	
13	C1TPOL	Comparator Trigger output polarity(to trigger other IP)	
		0	output normal (comparator out high activates trigger)
		1	output inverted (XOR)
12	C0TPOL	Comparator Trigger output polarity(to trigger other IP)	
		0	output normal (comparator out high activates trigger)
		1	output inverted (XOR)
9	C1IPOL	Comparator 1 interrupt polarity(level mode)	
		0	interrupt at comparator out high
		1	interrupt at comparator out low
8	C0IPOL	Comparator 0 interrupt polarity(level mode)	
		0	interrupt at comparator out high
		1	interrupt at comparator out low
3 2	C1IMODE	Comparator 1 Interrupt Flag bit.	
		00	level interrupt (by IPOL1)
		01	rising edge interrupt
		10	falling edge interrupt
		11	both edge interrupt
1 0	C0IMODE	Comparator 0 Interrupt Flag bit.	
		00	level interrupt (by IPOL0)
		01	rising edge interrupt
		10	falling edge interrupt
		11	both edge interrupt

21.2.5 CMP_IEN: comparator interrupt enable register

CMP_IEN register is used to enable comparator interrupts. When CMP_IEN<COUTMON> bit is set to'1', the state bit flags, CMP_IST<C0COUT> and CMP_IST<C1COUT>, are enabled to show outputs of each comparator.

CMP_IEN=0x4000_3438

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7	6 5 4 3 2	1	0
Reserved			C1IEN	Reserved	C1IEN	C0IEN
-			0	-	0	0
-			RW	-	RW	RW

Bits	Name	Function	
7	COUTMON	Enable comparator output monitoring	
		0	Disable
		1	Enable
1	C1IEN	Comparator 1 enable	
		0	Disable
		1	Enable
0	C0IEN	Comparator 0 enable	
		0	Disable
		1	Enable

21.2.6 CMP_IST: comparator interrupt status register

CMP_IST register is used to get comparator interrupt status and supports to monitor the outputs of comparators.

C0COUT and C1COUT bits are able to read the corresponding bit of each comparator only if CMP_IEN<COUTMON> bit is set to '1'.

C0EPR and C0EPF bits are read only bits. When comparator0 generates interrupts on both edges, each bit is set to '1' on the rising edge and the falling edge. If the bit is set to '0', the status of the comparator0 is Don't Care. These bits can be cleared by setting corresponding bit of CMP_ICLR register to '1'.

C0DO and C1DO bits enable each comparator to output de-bounced value when CMP_DBNC<CnDBNC> register is set to have de-bounce value. The corresponding flag is enabled.

CMP_IST=0x4000_343C

31 30 29 28 27 26 25 24	23 22 21 20	19	18	17	16	15 14	13	12	11 10	9	8	7 6 5 4 3 2	1	0
Reserved		C1EPF	C1EPR	C0EPF	C0EPR	Reserved	C1COUT	C0COUT	Reserved	C1DO	C0DO	Reserved	C1IRQ	C0IRQ
-		0	0	0	0	-	0	0	-	0	0	-	0	0
-		RO	RO	RO	RO	-	RO	RO	-	RO	RO	-	RO	RO

Bits	Name	Function	
19	C1EPF	The falling edge polarity status of comparator 1 in the comparison state	
		0	None
		1	Comparator 0 edge polarity is falling
18	C1EPR	The rising edge polarity status of comparator 1 in the comparison state	
		0	None
		1	Comparator 0 edge polarity is rising
17	C0EPF	The falling edge polarity status of comparator 0 in the comparison state	
		0	None
		1	Comparator 0 edge polarity is falling
16	C0EPR	The rising edge polarity status of comparator 0 in the comparison state	
		0	None
		1	Comparator 0 edge polarity is rising
13	C1COUT	The current output status of comparator 1	
		0	Current output value is Low
		1	Current output value is High
12	C0COUT	The current output status of comparator 0	
		0	Current output value is Low
		1	Current output value is High

9	C1DO	The de-bounce output status of comparator 1	
		0	None
		1	De-bounced output
8	C0DO	The de-bounce output status of comparator 0	
		0	None
		1	De-bounced output
1	C1IRQ	Comparator 1 interrupt status	
		0	No comparator Interrupt
		1	Comparator interrupt asserted
0	C0IRQ	Comparator 0 interrupt status	
		0	No comparator interrupt
		1	Comparator interrupt asserted

21.2.7 CMP_ICLR: comparator interrupt clear register

CMP_IST register is used to clear comparator interrupt. Each interrupt status flag is cleared to have '0' by setting corresponding bit to '1'.

CMP_ICLR=0x4000_3440

31 30 29 28 27 26 25 24	23 22 21 20	19	18	17	16	15 14 13 12 11 10 9 8	7 6 5 4 3 2	1	0
Reserved		C1EPFFC	C1EPRFC	C0EPFFC	C0EPRFC	Reserved		C1ICLR	C0ICLR
-		0	0	0	0	-		0	0
-		WO	WO	WO	WO	-		RW	RW

Bits	Name	Function
19	C1EPFFC	Comparator 1 edge polarity falling flag clear (write "1" to clear CMP_IST<C1EPF>)
18	C1EPRFC	Comparator 1 edge polarity rising flag clear (write "1" to clear CMP_IST<C1EPR>)
17	C0EPFFC	Comparator 0 edge polarity falling flag clear (write "1" to clear CMP_IST<C0EPF>)
16	C0EPRFC	Comparator 0 edge polarity rising flag clear (write "1" to clear CMP_IST<C0EPR>)
1	C1ICLR	Comparator 1 Interrupt Clear (write "1" to clear CMP_IST<C1IRQ>)
0	C0ICLR	Comparator 0 Interrupt Clear (write "1" to clear CMP_IST<C0IRQ>)

21.3 Functional description

The comparator compares the voltages of the source (input +) signal and the reference (input -) signal and generates a digital output signal based on the comparison result. If the reference voltage is lower than the source voltage, the digital signal value is 0; otherwise, the signal value is 1. The comparator is enabled by writing a 1 to the CnEN bit of the CMP_CMPnCR register. You can check the comparator output at the CnIRQ bit of the CMP_IST. (n = 0 and 1)

21.3.1 Reference and source input configuration

The reference input can be set with the CnINNSEL bit of the CMP_CMPnCR register. You can write a value between 0 and 2 to the bit field and choose between three references

The source voltage can be set with the CnINPSEL bit of the CMP_CMPnCR register. You can write a value between 0 and 2 to the bit field and choose between max three sources. The A31G22x comparator module can add a hysteresis voltage to the source pin. To use this function, set the CnHYSEN bit in CMP_CMPnCR to 1 to enable the hysteresis voltage and then write to the CnHYYSEL bit to determine the hysteresis voltage value. (n = 0 and 1)

21.3.2 Interrupt function

The comparator interrupt occurs at every rising or falling edge of the comparator output. The comparator detects rising edges if the CnIPOL bit of the CMP_ICON register is set to 1; and falling edges if it is set to 0. You can check the status of this interrupt at the CnIRQ bit of the CMP_IST register. Writing a 1 to the bit clears the interrupt flag (n = 0 and 1).

22 LCD driver

The LCD driver is controlled by the LCD control register (LCD_CR) and LCD driver bias and contrast control register (LCD_BCCR). The LCLK[1:0] determines the frequency of COM signal scanning of each segment output. A RESET clears the LCD control register LCD_CR and LCD_BCCR values to logic '0'.

The LCD display can continue operating during SLEEP modes.

The clock and duty for LCD driver is automatically initialized by hardware, whenever LCD_CR register data value is rewritten. So, don't rewrite LCD_CR frequently.

Channel configuration of LCD Driver [COM x SEG]:

- 80-pin: 8 x 37 or 3 x 42
- 64-pin: 8 x 29 or 3 x 34
- 48-pin: 8 x 21 or 3 x 26

Table 97 introduces pins assigned for LCD Driver.

Table 97. Pin Assignment of LCD Driver: External Signal

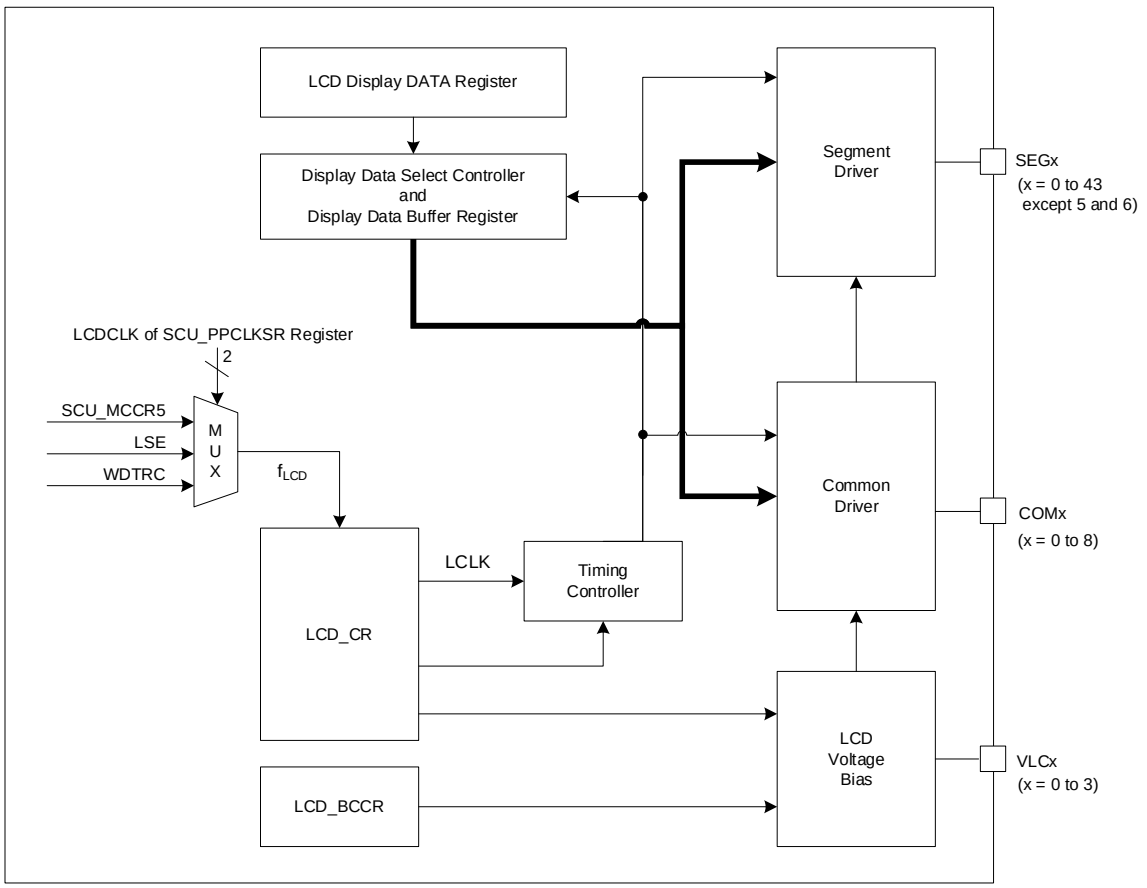
Pin name	Type	Description	A31G226MNN A31G226MLN A31G224MMN A31G224MLN (LQFP-80)	A31G226RMN A31G226RLN A31G224RMN A31G224RLN (LQFP-64)	A31G226CLN A31G224CLN (LQFP-48)
VLCx	A	LCD External Bias voltage input (x = 0 to 3)	O	O	X
COMx	O	LCD common signal outputs (x = 0 to 7)	O (Up to 8)	O (Up to 8)	O (Up to 8)
SEGx	O	LCD segment signal outputs (x = 0 to 43, except 5, 6*)	O (Up to 42)	O (Up to 34)	O (Up to 26)

***NOTE**: SEG5 and SEG6 pins in LQFP-80 package are not available. Please refer to 1.1 and 1.2 in A31G22x Errata sheet.

22.1 LCD Driver block diagram

Figure 137 describes the LCD block diagram.

Figure 137. LCD Driver Block Diagram



NOTES:

1. PE14 (SEG5) and PE15 (SEG6) pins do not support SEG alternative function. Also, LCD_DR5 and LCD_DR6 are not mapped for SEG control.
2. The pull-up resistors for PC1 (SEG23) and PC2 (SEG24) are initially enabled.

22.2 Registers

Base address of the LCD is introduced in the followings:

Table 98. Base Address of LCD

Name	Base address
LCD	0x4000_5000

Table 99. LCD Register Map

Name	Offset	Type	Description	Reset value	Ref.
LCD_CR	0x0000	RW	LCD driver control register	0x0000_0000	22.2.1
LCD_BCCR	0x0004	RW	LCD automatic bias and contrast control register	0x0000_0000	22.2.2
LCD_BSSR	0x0010	RW	LCD bias source selection register	0x0000_0000	22.2.3
LCD_DRn	0x0014	RW	LCD display data n register (n = 0 to 43, except 5, 6)	0x0000_0000	22.2.4

22.2.1 LCD_CR: LCD driver control register

LCD Driver Control Register is 32-bit register.

LCD_CR=0x4000_5000

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6	5 4 3	2 1	0
Reserved			IRSEL	DBS	LCLK	DISP
-			00	000	00	0
-			RW	RW	RW	RW

Bits	Name	Function	
7 6	IRSEL	Internal LCD bias dividing resistor selection bits.	
		00	RLCD3: 105/105/80[KΩ] @ (1/2)/(1/3)/(1/4) bias.
		01	RLCD1: 10/10/10[KΩ] @ (1/2)/(1/3)/(1/4) bias.
		10	RLCD2: 66/66/50[KΩ] @ (1/2)/(1/3)/(1/4) bias.
		11	RLCD4: 320/320/240[KΩ] @ (1/2)/(1/3)/(1/4) bias.
5 3	DBS	LCD duty and bias selection bits	
		000	1/8 duty, 1/4 bias.
		001	1/6 duty, 1/4 bias.
		010	1/5 duty, 1/3 bias.
		011	1/4 duty, 1/3 bias.
		100	1/3 duty, 1/3 bias.
		101	1/3 duty, 1/2 bias
		Others	Reserved.
2 1	LCLK	LCD clock divider selection bits. f_{LCD} (LCD clock source) is selected by SCU_PPCLKSR	
		00	f_{LCD} / 256 (128Hz @ f_{LCD} = 32.768KHz)
		01	f_{LCD} / 128 (256Hz @ f_{LCD} = 32.768KHz)
		10	f_{LCD} / 64 (512Hz @ f_{LCD} = 32.768KHz)
		11	f_{LCD} / 32 (1024Hz @ f_{LCD} = 32.768KHz)
0	DISP	LCD display control bit.	
		0	Display off.
		1	Normal display on.

22.2.2 LCD_BCCR: LCD automatic bias and contrast control register

LCD automatic bias and contrast control register is 32-bit register.

LCD_BCCR=0x4000_5004

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12	11	10 9 8	7 6	5	4	3 2 1 0
Reserved		LCDABC	Reserved	BMSEL	Reserved	LCTEN	Reserved	VLCD
-		0	-	000	-	0	-	0000
-		RW	-	RW	-	RW	-	RW

Bits	Name	Function	
12	LCDABC	LCD Automatic Bias Control bit.	
		0	LCD automatic bias is off.
		1	LCD automatic bias is on.
10 8	BMSEL	"Bias Mode A" time selection bits. Refer to the Figure 144. LCD automatic bias control timing diagram	
		000	"Bias Mode A" for 1-clock of $f_{LCD.}$
		001	"Bias Mode A" for 2-clock of $f_{LCD.}$.
		010	"Bias Mode A" for 3-clock of $f_{LCD.}$.
		011	"Bias Mode A" for 4-clock of $f_{LCD.}$.
		100	"Bias Mode A" for 5-clock of $f_{LCD.}$.
		101	"Bias Mode A" for 6-clock of $f_{LCD.}$.
		110	"Bias Mode A" for 7-clock of $f_{LCD.}$.
		111	"Bias Mode A" for 8-clock of $f_{LCD.}$.
5	LCTEN	LCD Driver Contrast Control bit.	
		0	Disable LCD driver contrast.
		1	Enable LCD driver contrast.
3 0	VLCD	VLC0 Voltage Control when the contrast is enabled.	
		0000	VLC0 = VDD x 16/31 step
		0001	VLC0 = VDD x 16/30 step
		0010	VLC0 = VDD x 16/29 step
		0011	VLC0 = VDD x 16/28 step
		0100	VLC0 = VDD x 16/27 step
		0101	VLC0 = VDD x 16/26 step
		0110	VLC0 = VDD x 16/25 step
		0111	VLC0 = VDD x 16/24 step
		1000	VLC0 = VDD x 16/23 step
		1001	VLC0 = VDD x 16/22 step
		1010	LVC0 = VDD x 16/21 step
		1011	LVC0 = VDD x 16/20 step

	1100	LVC0 = VDD x 16/19 step
	1101	LVC0 = VDD x 16/18 step
	1110	LVC0 = VDD x 16/17 step
	1111	LVC0 = VDD x 16/16 step
NOTE: The above LCD contrast step is based on 1/3 bias with 66kΩ RLCD and on 1/4 bias with 50kΩ RLCD.		

22.2.3 LCD_BSSR: LCD source selection register

LCD Bias source selection register is 32-bit register.

LCD_BSSR=0x4000_500C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12	11	10	9	8	7	6	5	4	3 2 1 0
Reserved			VLE_EN	Reserved	LCDDR	LCDEPEN	VLC3EN	VLC2EN	VLC1EN	VLC0EN	Reserved
-			0	-	0	0	0	0	0	0	-
-			RW	-	RW	RW	RW	RW	RW	RW	-

Bits	Name	Function
11	VLE_EN	External bias enable register. **NOTE:** To use LCD external bias VLCx, VLE_EN=1, LCDDR=1, LCDEPEN =1, VLCxEN = 1.
		0 External bias mode OFF
		1 External bias mode ON
9	LCDDR	LCD driving resistor for bias select
		0 Internal LCD driving resistors for bias
		1 External LCD driving resistors for bias
8	LCDEPEN	LCD external bias path enable bit
		0 disable
		1 Enable
7	VLC3EN	External bias VLC3 enable bit
		0 Disable VLC3
		1 Enable VLC3
6	VLC2EN	External bias VLC2 enable bit
		0 Disable VLC2
		1 Enable VLC2
5	VLC1EN	External bias VLC1 enable bit
		0 Disable VLC1
		1 Enable VLC1
4	VLC0EN	External bias VLC0 enable bit
		0 Disable VLC0
		1 Enable VLC0

22.2.4 LCD_DRn: LCD display data n register (n = 0 to 43)

LCD_DRn register is 8-bit size, and accesses to LCD Display Data register to update data. Please refer to Table 100 for information about LCD Display register address table.

LCD_DR0=0x4000_5010, … , LCD_DR43=0x4000_503B

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
DR(4*(n+3))	DR(4*(n+2))	DR(4*(n+1))	DR(4*(n+0))
0x00	0x00	0x00	0x00
RW	RW	RW	RW

Bits	Name	Function
31 24	DR(4*n+3)	LCD Display Data bits.
23 16	DR(4*n+2)	LCD Display Data bits.
15 8	DR(4*n+1)	LCD Display Data bits.
7 0	DR(4*n+0)	LCD Display Data bits.

NOTES:
1. PE14 (SEG5) and PE15 (SEG6) are not supported SEG alternative function.
2. The pull-up resistors for PC1 (SEG23) and PC2 (SEG24) are initially enabled.

Table 100. The base address table of LCD display registers by LCD segments

Segment	Register Name	Base Address	Segment	Register Name	Base Address
SEG0	DR0	0x4000_5010	SEG22	DR22	0x4000_5026
SEG1	DR1	0x4000_5011	SEG23	DR23	0x4000_5027
SEG2	DR2	0x4000_5012	SEG24	DR24	0x4000_5028
SEG3	DR3	0x4000_5013	SEG25	DR25	0x4000_5029
SEG4	DR4	0x4000_5014	SEG26	DR26	0x4000_502A
SEG5*	DR5*	0x4000_5015	SEG27	DR27	0x4000_502B
SEG6*	DR6*	0x4000_5016	SEG28	DR28	0x4000_502C
SEG7	DR7	0x4000_5017	SEG29	DR29	0x4000_502D
SEG8	DR8	0x4000_5018	SEG30	DR30	0x4000_502E
SEG9	DR9	0x4000_5019	SEG31	DR31	0x4000_502F
SEG10	DR10	0x4000_501A	SEG32	DR32	0x4000_5030
SEG11	DR11	0x4000_501B	SEG33	DR33	0x4000_5031
SEG12	DR12	0x4000_501C	SEG34	DR34	0x4000_5032
SEG13	DR13	0x4000_501D	SEG35	DR35	0x4000_5033
SEG14	DR14	0x4000_501E	SEG36	DR36	0x4000_5034
SEG15	DR15	0x4000_501F	SEG37	DR37	0x4000_5035
SEG16	DR16	0x4000_5020	SEG38	DR38	0x4000_5036
SEG17	DR17	0x4000_5021	SEG39	DR39	0x4000_5037
SEG18	DR18	0x4000_5022	SEG40	DR40	0x4000_5038
SEG19	DR19	0x4000_5023	SEG41	DR41	0x4000_5039
SEG20	DR20	0x4000_5024	SEG42	DR42	0x4000_503A
SEG21	DR21	0x4000_5025	SEG43	DR43	0x4000_503B

***NOTE**: DR5 (0x4000_5015) and DR6 (0x4000_5016) registers mapped to SEG5 pin and SEG6 pin do not support LCD control. Please refer to 1.1 and 1.2 in A31G22x Errata sheet.

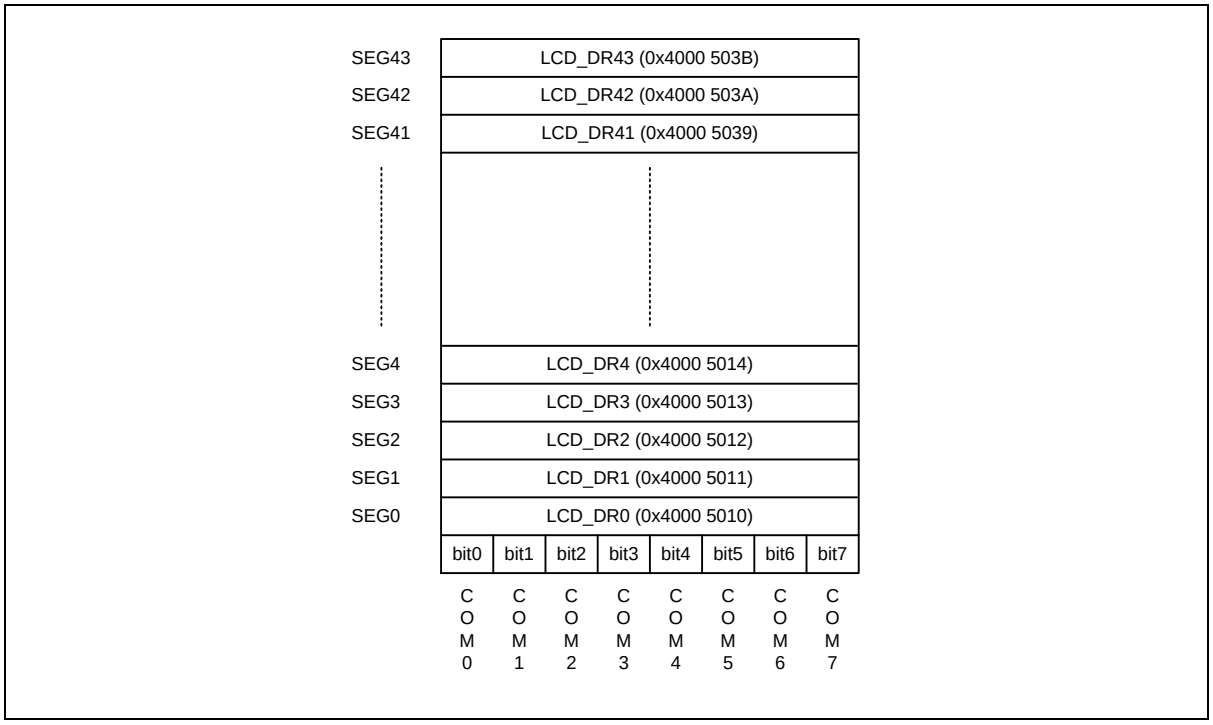
22.3 Functional description

22.3.1 LCD display RAM organization

The display data which stored to the display external data area (address 0x4000_5010-0x4000_503B) are read automatically and sent to the LCD driver by the hardware. The LCD driver generates the segment signals and common signals in accordance with the display data and drive method. Therefore, display patterns can be changed by only overwriting the contents of the display external data area with a program.

Figure 138 shows the correspondence between the display external data area and the COM/SEG pins. The LCD is turned on lights when the display data is "1" and turned off when "0".

Figure 138. LCD display RAM



22.3.2 LCD signal waveform

Figure 139. LCD Signal Waveforms (1/3 Duty, 1/3 Bias)

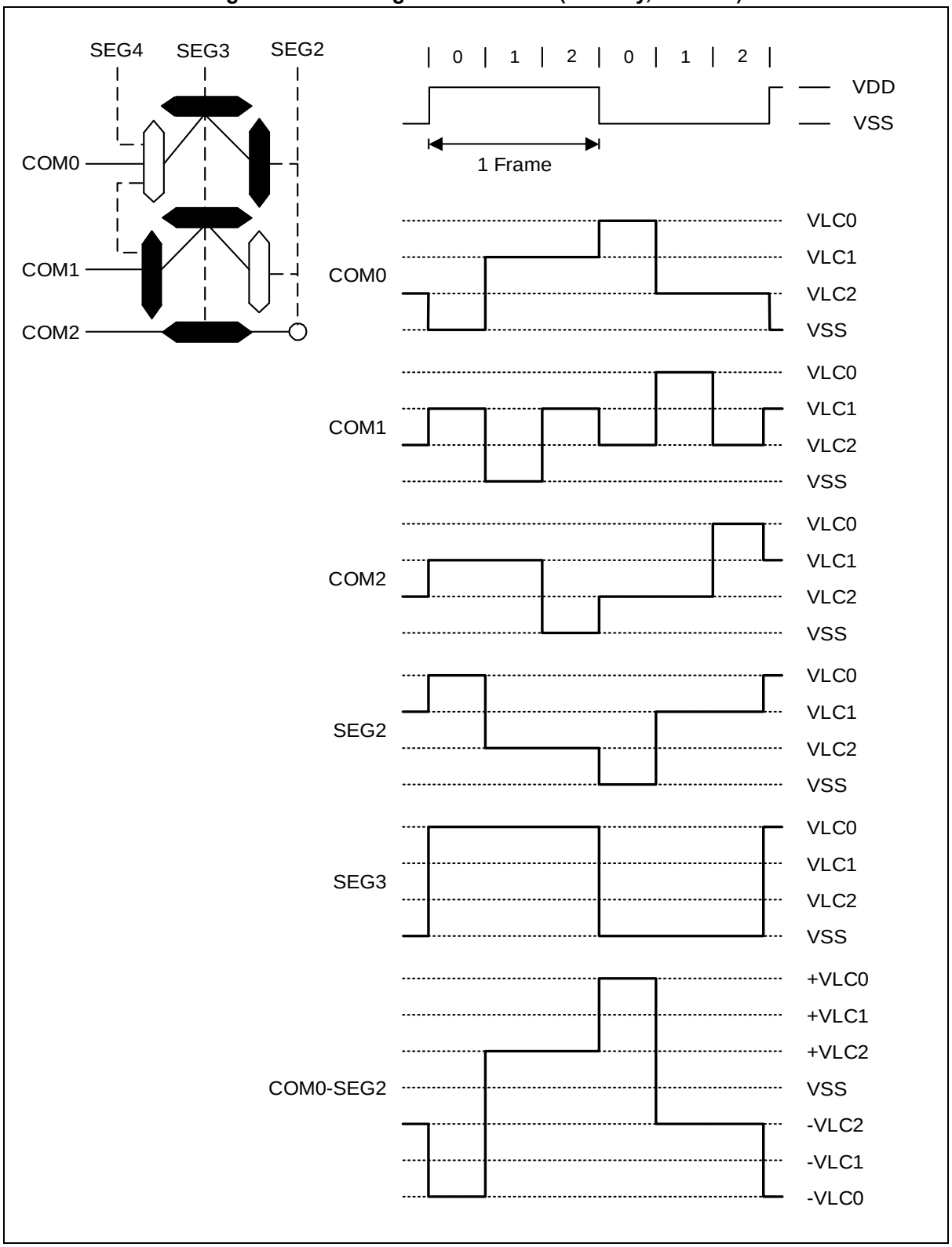


Figure 140. LCD Signal Waveforms (1/4 Duty, 1/3 Bias)

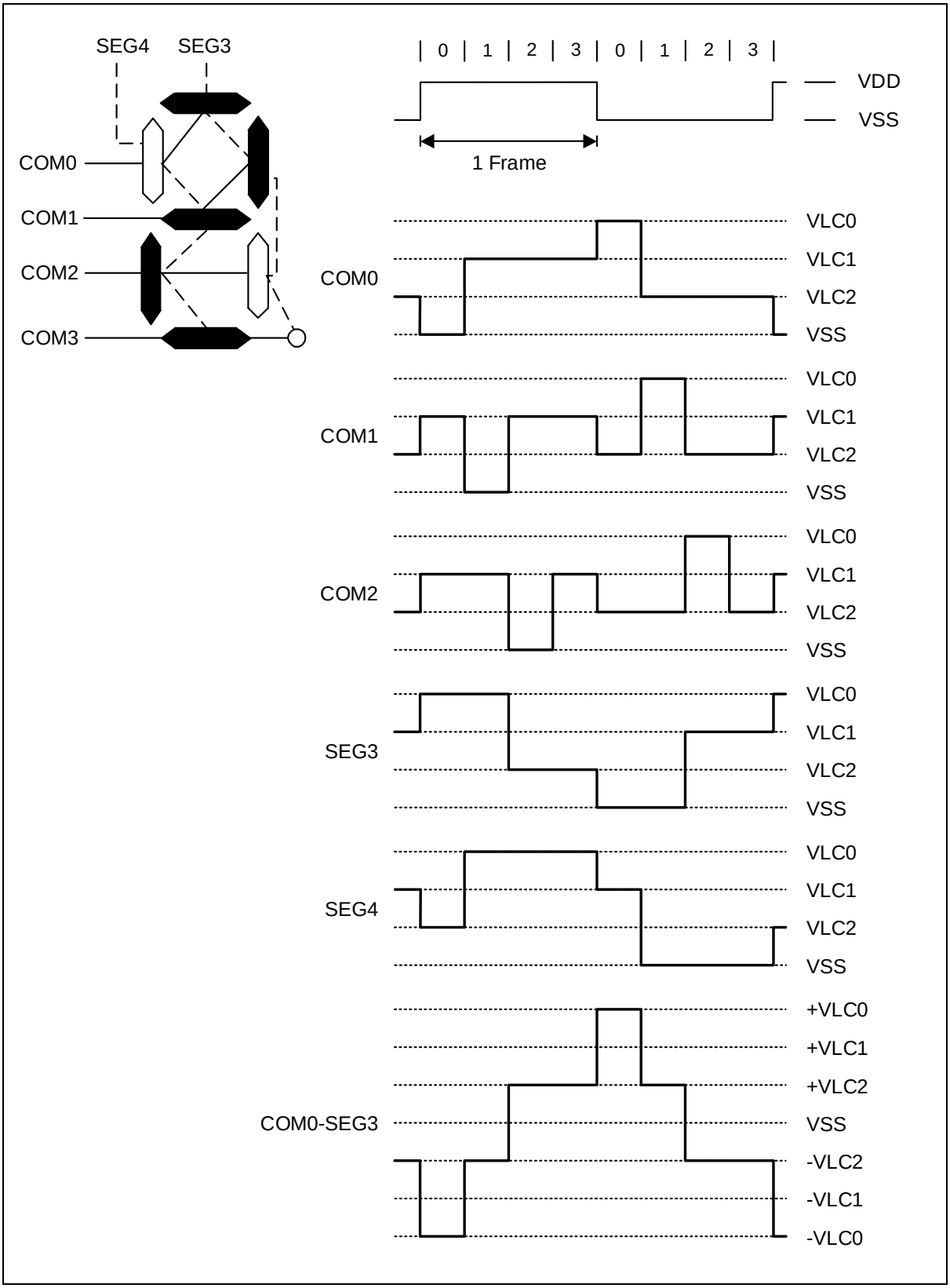
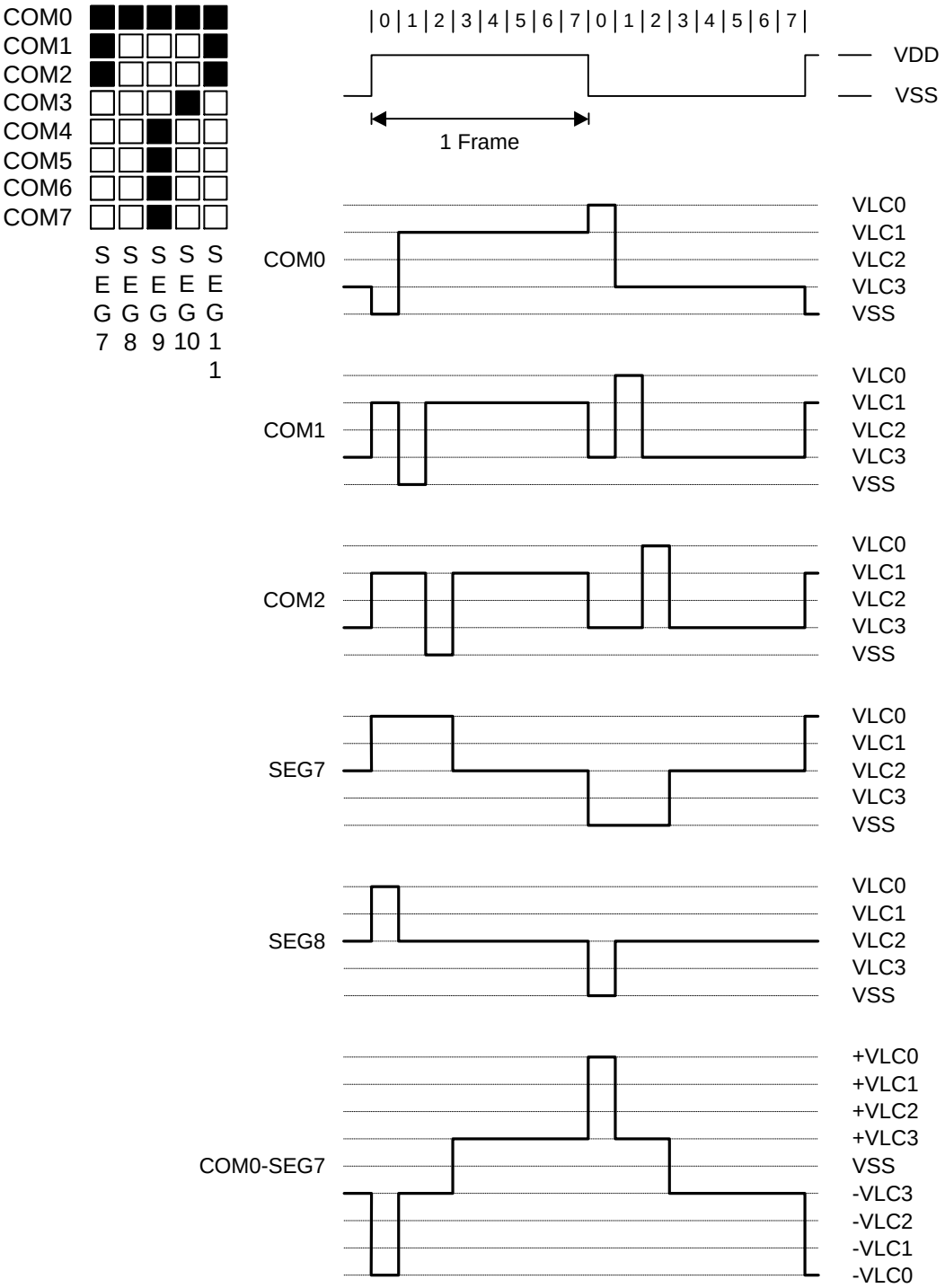
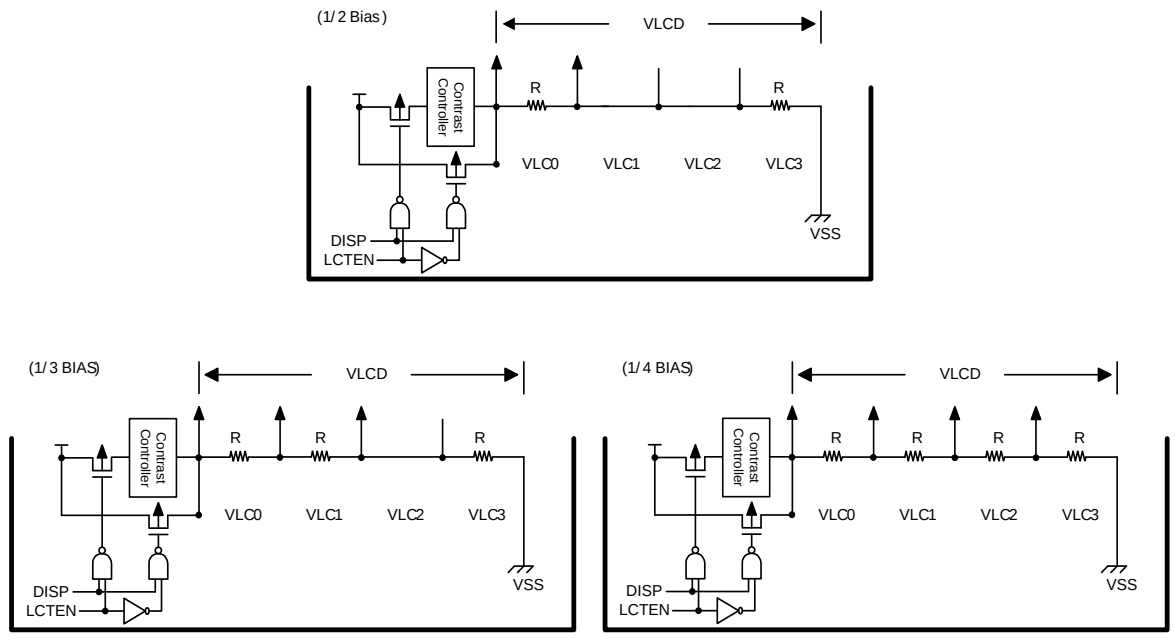


Figure 141. LCD Signal Waveforms (1/8 Duty, 1/8 Bias)



22.3.3 Internal resistor bias connection

Figure 142. LCD Signal Waveforms (1/8 Duty, 1/8 Bias)

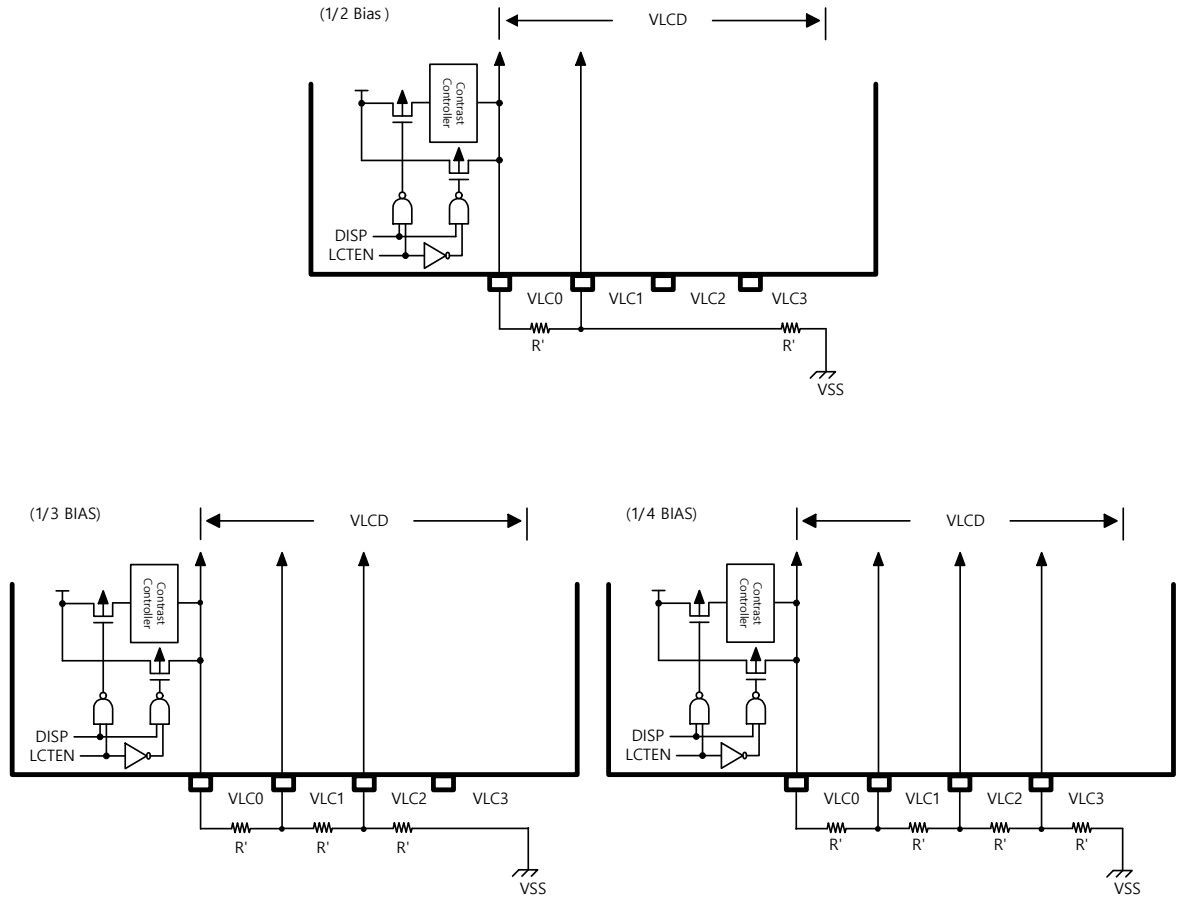


NOTES:

1. When 1/2 bias are selected, the VLC1, VLC2 and VLC3 are internally connected.
2. When 1/3 bias are selected, the VLC2 and VLC3 are internally connected.
3. When the internal resistors are selected, all the P73/VLC0, P72/VLC1, P71/VLC2 and P70/VLC3 pins can be used for normal I/O.

22.3.4 External resistor bias connection

Figure 143. Internal Resistor Bias Connection

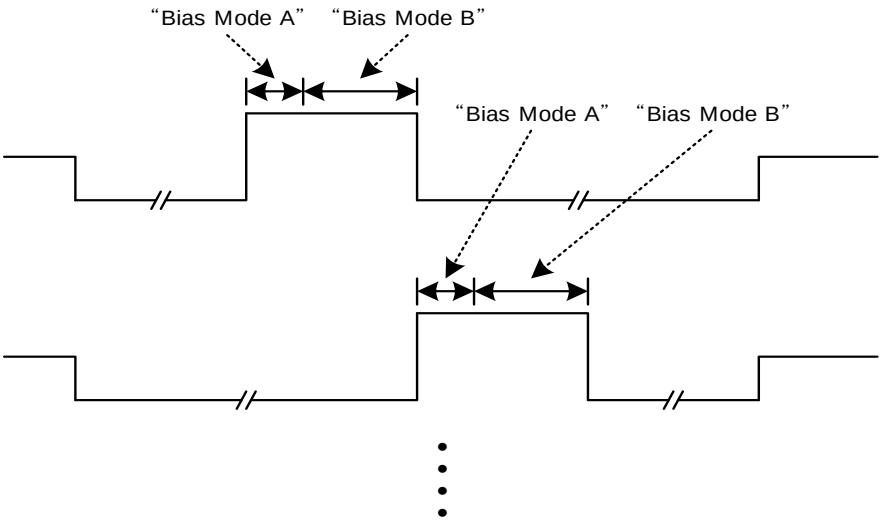


NOTES:

4. When the external resistor bias is selected, the internal resistors for bias are disconnected.
5. When the external resistor bias is selected, the dividing resistors should be connected like the above figure (Figure 143) and the needed bias pins should be selected as the LCD bias function pins (VLC0, VLC1, VLC2 and VLC3) by PE_MOD & PE_ASFR1 register.
6. It must set the LCDDR LCDEN VLCnEN of the LCD_BSSR register to '1'. And the others must be set to '0'.

22.3.5 LCD automatic bias control timing

Figure 144. LCD automatic bias control timing diagram



NOTES:

1. Refer to the below when the LCD automatic bias is on
2. "Bias Mode A": Always RLCD1 (10kΩ).
3. "Bias Mode B": A LCD bias dividing resistor which is selected by IRSEL[1:0] bits

23 Cyclic Redundancy Check (CRC) calculation unit

Cyclic redundancy check (CRC) generator is used to get a 16-bit CRC code from Flash ROM and a generator polynomial. Among other applications, CRC-based techniques are used to verify data transmission or storage integrity. In the scope of the functional safety standards, they offer a means of verifying the Flash memory integrity. CRC generator helps compute a signature of the software during runtime, to be compared with a reference signature.

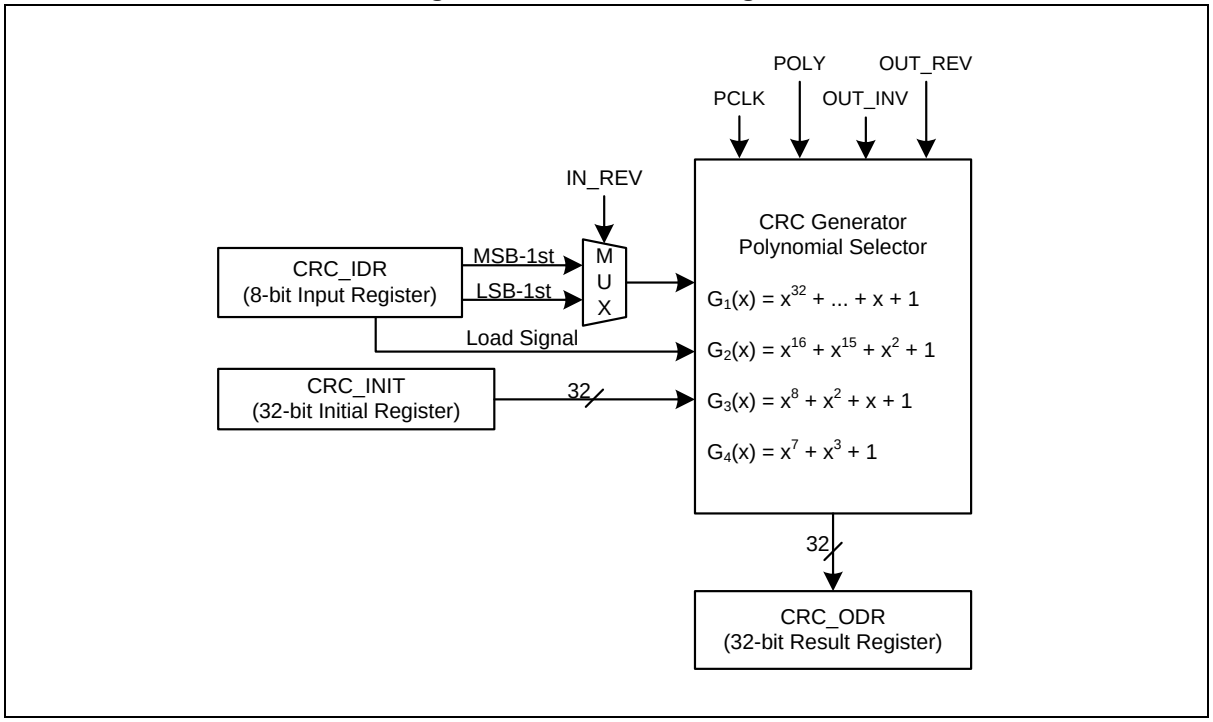
CRC generator of A31G22x series features the followings:

- Auto CRC (DMA) and User CRC Mode.
- Polynomial:
  - CRC-32 ($G_0(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$)
  - CRC-16 ($G_1(x) = x^{16} + x^{15} + x^2 + 1$),
  - CRC-8 ($G_2(x) = x^8 + x^2 + x + 1$)
  - CRC-7 ($G_3(x) = x^7 + x^3 + 1$)

23.1 CRC block diagram

Figure 145 describes the CRC block diagram.

Figure 145. CRC Block Diagram



23.2 Registers

Base address of the CRC is introduced in the followings:

Table 101. Base Address of CRC

Name	Base address
CRC	0x4000_0300

Table 102. CRC Register Map

Name	Offset	Type	Description	Reset value	Ref.
CRC_CR	0x00	RW, WO	CRC Control Register	0x0000_0000	23.2.1
CRC_INIT	0x04	RW	CRC Initial Data Register	0xFFFF_FFFF	23.2.2
CRC_IDR	0x08	WO	CRC Input Data Register	0x0000_0000	23.2.3
CRC_ODR	0x08	RO	CRC Output Data Register	0xFFFF_FFFF	23.2.4
CRC_SR	0x0C	RW	CRC Status Register	0x0000_0000	23.2.5

23.2.1 CRC_CR: CRC control register

CRC_CR is a 32-bit register, and able to do 32/16/8-bit access.

CRC_CR=0x4000_0300

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16				15 14 13 12 11 10 9 8		7 6 5 4 3 2 1 0		
Reserved	OUT_INV	OUT_REV	Reserved	IN_REV	Reserved	DMADINT	Reserved	POLY	INIT_EN
-	0	0	-	0	-	0	-	00	0
-	RW	RW	-	RW	-	RW	-	RW	WO

Bits	Name	Function	
21	OUT_INV	Whether to enable or disable CRC output data inversion	
		0	Disable
		1	Enable
20	OUT_REV	Whether to enable or disable CRC output data reverse	
		0	LSB-first
		1	MSB-first
16	IN_REV	Selects the first data bit to be calculated	
		0	LSB-first
		1	MSB-first
8	DMADINT	Whether to enable or disable the DMA done interrupt	
		0	Disables.
		1	Enables.
2 1	POLY	Polynomial selection	
		00	CRC32 (0x04C1_1DB7)
		01	CRC16 (0x8005)
		10	CRC8 (0x07)
		11	CRC7 (0x09)
0	INIT_EN	Whether or not to apply the CRC initial value register	
		0	No Effect
		1	Applies the initial value register value.

23.2.2 CRC_INIT: CRC initial data register

CRC Initial Data Register is 32-bit register The CRC initial value is written to CRC_INIT.

CRC_INIT=0x4000_0304

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
INIT			
0xFFFF_FFFF			
RW			

Bits	Name	Function
31 0	INIT	CRC initial data bits

NOTE: To write INIT data to the CRC register, the CRC_CR register's INIT_EN bit must be enabled. For example, write INIT = '0xFFFF' and write a '1' to INIT_EN bit, then this value is assigned to calculating CRC.

23.2.3 CRC_IDR: CRC input data register

CRC Input Data Register has an 8-bit width input data field for CRC calculations.

CRC_CR=0x4000_0308

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved			INPUT
-			0x00
-			WO

Bits	Name	Function
7 0	INPUT	CRC input data Once data is entered in this bit field, its polynomial result is automatically written at CRC_OUTPUT register.

NOTE: When CRC operation is performed with DMA, the value of DMAn_CR <SIZE> bit should be selected as '0x00' so that the data size must be one byte.

23.2.4 CRC_ODR: CRC output data register

CRC Output Data Register has a 32-bit width output data field that returns a calculated CRC value.

CRC_ODR=0x4000_0308

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
OUTPUT			
0xFFFF_FFFF			
RO			

Bits	Name	Function
31 0	INPUT	CRC output data

23.2.5 CRC_SR: CRC status register

CRC Status Register is a 32-bit register and displays the operating status of CRC.

CRC_CR=0x4000_030C

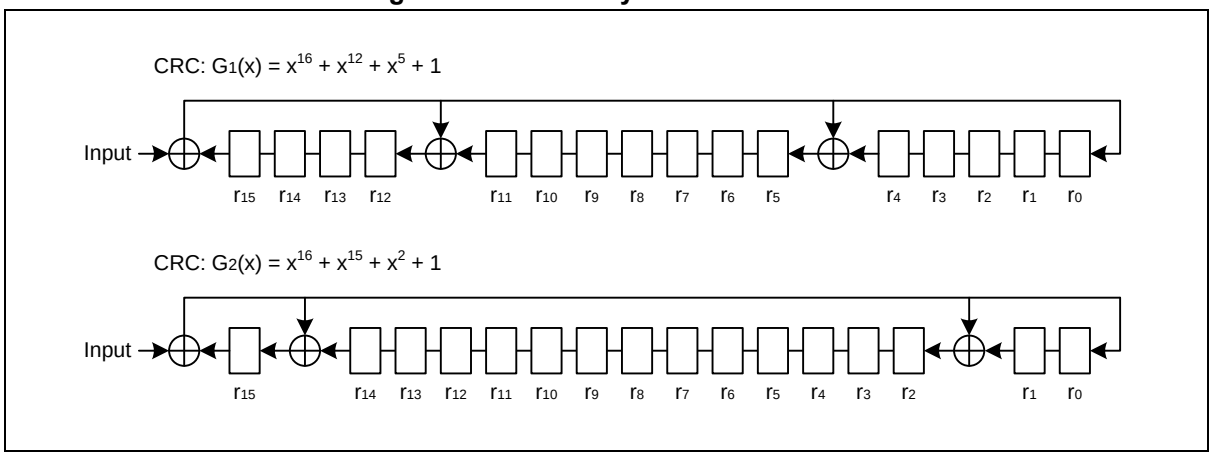
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9	8	7 6 5 4 3 2 1 0
Reserved			DMADINT	Reserved
-			0	-
-			RW	-

Bits	Name	Function	
8	DMADINT	CRC DMA done interrupt flag	
		0	The DMA transfer is not complete.
		1	Done DMA transfer.

23.3 Functional description

23.3.1 CRC polynomial structure

Figure 146. CRC Polynomial Structure



23.3.2 Initial configuration of CRC32

SCU_PER2.CRC = "1" : Enables the CRC peripheral.

SCU_PCER2.CRC = "1" : Enables the CRC peripheral clock.

CRC_CR.POLY = "0" : Selects CRC32 operating mode.

CRC_INIT = "0xFFFFFFFF" : Sets the CRC initial value to 0xFFFFFFFF.

CRC_CR.INIT_EN = "1" : Applies the CRC initial value.

CRC_IDN = "0x30" : Sets the CRC input value to 0x30

CRC_IDN = "0x31" : Sets the CRC input value to 0x31

read_data = CRC_ODR; : Stores the CRC result in read_data.

23.3.3 DMA transfer configuration for CRC

SCU_PER1.DMA = "1" : Enables the DMA peripheral.

SCU_PCER1.DMA = "1" : Enables the DMA peripheral clock.

DMA_CR.TRANSCNT = "0x005" : Sets the transfer counter number.

DMA_CR.SIZE = "00" : Sets the DMA transmit size to byte size.

DMA_CR.PERISEL = "0x05" : Selects the CRC module as the peripheral for DMA transfer.

DMA_CR.DIR = "0" : Sets the DMA direction to memory -> peripheral (Tx).

DMA_PAR = "0x4000_0308" : Sets the peripheral address (CRC_IDR).

DMA_MAR = "0x20000200" : Sets the memory address.

DMA_SR.DMAEN = "1" : Enables DMA.

24 Temp sensor

Temp sensor is to use the internal oscillator LSITS by default, which has a large temperature variation. The temperature-dependent LSITS frequency can be calculated based on a precisely trimmed internal oscillator or an external clock.

Reference clock and sense clock of the temp sensor can be changed by configuring TS_CR register. When selecting the clock, frequency of the reference clock must be faster than the sense clock frequency. In SCU, each clock must be activated by configuring corresponding register.

If value of RCCV using reference clock matches the RCCV in TS_RCCNT register set by the user, a match flag is generated and frequency of the sense clock is calculated by reading the value of SCCV in TS_SCCNT register at this time. Match flags can be used as interrupt sources.

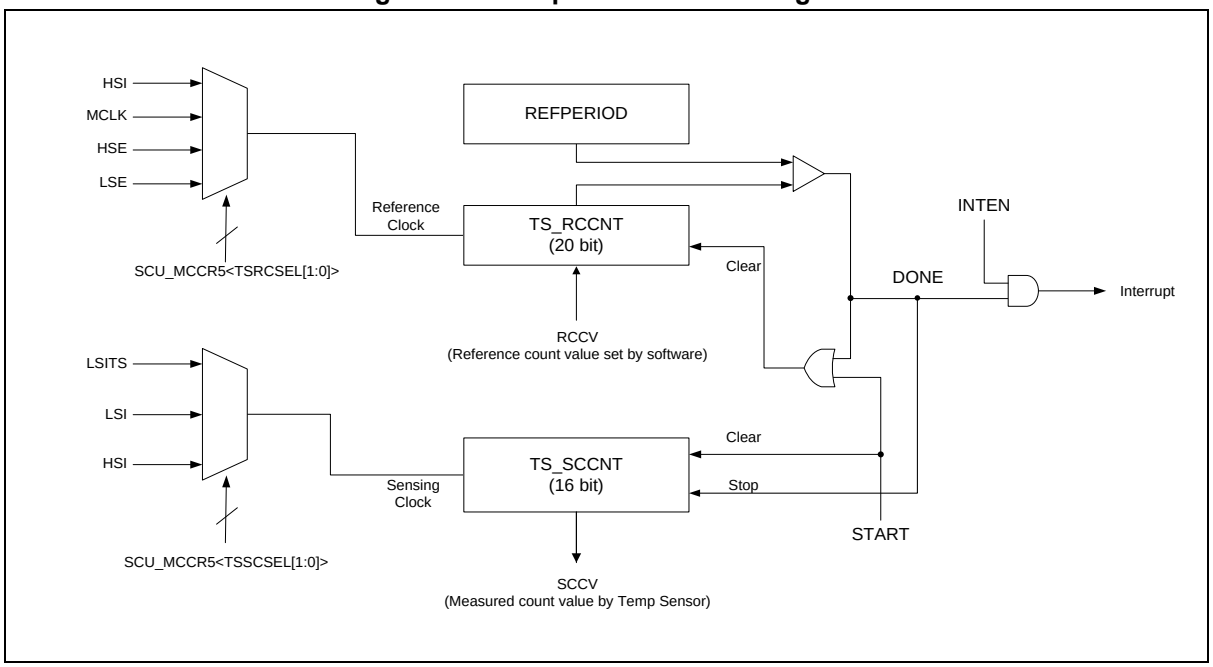
Glossary for this chapter

- HSI: HSI clock set by SCU_CSCR
- MCLK: System clock set by SCU_SCCR
- LSE: External sub oscillator
- LSITS: Internal temp sense oscillator
- LSI: Internal 500KHz oscillator

NOTE: If the TS clock source to be used in the operation of TS_SCCNT is selected as LSI or HSI instead of LSITS, the temperature sensor becomes a normal timer counter operation. Therefore, it is recommended to use a clock source with LSITS for temperature measurement.

24.1 Temp sensor block diagram

Figure 147. Temp Sensor Block Diagram



24.2 Registers

Base address of a temp sensor block is introduced in the followings:

Table 103. Base Address of TSENSE Interface

Name	Base address
TS (TEMP SENSOR)	0x4000_6300

Table 104. TSENSE Register Map

Name	Offset	Type	Description	Reset value	Ref.
TS_CR	0x0000	RW	Temp Sensor Control Register	0x0000_0000	24.2.1
TS_RCCNT	0x0004	RW	Temp Sensor Reference Clock Counter Register	0x0000_0000	24.2.2
TS_SCCNT	0x0008	RO	Temp Sensor Sensing Clock Counter Register	0x0000_0000	24.2.3
TS_SR	0x000C	RW	Temp Sensor Status Register	0x0000_0000	24.2.4

24.2.1 TS_CR: temp sensor control register

TS_CR register is temperature sensor control register. The operation and interrupt settings of the temperature sensor can be configured in this register.

During Temp Sensor operation, if internal counter increases to the value of reference period, TS_CR register enable a match flag. If INTEN bit in this register is set to '1' to enable interrupts, both the Temp Sensor interrupt and the match flag are generated.

TS_CR =0x4000_6300

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		INTEN	Reserved START
-		0	- 0
-		RW	- RW

Bits	Name	Function	
8	INTEN	Temp sensor interrupt enable	
		0	Disable temp sensor interrupt
		1	Enable temp sensor interrupt. When TS_SR<CNTDONE> flag is set, the interrupt of temp sensor is occurred.
0	START	Start reference clock counting	
		0	Stop
		1	Start, auto clear this bit to '0' after 1 set.

24.2.2 TS_RCCNT: temp sensor reference clock counter register

TS_RCCNT register has a 20-bit width field that sets the reference clock period for the temperature sensor. The reference clock is set by SCU_MCCR5 register.

TS_REFPERIOD =0x4000_6304

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		RCCV	
-		0x00000	
-		RW	

Bits	Name	Function
20 0	RCCV	Reference clock counter initial value. The reference clock counter counts down to zero from the RCCV setting value.

24.2.3 TS_SCCNT: temp sensor sensing clock counter register

TS_SCCNT has 16-bit counter and displays current count value of temp sensor based on the reference period configured by TS_RCCNT register.

TS_SENSECON =0x4000_6308

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Reserved		SCCV	
-		0x0000	
-		RO	

Bits	Name	Function
16 0	SCCV	Temp sensor counter value.

24.2.4 TS_SR: Temp sensor status register

TS_SR has 32-bit counter and displays current count value based on the reference period configured by TS_RCCNT register.

TS_SR =0x4000_630C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0	
Reserved			DONE	Reserved	BUSY
-			0	-	0
-			RW	-	RO

Bits	Name	Function	
8	DONE	Reference clock counting done flag	
		0	Reference clock counter is running.
		1	Reference clock counting is done. When reference clock counter is done, this bit automatically set to '1'. This value is cleared by writing '1' to this bit.
0	BUSY	The Status of the reference clock counter	
		0	Reference clock counter is ready.
		1	Reference clock counter is busy. BUSY flag is set '1' during the reference counter is running and automatically clear '0' when reference down-counting is finished.

25 Electrical characteristics

25.1 Absolute maximum ratings

Absolute maximum ratings are limiting values of operating and environmental conditions which should not be exceeded under the worst possible conditions.

Table 105. Absolute maximum rating

Parameter	Symbol	Ratings	Unit	Remark
Supply Voltage	V_{DD}	-0.3 – +6.5	V	—
Normal Pin	V_I	-0.3 – VDD+0.3	V	Voltage on any pin with respect to V_{SS}.
	V_O	-0.3 – VDD+0.3	V	
	I_{OH}	-20	mA	Maximum output current sourced by per I/O pin.
	ΣI_{OH}	-100	mA	Total output current sourced by sum of all I/Os.
	I_{OL}	25	mA	Maximum output current sunk by per I/O pin except I_{OL7}^*.
	ΣI_{OL}	210	mA	Total output current sunk by sum of all I/Os.
Total Power Dissipation	T_P	300	mW	—
Storage Temperature	T_{STG}	-55 – +125	℃	—

NOTE: I_{OL7} is output low level current specification when high current output function is enabled. Refer 25.13.2

25.2 Recommended operating conditions

Table 106. Recommended Operating Condition

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Supply Voltage	V_{DD}	Core and Peripherals	1.8	—	5.5	V
		CMP	2.0	—	5.5	V
		ADC	2.1	—	5.5	V
		DAC, LCD	2.7	—	5.5	V
		HSE	1.8	—	5.5	V
		LSE	2.7	—	5.5	V
Operating Frequency	f_{OP}	PLL	1	—	48	MHz
		HSE	1	—	16	MHz
		LSE	—	32.768	—	KHz
		HSI	31.52	32.00	32.48	MHz
		LSI	400	500	600	KHz
Operating Temperature	T_{OP}	Top	-40	+25	+105	℃

25.3 POR (Power-on Reset) characteristics

Table 107. POR Electrical Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Operating Current	I_{DD}	–	-	0.5	-	uA
POR Set Level	V_{SET}	–	1.05	1.20	1.35	V
POR Reset Level	V_{RESET}	–	1.00	1.10	1.20	V
Supply Rising Rate	T_{rVDD}		-	-	10	V/ms
Supply Falling Rate	T_{fVDD}		-	-	10	V/ms

25.4 LVR (Low Voltage Reset) characteristics

Table 108. Operating Condition of Low Voltage Reset

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Temperature	T_A	-40	25	105	℃

Table 109. Low Voltage Reset Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Operating voltage	V_{DD}	–	0.8	5.0	5.5	V
LVR detection level	V_{FLVR}	Falling detection voltage	1.45	1.56	1.67	V
			1.58	1.70	1.82	
			1.66	1.79	1.92	
			1.78	1.91	2.04	
			1.86	2.00	2.14	
			1.98	2.13	2.28	
			2.15	2.31	2.47	
			2.31	2.48	2.65	
			2.49	2.68	2.87	
			2.83	3.05	3.27	
			2.97	3.19	3.41	
			3.35	3.60	3.85	
			3.48	3.74	4.00	
			3.76	4.04	4.32	
			3.92	4.22	4.52	
			4.18	4.50	4.82	
	V_{RLVR}	Rising detection voltage	1.49	1.60	1.71	V
			1.62	1.74	1.86	
			1.70	1.83	1.96	
			1.83	1.96	2.09	
			1.92	2.06	2.20	
			2.04	2.19	2.34	
			2.21	2.37	2.53	
			2.37	2.54	2.71	
			2.56	2.75	2.94	
			2.91	3.13	3.35	
			3.05	3.28	3.51	
			3.45	3.70	3.95	
			3.58	3.84	4.10	
			3.86	4.15	4.44	
			4.03	4.33	4.63	

			4.30	4.62	4.94	
Noise cancelling time	t_{NC}	–	-	2	-	us
Operation current	I_{DD}	–	-	3.5	5.0	uA
Operation current (DEEP-SLEEP mode)	$I_{DD, STOP}$	–	-	2.5	3.0	nA

NOTE: While the system is in DEEP-SLEEP mode, if LVD function is enabled, BGR function of VDC should be enabled by setting SCU_VDCCON<PDBGR> to '0'.

25.5 LVI (Low Voltage Indicator) characteristics

Table 110. Operating Condition of Low Voltage Reset

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Temperature	T_A	-40	25	105	℃

Table 111. Low Voltage Indicator Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Operating voltage	V_{DD}		0.8	5.0	5.5	V
LVI detection level	V_{FLVI}	Falling detection voltage	1.86	2.00	2.14	V
			1.98	2.13	2.28	
			2.15	2.31	2.47	
			2.31	2.48	2.65	
			2.49	2.68	2.87	
			2.83	3.05	3.27	
			2.97	3.19	3.41	
			3.35	3.60	3.85	
			3.48	3.74	4.00	
			3.76	4.04	4.32	
			3.92	4.22	4.52	
			4.18	4.50	4.82	
	V_{RLVI}	Rising detection voltage	1.92	2.06	2.20	V
			2.04	2.19	2.34	
			2.21	2.37	2.53	
			2.37	2.54	2.71	
			2.56	2.75	2.94	
			2.91	3.13	3.35	
			3.05	3.28	3.51	
			3.45	3.70	3.95	
			3.58	3.84	4.10	

			3.86	4.15	4.44	
			4.03	4.33	4.63	
			4.30	4.62	4.94	
Noise cancelling time	t_{NC}	–	-	2	-	us
Operation current	I_{DD}	–	-	3.5	5.0	uA
Operation current (DEEP-SLEEP mode)	$I_{DD, STOP}$	–	-	2.5	3.0	nA

NOTE: While the system is in DEEP-SLEEP mode, if LVD function is enabled, BGR function of VDC should be enabled by setting SCU_VDCCON<PDBGR> to '0'.

25.6 HSI (High Frequency Internal) RC oscillator characteristics

Table 112. Operating Condition of HSI

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	VDD	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 113. High Frequency Internal RC Oscillator Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Frequency	f_{HSI}	T_A = – 20 °C to + 85 °C (Commercial grade)	31.68	32.0	32.32	MHz
Frequency	f_{HSI}	T_A = – 40 °C to + 105 °C (Industrial grade)	31.52	32.0	32.48	MHz
Tolerance	–	T_A = – 20 °C to + 85 °C (Commercial grade)	-1.0	–	1.0	%
		T_A = – 40 °C to + 105 °C (Industrial grade)	-1.5	–	1.5	%
Clock Duty Ratio	T_{OD}	–	–	50	–	%
Stabilization Time	t_{HFS}	–	100		–	us
IRC Current	I_{HSI}	Enable	–	190	–	uA
		Disable	–	1	–	uA

25.7 LSI (Low Frequency Internal) RC oscillator characteristics

Table 114. Operating Condition of LSI

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	VDD	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 115. Low Frequency (500KHz) Internal RC Oscillator Characteristics

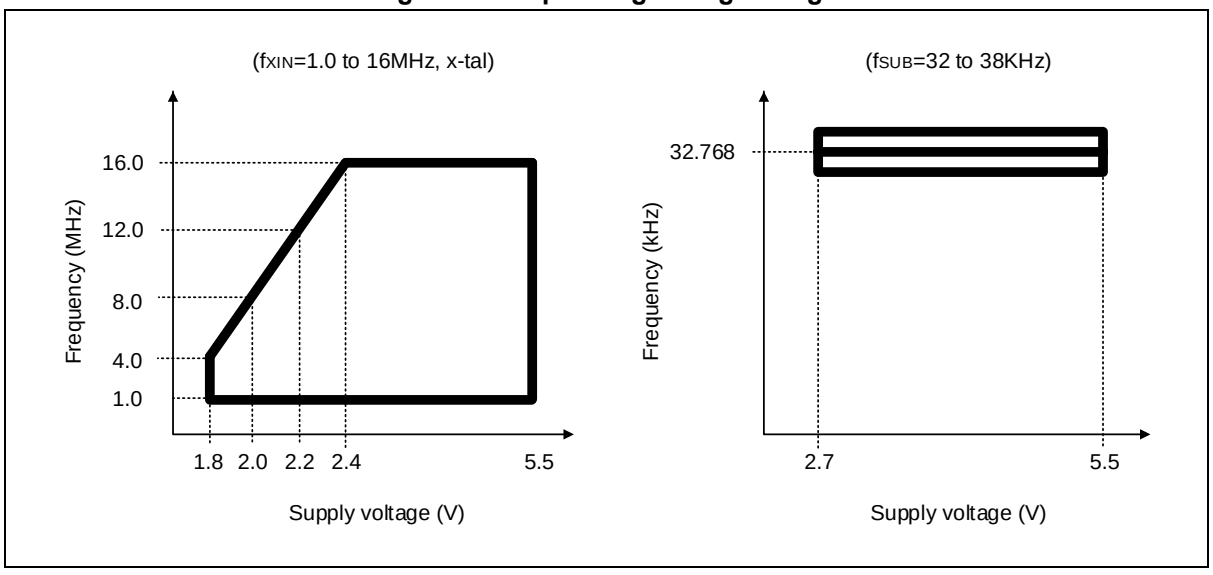
The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ	Max.	Units
Operating current NOTE	I_{LIRC}	Enable	—	1.5	2	uA
		Disable	—	1	20	nA
Frequency	f_{LSI}	V_{DD} = 1.8V to 5.5V T_A = – 40 °C to + 105 °C	400	500	600	KHz
Tolerance	–	V_{DD} = 1.8V to 5.5V T_A = – 40 °C to + 105 °C	-20	–	20	%

NOTE: LSI self-consumption current when LSI clock is not supplied to digital logic.

25.8 Operating voltage range

Figure 148. Operating Voltage Range



25.9 HSE (main oscillator) characteristics

Table 116. Operating Condition of HSE

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	VDD	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 117. Main Oscillator Characteristics

The specifications of the parameters are guaranteed by design.

Oscillator	Parameter	Conditions	Min.	Typ.	Max.	Units
Operating current	I_{DD}	—	—	—	2.5	mA
Power down current	I_{STOP}	—	—	0.2	30	nA
Output frequency	XOUT input frequency	VDDEXT ≥ 1.8V SCU_EOSCR<ISEL[1:0]>='11' SCU_EOSCR<NCOPT[1:0]>='00'	1.0	—	4.0	MHz
		VDDEXT ≥ 2.0V SCU_EOSCR<ISEL[1:0]>='10' SCU_EOSCR<NCOPT[1:0]>='01'	1.0	—	8.0	MHz
		VDDEXT ≥ 2.2V SCU_EOSCR<ISEL[1:0]>='01' SCU_EOSCR<NCOPT[1:0]>='10'	1.0	—	12.0	MHz
		VDDEXT ≥ 2.4V SCU_EOSCR<ISEL[1:0]>='00' SCU_EOSCR<NCOPT[1:0]>='11'	1.0	—	16	MHz
Start-up time	T_{start}	—	—	2	—	ms
Crystal input (low)	V_{IL}	—	—	—	0.2V DD	V
Crystal input (high)	V_{IH}	—	0.8V DD	—	—	V
Crystal out (low)	V_{OL}	—	—	—	0.2V DD	V
Crystal out (high)	V_{OH}	—	0.8V DD	—	—	V
External load cap	C_L	1MHz < f_{OUT} < 4MHz	18	30	35	pf
		4MHz < f_{OUT} < 12MHz	10	22	30	pf
		12MHz < f_{OUT} < 16MHz	7	18	22	pf
Feedback resistance	R_{FB}	VDDEXT=5V	0.7	1.0	1.3	MΩ

Figure 149. Crystal/Ceramic Oscillator

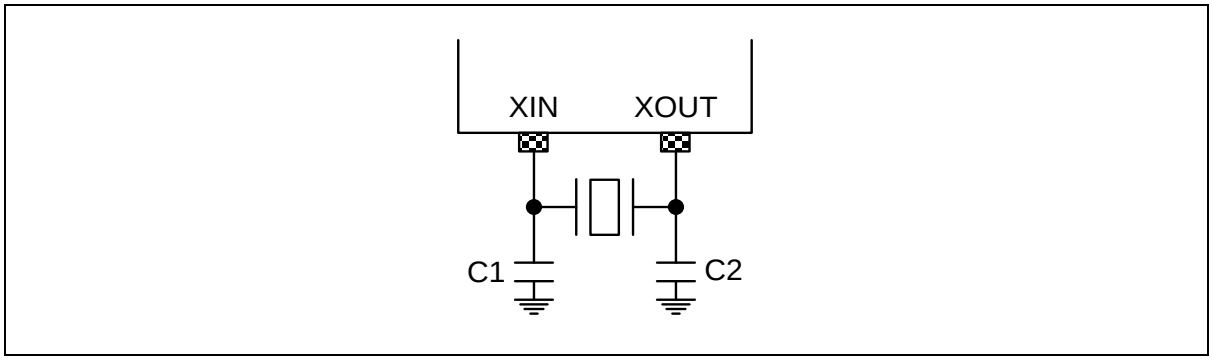


Figure 150. External Clock

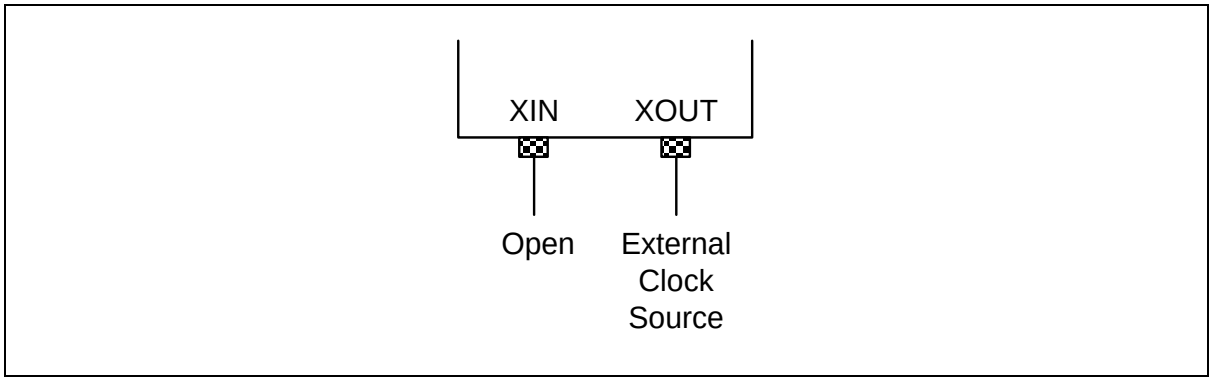
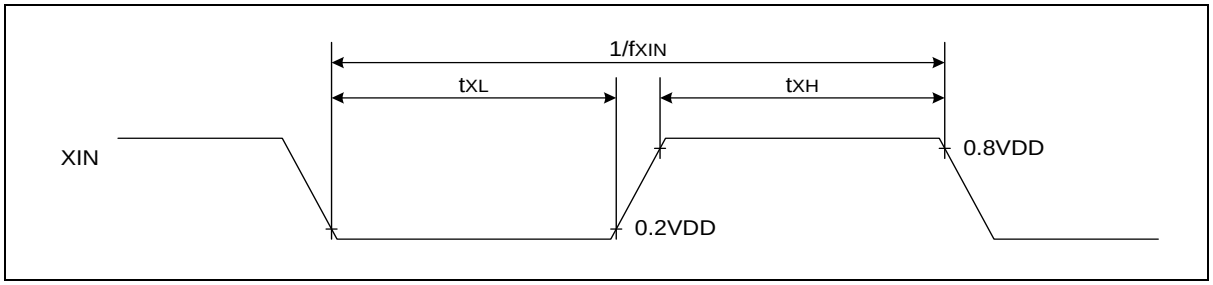


Figure 151. Clock Timing Measurement at XIN



25.10 LSE (sub oscillator) characteristics

Table 118. Operating Condition of LSE

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	2.7	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 119. Sub Oscillator Characteristics

The specifications of the parameters are guaranteed by design.

Oscillator	Parameter	Conditions	Min.	Typ	Max.	Units
Operating current	I_{DD}			3.0	5.0	uA
Power down current	I_{STOP}	—	—	0.2	15.0	nA
Output frequency	f_{SUB}	—	—	32.768	—	KHz
Start-up time	T_{start}	—	—	2	—	s
Crystal input (low)	V_{IL}	—	—	—	0.2VDD	V
Crystal input (High)	V_{IH}	—	0.8VDD	—	—	V
Crystal out (low)	V_{OL}	—	—	—	0.2VDD	V
Crystal out (high)	V_{OH}	—	0.8VDD	—	—	V
External load cap	R_{FB}	—	5	15	35	pF
Feedback resistance	C_L	—	7	12	24	MΩ

Figure 152. Crystal Oscillator

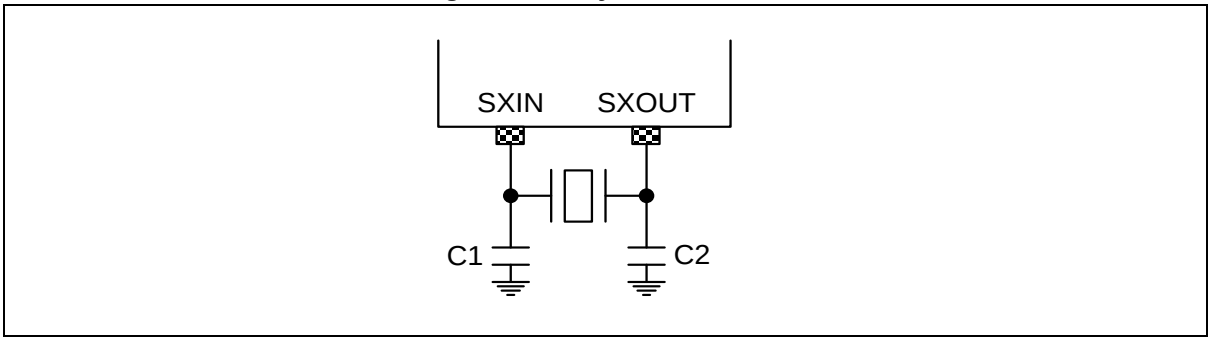
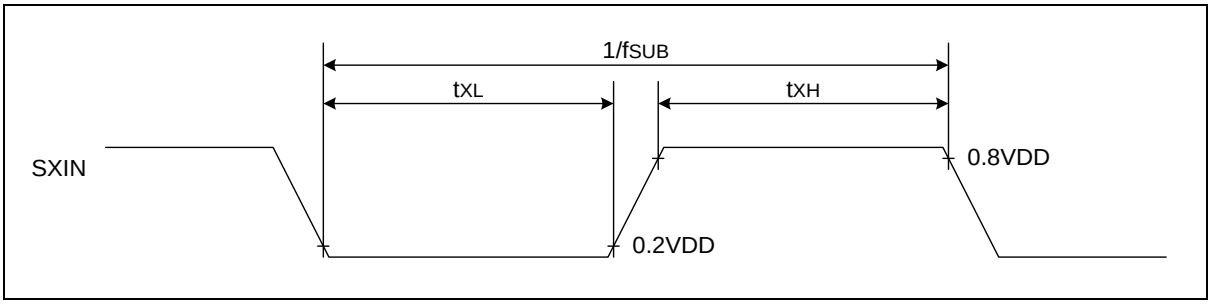


Figure 153. Clock Timing Measurement at SXIN



25.11 PLL electrical characteristics

Table 120. Operating Condition of PLL

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 121. PLL Electrical Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Operating current	I_{DD}	Enable	—	—	1	mA
		Disable	—	5	500	nA
Output frequency	f_{OUT}	—	1	—	48	MHz
Duty	f_{DUTY}		40	—	60	%
VCO Frequency	f_{VCO}		10	—	240	MHz
Frequency Peak-to-Peak Jitter	$f_{JITTER(P-P)}$				500	ps
VCO Linear Range	f_{VCO_LIN}		50	—	150	MHz
Input Frequency	$f_{PLLINCLK}$		4	8	16	MHz
Locking Time*	t_{LOCK}		—	60	100	us
Input Bandwidth	f_{IN}		1	2	3	MHz

NOTE: The tolerance of PLL output frequency is reflected based on PLL input clock source selected by PLLINCLKSEL in the SCU_SCCR[2] register.

25.12 Supply current characteristics

Table 122. Operating Condition of Supply Current

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 123. Supply Current Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions		Typ.	Max	Units
Supply current	I_{DD1} (Run)	f_{XIN} = 8MHz	VDD = 5V	4.0	12.0	mA
		f_{HSI} = 32MHz		10.0	30.0	
		f_{HSI} = 12MHz		3.5	10.0	
		f_{LSI} = 500KHz		200	600	uA
		f_{LSE} = 32.768KHz		140	300	
	I_{DD2} (SLEEP)	f_{XIN} = 8MHz	VDD = 5V	5	15	mA
		f_{HIRC} = 32MHz		6	18	
		f_{HIRC} = 12MHz		2	6	
		F_{LIRC} = 500KHz		180	500	uA
		F_{LSE} = 32.768KHz		130	400	
	I_{DD3} (DEEP-SLEEP)	WDT(WDTRC) = ON, LSIAON=ENABLE NOTE3, LVD = ON, T_A = 25 °C	VDD = 5V	30	—	uA
	I_{DD4} (DEEP-SLEEP)	WDT(WDTRC) = ON, LSIAON=ENABLE NOTE3, LVD = OFF NOTE4, T_A = 25 °C		25	—	uA
	I_{DD5} (DEEP-SLEEP)	WDT(WDTRC) = OFF, LVD = ON, T_A = 25 °C		5	—	uA
	I_{DD6} (DEEP-SLEEP)	WDT(WDTRC) = OFF, LVD = OFF NOTE4, T_A = 25 °C		2	—	uA
		WDT(WDTRC) = OFF, LVD = OFF NOTE4, TA = 85 °C		10	—	uA
		WDT(WDTRC) = OFF, LVD = OFF N OTE4, TA = 105 °C		30	—	uA

NOTES:

1. All supply current items do not include the current of a low frequency internal RC oscillator and a peripheral block.
2. All supply current items include the current of the power-on reset (POR) block.
3. SCU_SMR<LSIAON> bit must be enabled in order to use the WDT as a wake up source in DEEP-SLEEP mode.

4. 'LVD = OFF' indicates that LVR reset function , LVR block and LVI block are disabled.
SCU_RSER<LVRRST> = 0
SCULV_LVRCR<LVREN> = 0x55
SCULV_LVICR<LVIEN> = 0

25.13 I/O Port characteristics

Table 124. Operating Condition of I/O Electrical Characteristics

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

25.13.1 General I/O characteristics

The parameters given in Table 125 for I/O static characteristics are derived from tests performed under the ambient temperature, and VDD supply voltage conditions summarized in Table 124.

Table 125. I/O static characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Input high voltage	V_{IH1}	PA, PB, PC, PD, PE, PF, nBOOT(PB3)	0.8*V_{DD}	—	V_{DD}	V
Input low voltage	V_{IL1}	PA, PB, PC, PD, PE, PF, nBOOT(PB3)	V_{GND}	—	0.2*V_{DD}	V
1.8 V Input high voltage	V_{IH2}	PF5, PF6, PF7 1.8 V input level in PCU_PLSR[2:0]	0.8*1.8	—	1.8	V
1.8 V Input low voltage	V_{IL2}	PF5, PF6, PF7 1.8 V input level in PCU_PLSR[2:0]	V_{GND}	—	0.2*1.8	V
Input high leakage current	I_{IHLKG}	All Input ports	—	—	1	uA
Input low leakage current	I_{ILLKG}	All Input ports	-1	—	—	uA
Pull-up resistor	R_{PU}	All Input pins, nBOOT(PB3)	40	—	70	KΩ
Pull-down resistor	R_{PD}	All Input pins, nBOOT(PB3)	40	—	70	KΩ
I/O pin capacitance	C_{IO}	—	—	—	10	pF

25.13.2 Output driving current

A31G22x GPIOs can sink or source.

In the user application, the number of I/O pins which can drive current must be limited to respect the absolute maximum rating specified in 25.1:

• The sum of the currents sourced by all the I/Os on V_{DD}, plus the maximum run consumption of the MCU sourced on V_{DD}, cannot exceed the absolute maximum rating ΣI_{OH}.

• The sum of the currents sunk by all the I/Os on V_{SS} plus the maximum run consumption of the MCU sunk on V_{SS} cannot exceed the absolute maximum rating ΣI_{OL}.

Table 126. Output voltage Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Output High Level Voltage 1	V_{OH1}	$V_{DD} \geq 1.8$ V, $I_{OH1} = -1$ mA PA[5:0], PA[11:8], PF[4:0], PF[11:8] PE[15:8]	1.44	—	1.80	V
		$V_{DD} \geq 3.3$ V, $I_{OH1} = -2$ mA PA[5:0], PA[11:8], PF[4:0], PF[11:8] PE[15:8]	2.64	—	3.30	V
		$V_{DD} \leq 5.5$ V, $I_{OH1} = -5$ mA PA[5:0], PA[11:8], PF[4:0], PF[11:8] PE[15:8]	4.40	—	5.50	V
Output low level Current 1	V_{OL1}	$V_{DD} \geq 1.8$ V, $I_{OL1} = +1$ mA PA[5:0], PA[11:8], PF[4:0], PF[11:8] PE[15:8]	0.00	—	0.36	V
		$V_{DD} \geq 3.3$ V, $I_{OL1} = +3$ mA PA[5:0], PA[11:8], PF[4:0], PF[11:8] PE[15:8]	0.00	—	0.66	V
		$V_{DD} \leq 5.5$ V, $I_{OL1} = +7$ mA PA[5:0], PA[11:8], PF[4:0], PF[11:8] PE[15:8]	0.00	—	1.10	V
Output High Level Voltage 2	V_{OH2}	$V_{DD} \geq 1.8$ V, $I_{OH2} = -1$ mA PA[7:6], PB[15:3], PC0, PC1, PC[12:6], PD0, PD1, PD2, PD3, PD4, PD5	1.44	—	1.80	V
		$V_{DD} \geq 3.3$ V, $I_{OH2} = -3$ mA PA[7:6], PB[15:3], PC0, PC1, PC[12:6], PD0, PD1, PD2, PD3, PD4, PD5	2.64	—	3.30	V
		$V_{DD} \leq 5.5$ V, $I_{OH2} = -8$ mA PA[7:6], PB[15:3], PC0, PC1, PC[12:6], PD0, PD1, PD2, PD3, PD4, PD5	4.40	—	5.50	V
Output low level Current 2	V_{OL2}	$V_{DD} \geq 1.8$ V, $I_{OL2} = +1$ mA PA[7:6], PB[15:3], PC0, PC1, PC[12:6], PD0, PD1,	0.00	—	0.36	V

		PD2, PD3, PD4, PD5				
		V_{DD} ≥ 3.3 V, I_{OL2} = +3 mA PA[7:6], PB[15:3], PC0, PC1, PC[12:6], PD0, PD1, PD2, PD3, PD4, PD5	0.00	—	0.66	V
		V_{DD} ≤ 5.5 V, I_{OL2} = +7 mA PA[7:6], PB[15:3], PC0, PC1, PC[12:6], PD0, PD1, PD2, PD3, PD4, PD5	0.00	—	1.10	V
Output High Level Voltage 3	V_{OH3}	V_{DD} ≥ 1.8 V, I_{OH3} = -2 mA PE0, PE1, PE[7:2]	1.44	—	1.80	V
		V_{DD} ≥ 3.3 V, I_{OH3} = -8 mA PE0, PE1, PE[7:2]	2.64	—	3.30	V
		V_{DD} ≤ 5.5 V, I_{OH3} = -19 mA PE0, PE1, PE[7:2]	4.40	—	5.50	V
Output low level Current 3	V_{OL3}	V_{DD} ≥ 1.8 V, I_{OL3} = +2 mA PE0, PE1, PE[7:2]	0.00	—	0.36	V
		V_{DD} ≥ 3.3 V, I_{OL3} = +6 mA PE0, PE1, PE[7:2]	0.00	—	0.66	V
		V_{DD} ≤ 5.5 V, I_{OL3} = +14 mA PE0, PE1, PE[7:2]	0.00	—	1.10	V
Output High Level Voltage 4	V_{OH4}	V_{DD} ≥ 1.8 V, I_{OH4} = Don't Care PF5, PF6, PF7	1.44	—	1.80	V
		V_{DD} ≥ 3.3 V, I_{OH4} = Don't Care PF5, PF6, PF7	2.64	—	3.30	V
		V_{DD} ≤ 5.5 V, I_{OH4} = Don't Care PF5, PF6, PF7	4.40	—	5.50	V
Output low level Current 4	V_{OL4}	V_{DD} ≥ 1.8 V, I_{OL4} = +1 mA PF5, PF6, PF7	0.00	—	0.36	V
		V_{DD} ≥ 3.3 V, I_{OL4} = +3 mA PF5, PF6, PF7	0.00	—	0.66	V
		V_{DD} ≤ 5.5 V, I_{OL4} = +7 mA PF5, PF6, PF7	0.00	—	1.10	V
Output High Level Voltage 5	V_{OH5}	V_{DD} ≥ 1.8 V, I_{OH5} = -1 mA PC2, PC3, PC4	1.44	—	1.80	V
		V_{DD} ≥ 3.3 V, I_{OH5} = -2 mA PC2, PC3, PC4	2.64	—	3.30	V
		V_{DD} = 5.5 V, I_{OH5} = -5 mA PC2, PC3, PC4	4.40	—	5.50	V
		Strength On[NOTE1] V_{DD} ≥ 1.8 V, I_{OH5} = -2 mA PC2, PC3, PC4	1.44	—	1.80	V
		Strength On[NOTE1] V_{DD} ≥ 3.3 V, I_{OH5} = -7 mA PC2, PC3, PC4	2.64	—	3.30	V

		Strength On[NOTE1] V_{DD} ≤ 5.5 V, I_{OH5} = -16 mA PC2, PC3, PC4	4.40	—	5.50	V
Output low level Current 5	V_{OL5}	V_{DD} ≥ 1.8 V, I_{OL5} = +1 mA PC2, PC3, PC4	0.00	—	0.36	V
		V_{DD} ≥ 3.3 V, I_{OL5} = +3 mA PC2, PC3, PC4	0.00	—	0.66	V
		V_{DD} ≤ 5.5 V, I_{OL5} = +7 mA PC2, PC3, PC4	0.00	—	1.10	V
		Strength On[NOTE1] V_{DD} ≥ 1.8 V, I_{OL5} = +2 mA PC2, PC3, PC4	0.00	—	0.36	V
		Strength On[NOTE1] V_{DD} ≥ 3.3 V, I_{OL5} = +8 mA PC2, PC3, PC4	0.00	—	0.66	V
		Strength On[NOTE1] V_{DD} ≤ 5.5 V, I_{OL5} = +19 mA PC2, PC3, PC4	0.00	—	1.10	V
Output High Level Voltage 6	V_{OH6}	V_{DD} ≥ 1.8 V, I_{OH6} = -1 mA PB0, PB1, PB2	1.44	—	1.80	V
		V_{DD} ≥ 3.3 V, I_{OH6} = -2 mA PB0, PB1, PB2	2.64	—	3.30	V
		V_{DD} ≤ 5.5 V, I_{OH6} = -5 mA PB0, PB1, PB2	4.40	—	5.50	V
		Strength On[NOTE1] V_{DD} ≥ 1.8 V, I_{OH6} = -2 mA PB0, PB1, PB2	1.44	—	1.80	V
		Strength On[NOTE1] V_{DD} ≥ 3.3 V, I_{OH6} = -7 mA PB0, PB1, PB2	2.64	—	3.30	V
		Strength On[NOTE1] V_{DD} ≤ 5.5 V, I_{OH6} = -16 mA PB0, PB1, PB2	4.40	—	5.50	V
Output low level Current 6	V_{OL6}	V_{DD} ≥ 1.8 V, I_{OL6} = +1 mA PB0, PB1, PB2	0.00	—	0.36	V
		V_{DD} ≥ 3.3 V, I_{OL6} = +5 mA PB0, PB1, PB2	0.00	—	0.66	V
		V_{DD} ≤ 5.5 V, I_{OL6} = +11 mA PB0, PB1, PB2	0.00	—	1.10	V
		Strength On[NOTE1] V_{DD} ≥ 1.8 V, I_{OL6} = +2 mA PB0, PB1, PB2	0.00	—	0.36	V
		Strength On[NOTE1] V_{DD} ≥ 3.3 V, I_{OL6} = +8 mA PB0, PB1, PB2	0.00	—	0.66	V
		Strength On[NOTE1] V_{DD} ≤ 5.5 V, I_{OL6} = +19 mA	0.00	—	1.10	V

		PB0, PB1, PB2				
Output low level Current 7	V_{OL7}	High Current Output (Sink)[NOTE2] V_{DD} ≥ 1.8 V, I_{OL7} = +28 mA PD2, PD3, PD4, PD5, PE2, PE3, PE4, PE5, PE6, PE7	0.00	—	0.36	V
		High Current Output (Sink)[NOTE2] V_{DD} ≥ 3.3 V, I_{OL7} = +53mA PD2, PD3, PD4, PD5, PE2, PE3, PE4, PE5, PE6, PE7	0.00	—	0.66	V
		High Current Output (Sink)[NOTE2] V_{DD} ≤ 5.5 V, I_{OL7} = +73mA PD2, PD3, PD4, PD5, PE2, PE3, PE4, PE5, PE6, PE7	0.00	—	1.10	V

NOTES:

1. GPIO's strength function is controlled by Px_STR register.
2. The sink type high current output driving of GPIOs is controlled by PCU2_ISEGPEN and PCU2_ISEGR, PCU2_ISEGIR registers.

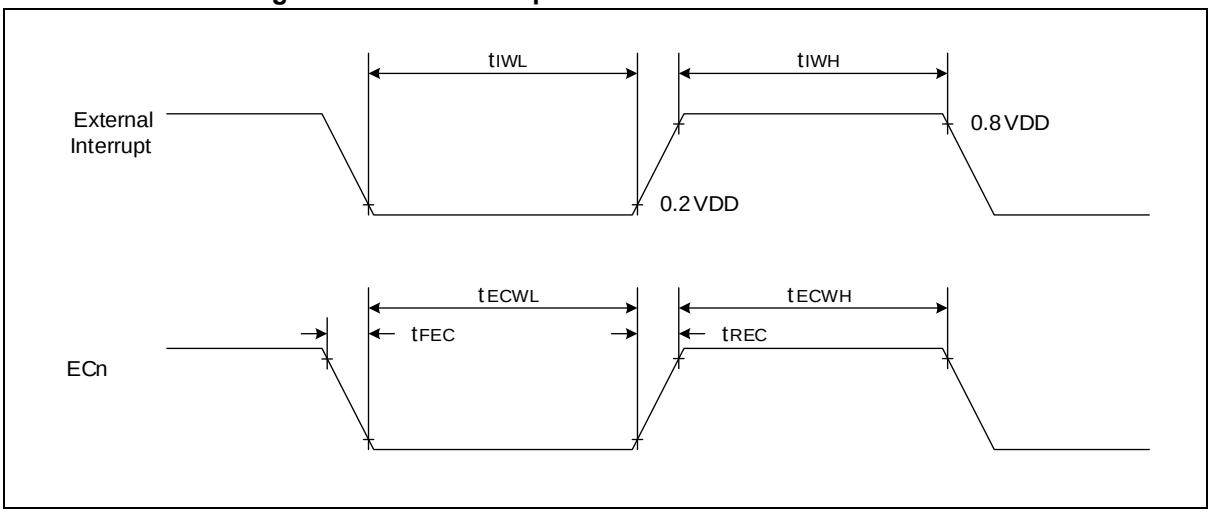
25.13.3 I/O AC characteristics

The definition and values of external input AC characteristics are given in Table 127 and Figure 154.

Table 127. External Input AC Characteristics

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Interrupt input high low width	t_{IWH}, t_{IWL}	All external interrupts 1.8 V≤V_{DD}≤ 5.5 V -40 °C≤T_A ≤105 °C	100	—	—	ns
External counter input high low pulse width	t_{ECWH}, t_{ECWL}	ECn, All external counter input 1.8 V≤V_{DD}≤ 5.5 V -40 °C≤T_A ≤105 °C	100	—	—	
External counter transition time	t_{REC}, t_{FEC}	ECn, All external counter input 1.8 V≤V_{DD}≤ 5.5 V -40 °C≤T_A ≤105 °C	—	—	20	

Figure 154. External Input AC Characteristics definitions



25.14 nRESET pin characteristics

The nRESET pin input is connected to a internal permanent pull-up resistor, R_{PU}.

The parameters given in Table 128 are derived from tests performed under the ambient temperature and VDD supply voltage conditions summarized in Table 124.

Table 128. nRESET pin Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
nRESET input high voltage	$V_{IH(nRST)}$	1.8 V ≤ V_{DD} ≤ 5.5 V	0.8*V_{DD}	—	V_{DD}	V
nRESET input low voltage	$V_{IL(nRST)}$	1.8 V ≤ V_{DD} ≤ 5.5 V	V_{GND}	—	0.2*V_{DD}	V
nRESET input hysteresis	$V_{HYS(nRST)}$	1.8 V ≤ V_{DD} ≤ 5.5 V	100	200	—	mV
nRESET Pull-up resistor	$R_{PU(nRST)}$	V_{IN} = V_{SS}	150	250	400	KΩ
nRESET input low width	t_{RST}	1.8 V ≤ V_{DD} ≤ 5.5 V	10	—	—	us

Figure 155. nRESET pin low width definitions

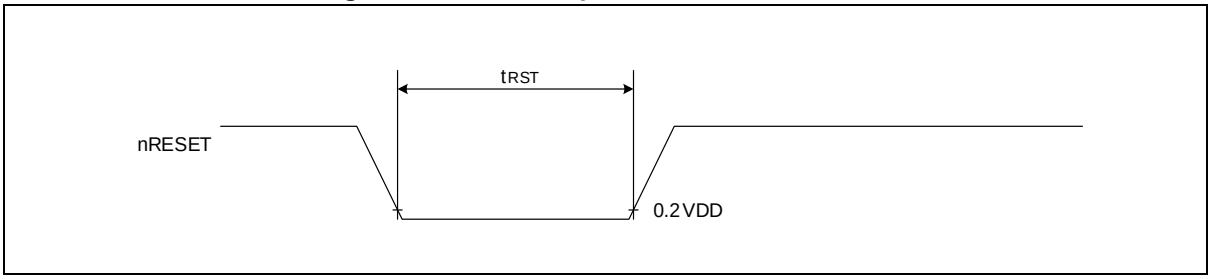
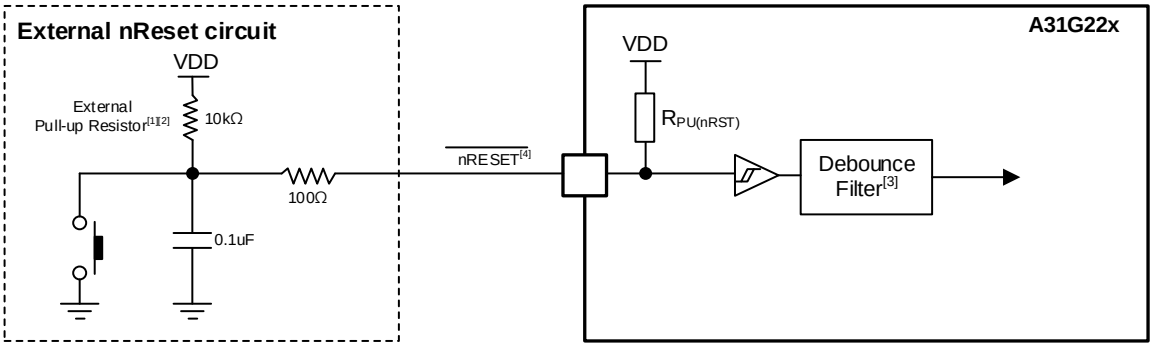


Figure 156. Recommended nRESET pin protection



NOTES:

[1] It is recommended to configure the circuit by connecting an external pull-up resistor to the nRESET pin.
[2] The external pull-up resistor should be less than the internal pull-up R_{PU} of nRESET.
[3] It is recommended to activate the nRESET debounce filter to prevent external noise from entering.
[4] The user must make sure that the level of the RESET pin can drop below the $V_{IL(nRST)}$ maximum level.

25.15 UART characteristics

Table 129. Operating Condition of UART

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

25.16 SPI characteristics

The parameters given in Table 131 for SPI are derived from tests performed under the ambient temperature, and f_{PCLK} frequency and V_{DD} supply voltage conditions summarized in Table 130.

Table 130. Operating Condition of SPIn (n = 20, 21)

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 131. SPIn characteristics (n = 20, 21)

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Condition	Min.	Typ.	Max.	Units
SPI clock frequency with strength on	f_{SCK1}	SPI20, SPI21 V_{DD}≤ 1.8 V	—	—	12	MHz
	f_{SCK2}	SPI20, SPI21 V_{DD}≤ 3.3 V	—	—	12	MHz
	f_{SCK3}	SPI20, SPI21 V_{DD}≤ 5.5 V	—	—	12	MHz
SPI clock frequency with strength off	f_{SCK4}	SPI20, SPI21 V_{DD}≤ 1.8 V	—	—	5	MHz
	f_{SCK5}	SPI20, SPI21 V_{DD}≤ 3.3 V	—	—	8	MHz
	f_{SCK6}	SPI20, SPI21 V_{DD}≤ 5.5 V	—	—	10	MHz
Duty cycle of SPI frequency (SCK)	Duty	Slave Mode	30	50	70	%

25.17 USART SPI characteristics

Table 132. Operating Condition of USART SPI

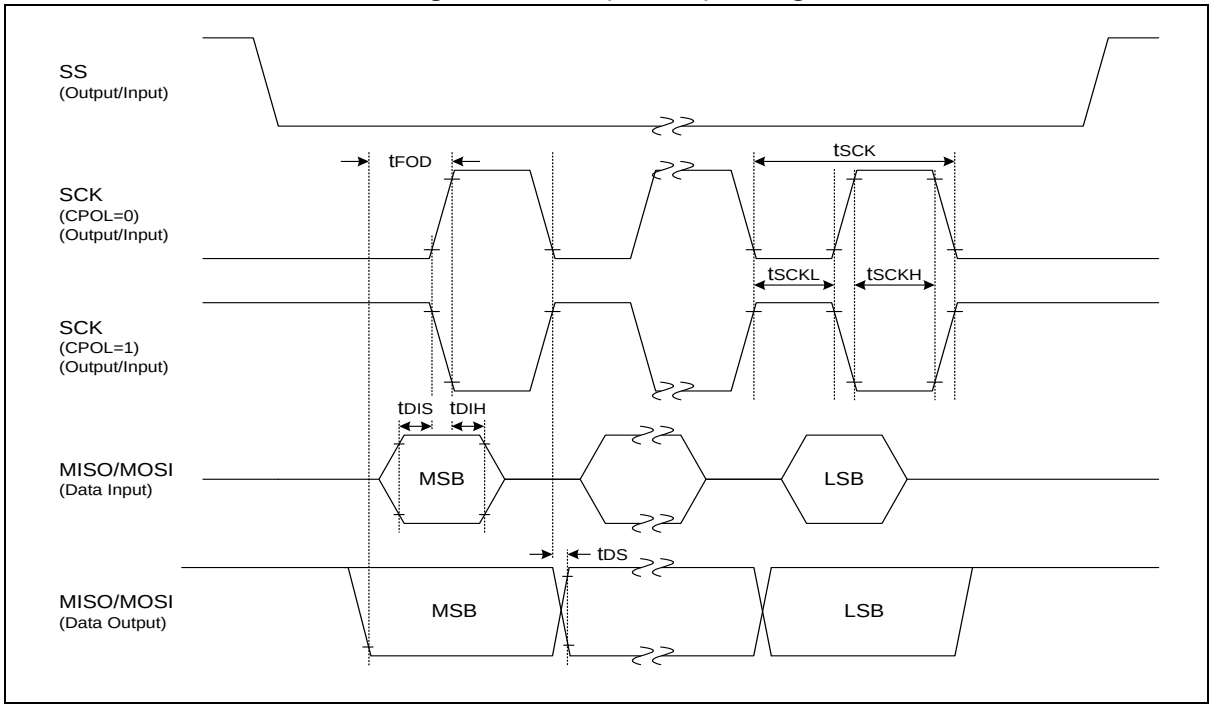
Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 133. USART SPI Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Output clock pulse period	t_{SCK}	Internal SCK source	400	—	—	ns
Input clock pulse period		External SCK source	400	—	—	
Output clock high, low pulse width	t_{SCKH}, t_{SCKL}	Internal SCK source	180	—	—	
Input clock high, low pulse width		External SCK source	180	—	—	
First output clock delay time	t_{FOD}	Internal/external SCK source	200	—	—	
Output clock delay time	t_{DS}	—	—	—	100	
Input setup time	t_{DIS}	—	180	—	—	
Input hold time	t_{DIH}	—	180	—	—	

Figure 157. SPI (USART) Timing



25.18 USART UART timing characteristics

Table 134. Operating Condition of USART UART

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 135. USART UART Timing Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Min.	Typ	Max.	Units
Serial port clock cycle time	t_{SCK}	1250	t_{CPU} x 16	1650	ns
Output data setup to clock rising edge	t_{S1}	590	t_{CPU} x 13	—	
Clock rising edge to input data valid	t_{S2}	—	—	590	
Output data hold after clock rising edge	t_{H1}	t_{CPU} – 50	t_{CPU}	—	
Input data hold after clock rising edge	t_{H2}	0	—	—	
Serial port clock High, Low level width	t_{HIGH}, t_{LOW}	470	t_{CPU} x 8	970	

Figure 158. Waveform of UART Timing Characteristics

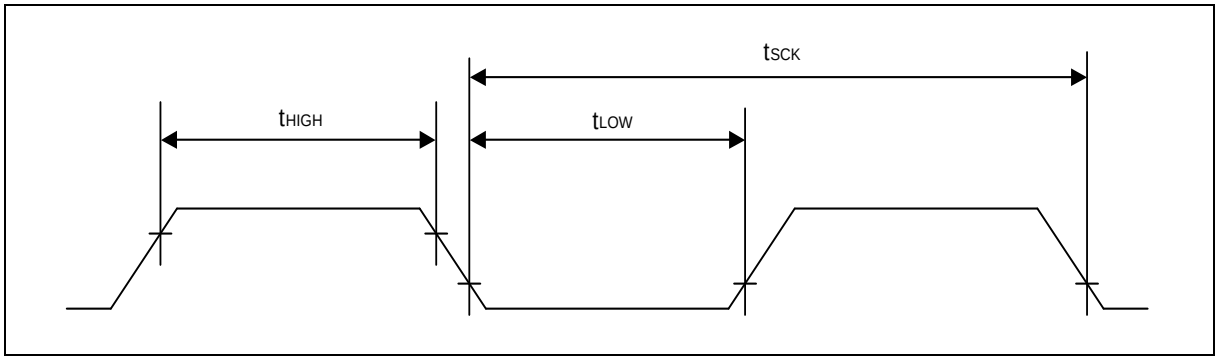
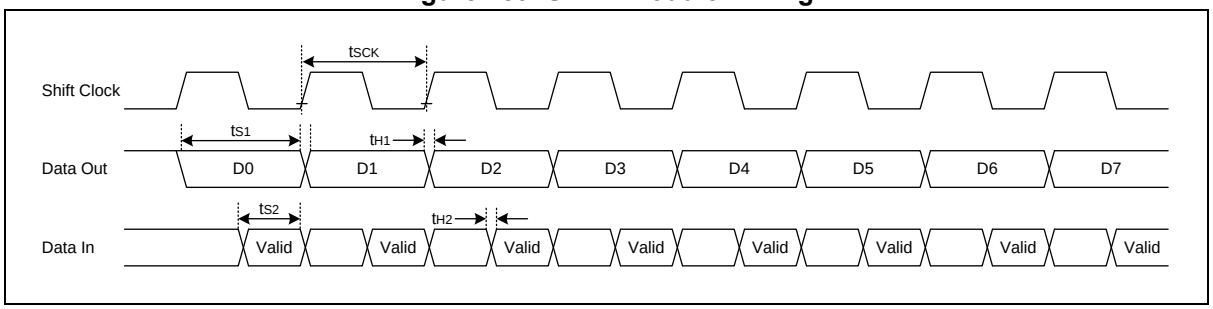


Figure 159. UART Module Timing



25.19 I2C characteristics

Table 136. Operating Condition of I2C

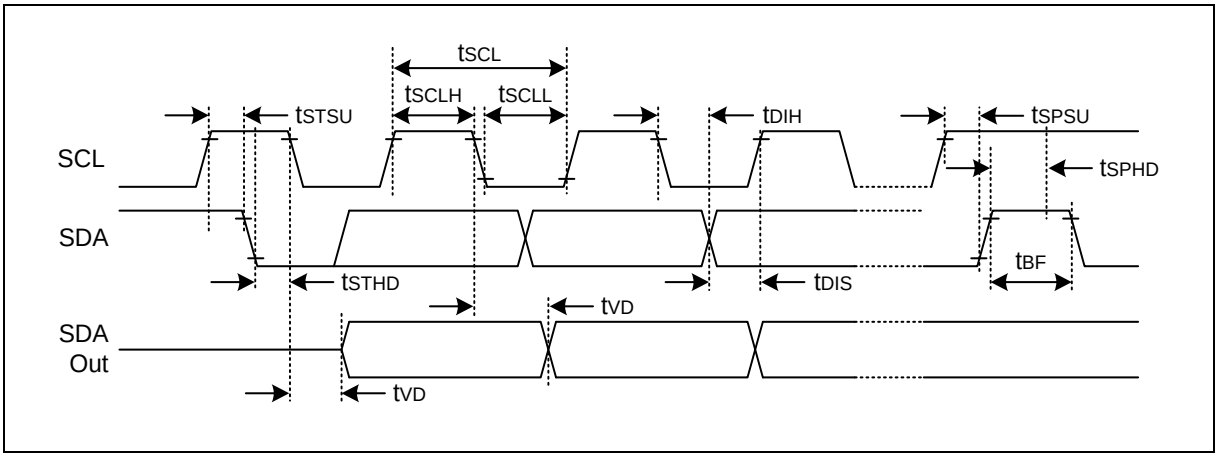
Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 137. I2C Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Standard		Fast		Units
		Min.	Max.	Min.	Max.	
Clock frequency	t_{SCL}	0	100	0	400	KHz
Clock high pulse width	t_{SCLH}	4.0	—	0.6	—	us
clock low pulse width	t_{SCLL}	4.7	—	1.3	—	
Bus free time	t_{BF}	4.7	—	1.3	—	
Start condition setup time	t_{STSU}	4.7	—	0.6	—	
Start condition hold time	t_{STHD}	4.0	—	0.6	—	
Stop condition setup time	t_{SPSU}	4.0	—	0.6	—	
Stop condition hold time	t_{SPHD}	4.0	—	0.6	—	
Output valid from clock	t_{VD}	0	—	0	—	
Data input hold time	t_{DIH}	0	—	0	1.0	
Data input setup time	t_{DIS}	250	—	100	—	ns

Figure 160. I2C Timing



25.20 Internal Code flash characteristics

Table 138. Operating Condition of Internal Code Flash

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 139. Internal Code Flash Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Max. available clock frequency	-	0-wait	-	-	20	MHz
Reset Cycle Time	$f_{RSTBUSY}$	—	5.6	8	10.4	us
Fuse Program Cycle Time	$f_{FRDBUSY}$	—	4.2	6	7.8	us
Normal Program Cycle Time	$t_{PGMBUSY}$	—	21	30	42	us
Normal Page Erase Cycle Time	$t_{PERSBUSY}$	—	2.8	4	5.2	ms
Sector Erase Cycle Time	$t_{SERSBUSY}$	—	2.8	4	5.2	ms
Chip Erase Cycle Time	$t_{MERSBUSY}$	—	5.6	8	10.4	ms
Endurance of write/erase	N_{FWE}	T_A=25 ˚C, Page unit	10,000	—	—	Times
Retention time	t_{FRT}	—	10	—	—	Years

25.21 Internal Data flash characteristics

Table 140. Operating Condition of Internal Data Flash

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	1.8	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 141. Internal Data Flash Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
Max. available clock frequency	-	0-wait	-	-	20	MHz
Reset Cycle Time	$f_{RSTBUSY}$	—	5.6	8	10.4	us
Fuse Program Cycle Time	$f_{FRDBUSY}$	—	4.2	6	7.8	us
Normal Program Cycle Time	$t_{PGMBUSY}$	—	21	30	42	us
Normal Page Erase Cycle Time	$t_{PERSBUSY}$	—	2.8	4	5.2	ms
Sector Erase Cycle Time	$t_{SERSBUSY}$	—	2.8	4	5.2	ms
Chip Erase Cycle Time	$t_{MERSBUSY}$	—	5.6	8	10.4	ms
Endurance of write/erase	N_{FWE}	T_A=25 ˚C, Page unit	100,000	—	—	Times
Retention time	t_{FRT}	—	10	—	—	Years

25.22 ADC characteristics

Table 142. Operating Condition of ADC

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	2.1*	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

NOTE: If 12-bit ADC uses voltage of less than 2.0V, measurement error may be big. To avoid this, it is recommended to use voltage ranging from 2.1V to 5.5V for the ADC which is used for precision sensing of analog voltage.

Table 143. ADC Electrical Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Condition	Min.	Typ.	Max.	unit
Resolution	—	—	—	—	12	Bit
Number of Channel	—	—	14	18	—	CH
Analog input range	V_{AVREF}	V_{AVREF}=V_{DD} V_{SS}=V_{GND}	VSS	—	AVREF	V
Operating Current	I_{DD}	V_{DD} = 5.0V MCLK=20MHz Input buffer OFF	—	2.4	—	mA
		V_{DD} = 5.0V MCLK=20MHz Input buffer ON	—	3.0	—	mA
Standby Current	I_{ST}	—	—	50	—	nA
Full Scale Input Range	V_{IN}	V_{AVDD}=V_{DD}	V_{GND}		V_{AVDD}	V
Differential nonlinearity	DNL	—	—	±1	±2	LSB
Integral nonlinearity	INL	—	—	±2	±4	LSB
Zero offset error	ZOE	—	—	±2	—	LSB
Full scale error	FSE	—	—	±2	—	LSB
Input clock frequency	MCLK	VDD > 3.3V	—	—	20	MHz
		VDD > 2.7V	—	—	13.3	
		VDD > 2.1V	—	—	3	
Conversion time	t_{CONV}	TSMP_I<4:0> 2~32*MCLK	15*MCLK	—	45*MCLK	us
Conversion frequency ratio	C_{RATE}	VDD > 3.3V	—	—	1.0	Msps
		VDD > 2.7V	—	—	0.7	
		VDD > 2.1V	—	—	0.15	
Sampling time	t_{SAMPLE}	VDD > 3.3V	—	—	10	us
		VDD > 2.7V	—	—	10	
		VDD > 2.1V	—	—	10	

25.23 DAC characteristics

Table 144. Operating Condition of DAC

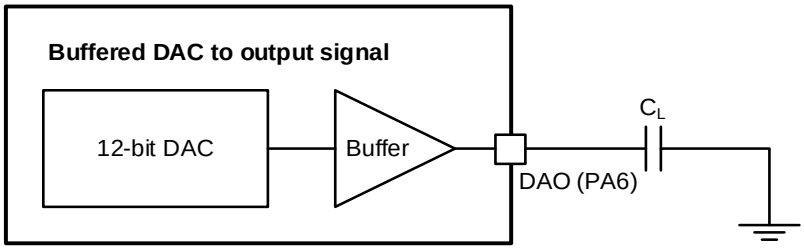
Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	2.7	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 145. DAC Electrical Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Condition	Min.	Typ.	Max.	unit
Resolution			—	—	12	Bit
DAC Output Voltage	D_{AOUT}	0x0A3(min) ~ 0xF5B(max) (@AVDD=5V) 0x12F(min) ~ 0xECF(max) (@AVDD=2.7V)	0.2	—	VDD-0.2	V
Operating Current	$I_{AVDD,rms}$	No load, middle code(0x800)	—	0.95	1.48	mA
DNL	—	No R-Load, DAC Output Voltage	—	2	8	LSB
INL	—	Integral non linearity (difference between measured value at Code i and the value at Code i on a line drawn between Code 0 and last Code 4095)	—	4	8	LSB
Offset	—	Offset Error is difference between measured value at Code (0x800) and the ideal value(AVDD/2)	—	—	10	mV
Conversion Time	$t_{SETTLINB}$	—	—	2	4	us
Capacitive load	C_L	Maximum capacitive load at DAC_OUT pin when Buffer is enabled.	—	—	50	pF

Figure 161. 12-bit buffered DAC



NOTE: The DAC integrates an output buffer that can be used to reduce the output impedance and to drive external loads directly without the use of an external current load. The buffer can be enabled by configuring the DAC_BUF_EN[10] bit in the DAC_CR register.

25.24 Comparator characteristics

Table 146. Operating Condition of Comparator

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	2.0	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 147. Comparator Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ	Max.	Units
Operating Current	$I_{DD(RMS)}$	-	—	70	100	uA
Input Voltage Range	V_{ICM}	-	V_{GND}+50 mV	—	V_{DD}-50mV	V
Propagation Delay	tPD (t_{PHL}, t_{PLH})	V_{OV} > 10mV	—	0.5	2	us
Input Offset	V_{OS}	VDDEDXT=4.5V, VIN=1/2 VDDEXT, Before Offset Calibration	—	±10	±20	mV
		VDDEDXT=4.5V, VIN=1/2 VDDEXT, After Offset Calibration	—	—	±5	mV
Hysteresis	V_{HYS}	V_{DD}=4.5V, HYSSEL=0	—	±5	±25	mV
		V_{DD}=4.5V, HYSSEL=1	—	±20	±60	mV

25.25 LCD driver characteristics

Table 148. Operating Condition of LCD Driver

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	2.7	5.0	5.5	V
Operating Temperature	T_A	-40	25	105	℃

Table 149. LCD Driver Characteristics

The specifications of the parameters are guaranteed by design.

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units
LCD Voltage	VLC0	LCD contrast = DISABLED, 1/4 bias	—	—	—	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x00	Typ.x0.94	V_{DD}x16/31	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x01	Typ.x0.94	V_{DD}x16/30	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x02	Typ.x0.94	V_{DD}x16/29	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x03	Typ.x0.94	V_{DD}x16/28	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x04	Typ.x0.94	V_{DD}x16/27	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x05	Typ.x0.94	V_{DD}x16/26	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x06	Typ.x0.94	V_{DD}x16/25	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x07	Typ.x0.94	V_{DD}x16/24	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x08	Typ.x0.94	V_{DD}x16/23	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x09	Typ.x0.94	V_{DD}x16/22	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x0A	Typ.x0.94	V_{DD}x16/21	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load	Typ.x0.94	V_{DD}x16/20	Typ.x1.06	V

		VLCD[3:0] = 0x0B				
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x0C	Typ.x0.94	V_{DD}x16/19	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x0D	Typ.x0.94	V_{DD}x16/18	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x0E	Typ.x0.94	V_{DD}x16/17	Typ.x1.06	V
	VLC0	LCD contrast = ENABLED, 1/4 bias, No panel load VLCD[3:0] = 0x0F	Typ.x0.94	V_{DD}x16/16	Typ.x1.06	V
LCD Mid Bias Voltage[NOTE]	VLC1	V_{DD} = 2.7V to 5.5V, LCD clock = 0Hz, 1/4 bias, No panel load	Typ-0.2	3/4xVLC0	Typ+0.2	V
	VLC2		Typ-0.2	2/4xVLC0	Typ+0.2	V
	VLC3		Typ-0.2	1/4xVLC0	Typ+0.2	V
LCD Driver Output Impedance	RLO	VLCD = 3.0V	–	5	10	kΩ
LCD Bias Dividing Resistor	RLCD1	1/4 bias, T_A = 25°C	7.5	10	12.5	kΩ
	RLCD2		38	50	62	
	RLCD3		60	80	100	
	RLCD4		180	240	300	

NOTE: It is middle output voltage when the VDD and the VLC0 node are connected.

25.26 TS characteristics

The specifications of the parameters are guaranteed by design.

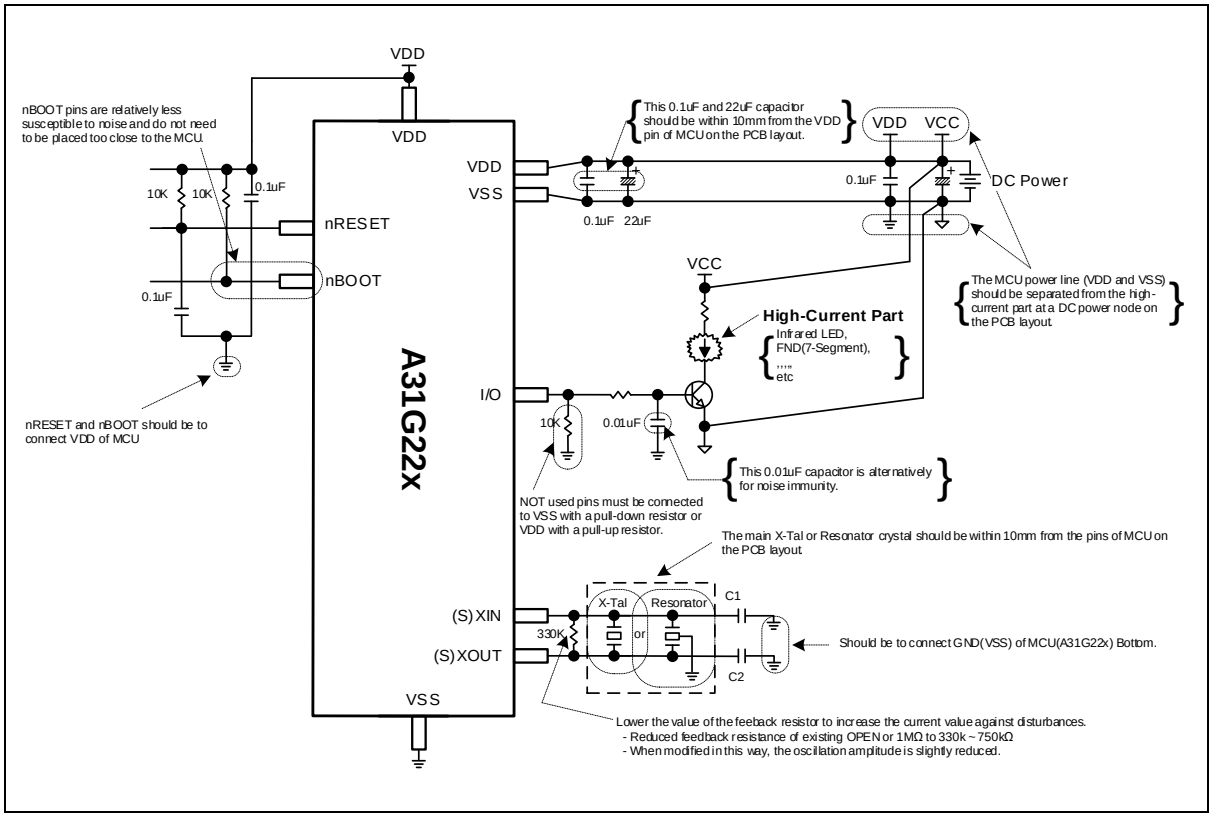
Table 150. Operating condition and characteristics of Temp Sensor

Parameter	Symbol	Min.	Typ.	Max.	Units
Operating Voltage	V_{DD}	2.7	5.0	5.5	V
Sensing Temperature	T_S	-40	25	105	℃
Temperature Accuracy Error	$T_{SENSEACC}$	—	±10	—	℃

26 Recommended circuit and layouts

26.1 Recommended circuit layout

Figure 162. A31G22x recommended circuit layout



27 Development tools

27.1 Compiler

ABOV semiconductor does not provide any compiler for A31G22x. However, since A31G22x have ARM's high-speed 32-bit Cortex-M0+ Cores for their CPU, you can use all kinds of third party's standard compiler such as Keil C Compiler. These compilers' output debug information can be integrated with our A-Link and A-Link Pro. Please visit our website www.abovsemi.com for more information regarding the A-Link and A-Link Pro.

27.2 Debugger

The A-Link and A-Link Pro support ABOV Semiconductor's A31G22x MCU emulation in SWD Interface. The A-Link and A-Link Pro use two wires interfacing between PC and MCU, which is attached to user's system. The A-Link and A-Link Pro can read or change the value of MCU's internal memory and I/O peripherals. In addition, the A-Link and A-Link Pro control MCU's internal debugging logic. This means A-Link and A-Link Pro control emulation, step run, monitoring and many more functions regarding debugging.

The A-Link and A-Link Pro run underneath MS operating system such as MS-Windows NT/ 2000/ XP/ Vista/ 7/ 8/ 8.1/ 10 (32-bit, 64-bit).

Programming information using the A-Link and A-Link Pro are provided in Figure 163. More detailed information about the A-Link and A-Link Pro, please visit our website www.abovsemi.com and download the debugger S/W and documents.

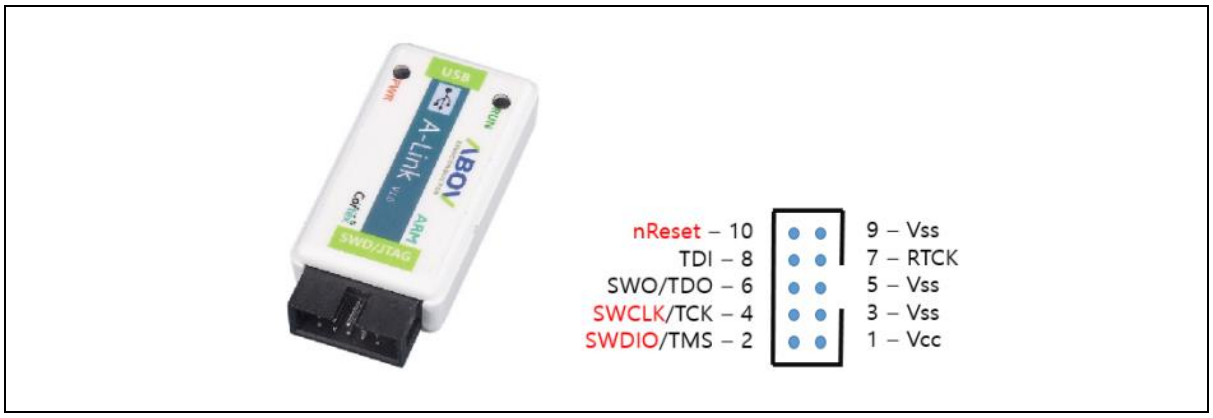


Figure 163. A-Link and Pin Descriptions

27.3 Programmer

27.3.1 E-PGM+

E-PGM+ is a single programmer, and allows a user to program on the device directly.

- Support ABOV devices
- 2~5 times faster than S-PGM+
- Main controller: 32-bit MCU @72MHz
- Buffer memory: 1MB

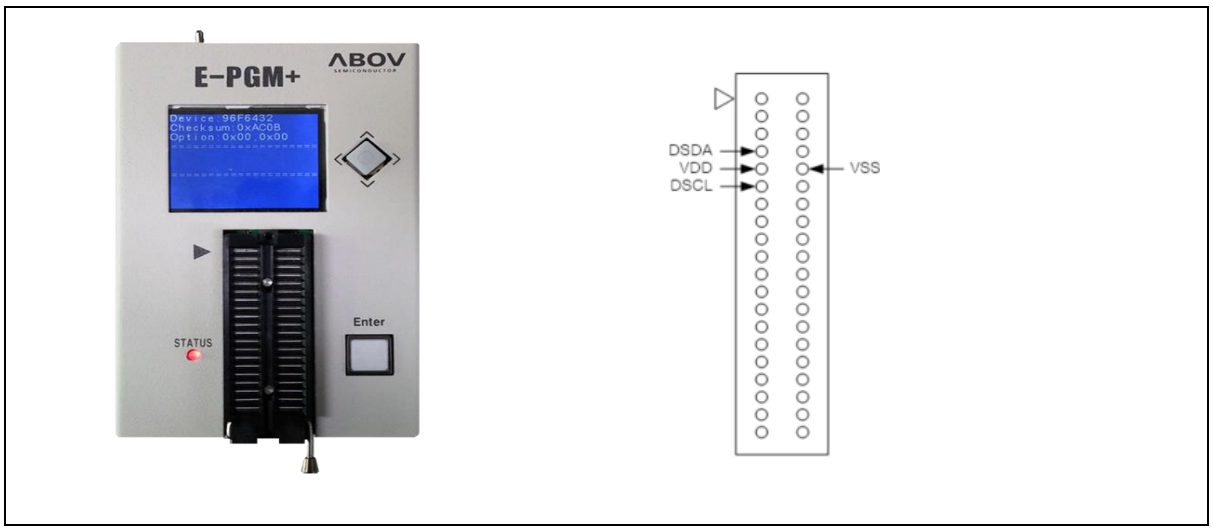


Figure 164. E-PGM+ (Single Writer) and Pin Descriptions

27.3.2 Gang programmer

E-Gang4 and E-Gang6 allows a user to program on multiple devices at a time. They run not only in PC controlled mode but also in standalone mode without PC control. USB interface is available and it is easy to connect to the handler.

Figure 165. E-Gang4 and E-Gang6 (for Mass Production)

27.4 SWD debug mode and E-PGM+ connection

Connections for SWD debugger interface or EPGM+ is described in "*Figure 10. Connection Diagram of E-PGM+ and SWD Port*".

28 Package information

This chapter provides A31G22x series package information.

28.1 80 LQFP (14x14) package information

Figure 166. 80 LQFP (14x14) Package

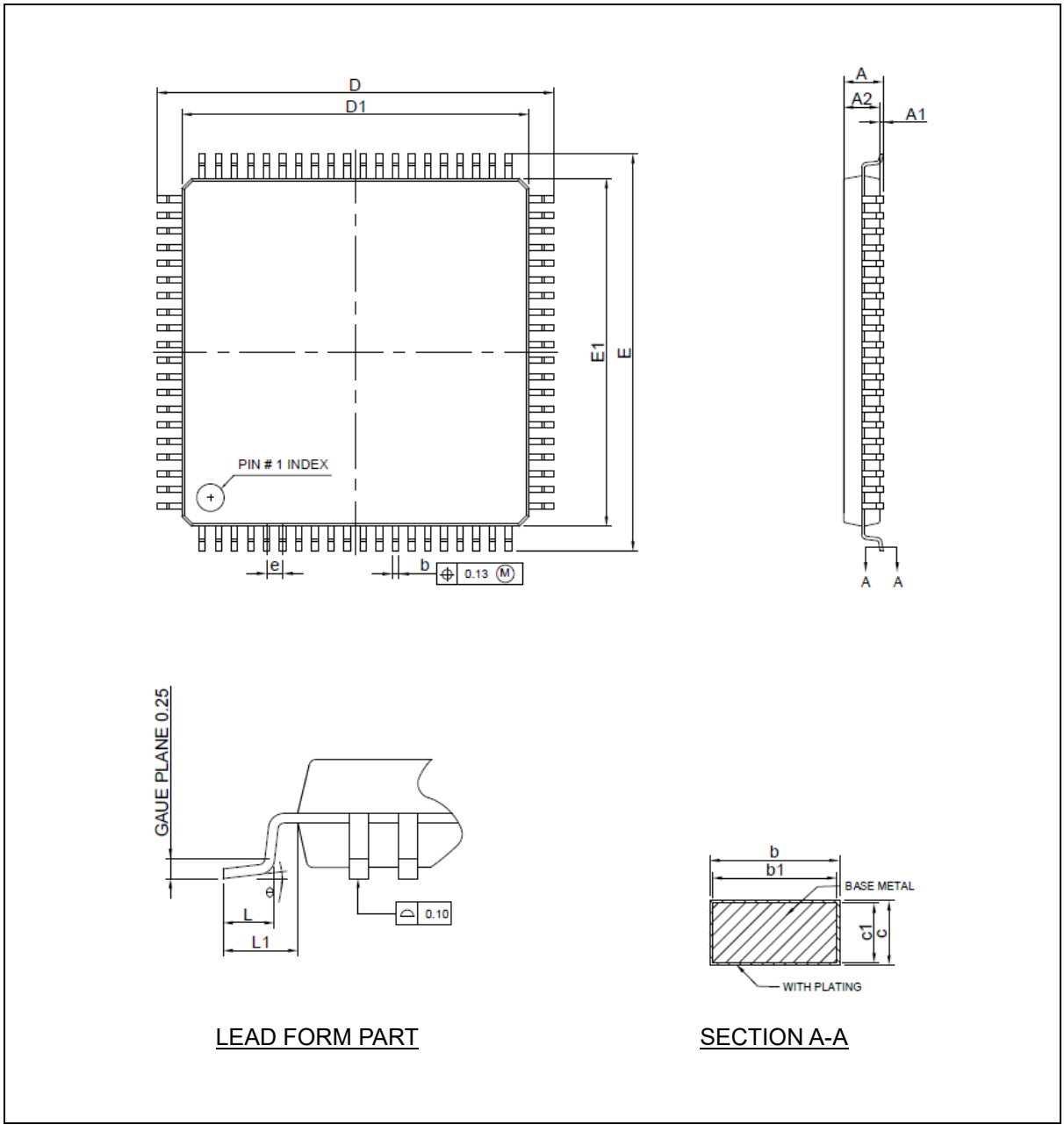


Table 151. 80 LQFP (14 x 14) Package Mechanical Data

Symbol	Dimension (mm)		
	Min.	Nom.	Max.
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.22	0.32	0.38
b1	0.22	0.30	0.33
c	0.09	—	0.20
c1	0.09	—	0.16
D	15.80	16.00	16.20
D1	13.80	14.00	14.20
E	15.80	16.00	16.20
E1	13.80	14.00	14.20
e	0.65 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
Θ	0°	3.5°	7°

NOTES:

1. All dimensions refer to JEDEC standard MS-026-BEC.
2. Dimension 'D1' and 'E1' do not include mold protrusions. Allowable protrusion is 0.25mm per side. 'D1' and 'E1' are maximum plastic body size dimensions including mold mismatch.
3. Dimension 'b' does not include dambar protrusion. Allowable dambar protrusion shall not cause the lead width to exceed the maximum 'b' dimension by more than 0.08mm.
4. 'A1' is defined as the distance from the seating plane to the lowest point on the package body.

28.2 80 LQFP (12x12) package information

Figure 167. 80 LQFP (12x12) Package

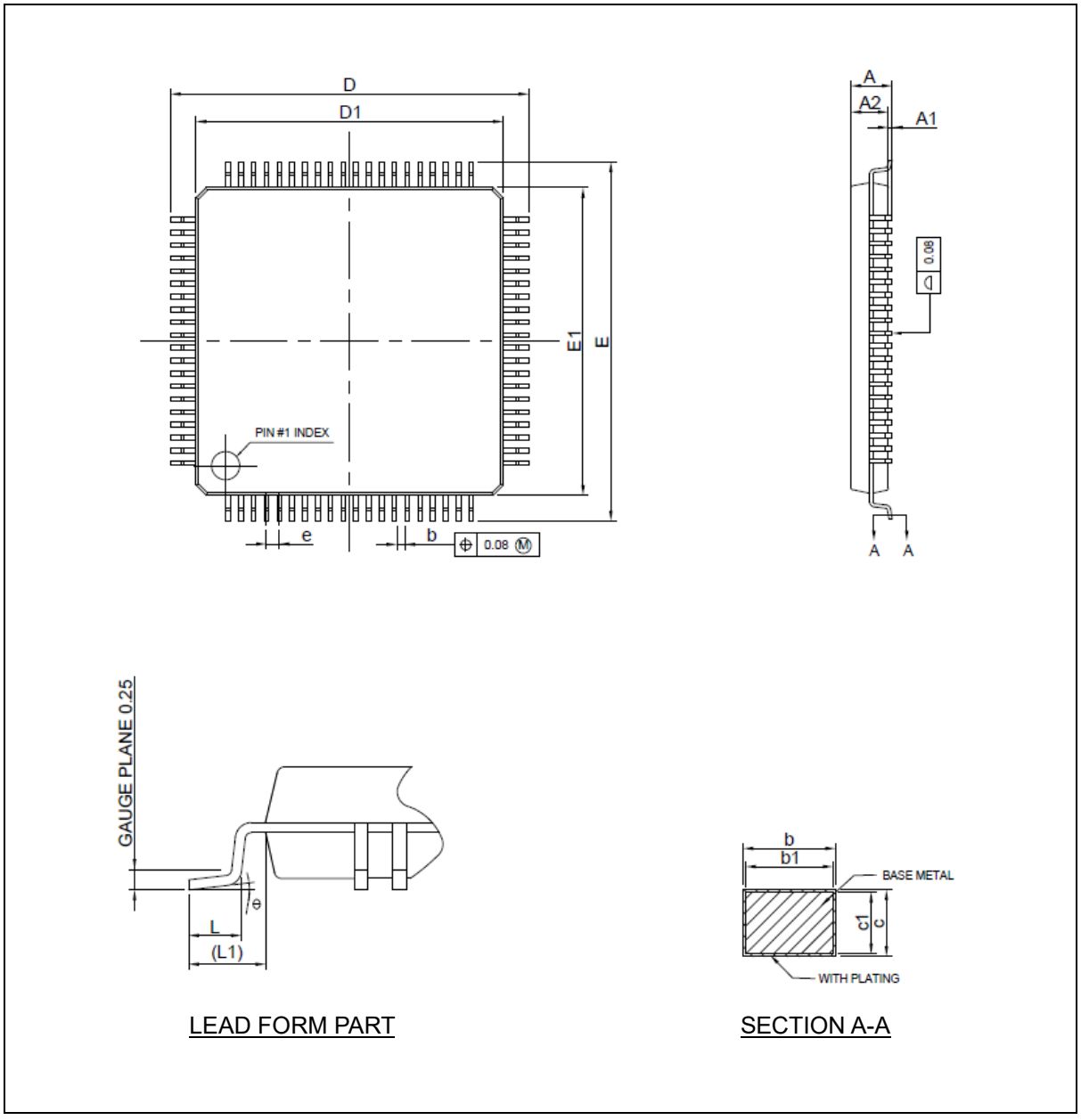


Table 152. 80 LQFP (12 x 12) Package Mechanical Data

Symbol	Dimension (mm)		
	Min.	Nom.	Max.
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
b1	0.17	0.20	0.23
c	0.09	—	0.20
c1	0.09	—	0.16
D	13.80	14.00	14.20
D1	11.80	12.00	12.20
E	13.80	14.00	14.20
E1	11.80	12.00	12.20
e	0.50 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
Θ	0°	3.5°	7°

NOTES:

1. All dimensions refer to JEDEC standard MS-026-BDD.
2. Dimension 'D1' and 'E1' do not include mold protrusions. Allowable protrusion is 0.25mm per side. 'D1' and 'E1' are maximum plastic body size dimensions including mold mismatch.
3. Dimension 'b' does not include dambar protrusion. Allowable dambar protrusion shall not cause the lead width to exceed the maximum 'b' dimension by more than 0.08mm.
4. 'A1' is defined as the distance from the seating plane to the lowest point on the package body.

28.3 64 LQFP (12x12) package information

Figure 168. 64 LQFP (12x12) Package

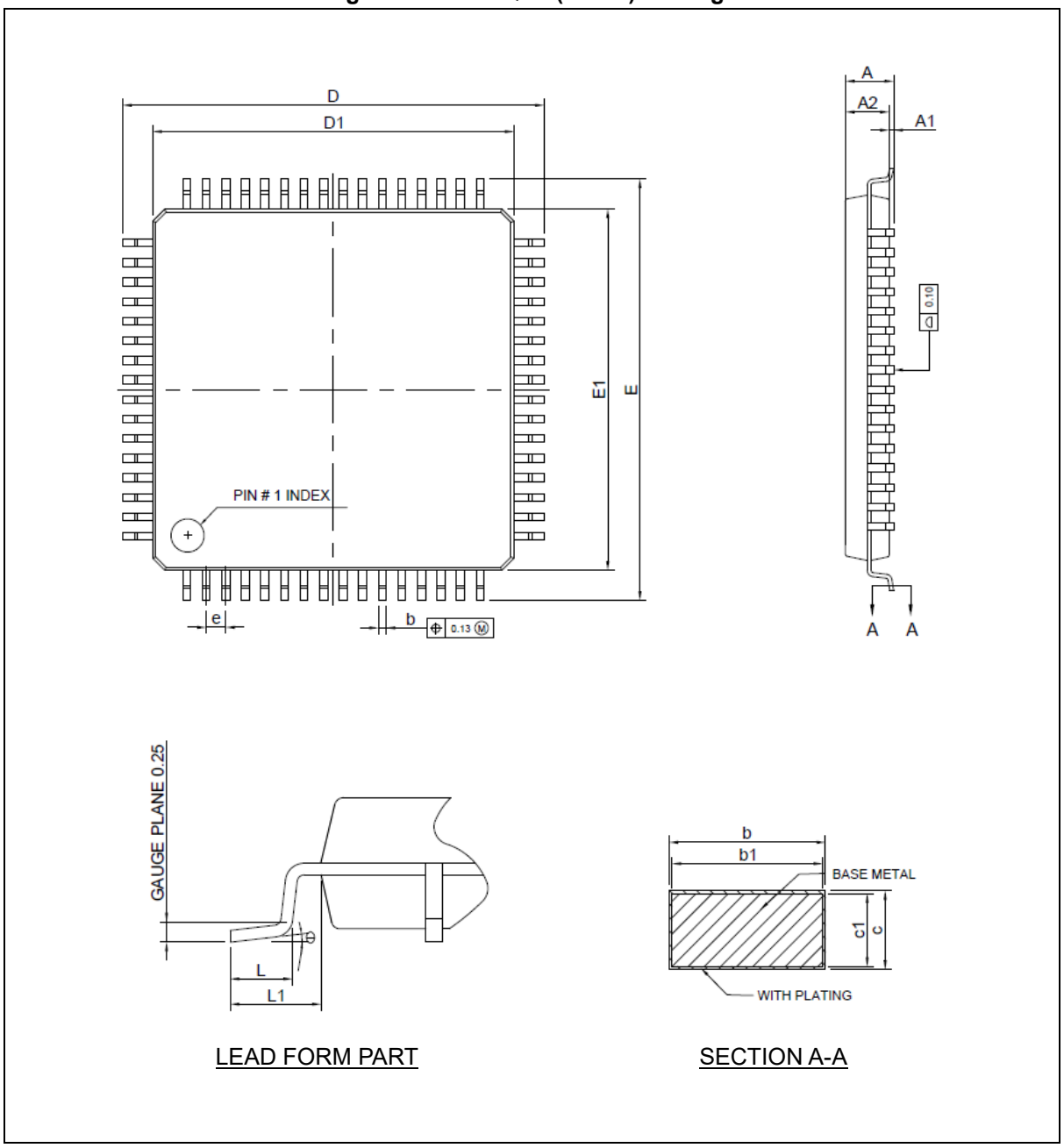


Table 153. 64 LQFP (12 x 12) Package Mechanical Data

Symbol	Dimension (mm)		
	Min.	Nom.	Max.
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.22	0.32	0.38
b1	0.25	0.30	0.33
c	0.09	—	0.20
c1	0.09	—	0.16
D	13.80	14.00	14.20
D1	11.80	12.00	12.20
E	13.80	14.00	14.20
E1	11.80	12.00	12.20
e	0.65 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
Θ	0°	3.5°	7°

NOTES:

1. All dimensions refer to JEDEC standard MS-026-BDC.
2. Dimension 'D1' and 'E1' do not include mold protrusions. Allowable protrusion is 0.25mm per side. 'D1' and 'E1' are maximum plastic body size dimensions including mold mismatch.
3. Dimension 'b' does not include dambar protrusion. Allowable dambar protrusion shall not cause the lead width to exceed the maximum 'b' dimension by more than 0.08mm.
4. 'A1' is defined as the distance from the seating plane to the lowest point on the package body.

28.4 64 LQFP (10x10) package information

Figure 169. 64 LQFP (10x10) Package

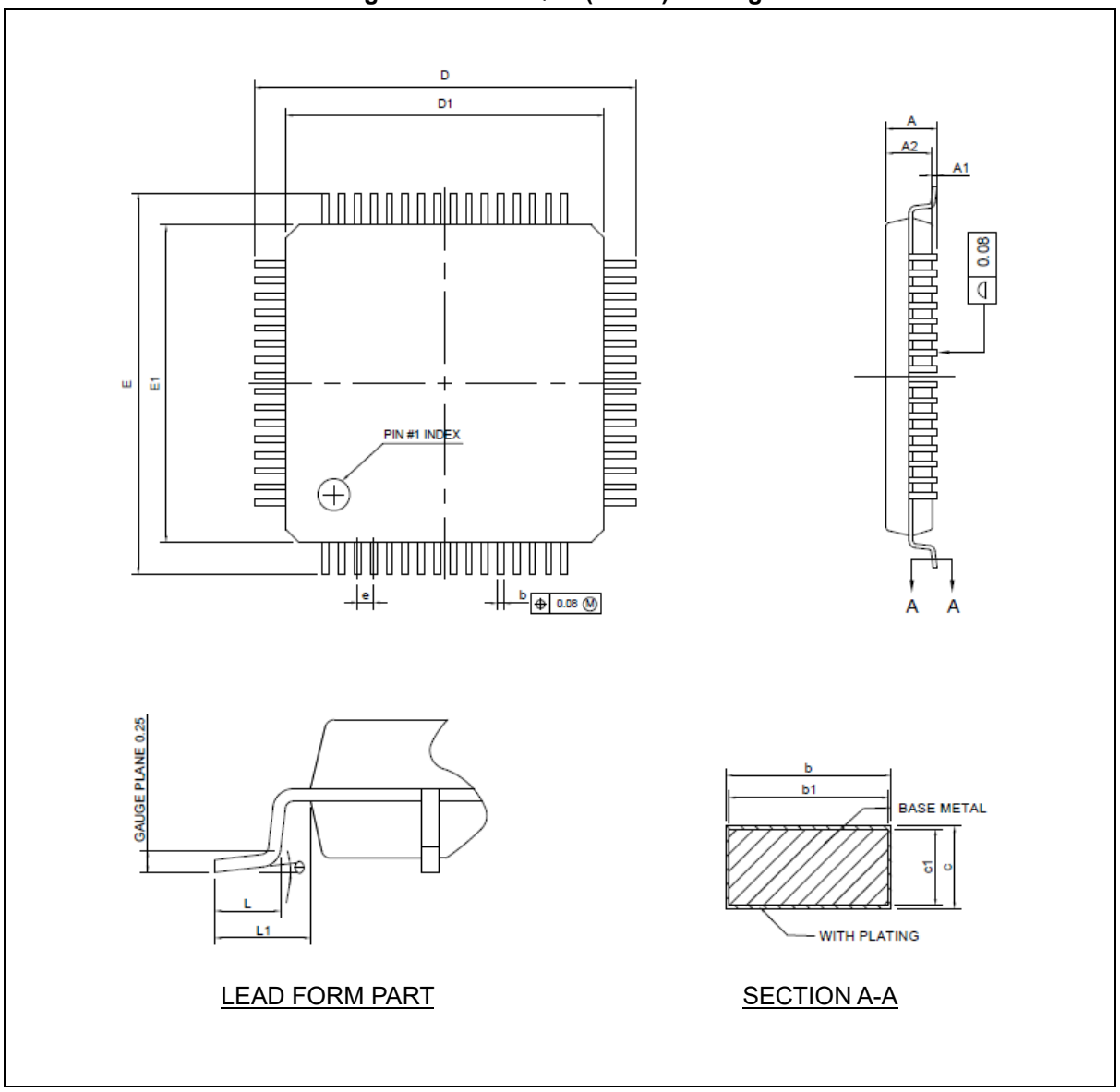


Table 154. 64 LQFP (10 x 10) Package Mechanical Data

Symbol	Dimension (mm)		
	Min.	Nom.	Max.
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
b1	0.17	0.20	0.23
c	0.09	—	0.20
c1	0.09	—	0.16
D	11.80	12.00	12.20
D1	9.80	10.00	10.20
E	11.80	12.00	12.20
E1	9.80	10.00	10.20
e	0.50 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
Θ	0°	3.5°	7°

NOTES:

1. All dimensions refer to JEDEC standard MS-026-BCD.
2. Dimension 'D1' and 'E1' do not include mold protrusions. Allowable protrusion is 0.25mm per side. 'D1' and 'E1' are maximum plastic body size dimensions including mold mismatch.
3. Dimension 'b' does not include dambar protrusion. Allowable dambar protrusion shall not cause the lead width to exceed the maximum 'b' dimension by more than 0.08mm.
4. 'A1' is defined as the distance from the seating plane to the lowest point on the package body.

28.5 48 LQFP (7 x 7) package information

Figure 170. 48 LQFP (7 x 7) Package

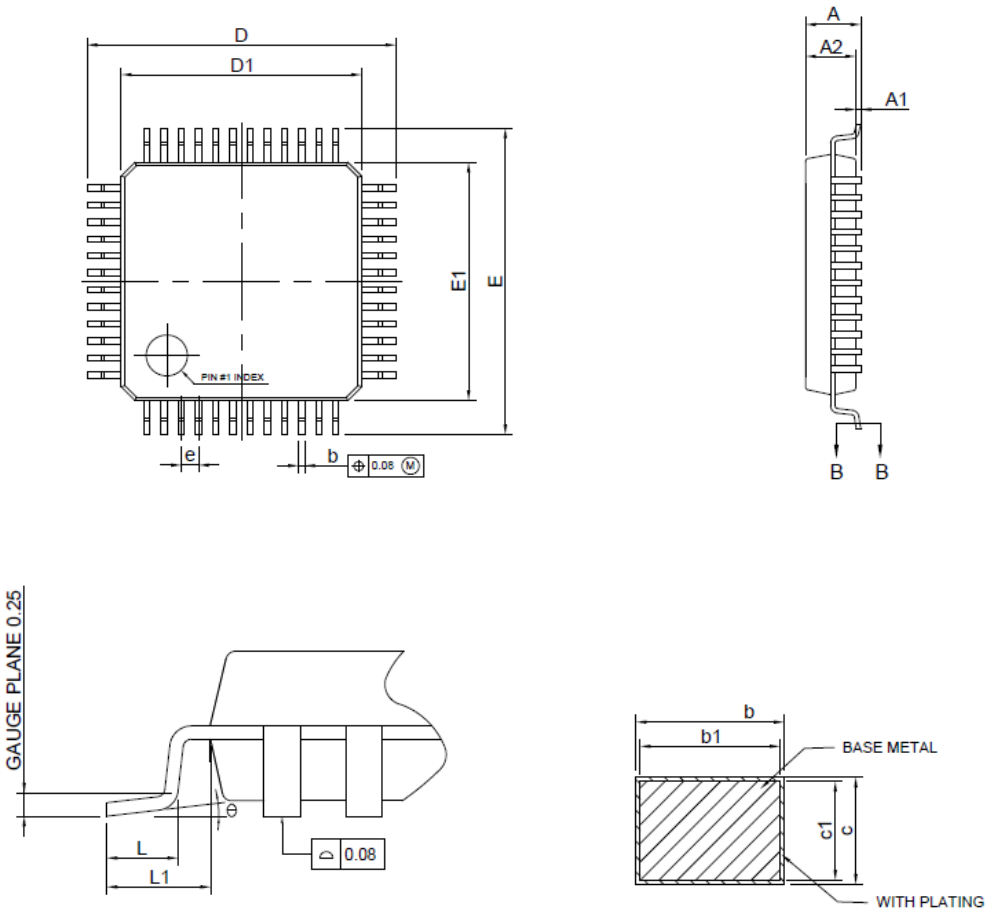


Table 155. 48 LQFP (7 x 7) Package Mechanical Data

Symbol	Dimension (mm)		
	Min.	Nom.	Max.
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
b1	0.17	0.20	0.23
c	0.09	—	0.20
c1	0.09	—	0.16
D	8.80	9.00	9.20
D1	6.80	7.00	7.20
E	8.80	9.00	9.20
E1	6.80	7.00	7.20
e	0.50 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
Θ	0°	3.5°	7°

NOTES:

1. All dimensions refer to JEDEC standard MS-026-BBC.
2. Dimension 'D1' and 'E1' do not include mold protrusions. Allowable protrusion is 0.25mm per side. 'D1' and 'E1' are maximum plastic body size dimensions including mold mismatch.
3. Dimension 'b' does not include dambar protrusion. Allowable dambar protrusion shall not cause the lead width to exceed the maximum 'b' dimension by more than 0.08mm.
4. 'A1' is defined as the distance from the seating plane to the lowest point on the package body.

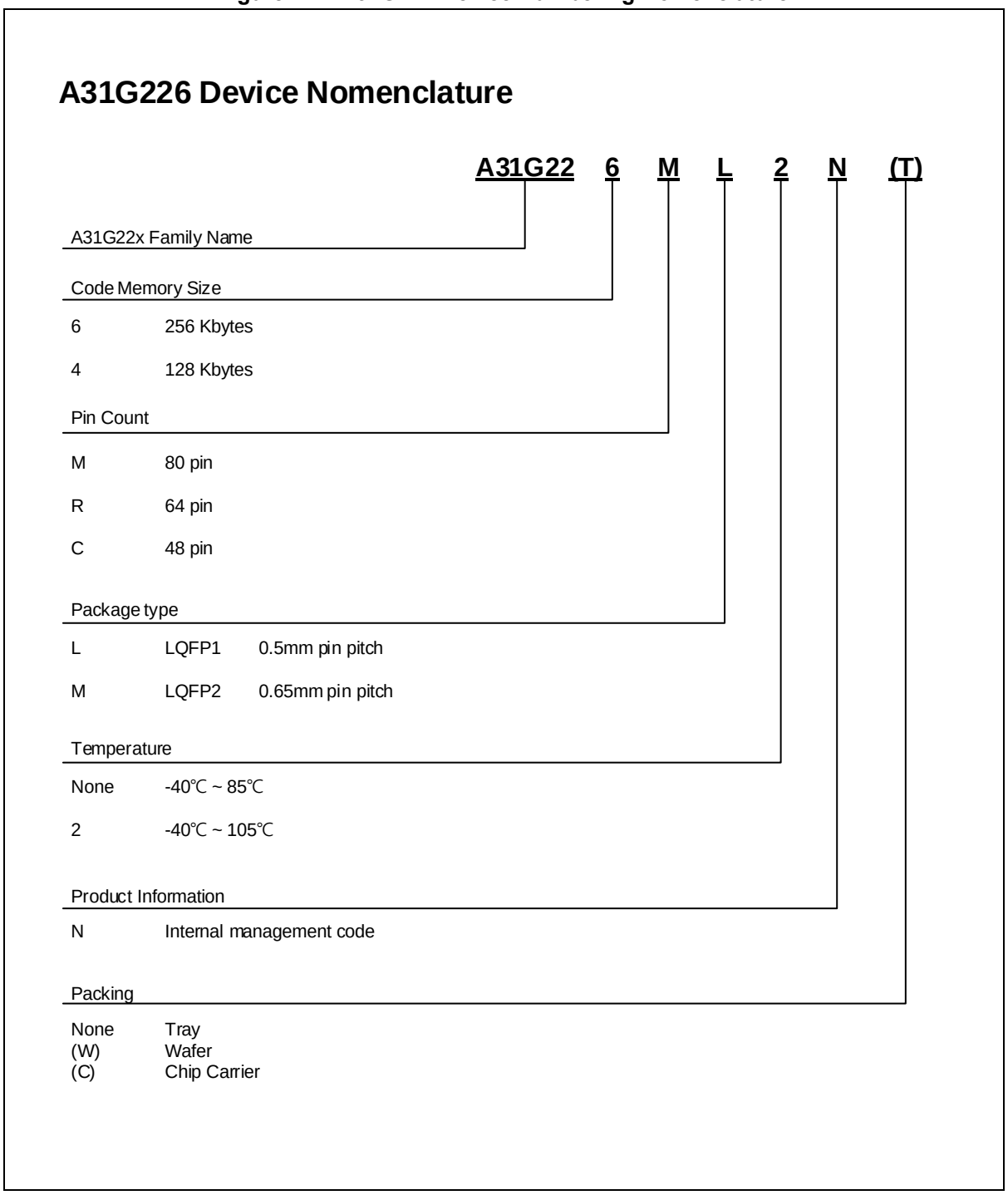
29 Ordering information

Table 156. A31G22x Device Ordering Information

Device name	Code Flash [KB]	Data Flash [KB]	SRAM [KB]	I2C [ch]	UART [ch]	USART [ch]	SPI [ch]	TIMER [ch]	PWM [ch]	12-bit ADC [ch]	LCD Driver [COM x SEG]	I/O ports [ch]	Op. Temp. [℃]	Package Type
A31G226ML2N	256	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~105	80LQFP12
A31G226MM2N*	256	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~105	80LQFP14
A31G226RM2N*	256	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~105	64LQFP12
A31G226RL2N*	256	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~105	64LQFP10
A31G226CL2N*	256	32	20	2	2	2	2	7(16bit)/ 2(32bit)	3	14	8 x 21 or 3 x 26	43	-40 ~105	48LQFP7
A31G224MM2N*	128	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~105	80LQFP14
A31G224ML2N*	128	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~105	80LQFP12
A31G224RM2N*	128	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~105	64LQFP12
A31G224RL2N	128	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~105	64LQFP10
A31G224CL2N*	128	32	20	2	2	2	2	7(16bit)/ 2(32bit)	3	14	8 x 21 or 3 x 26	43	-40 ~105	48LQFP7
A31G226MLN*	256	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~85	80LQFP12
A31G226MMN*	256	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~85	80LQFP14
A31G226RMN*	256	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~85	64LQFP12
A31G226RLN*	256	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~85	64LQFP10
A31G226CLN*	256	32	20	2	2	2	2	7(16bit)/ 2(32bit)	3	14	8 x 21 or 3 x 26	43	-40 ~85	48LQFP7
A31G224MMN*	128	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~85	80LQFP14
A31G224MLN*	128	32	20	3	2	4	2	7(16bit)/ 2(32bit)	3	18	8 x 37 or 3 x 42	75	-40 ~85	80LQFP12
A31G224RMN*	128	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~85	64LQFP12
A31G224RLN*	128	32	20	3	2	3	2	7(16bit)/ 2(32bit)	3	18	8 x 29 or 3 x 34	59	-40 ~85	64LQFP10
A31G224CLN*	128	32	20	2	2	2	2	7(16bit)/ 2(32bit)	3	14	8 x 21 or 3 x 26	43	-40 ~85	48LQFP7

: For available options for further information on the device with a '' mark, please contact the ABOV Sales Offices.

Figure 171. A31G22x Device Numbering Nomenclature



Revision history

Date	Revision	Description
Dec. 22, 2020	1.00	First release.
Jan. 11, 2021	1.01	Modified the descriptions of CLKDIV and EXTCLK in ADC_CCR. The symbol and unit of conversion frequency ratio in Table 141. ADC Electrical Characteristics have been modified.
Feb. 02. 2021	1.02	Modified the figures and the descriptions in 16.3.1, 16.3.5 Updated the register information in 16.2.4 UARTn_IIR Modified Table 75. Updated the register information in 18.2.1 I2Cn_CR Corrected typos.
Apr. 02, 2021	1.03	Updated block diagram of the introduction. Added notes on pin function setting to Pn_AFSR1, Pn_AFSR2. Updated the number of channels of LCD driver.
Jul. 12, 2021	1.04	Modified typo 'I_{OL3}' to 'I_{OL7}' of V_{OL7} and added notification to Table 105. Added VLE_EN bit and notification to 22.2.3 for setting external bias VLC.
Jul. 27, 2021	1.05	Removed Figure135. Removed AVREF signal in Figure 1, Figure 2, Figure 3. Changed the SFLAG bit access type of the CFMC_SFSR register from RW to RO. Added 6.2.4 Write protection with memory bank swap. Added a comment to 6.1.8 to be referenced 6.2.4.
Aug. 10, 2021	1.06	Fixed broken cross-reference links in Figure 79 and Figure 82. Modified the terminology, block diagram and descriptions of the Temp Sensor chapter.
Aug. 24, 2021	1.07	The register name and bit name are the same. - 4.5.8, 4.5.9: Changed bit name LVRRST to LVDRST to the use of words with duplicate meanings. - 4.5.16: Corrected LCDCLK offset value display error - 4.5.20: LVRSTS --> LVDSTS - 4.5.32: TSLSIEN --> TSLSITSEN - 12.2.1: 'T1n' prefix removed - 13.2.1: 'T2n' prefix removed - 13.3.1: Added the number of TIMER2n unit (n = 0 to 1) - 14.2.1, 14.2.11, 14.2.12, 14.2.14: Removed 'T30' prefix. - Modified the 'Figure 162. A31G22x recommended circuit layout'
Oct. 5, 2021	1.08	Modified the descriptions of register bit '10' in Pn_ICR and Pn_ISR registers. Corrected typo of Figure 8 and Table 1 Modified the terminology of operating modes.
Nov. 23, 2021	1.09	Updated that SEG5 and SEG6 pins for LCD driver control are not supported. - Modified Figure 4, Figure 6, Figure 138, Figure 141

		- Modified Table1, Table2, Table20, Table 95, Table 97 - Added the notifications to Table 98 - Modified the description of Ch 22 - Modified Table 154 Updated Figure 171 Updated Figure 137 Corrected Typo - 4.4.1, 4.4.4, Table 91 Added the notification to 17.2.5
Dec. 14, 2021	1.10	Updated Figure 162 Modified The start-up time of HSE Characteristics from Typ. 200 ms to Typ. 2 ms. Modified Stabilization time of HSI characteristics from Max. 100 us to Min. 100 us. Updated Figure 80 Modified Table 28 and Table 36 Changed the title from 'CRC and Checksum' to 'CRC' because CRC checksum is not supported. Added the notifications and pull-up state of PC1, PC2 pin in Table 3. Updated Figure 43 LCD block diagram and added the notifications. Added the notification about PC1 and PC2 to 5.3.5 Pn_PUPD register description table. Modified Figure 66 and Figure 67 Modified the description of 4.4 Corrected typo
Apr. 13, 2022	1.11	Modified Table 157 USART13_RXD13 -> USRAT13_TXD13, USART13_TXD13 -> USART13_RXD13 Removed RTC function
Nov. 04, 2022	1.12	Revision the font of this document
Jan. 18. 2023	1.13	- Modified "Table 125. I/O static characteristics" I/O pin capacitance to max. 10pF. -Change pull-up min/max value in Table 91 DC Electrical Characteristics - Removed "6.21 Code Flash erase and program examples" Removed of content : "In addition, a user must enable 48MHz internal oscillator first to erase or program flash". - Removed "7.2.1 Data flash erase and program examples" Removed of content : "In addition, a user must enable 48MHz internal oscillator first to erase or program flash".
Jan. 19. 2023	1.14	Modified Figure 93 Modified the description of 4.5.6
Mar. 09. 2023	1.15	Modified the description of 4.4.2 Modified Table 100
Apr. 06. 2023	1.16	Added the note of 15.2.1 Modified Table 131(deleted capacitance load)
Apr. 10. 2023	1.17	Modified the description of 4.4.2

Korea

Regional Office, Seoul
R&D, Marketing & Sales
8th Fl., 330, Yeongdong-daero,
Gangnam-gu, Seoul,
06177, Korea

Tel: +82-2-2193-2200
Fax: +82-2-508-6903
www.abovsemi.com

HQ, Ochang
R&D, QA, and Test Center
93, Gangni 1-gil, Ochang-eup,
Cheongwon-gun,
Chungcheongbuk-do,
28126, Korea

Tel: +82-43-219-5200
Fax: +82-43-217-3534
www.abovsemi.com

Domestic Sales Manager
Tel: +82-2-2193-2206
Fax: +82-2-508-6903
Email: sales_kr@abov.co.kr

Global Sales Manager
Tel: +82-2-2193-2281
Fax: +82-2-508-6903
Email: sales_gl@abov.co.kr

China Sales Manager
Tel: +86-755-8287-2205
Fax: +86-755-8287-2204
Email: sales_cn@abov.co.kr