AC33M4064T,AC33M3064T
User's Manual

32-bit Cortex-M3 based
Programmable Motor Controller

User's Manual Version 1.04

Introduction

AC33Mx064T is a special purpose microcontroller for a motor application. This microcontroller brings high-performance 32-bit computing to a low cost system solution.

AC33Mx064T provides 3-phase PWM generator units which are suitable for an inverter motor drive system. A built-in 3-phase PWM generator controls one inverter motor.

Two 12-bit high speed ADC units with 11-channel analog multiplexed inputs support to get feedback information from a motor. They can control up to one inverter motor or one inverter motor and PFC (Power Factor Correction) function simultaneously.

Powerful and various external serial interfaces help to communicate with on-board sensors and devices. In addition, development environment is compatible with the Cortex-M family and supports the SWD.

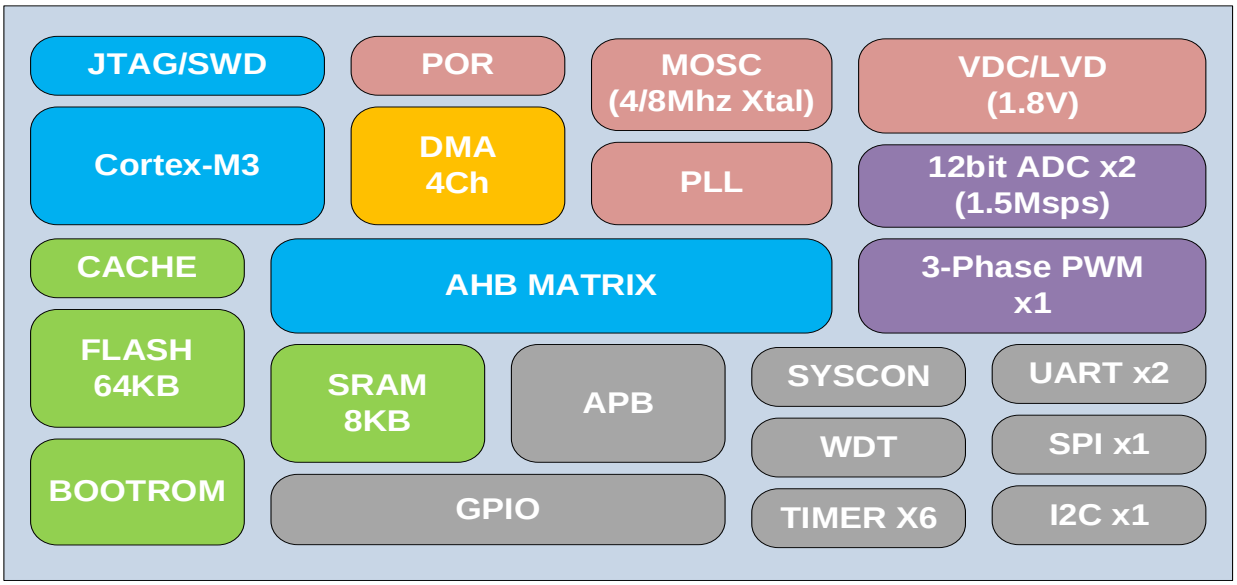


Figure 1. AC33Mx064T

Reference document

- Document DDI337 is provided by ARM, and contains detail information of Cortex-M3.
- AC33Mx064T Datasheet is provided by ABOV, and available at www.abovsemi.com.

.

Contents

List of figures

List of tables

1 Description

AC33Mx064T's CPU core is supported by ARM Cortex-M3 processor which provides a high-performance and low-cost platform.

1.1 Device overview

In this section, features of AC33Mx064T series and peripheral counts are introduced.

Table 1. AC33Mx064T Series Features and Peripheral Counts

Peripherals		Description
Core	CPU	• Maximum operating frequency: 48MHz • 32-bit ARM Cortex-M3 CPU
	Interrupt	• NVIC (Nested-Vectored Interrupt Controller)
Memory	Code flash	• 64Kbytes code flash memory
	BOOT ROM	• UART, SPI boot modes • In-system programming
	SRM	• 8 KB
System Control Unit (SCU)	Operating frequency	• 48MHz
	Clock	• MainOSC : X-TAL(4MHz~8MHz) • PLL Clock : 4MHz ~ 48MHz • Internal RING OSC : 1MHz
	Clock monitoring	• System Fail-Safe function by Clock Monitoring
	Operating mode	• RUN mode • SLEEP mode
	Reset	• nRESET pin reset • Core reset • Software reset
	LDO	• Low-dropout (LDO) regulator built in for low-voltage operation
	POR	• Power On Reset
	LVD	• Programmable Low Voltage Detector (Brown-Out Detector)
General Purpose I/O (GPIO)		• General Purpose I/O (GPIO) — 44Ports : 48-Pin — 28rts: 32-Pin

Table 1. AC33Mx064T Series Features and Peripheral Counts (continued)

Peripherals		Description
TIMER	16-bit Timer	• 6channels • Periodic, One-shot, PWM, Capture mode • Multi-Timer Synchronization Option
	WDT	• 1 channels
Serial interface	UART	• 2 channels supported
	SPI	• 1 channels supported
	I2C	• 1 channels supported
Motor Pulse-Width Modulation	MPWM	• 3-Phase Motor PWM with ADC triggering function • 1 channel
12-bit A/D Converter	ADC	• 1.5Msps high-speed ADC with sequential conversion function • 2 units with 11 channel Inputs
Operating voltage		• 3.0V to 5.5V
Operating temperature		• Commercial grade (-40℃ to +105℃)
Package		• Three types of package options — 48-pin LQFP — 32-pin LQFP

1.2 Block diagram

In this section, the AC33Mx064T series with peripherals is described in block diagram.

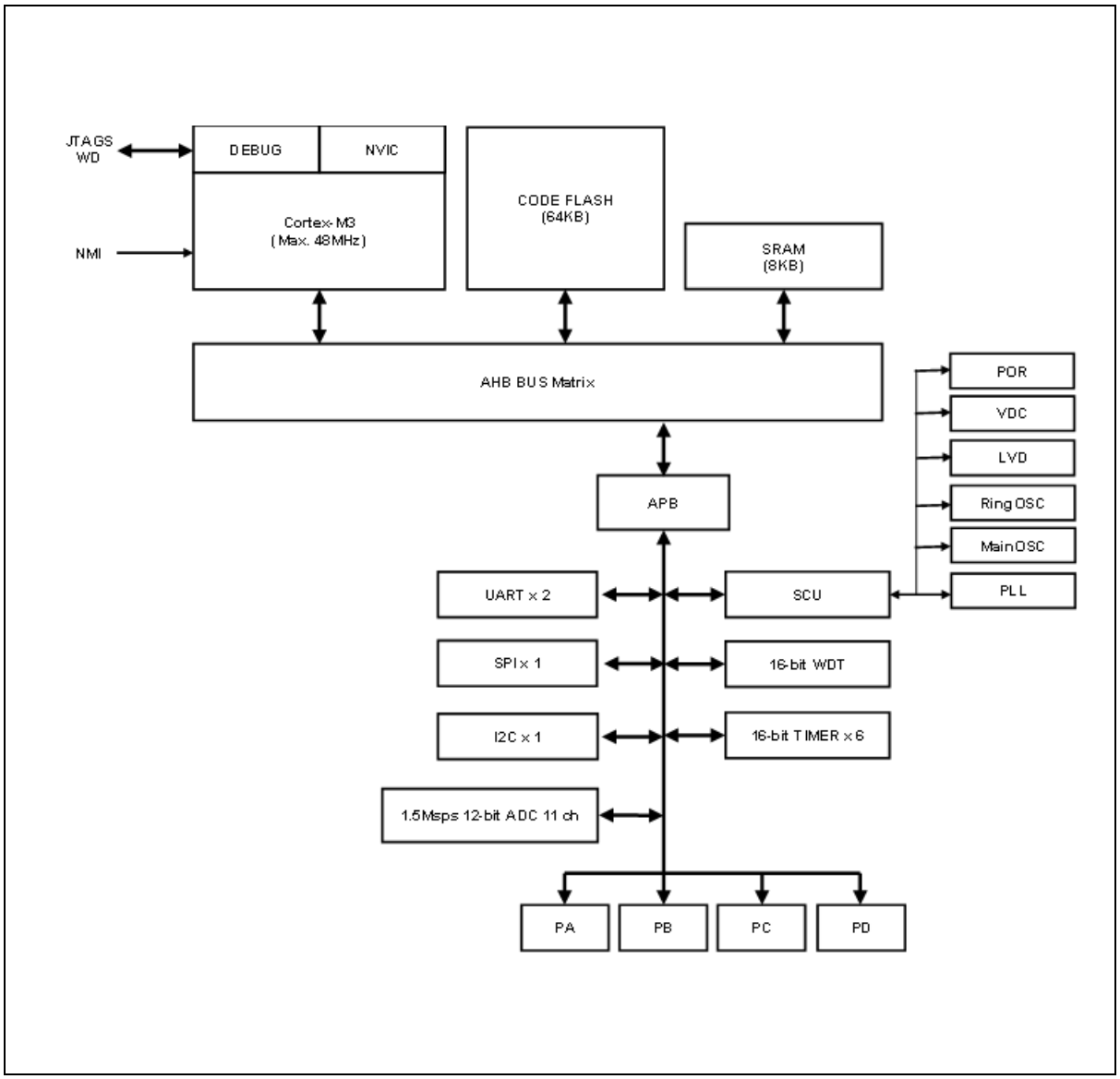


Figure 2. AC33Mx064T Block Diagram

2 Pinouts and pin descriptions

In this chapter, pinouts and pin descriptions of the AC33Mx064T series are introduced.

2.1 Pinouts

2.1.1 AC33M4064T (LQFP-48)

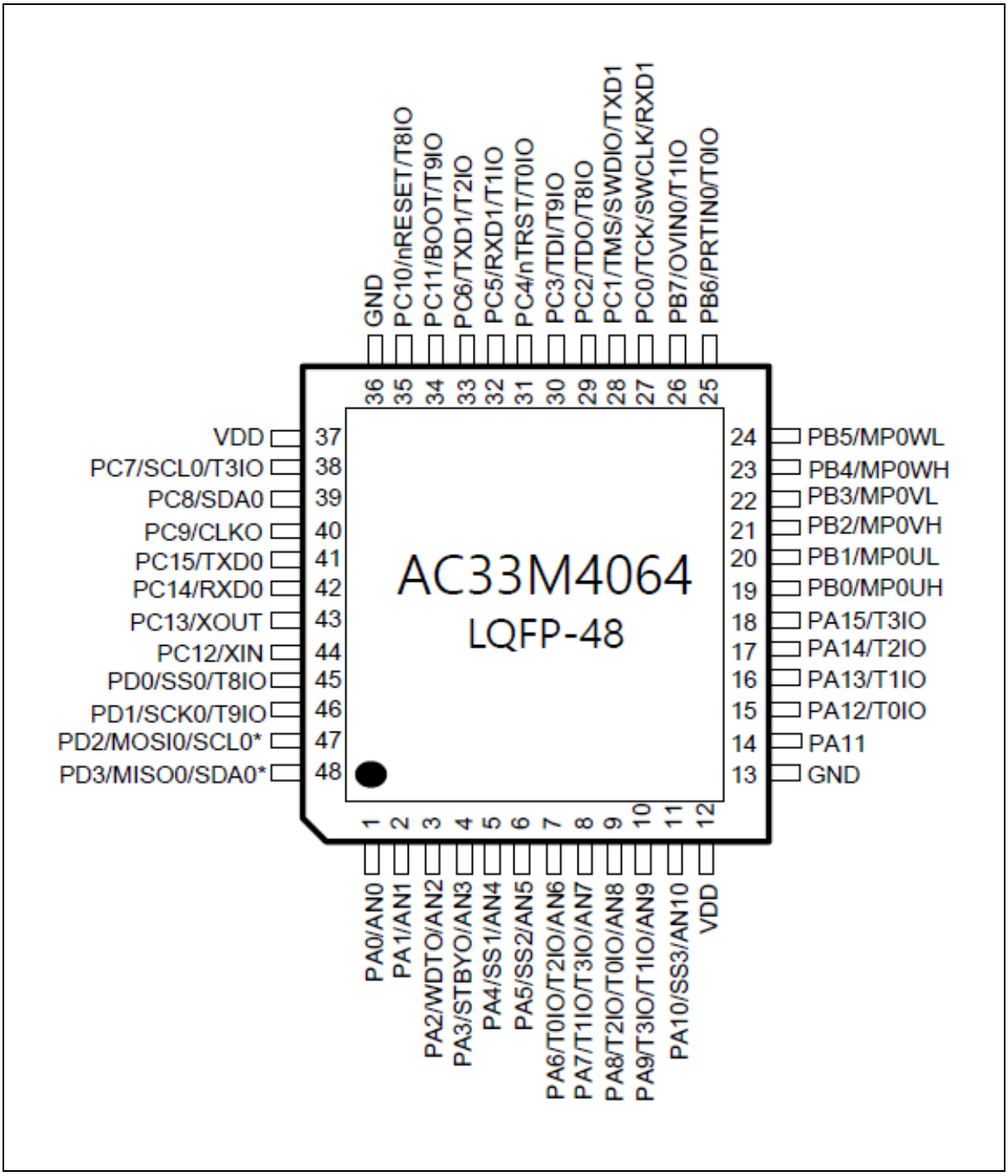


Figure 3. LQFP 48 Pinouts

2.1.2 AC33M3064T (LQFP-32)

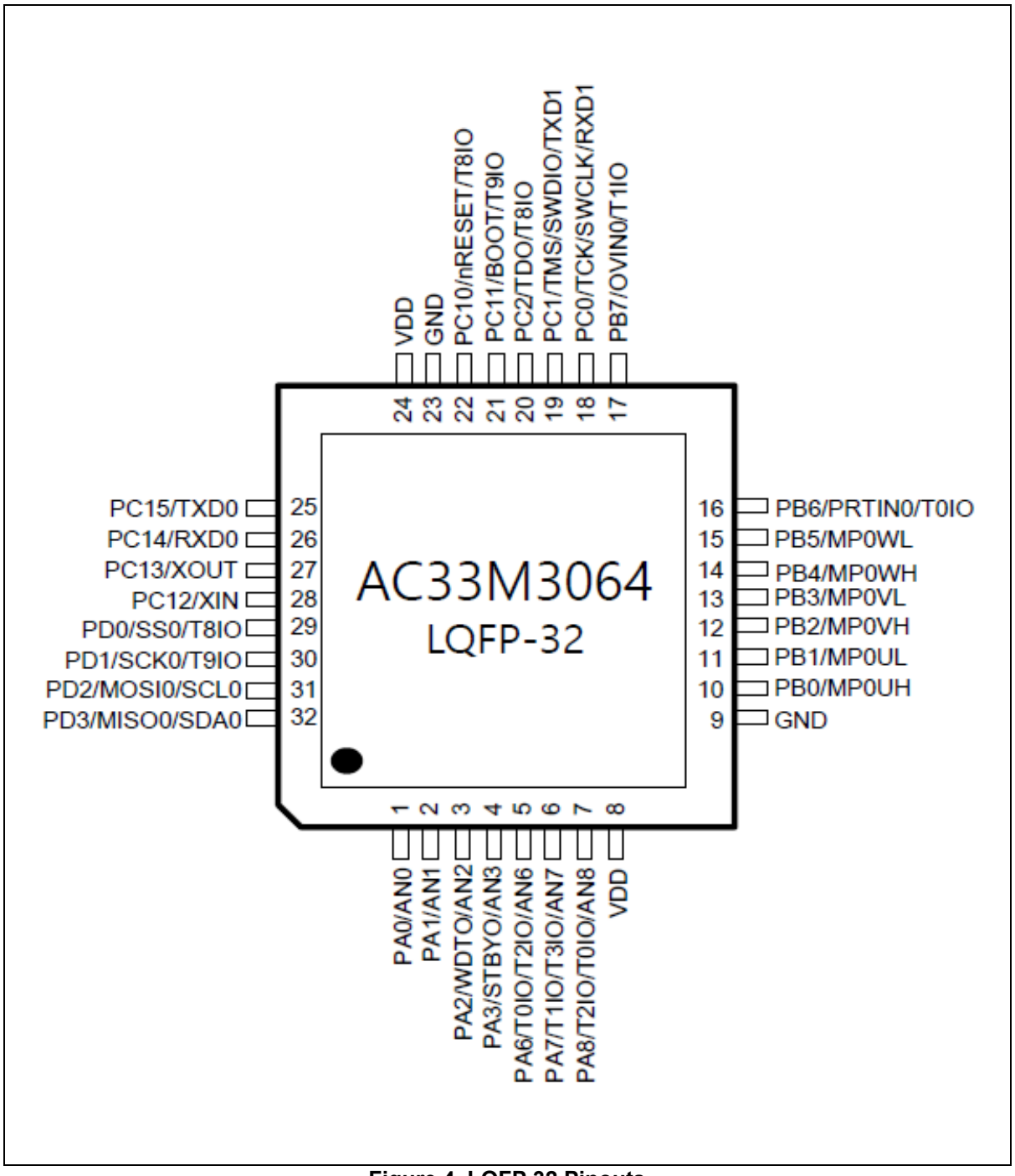


Figure 4. LQFP 32 Pinouts

2.2 Pin description

Pin configuration information in Table 2 contains two pairs of power/ground and other dedicated pins. These multi-function pins have up to five selections of functions including GPIO. Configuration including pin ordering can be changed without notice.

Table 2. Pin Description

Pin No		Pin Name	Type	Description	Remark
LQFP48	LQFP32				
1	1	PA0*	IOUS	PORT A Bit 0 Input/Output	
		AN0	IA	Analog Input 0	
2	2	PA1*	IOUS	PORT A Bit 1 Input/Output	
		AN1	IA	Analog Input1	
3	3	PA2*	IOUS	PORT A Bit 2 Input/Output	
		WDTO	O	Watchdog timer overflow output	
		AN2	IA	Analog Input2	
4	4	PA3*	IOUS	PORT A Bit 3 Input/Output	
		STBO	O	Stop mode output	
		AN3	IA	Analog Input 3	
5	-	PA4*	IOUS	PORT A Bit 4 Input/Output	
		SS1	I/O	Slave Select 1 for SPI0	
		AN4	IA	Analog Input 4	
6	-	PA5*	IOUS	PORT A Bit 5 Input/Output	
		SS2	I/O	Slave Select 2 for SPI0	
		AN5	IA	Analog Input 5	
7	5	PA6*	IOUS	PORT A Bit 6 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	
		AN6	IA	Analog Input 6	
8	6	PA7*	IOUS	PORT A Bit 7 Input/Output	
		T1IO	I/O	Timer 1 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
		AN7	IA	Analog Input 7	
9	7	PA8*	IOUS	PORT A Bit 8 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
		AN8	IA	Analog Input 8	

Table 2. Pin Description (continued)

Pin No		Pin Name	Type	Description	Remark
LQFP48	LQFP32				
10	-	PA9*	IOUS	PORT A Bit 9 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
		T1IO	I/O	Timer 1 Input/Output	
		AN9	IA	Analog Input 9	
11	-	PA10*	IOUS	PORT A Bit 10 Input/Output	
		SS3	Output	Slave Select 3 for SPI0	
		AN10	IA	Analog Input 10	
12	8	VDD	P	VDD	
13	9	GND	P	Ground	
14	-	PA11*	IOUS	PORT A Bit 11 Input/Output	
15	-	PA12*	IOUS	PORT A Bit 12 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
16	-	PA13*	IOUS	PORT A Bit 13 Input/Output	
		T1IO	I/O	Timer 1 Input/Output	
17	-	PA14*	IOUS	PORT A Bit 14 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	
18	-	PA15*	IOUS	PORT A Bit 15 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
19	10	PB0	IOUS	PORT B Bit 0 Input/Output	
		PWM0UH	Output	PWM0 UH Output	
20	11	PB1	IOUS	PORT B Bit 1 Input/Output	
		PWM0UL	Output	PWM0 UL Output	
21	12	PB2	IOUS	PORT B Bit 2 Input/Output	
		PWM0VH	Output	PWM0 VH Output	
22	13	PB3	IOUS	PORT B Bit 3 Input/Output	
		PWM0VL	Output	PWM0 VL Output	
23	14	PB4	IOUS	PORT B Bit 4 Input/Output	
		PWM0WH	Output	PWM0 WH Output	
24	15	PB5	IOUS	PORT B Bit 5 Input/Output	
		PWM0WL	Output	PWM0 WL Output	
25	16	PB6	IOUS	PORT B Bit 6 Input/Output	
		PRTIN0	Input	PWM0 Protection Input signal 0	
		T0IO	I/O	Timer 0 Input/Output	
26	17	PB7	IOUS	PORT B Bit 7 Input/Output	
		OVIN0	Input	PWM0 Over-voltage input signal 0	
		T1IO	I/O	Timer 1 Input/Output	
27	18	PC0	IOUS	PORT C Bit 0 Input/Output	
		TCK/SWCK	Input	JTAG TCK, SWD Clock Input	
		RXD1	Input	UART0 Rx Data Input	
28	19	PC1	IOUS	PORT C Bit 1 Input/Output	
		TMS/SWDIO	I/O	JTAG TMS, SWD Data Input/Output	
		TXD1	Input	UART0 Tx Data Output	
29	20	PC2	IOUS	PORT C Bit 2 Input/Output	
		TDO/SWO	Output	JTAG TDO, SWO Output	
		T8IO	I/O	Timer 8 Input/Output	
30	-	PC3	IOUS	PORT C Bit 3 Input/Output	
		TDI	Input	JTAG TDI Input	
		T9IO	I/O	Timer 9 Input/Output	
31	-	PC4	IOUS	PORT C Bit 4 Input/Output	
		nTRST	Input	JTAG nTRST Input	
		T0IO	Input	Timer 0 Input/Output	
32	-	PC5	IOUS	PORT C Bit 5 Input/Output	
		RXD1	Input	UART1 RXD Input	
		T1IO	I/O	Timer 1 Input/Output	

Table 2. Pin Description (continued)

Pin No		Pin Name	Type	Description	Remark
LQFP48	LQFP32				
33	-	PC6	IOUS	PORT C Bit 6 Input/Output	
		TXD1	Output	UART1 TXD Output	
		T2IO	I/O	Timer 2 Input/Output	
34	21	PC11	IOUS	PORT C Bit 11 Input/Output	
		BOOT	Input	Boot mode Selection Input	
		T9IO	I/O	Timer 9 Input/Output	
35	22	PC10	IOUS	PORT C Bit 10 Input/Output	
		nRESET	Input	External Reset Input	Pull-up
		T8IO	I/O	Timer 8 Input/Output	
36	23	GND	P	Ground	
37	24	VDD	P	VDD	
38	-	PC7	IOUS	PORT C Bit 7 Input/Output	
		SCL0	Output	I²C Channel 0 SCL In/Out	
		T3IO	I/O	Timer 3 Input/Output	
39	-	PC8	IOUS	PORT C Bit 8 Input/Output	
		SDA0	Output	I²C Channel 0 SDA In/Out	
40	-	PC9	IOUS	PORT C Bit 9 Input/Output	
		CLKO	Output	System Clock Output	
41	25	PC15	IOUS	PORT C Bit 15 Input/Output	
		TXD0	Output	UART0 TXD Output	
		MISO0	I/O	SPI0 Master-Input/Slave-Output	
42	26	PC14	IOUS	PORT C Bit 14 Input/Output	
		RXD0	Input	UART0 RXD Input	
		MOSI0	I/O	SPI0 Master-Output/Slave-Input	
43	27	PC13	IOUS	PORT C Bit 13 Input/Output	
		XOUT	OA	External Crystal Oscillator Output	
44	28	PC12	IOUS	PORT C Bit 12 Input/Output	
		XIN	IA	External Crystal Oscillator Input	
45	29	PD0	IOUS	PORT D Bit 0 Input/Output	
		SS0	I/O	SPI1 Slave Select	
		T8IO	I/O	Timer 8 Input/Output	
46	30	PD1	IOUS	PORT D Bit 1 Input/Output	
		SCK0	I/O	SPI0 Clock Input/Output	
		T9IO	I/O	Timer 9 Input/Output	
47	31	PD2	IOUS	PORT D Bit 2 Input/Output	
		MOSI0	I/O	SPI Channel 0 Master Out / Slave In	
		SCL0	Output	I²C Channel 0 SCL In/Out	
48	32	PD3*	IOUS	PORT D Bit 3 Input/Output	
		MISO0	I/O	SPI Channel 0 Master In / Slave Out	
		SDA0	Output	I²C Channel 0 SDA In/Out	

NOTES:

1. * Notation: I=Input, O=Output, U=Pull-up, D=Pull-down, S=Schmitt-Trigger Input Type, C=CMOS Input Type, A=Analog, P=Power
2. (*) Selected pin function after reset condition
3. Pin order may be changed with revision notice

3 System and memory overview

3.1 System architecture

Main system of AC33Mx064T series consists of the followings:

- ARM® Cortex® -M3 core
- Internal SRAM, Flash memory

3.1.1 Cortex-M3 core

ARM powered Cortex-M3 Core based on ARMv7M architecture which is optimized for small size and low power system. On core system timer (SYSTICK) provides a simple 24 bit timer easy to manage the system operation Thumb-compatible Thumb-2 only instruction set processor core makes code high-density. Hardware division and single-cycle multiplication is present Integrated Nested Vectored Interrupt Controller (NVIC) provides deterministic interrupt handling. JTAG and SWD debugging features are provided. Max 48MHz operating frequency with zero wait execution

3.1.2 Interrupt controller

Table 3. 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	MPU Fault Handler
-11	0x0000_0014	BUS Fault Handler
-10	0x0000_0018	Usage Fault Handler
-9	0x0000_001C	Reserved
-8	0x0000_0020	
-7	0x0000_0024	
-6	0x0000_0028	
-5	0x0000_002C	SVCall Handler
-4	0x0000_0030	Debug Monitor Handler
-3	0x0000_0034	Reserved
-2	0x0000_0038	PenSV Handler
-1	0x0000_003C	SysTick Handler

Table 3. Interrupt Vector Map (continued)

Priority	Vector address	Interrupt source
0	0x0000_0040	LVDFAIL
1	0x0000_0044	SYSCLKFAIL
2	0x0000_0048	XOSCFAIL
3	0x0000_004C	WDT
4	0x0000_0050	Reserved
5	0x0000_0054	TIMER0
6	0x0000_0058	TIMER1
7	0x0000_005C	TIMER2
8	0x0000_0060	TIMER3
9	0x0000_0064	Reserved
10	0x0000_0068	
11	0x0000_006C	
12	0x0000_0070	
13	0x0000_0074	TIMER8
14	0x0000_0078	TIMER9
15	0x0000_007C	Reserved
16	0x0000_0080	GPIOAE
17	0x0000_0084	GPIOAO
18	0x0000_0088	GPIOBE
19	0x0000_008C	GPIOBO
20	0x0000_0090	GPIOCE
21	0x0000_0094	GPIOCO
22	0x0000_0098	GPIODE
23	0x0000_009C	GPIODO
24	0x0000_00A0	MPWM0
25	0x0000_00A4	MPWM0PROT
26	0x0000_00A8	MPWM0OVV
27	0x0000_00AC	Reserved
28	0x0000_00B0	
29	0x0000_00B4	
30	0x0000_00B8	
31	0x0000_00BC	
32	0x0000_00C0	SPI0
33	0x0000_00C4	Reserved
34	0x0000_00C8	
35	0x0000_00CC	
36	0x0000_00D0	I2C0
37	0x0000_00D4	Reserved
38	0x0000_00D8	UART0
39	0x0000_00DC	UART1
40	0x0000_00E0	Reserved
41	0x0000_00E4	
42	0x0000_00E8	
43	0x0000_00EC	ADC0
44	0x0000_00F0	ADC1
45	0x0000_00F4	Reserved
46	0x0000_00F8	
47	0x0000_00FC	
48	0x0000_0100	
49	0x0000_0104	
50	0x0000_0108	
51	0x0000_010C	
52	0x0000_0110	
53	0x0000_0114	
54	0x0000_0118	

Table 3. Interrupt Vector Map (continued)

Priority	Vector address	Interrupt source
55	0x0000_011C	
56	0x0000_0120	
57	0x0000_0124	
58	0x0000_0128	
59	0x0000_012C	
60	0x0000_0130	
61	0x0000_0134	
62	0x0000_0138	
63	0x0000_013C	

NOTE: Each external interrupt has an associated priority-level register. Each of them is 3 bits wide, occupying the three MSBs of the Interrupt Priority Level Registers. Each Interrupt Priority Level Register occupies 1 byte (8 bits). NVIC registers in the Cortex-M3 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 = 3

3.2 Memory organization

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

3.2.1 Memory map

Figure 5 shows addressable memory space in memory map.

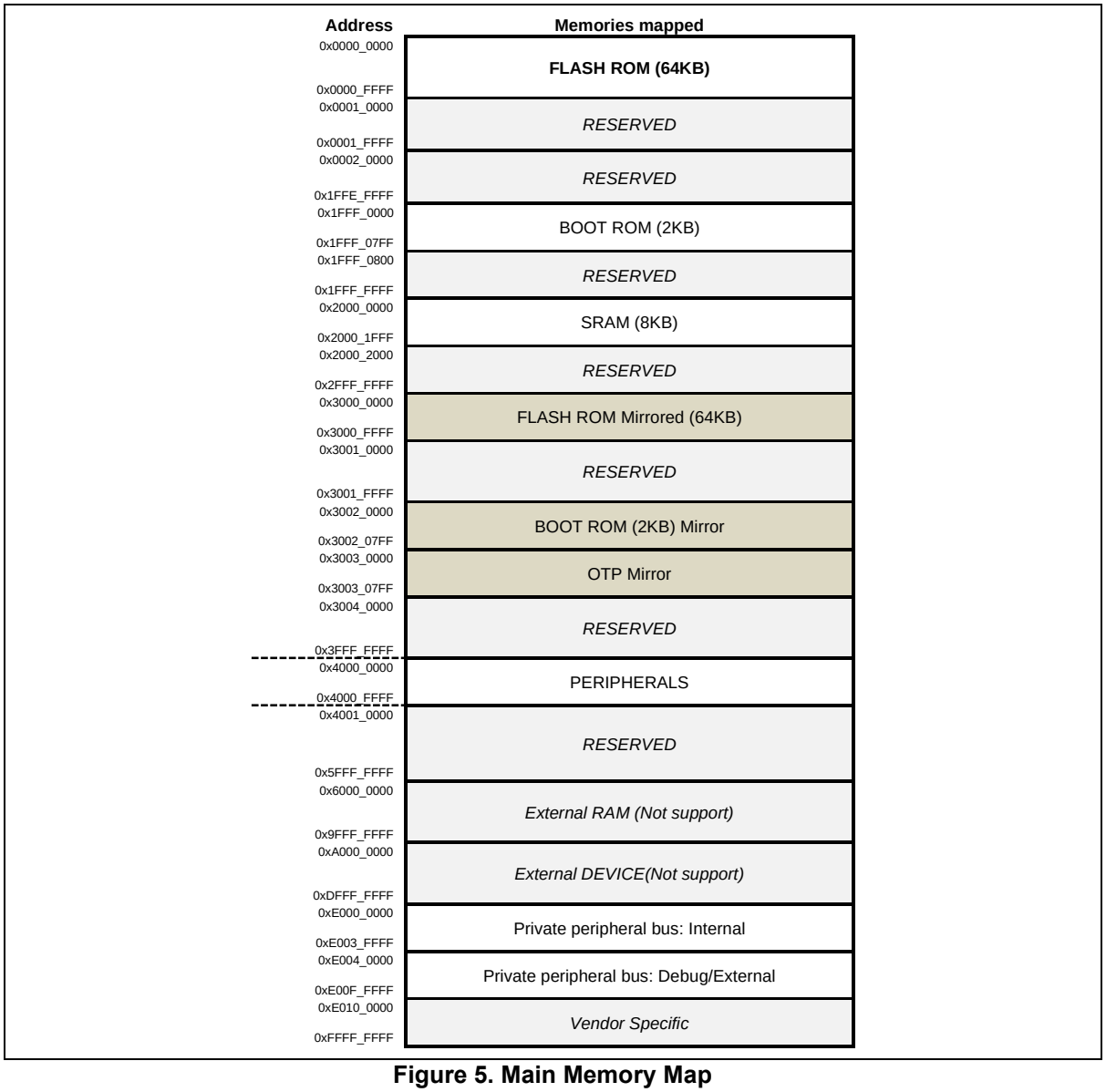


Figure 5. Main Memory Map

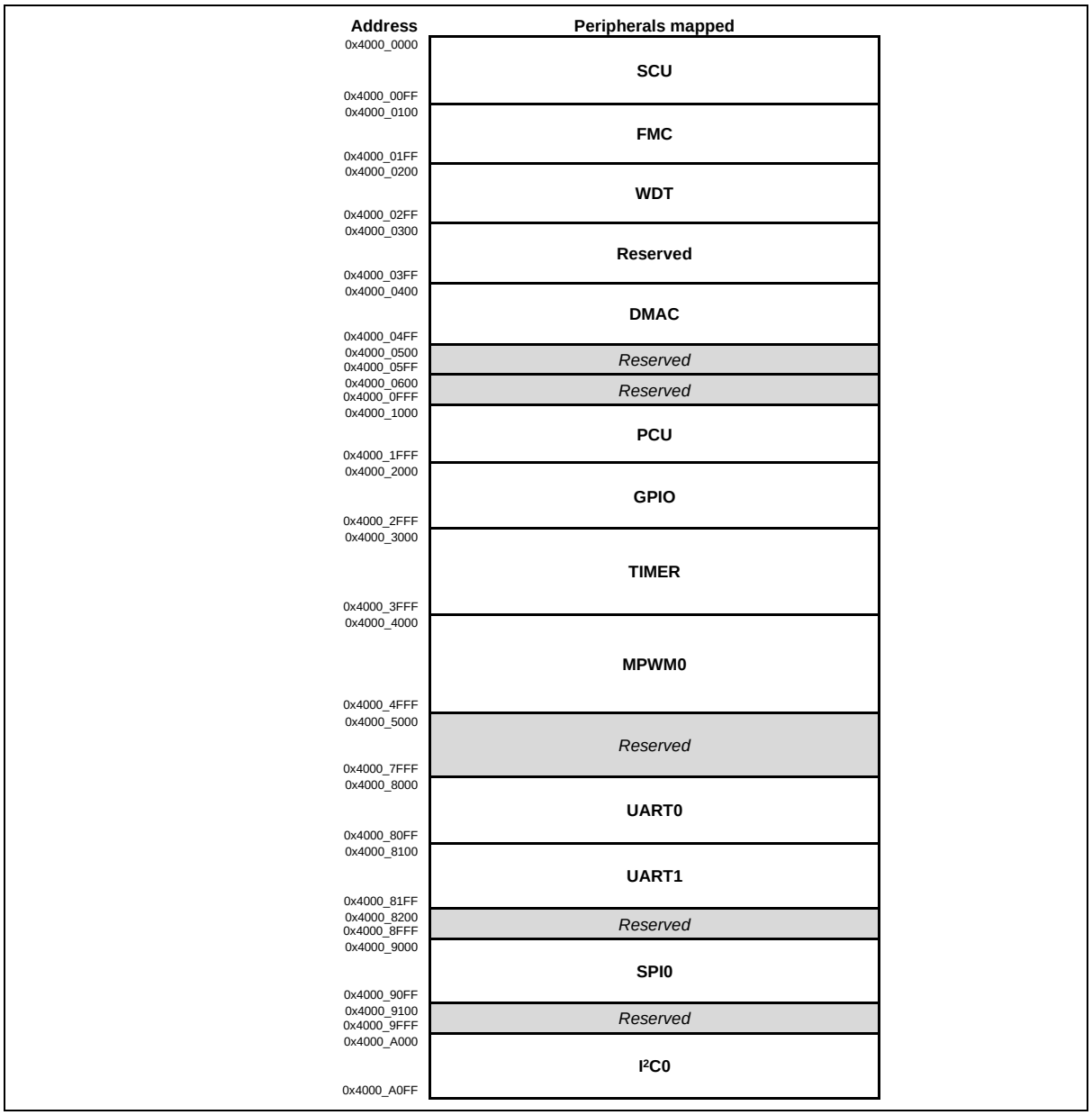


Figure 6. Peripheral Memory Map

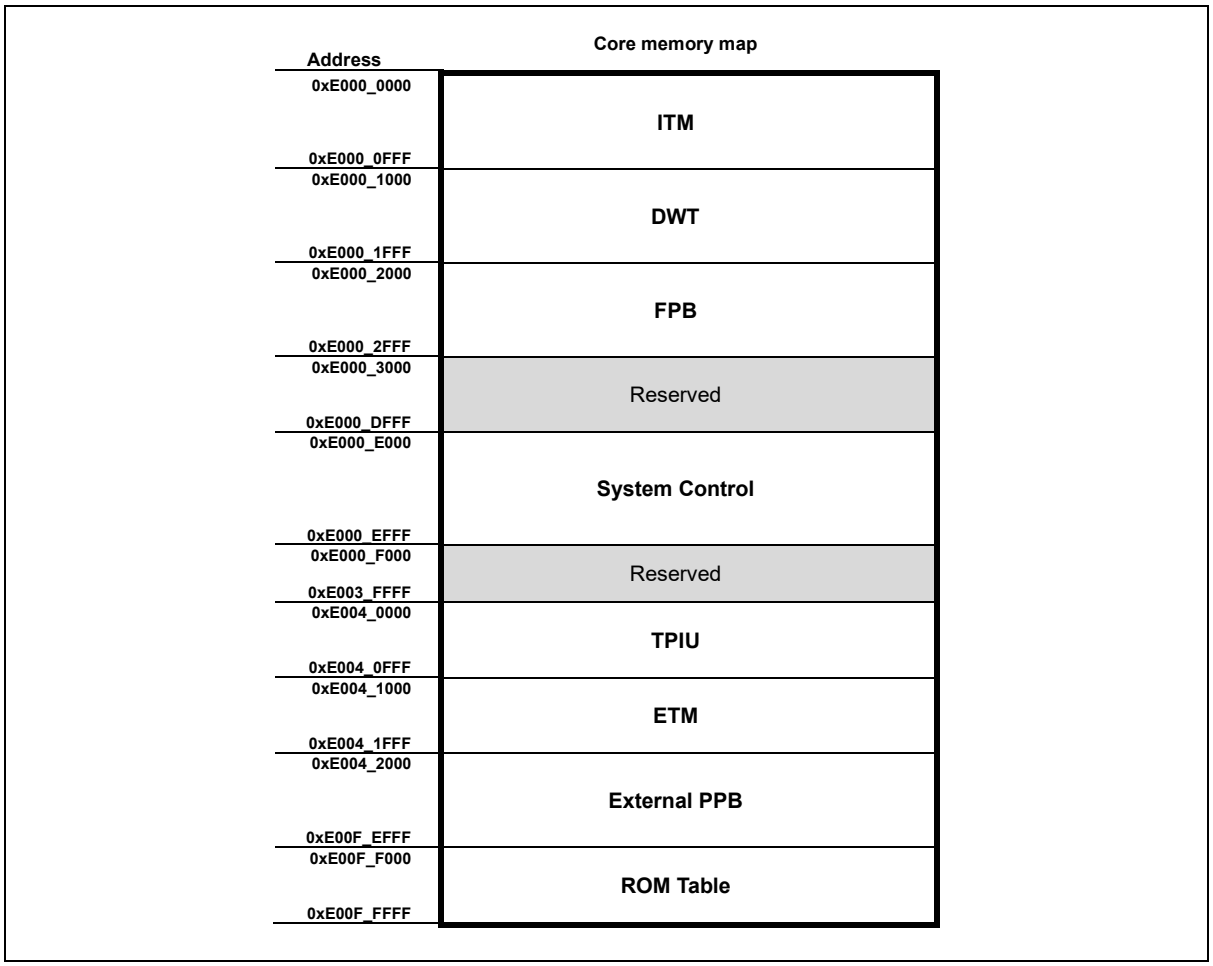


Figure 7. Cortex-M3 Private Memory Map

3.2.2 Embedded SRAM

On chip 8KB 0-wait SRAM can be used for working memory space and program code can be loaded on this SRAM.

3.2.3 Flash memory overview

The AC33Mx064T provides internal 64KB code flash memory and its controller. This is enough to program motor algorithm and general control the system. Self-programming is available and ISP and JTAG programming is also supported in boot or debugging mode.

Instruction and data cache buffer are present and overcome the low bandwidth flash memory.

3.3 Boot mode

3.3.1 Boot mode pins

AC33Mx064T series has a boot mode option to program internal flash memory. Boot mode can be entered by setting BOOT pin to 'L' at reset timing (Normal state is 'H').

Boot mode supports both of UART/SPI boot:

- UART boot and SPI boot uses TXD0/RXT0 ports.

Pins for the boot mode are listed in Table 4.

Table 4. Boot Mode Pin List

Block	Pin name	Dir	Description
SYSTEM	nRESET/PC10	I	Reset Input signal
	BOOT/PC11	I	'0' to enter Boot mode
UART0	RXD0/PC14	I	UART Boot Receive Data
	TXD0/PC15	O	UART Boot Transmit Data
SPI	SS0/PD0	I	SPI Boot Slave Select
	SCK0/PD1	I	SPI Boot Clock Input
	MOSI0/PD2	I	SPI Boot Data Input
	MISO0/PD3	O	SPI Boot Data Output

3.3.2 Boot mode connections

Users can design a target board using any of boot mode ports such as UART mode of UART0. Sample connection diagrams of boot mode are introduced in the following figures:

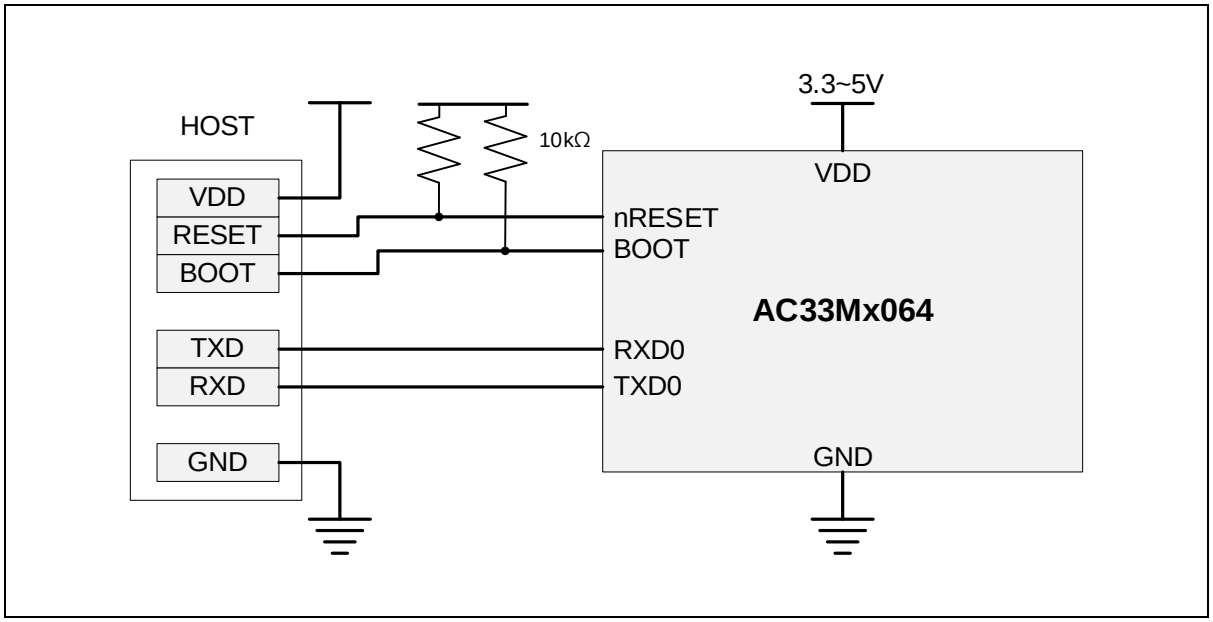


Figure 8. Connection Diagram of UART0 Boot

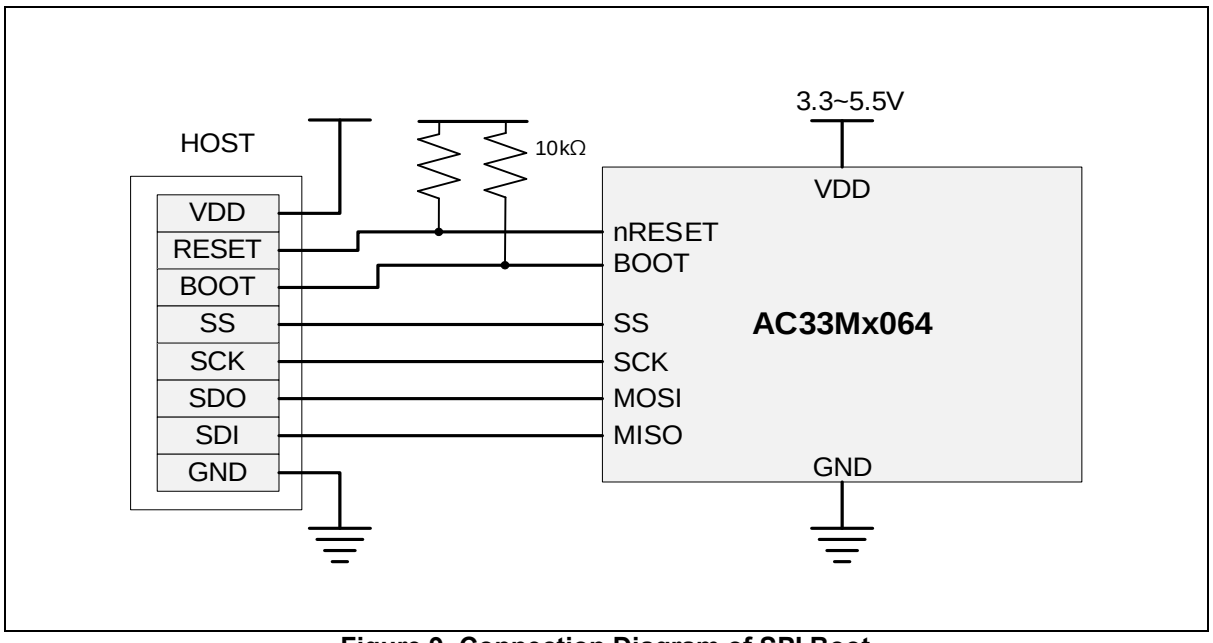


Figure 9. Connection Diagram of SPI Boot

4 System Control Unit (SCU)

AC33Mx064T 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.

Table 5 are assigned for SCU block

Table 5. SCU Pins

Pin name	Type	Description
nRESET	I	External reset input
XIN/XOUT	OSC	External crystal oscillator
CLKO	O	Clock output monitoring signal

4.1 SCU block diagram

In this subsection, SCU block diagram is introduced in Figure 10.

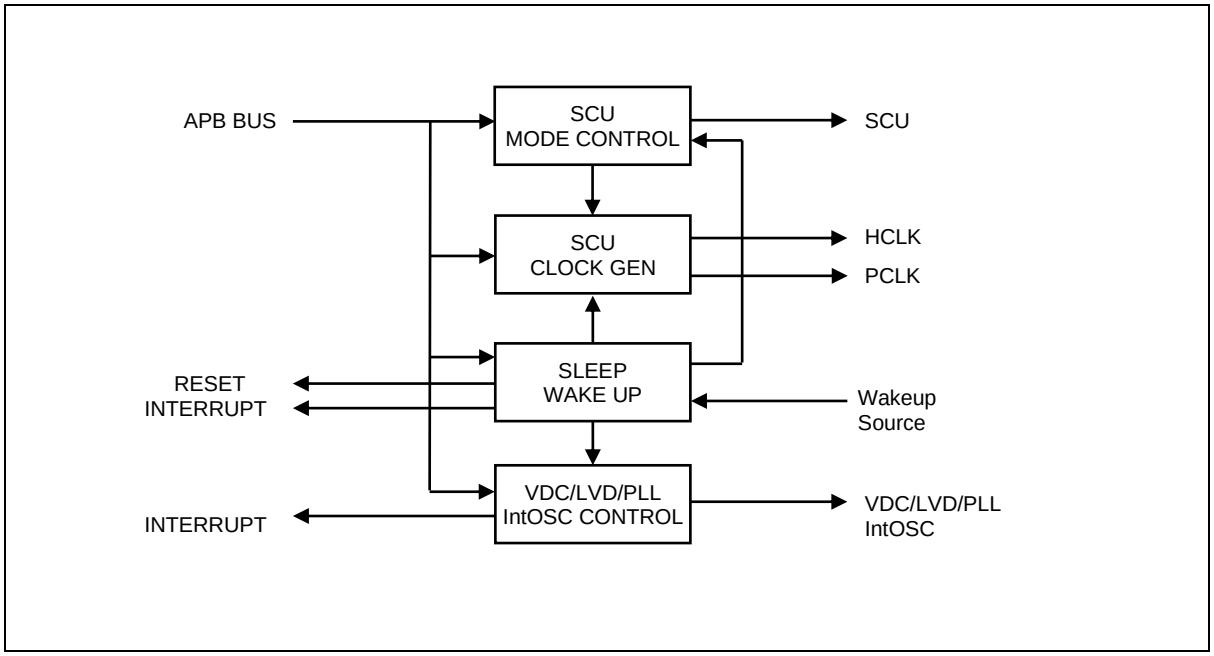
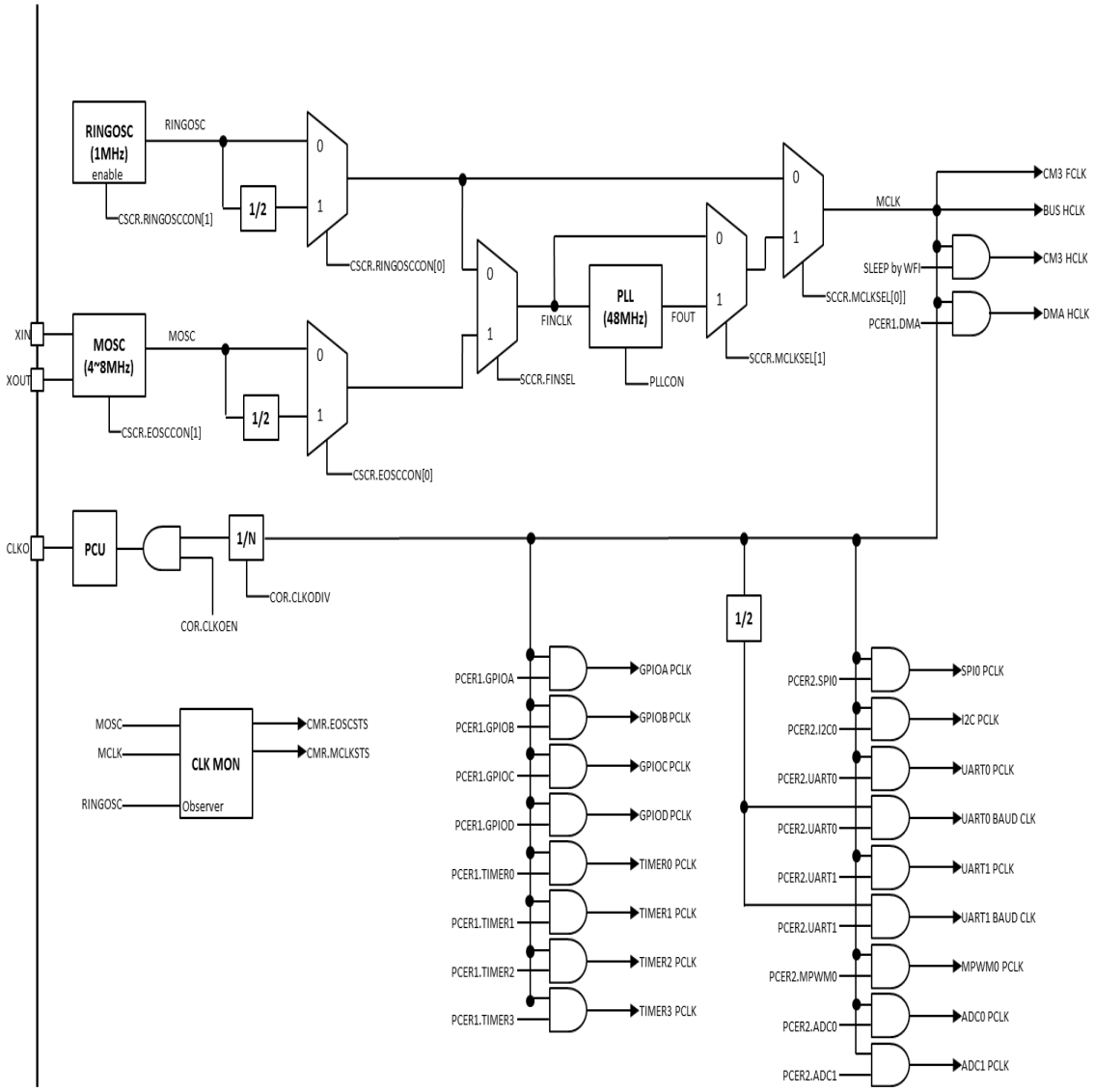


Figure 10. SCU Block Diagram

4.2 Clock system

AC33Mx064T 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 11 and Table 6, users can learn about the clock system of AC33Mx064T devices and clock sources.



NOTE : The PLL clock cannot be used as a RINGOSC (1MHz), because the PLL input Frequency ranges from 4MHz to 8MHz.

Figure 11. Clock Tree Configuration

A clock monitoring block is provided for security purpose. The RING OSC clock is a source clock for monitoring other clock sources. The clock monitoring block observes the status of MCLK clock and MOSC clock.

All muxes switching clock sources have glitch-free circuits internally. So clock can be switched without glitch risks. When you try to change the clock mux control, both of clock sources should be alive. If one of them is not alive, clock change operation is stopped and system will be halted and not be recovered.

Table 6. Clock Sources

Clock name	Frequency	Description
MainOSC	X-TAL(4MHz~8MHz)	External Crystal IOSC
PLL Clock	4MHz ~ 48MHz	On Chip PLL
ROSC	1MHz(±50%)	Internal RING OSC

The PLL can synthesize PLLCLK clock up to 48MHz with FIN reference clock. It also has internal pre-divider and post-divider.

4.2.1 Configuration of miscellaneous clocks

4.2.2 HCLK clock domain

HCLK clock feeds the clock to the CPU and AHB bus. Cortex-M3 CPU requires 2 clocks related with HCLK clock. FCLK and HCLK. FCLK is free running clock and it is always running except power down mode. HCLK can be stopped in the sleep mode.

BUS system and memory systems operated by HCLK clock. Max bus operating clock speed is 48MHz. HCLK frequency should be controlled under 48MHz frequency.

4.2.3 Miscellaneous clock domain

Various clock sources are required for each functional blocks. The SCU provide clock source selection function with its dedicated pre-scaler for each functional blocks. The clock selection mux cannot provide glitch-free function, so the clock is unpredictable at clock selection changing time. Figure 12 shows miscellaneous clock configurations.

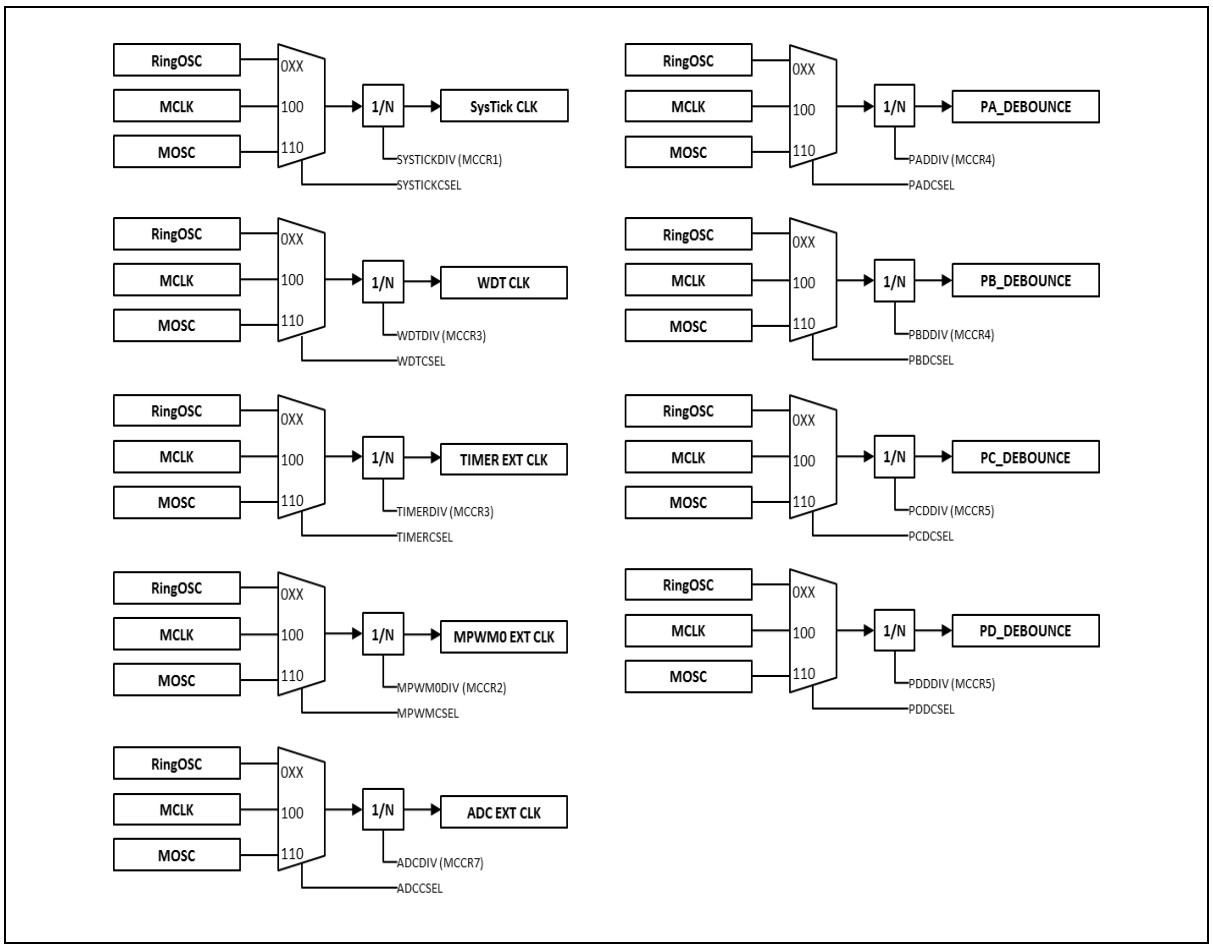


Figure 12. Miscellaneous Clock Configuration

4.2.4 PCLK clock domain

PCLK is a master clock of all peripherals. It can be stopped in power down modes. Each peripheral clock is generated by the PCER1 and PCER2 register set. PCLK clock distributions are showed in Figure 12. Before enabling the PCLK clock of each block, it can't be accessible even when reading its registers.

4.2.5 Clock configuration procedure

After power up, the default system clock is feed by RING OSC (1MHz) clock. RING OSC is default enabled at power up sequence. The other clock sources will be enabled by user controls with the RING OSC system clock.

MOSC clock can be enabled by CSCR register. Before enable MOSC block, the pin mux configuration should be set for XIN, XOUT function. PC12 and PC13 pins are shared with MOSC's XIN and XOUT function - PCCMR and PCCCR registers should be configured properly. After enabling the MOSC block, you must wait for more than 1msec time to ensure stable operation of crystal oscillation.

PLL clock can be enabled by PLLCON register. After enabling the PLL block, you must wait for PLL lock flag. PLL output clock is stable, you can select MCLK for your system requirement. Before changing the system clock, flash access wait should be set to the maximum value. After the system clock is changed, you will need to set flash access wait that you want if necessary.

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

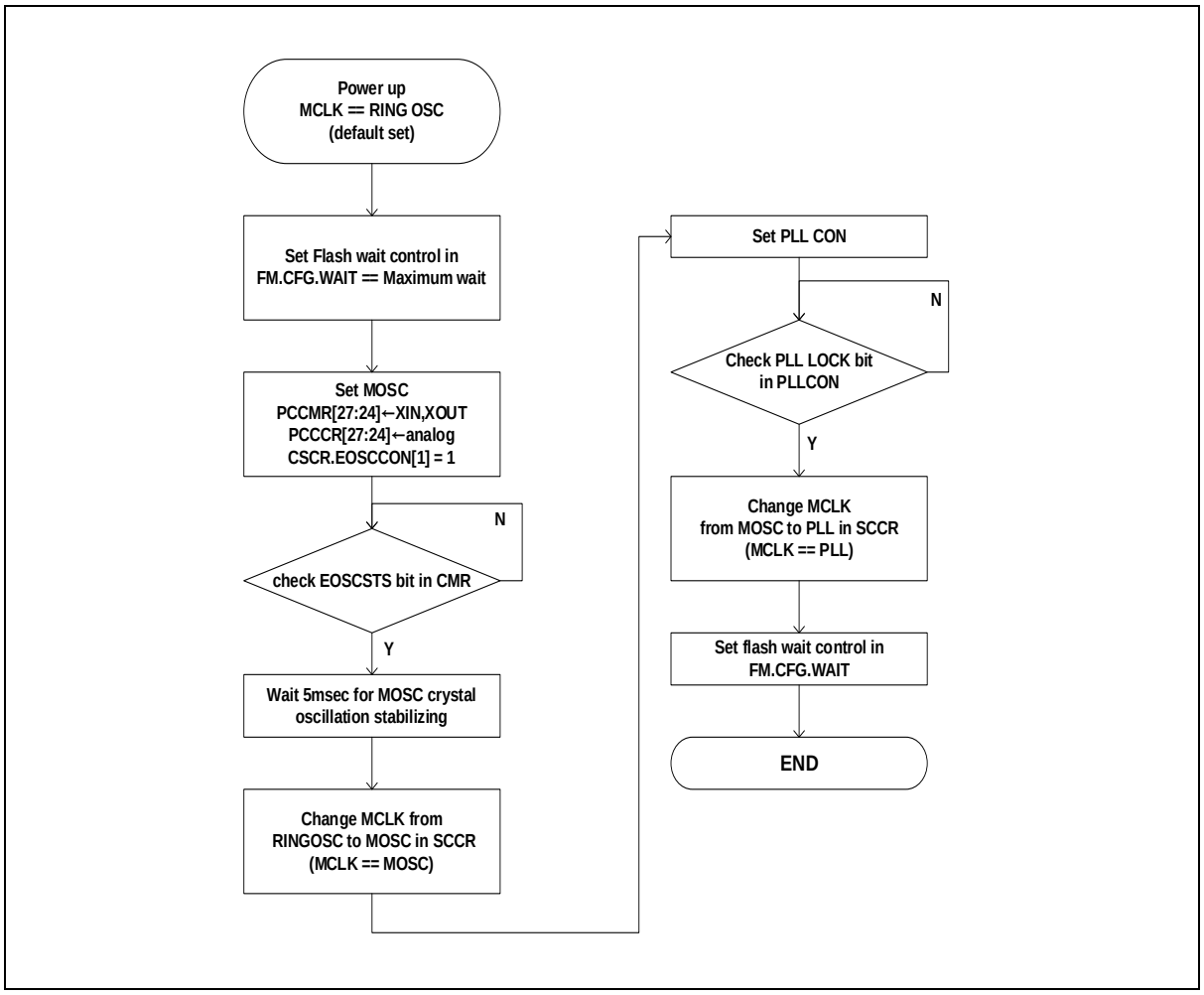


Figure 13. Clock Change Procedure

When you speed up the system clock until max operating frequency, you should check flash wait control configuration. Flash read access time is one of limitation factor for the performance. The wait control recommendation is provided in Table 7.

Table 7. Flash Wait Control Recommendation

FM.CFG.WAIT	FLASH Access Wait	Available Max System clock frequency
000	0 clock wait	Up to 16MHz
001	1 clock wait	Up to 32MHz
010	2 clock wait	Up to 48MHz
011	3 clock wait	Up to 48MHz

4.3 Reset

AC33Mx064T series has two system reset options. One is a cold reset that is effective during power up or down sequence. The other is a 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 8.

Table 8. Reset Sources

	Reset
Reset sources	• nRESET pin • WDT reset • LVD reset • MCLK Fail reset • MOSC Fail reset • S/W reset • CPU request reset

4.3.1 Cold reset

The cold reset is important feature of the chip when power is up. This characteristic will globally affect the system boot. Internal VDC is enabled when VDD power is turn on. Internal VDD level slope will follow by External VDD power slope. Internal PoR trigger level is 1.4V of internal VDC voltage out level. At this time, boot operation is started. The RING OSC clock is enabled and counts 4msec time for internal VDC level stabilizing. In this time, external VDD voltage level should be over than initial LVD level (2.3V). After 4msec counting, the CPU reset is released and start the operation.

Figure 14 shows the power-up process and the initial reset waveforms.

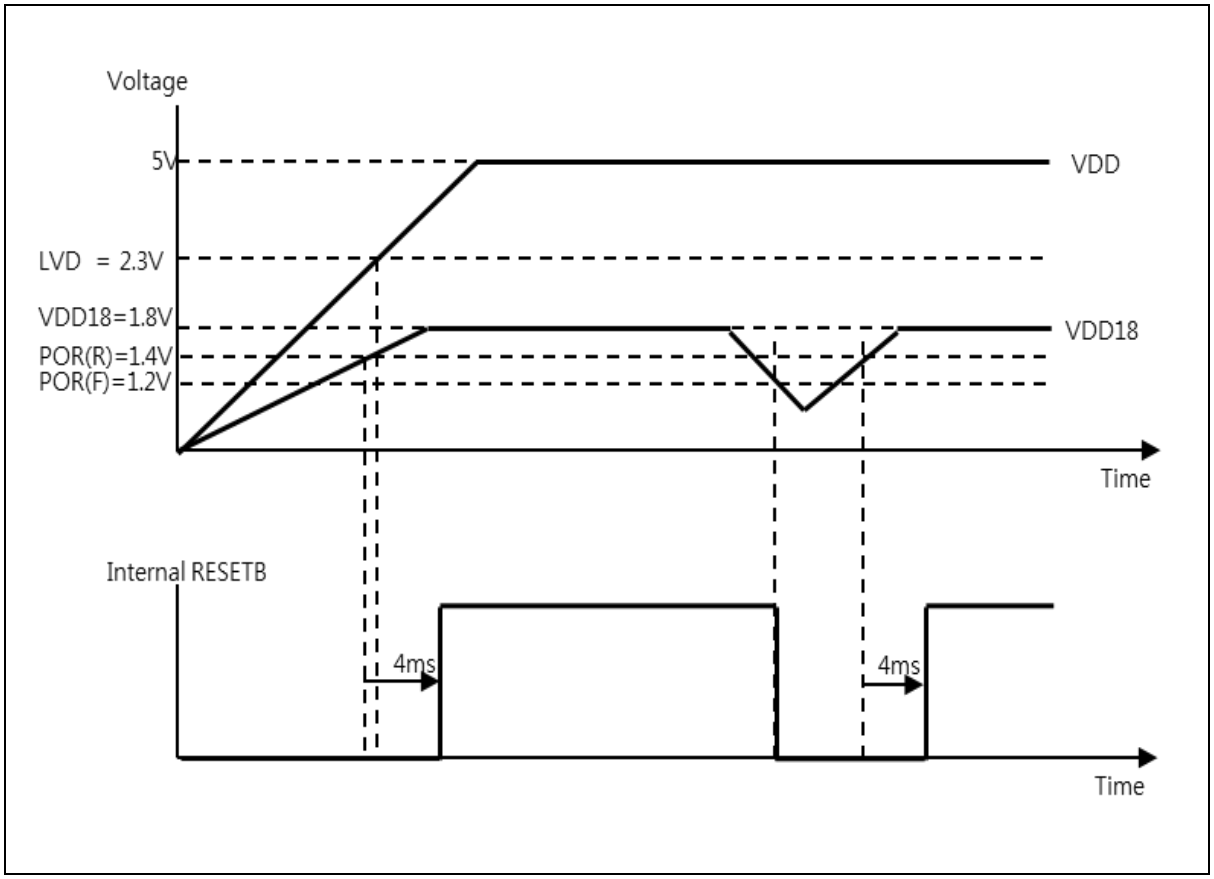


Figure 14. Power-Up POR Sequence

RSSR register shows the POR reset status. The last reset is come from POR, RSSR.PORST is set to "1". After power up, this bit is always "1". If abnormal internal voltage drop is occurred during normal operation, the system will be reset and this bit also set to "1".

When the cold reset applied, all the chip returns to initial state.

4.3.2 Warm reset

The warm reset event has several reset sources and some parts of chip returns to initial state when warm reset condition is occurred. The warm reset source is controlled by RSER register and the status is appeared in RSSR register. The reset for each peripheral blocks is controlled by PRER register. The reset can be masked independently.

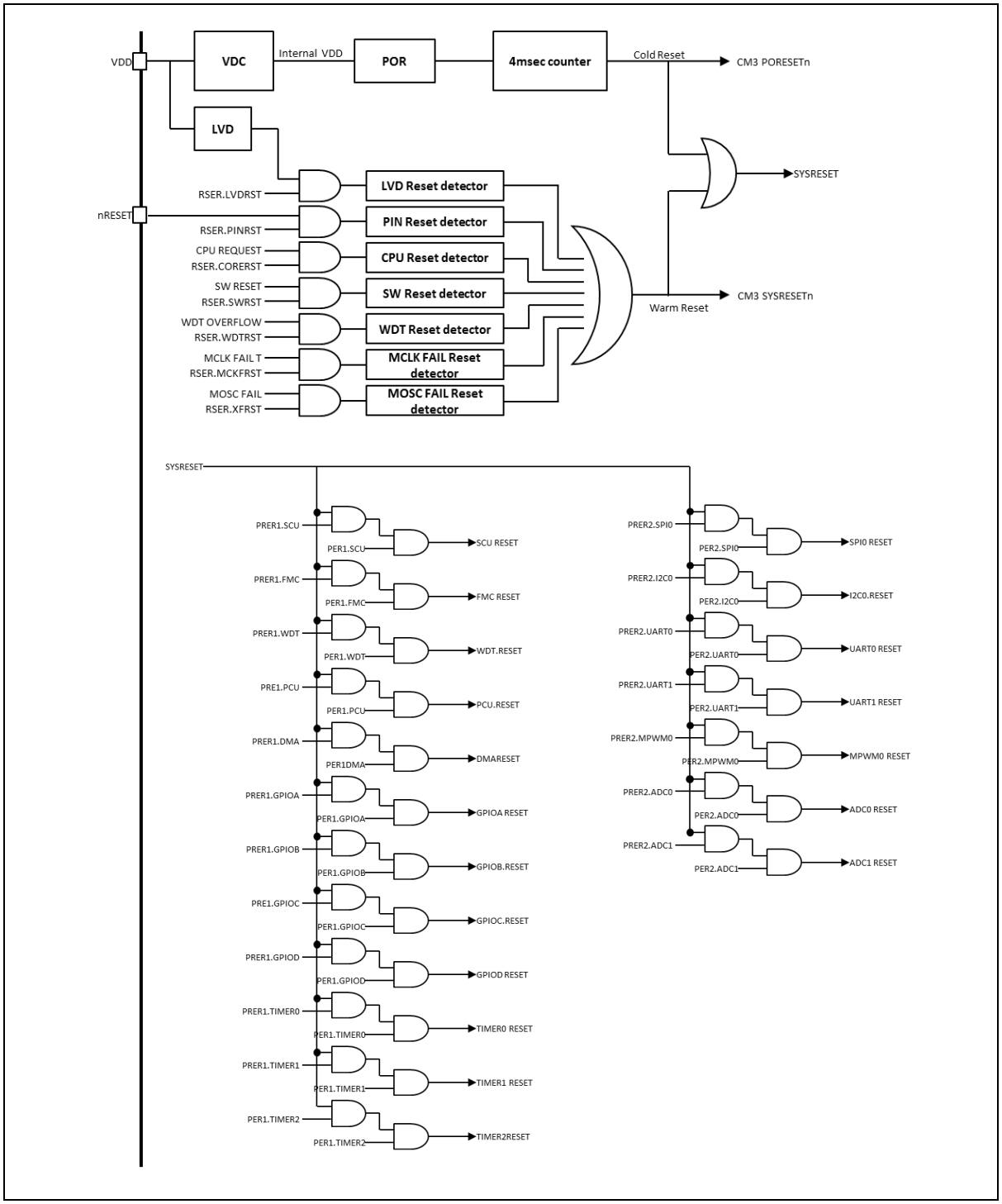


Figure 15. Reset Configuration

4.4 Operation mode

The INIT mode is initial state of the chip when reset is asserted. The RUN mode is max performance of the CPU with high-speed clock system. And the SLEEP mode can be used as the low power consumption mode. The low power consumption is achieved by halting processor core and unused peripherals.

Figure 16 shows the operation mode transition diagram.

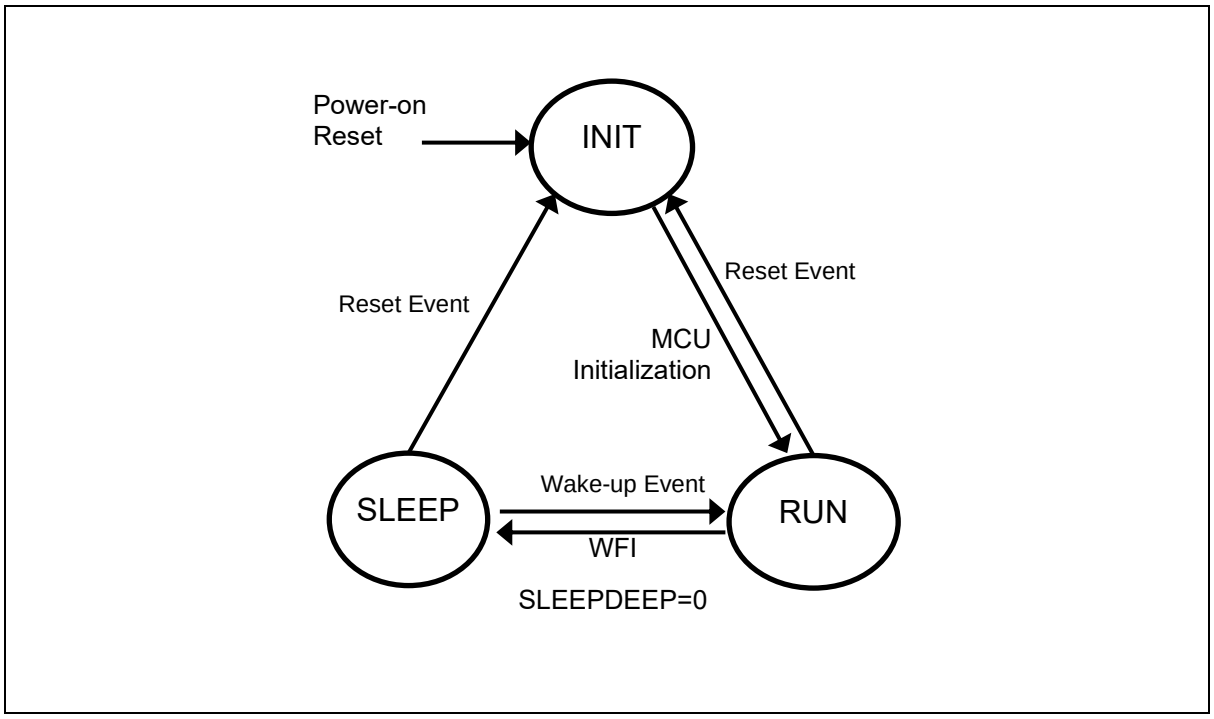


Figure 16. Operation Modes

4.4.1 RUN mode

This mode is to operate the CPU core and the peripheral hardware by using the high-speed clock. After reset followed by INIT state, it is entered into RUN mode.

4.4.2 SLEEP mode

Only the CPU is stopped in this mode. Each peripheral function can be enabled by the function enable and clock enable bit in the PER and PCER register.

4.5 Registers

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

Table 9. Base Address of SCU

Name	Base address
SCU	0x4000_0000

Table 10. SCU Register Map

Name	Offset	Type	Description	Reset value	Reference
CIDR	0x0000	R	CHIP ID Register	AC33_4064	4.5.1
SMR	0x0004	RW	System Mode Register	0000_0000	4.5.2
SRCR	0x0008	RW	System Reset Control Register	0000_0000	4.5.3
WUER	0x0010	RW	Wake up source enable register	0000_0000	4.5.4
WUSR	0x0014	RW	Wake up source status register	0000_0000	4.5.5
RSER	0x0018	RW	Reset source enable register	0000_0049	4.5.6
RSSR	0x001C	RW	Reset source status register	0000_0080*	4.5.7
PRER1	0x0020	RW	Peripheral reset enable register 1	030F_0F3F*	4.5.8
PRER2	0x0024	RW	Peripheral reset enable register 2	0031_0311*	4.5.9
PER1	0x0028	RW	Peripheral enable register 1	0000_000F*	4.5.10
PER2	0x002C	RW	Peripheral enable register 2	0000_0101*	4.5.11
PCER1	0x0030	RW	Peripheral clock enable register 1	0000_000F*	4.5.12
PCER2	0x0034	RW	Peripheral clock enable register 2	0000_0101*	4.5.13
CSCR	0x0040	RW	Clock Source Control register	0000_0020	4.5.14
SCCR	0x0044	RW	System Clock Control register	0000_0000	4.5.15
CMR	0x0048	RW	Clock Monitoring register	0000_0090	4.5.16
NMIR	0x004C	RW	NMI control register	0000_0000	4.5.17
COR	0x0050	RW	Clock Output Control register	0000_000F	4.5.18
PLLCON	0x0060	RW	PLL Control register	0000_0000	4.5.19
VDCCON	0x0064	RW	VDC Control register	0000_000F	4.5.20
LVDCON	0x0068	RW	LVD Control register	0000_0001	4.5.21
EOSCR	0x0080	RW	External Oscillator control register	0000_0300	4.5.22

Table 10. SCU Register Map (continued)

Name	Offset	Type	Description	Reset value	Reference
EMODR	0x0084	RW	External mode pin read register	0000_000X	4.5.23
MCCR1	0x0090	RW	Misc Clock Control register 1	0000_0000	4.5.24
MCCR2	0x0094	RW	Misc Clock Control register 2	0000_0000	4.5.25
MCCR3	0x0098	RW	Misc Clock Control register 3	0000_0001	4.5.26
MCCR4	0x009C	RW	Misc Clock Control register 4	0001_0001	4.5.27
MCCR5	0x00A0	RW	Misc Clock Control register 5	0001_0001	4.5.28
MCCR7	0x00A8	RW	Misc Clock Control register 6	0001_0000	4.5.29

4.5.1 CIDR Chip ID Register

CHIP ID Register shows chip identification information.

This register is 32-bit read-only register.

CIDR=0x4000_0000

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																															
0xAC33_4064																															
RO																															

31 0	CHIPID	Device ID 0xAC33_4064

CIDR=0x4000_000C

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
CHIPID2																															
0x0000_0000																															
RO																															

31 0	CHIPID2	Revision ID 0x0000_0000

4.5.2 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

System Mode Register is 16-bit register.

SMR=0x4000_0004

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
										PREVMODE					
0	0	0	0	0	0	0	0	0	0	00		0	0	0	0
										R					

5 4	PREVMODE	Previous operating mode before current reset event.	
		00	Previous operating mode was RUN mode
		01	Previous operating mode was SLEEP mode
		10	Previous operating mode was Power Down mode
		11	Previous operating mode was INIT mode

4.5.3 SRCR System Reset Control Register

System reset control register is 8-bit register.

SRCR=0x4000_0008

7	6	5	4	3	2	1	0
							SWRST
0	0	0	0	0	0	0	0
							W

1	SWRST	Internal soft reset activation bit	
		0	Normal operation
		1	Internal soft reset is applied and auto cleared

4.5.4 WUER Wakeup Source Enable Register

Enable wakeup source when the chip is in the Power Down 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'.

This register is 16-bit register.

WUER=0x4000_0010

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				GPIODWUE	GPIOCWUE	GPIOBWUE	GPIOAWUE							WDTWUE	LVDWUE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
				RW	RW	RW	RW							RW	RW

12	GPIOEWUE	Enable wakeup source of GPIOE port pin change event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
11	GPIODWUE	Enable wakeup source of GPIOD port pin 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
1	WDTWUE	Enable wakeup source of watchdog timer event
		0 Not used for wakeup source
		1 Enable the wakeup event generation
0	LVDWUE	Enable wakeup source of LVD event
		0 Not used for wakeup source
		1 Enable the wakeup event generation

4.5.5 WUSR Wakeup Source Status Register

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

GPIOWU status will be cleared by clearing interrupt status bit in PCn.ISR register of PCU block. WDTWU status will be cleared by clearing the interrupt status bit in WUF.WDT.CON register of WDT block. LDVWU status bit will be cleared when VDD level is over than LVD trigger level.

WUSR=0x4000_0014

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				GPIODWU	GPIOCWU	GPIOBWU	GPIOAWU							WDTWU	LVDWU
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
				R	R	R	R							R	R

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
1	WDTWU	Status of wakeup source of watchdog timer event
		0 No wakeup event
		1 Wakeup event was generated
0	LVDWU	Status of wakeup source of LVD event
		0 No wakeup event
		1 Wakeup event was generated

4.5.6 RSER Reset Source Enable Register

The reset source to the CPU can be selected by 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.

RSER=0x4000_0018

7	6	5	4	3	2	1	0
	PINRST	CORERST	SWRST	WDTRST	MCKFRST	XFRST	LVDRST
0	1	0	0	1	0	0	1
	RW	RW	RW	RW	RW	RW	RW

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	XFRST	External OSC Clock fail reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
0	LVDRST	LVD reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled

4.5.7 RSSR Reset Source Status Register

The RSSR shows the reset source information when reset event is occurred. "1" shows reset event was exist and "0" shows reset event is not exist for corresponding reset source. When reset source is founded, write "1" into the corresponding bit will clear the reset status. This register is 8-bit register

RSSR=0x4000_001C

7	6	5	4	3	2	1	0
PORST	PINRST	CORERST	SWRST	WDTRST	MCKFRST	XFRST	LVDRST
1	0	0	0	0	0	0	0
RC1	RC1	RC1	RC1	RC1	RC1	RC1	RC1

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	MCLKFRST	MCLK 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	XFRST	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

4.5.8 PRER1 Peripheral Reset Enable Register 1

The reset of each peripheral by event reset, can be masked by user setting. PRER1/PRER2 register will control the enable of the event 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.

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
						TIMER9	TIMER8					TIMER3	TIMER2	TIMER1	TIMER0					GPIOD	GPIOC	GPIOB	GPIOA				DMA	PCU	WDT	FMC	SCU
0	0	0	0	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	1	1	1	1	1
						RW	RW					RW	RW	RW	RW					RW	RW	RW	RW				RW	RW	RW	RW	RW

25	TIMER9	TIMER9 reset mask
24	TIMER8	TIMER8 reset mask
19	TIMER3	TIMER3 reset mask
18	TIMER2	TIMER2 reset mask
17	TIMER1	TIMER1 reset mask
16	TIMER0	TIMER0 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 Control Unit reset mask
2	WDT	Watchdog Timer reset mask
1	FMC	Flash memory controller reset mask
0	SCU	System Control Unit reset mask

4.5.9 PRER2 Peripheral Reset Enable Register 2

Peripheral Reset Enable Register 2 is 32-bit register.

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
										ADC1	ADC0				MWPM0							UART1	UART0				I2C0				SPI0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1
										RW	RW				RW							RW	RW				RW				RW

21	ADC1	ADC1 reset enable
20	ADC0	ADC0 reset enable
16	MPWM0	MPWM0 reset enable
9	UART1	UART1 reset enable
8	UART0	UART0 reset enable
4	I2C0	I^2C0 reset enable
0	SPI0	SPI0 reset enable

4.5.10 PER1 Peripheral Enable Register 1

To use peripheral unit, it should be activated by writing "1"to the correspond bit in the PER1/PER2 register. Before the activation, the peripheral will stay in reset state.

All the peripherals enabled by default. To disable the peripheral unit, write "0" to the correspond bit in the PER0/PER1 register, and then the peripheral enter the reset state.

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
						TIMER9	TIMER8					TIMER3	TIMER2	TIMER1	TIMER0					GPIOD	GPIOC	GPIOB	GPIOA				DMA				
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
						RW	RW					RW	RW	RW	RW					RW	RW	RW	RW				RW	R	R	R	R

25	TIMER9	TIMER9 function enable
24	TIMER8	TIMER8 function enable
19	TIMER3	TIMER3 function enable
18	TIMER2	TIMER2 function enable
17	TIMER1	TIMER1 function enable
16	TIMER0	TIMER0 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
3		Reserved
2		
1		
0		

4.5.11 PER2 Peripheral Enable Register 2

Peripheral Enable Register 2 is 32-bit register.

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
										ADC1	ADC0				MWPM0							UART1	UART0				I2C0				SPI0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
										RW	RW				RW							RW	RW				RW				RW

21	ADC1	ADC1 function enable
20	ADC0	ADC0 function enable
16	MPWM0	MPWM0 function enable
9	UART1	UART1 function enable
8	UART0	UART0 function enable
4	I2C0	I^2C0 function enable
0	SPI0	SPI0 function enable

4.5.12 PCER1 Peripheral Clock Enable Register 1

To use peripheral unit, its clock should be activated by writing '1'to the corresponding bit in the PCER1/PCER2 register. Before enabling its clock, the peripheral won't operate properly.

To stop the clock of the peripheral unit, write '0'to the correspond bit in the PCER1/PCER2 register, and then the clock of the peripheral is stopped.

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
						TIMER9	TIMER8					TIMER3	TIMER2	TIMER1	TIMER0					GPIOD	GPIOC	GPIOB	GPIOA				DMA				
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
						RW	RW					RW	RW	RW	RW					RW	RW	RW	RW				RW	R	R	R	R

25	TIMER9	TIMER9 clock enable
24	TIMER8	TIMER8 clock enable
19	TIMER3	TIMER3 clock enable
18	TIMER2	TIMER2 clock enable
17	TIMER1	TIMER1 clock enable
16	TIMER0	TIMER0 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
3		Reserved
2		
1		
0		

4.5.13 PCER2 Peripheral Clock Enable Register 2

To use peripheral unit, its clock should be activated by writing '1' to the correspond

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
										ADC1	ADC0				MWPM0							UART1	UART0				I2C0				SPI0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
										RW	RW				RW							RW	RW				RW				RW

21	ADC1	ADC1 clock enable
20	ADC0	ADC0 clock enable
16	MPWM0	MPWM0clock enable
9	UART1	UART1 clock enable
8	UART0	UART0 clock enable
4	I2C0	I^2C0 clock enable
0	SPI0	SPI0 clock enable

4.5.14 CSCR Clock Source Control Register

The AC33Mx064T has multiple clock sources to generate internal operating clocks. Each clock sources can be controlled by CSCR register.

This register is 8-bit register.

CSCR=0x4000_0040

7	6	5	4	3	2	1	0
-		RINGOSCCON		-		EOSCCON	
00		10		00		00	
R		RW		R		RW	

5	RINGOSCCON	Internal ring oscillator control	
4		0X	Stop internal sub oscillator
		10	Enable internal sub oscillator
		11	Enable internal sub oscillator divide by 2
1	EOSCON	External crystal oscillator control	
0		0X	Stop External Crystal oscillator
		10	Enable External Crystal oscillator
		11	Enable External Crystal divide by 2

4.5.15 SCCR System Clock Control Register

The AC33Mx064T has multiple clock sources to generate internal operating clocks. Each clock sources can be controlled by SCUCSCR register.

SCCR=0x4000_0044

7	6	5	4	3	2	1	0
-					FINSEL	MCLKSEL	
0000					0	00	
R					RW	RW	

2	FINSEL	PLL input source FIN select register	
		0	IOSC clock is used as FIN clock
		1	MOSC clock is used as FIN clock
1 0	MCLKSEL	System clock select register	
		0X	Internal sub oscillator
		10	PLL bypassed clock
		11	PLL output clock

When change FINSEL, both of internal OSC and external OSC should be alive, otherwise the chip will do mal function

4.5.16 CMR Clock Monitoring Register

Internal clock can be monitored by internal sub oscillator for security purpose.

Clock Monitoring Register is 16-bit register.

CMR=0x4000_0048

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCLKREC								MCLKMNT	MCLKIE	MCLKFAIL	MCLKSTS	EOSCMNT	EOSCIE	EOSCFAIL	EOSCSTS
0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R								RW	RW	RC1	RC1	RW	RW	RC1	RC1

15	MCLKREC	MCLK fail auto recovery
		0 MCLK is changed to RING OSC by default when MCLKFAIL issued
		1 MCLK auto recovery is disabled
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	EOSCMNT	External oscillator monitoring enable
		0 External oscillator monitoring disabled
		1 External oscillator monitoring enabled
2	EOSCIE	External oscillator fail interrupt enable
		0 External oscillator fail interrupt disabled
		1 External oscillator fail interrupt enabled
1	EOSCFAIL	External oscillator fail interrupt
		0 External oscillator fail interrupt not occurred
		1 Read : External oscillator fail interrupt is pending Write : Clear pending interrupt
0	EOSCSTS	External oscillator status
		0 Not oscillate
		1 External oscillator is working normally

Clock monitoring function cannot cover all mal function cases. It is just used for the reference. Fig1.8 shows the operational diagram for clock monitoring function.

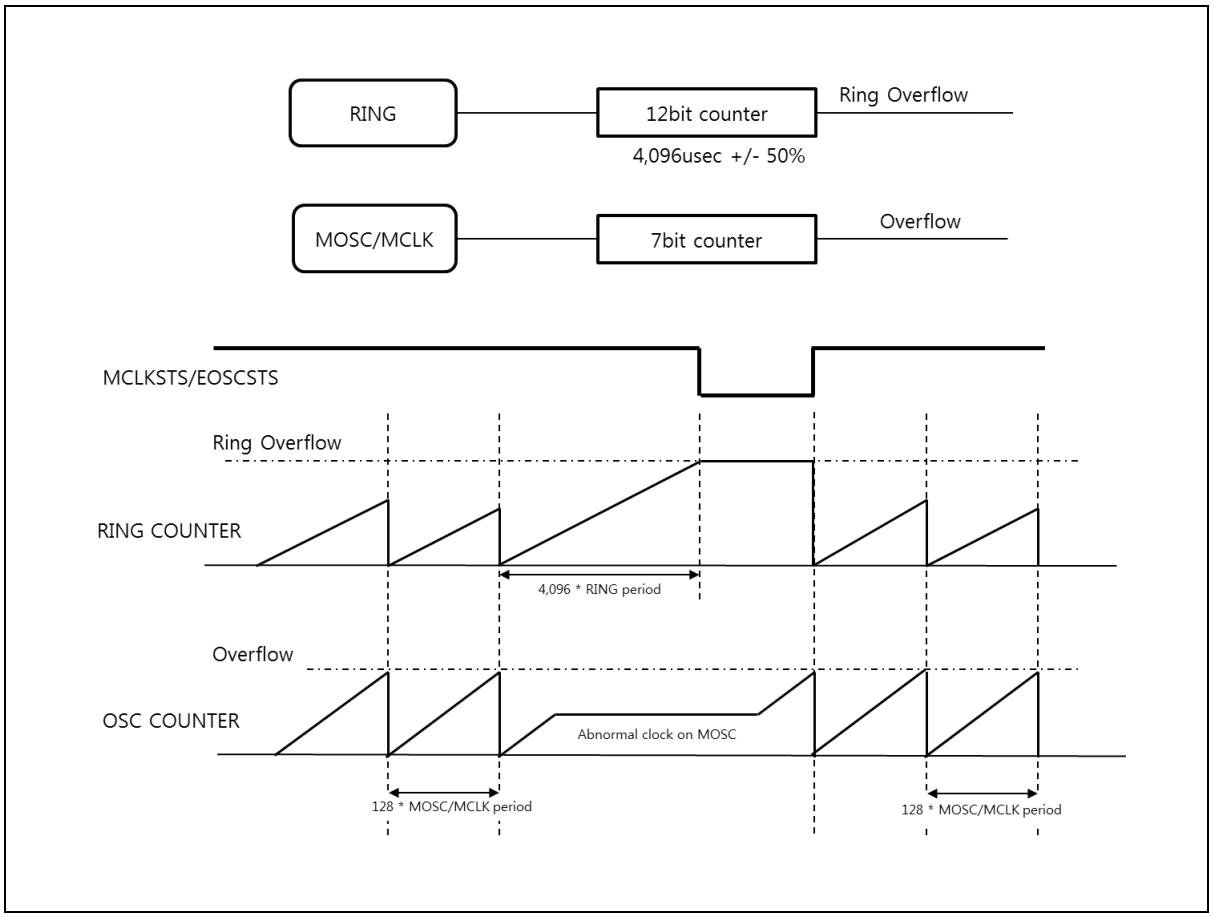


Figure 17. Clock Monitoring Function Diagram

4.5.17 NMIR NMI Control Register

NMIR is the non-maskable interrupt configuration register which can be set by software. There are five kinds of interrupt sources from MPWM, WDT and SCU.

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

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																			PROTSTS	OVPSTS	WDTSTS	MCLKFAILST	LVDSTS				PROTEN	OVPEN	WDTEN	MCLKFAILEN	LVDEN
-																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
WO																			R	R	R	R	R				RW	RW	RW	RW	RW

31 16	ACCESSCODE	This field enables writing access to this register. Writing 0xA32C is to enable writing.
12	PROTSTS	Protection condition status bit. This bit can't invoke nmi interrupt without enable bit
		0 Not occurred
		1 Event occurred
11	OVPSTS	Over Voltage Protection condition status bit This bit can't invoke nmi interrupt without enable bit
		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
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 condition status bit This bit can't invoke nmi interrupt without enable bit
		0 Not occurred
		1 Event occurred
4	PROTEN	Protection condition enable for NMI interrupt
		0 Disable
		1 Enable
3	OVPEN	Over Voltage Protection condition enable for NMI interrupt
		0 Disable
		1 Enable
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	LVDEN	LVD Fail condition enable for NMI interrupt
		0 Disable
		1 Enable

4.5.18 COR Clock Output Register

The AC33Mx064 can drive the clock from internal MCLK clock with dedicated post divider. Clock Output Register is 8-bit register.

COR=0x4000_0050

7	6	5	4	3	2	1	0
-			CLKOEN	CLKODIV			
000			0	1111			
R			RW	RW			

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

4.5.19 PLLCON PLL Control Register

Integrated PLL will synthesize high speed clock for extremely high performance of the CPU. The PLL controlled by register setting. PLL Control Register is 16-bit register.

PLLCON=0x4000_0060

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PLLRSTB	PLLEN	BYPASS	LOCKSTS				PREDIV	FBCTRL				POSTDIV			
0	0	0	0	0	0	0	0	0000				0000			
RW	RW	RW	R				RW	RW				RW			

15	PLLRSTB	PLL reset			
		0	PLL reset is asserted		
		1	PLL reset is negated		
14	PLLEN	PLL enable			
		0	PLL is disabled		
		1	PLL is enabled		
13	BYPASS	FIN bypass			
		0	FOUT is bypassed as FIN		
		1	FOUT is PLL output		
12	LOCK	LOCK status			
		0	PLL is not locked		
		1	PLL is locked		
8	PREDIV	FIN predivider			
		0	FIN divided by 1		
		1	FIN divided by 2		
7	FBCTRL	Feedback control			
4		0000	M = 4	1000	M = 20
		0001	M = 6	1001	M = 24
		0010	M = 8	1010	M = 26
		0011	M = 10	1011	M = 34
		0100	M = 12	1100	Not available
		0101	M = 14	1101	
		0110	M = 16	1110	
		0111	M = 18	1111	
3	POSTDIV	Post divider control			
0		000	N = 1		
		001	N = 2		
		010	N = 3		
		011	N = 4		
		100	N = 6		
		101	N = 8		
		110	N = 3		
		111	N =16		

NOTES:

1. *M: Feedback control value, N: Post divider control value*
2. You can change PLL clock with these are equations.

   $\mathrm{PLL(MHz)} = \mathrm{FINCLK(MHz)} \times \frac{1}{1} \times M \times \frac{1}{N}$, (PREDIV[8] = 0).

   $\mathrm{PLL(MHz)} = \mathrm{FINCLK(MHz)} \times \frac{1}{2} \times M \times \frac{1}{N}$, (PREDIV[8] = 1).
3. If you want to make 48MHz PLL clock with 8MHz FINCLK, You can choose PREDIV[8] = 1, M = 12, N = 1.

   $(48(\mathrm{MHz}) = 8(\mathrm{MHz}) \times \frac{1}{2} \times 12 \times \frac{1}{1})$

4.5.20 VDCCON VDC Control Register

On chip VDC control register. VDCTRIM is used for the trim value of VDC output. To modify VDCTRIM bit, VDCTE should be write "1"simultaneously. VDCWDLY value can be written with writing "1" to VDCDE bit simultaneously.

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
RESERVE					RESERVED			VDCTE				VDCTRIM											VDCDE	VDCWDLY							
0	0	0	0	0	00			0	0	0	0	0000				0	0	0	0	0	0	0	0	0x7F							
W					RW			W				RW											W	RW							

31	RESERVE	Reference reserved write enable. 0 RESERVED field is not updated by writing 1 RESERVED filed can be updated by writing
26 24	RESERVED	Reserved data for the future
23	VDCTE	VDCTRIM value write enable. Write only with VDCTRIM value. 0 VDCTRIM field is not updated by writing 1 VDCTRIM filed can be updated by writing
19 16	VDCTRIM	VDC output voltage trim value Not recommended strongly to write into this field.
8	VDCDE	VDCWDLY value write enable. Write only with VDCWDLY value 0 VDCWDLY field is not updated by writing 1 VDCWDLY field can be updated by writing
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 can be defined with this register value 7F : 2msec

4.5.21 LVDCON LVD Control Register

On chip Brown-out detector control register. This register is 32-bit register.

LVDCON=0x4000_0068

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
								LVDTE						LVDTRIM		SELEN						SEL								LVDLVL	LVDEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	00		0	0	0	0	0	0	00		0	0	0	0	0	0	0	1
								W						RW		W						RW								R	RW

23	LVDTE	LVDTRIM value write enable. Write only with LVDTRIM value.
		0 LVDTRIM field is not updated by writing
		1 LVDTRIM filed can be updated by writing
17 16	LVDTRIM	LVD voltage level trim value Not recommended strongly to write into this field
15	SELEN	LVDSEL value write enable. Write only with LVDSEL value.
		0 SEL field is not updated by writing
		1 SEL filed can be updated by writing
9 8	LVDSEL	LVD detect level select
		00 LVD detect level is 1.8V
		01 LVD detect level is 2.2V
		10 LVD detect level is 2.7V
		11 LVD detect level is 4.3V
1	LVDLVL	LVD Status
		0 VDDEXT level is over than LVD level
		1 VDDEXT level is under than LVD level
0	LVDEN	LVD Function enable
		0 LVD is not enabled
		1 LVD is enabled

4.5.22 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. This register is 16-bit register.

EOSCR=0x4000_0080

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISELEN						ISEL									
0	0	0	0	0	0	11		0	0	0	0	0	0	0	0
W						RW									

15	ISELEN	Write enable of bit field ISEL.	
		0	Write access of ISEL field is masked
		1	Write access of ISEL field is accepted
9 8	ISEL	Select current.	
		00	Minimum current driving option
		01	Low current driving option
		10	High current driving option
		11	Maximum current driving option

4.5.23 EMODRExternal Mode Status Register

External Mode Status Register shows external mode pin status while booting. This register is 8-bit register.

EMODR=0x4000_0084

7	6	5	4	3	2	1	0
					Reserved	Reserved	BOOT
0x0					0	0	-
R					R	R	R

0	BOOT	BOOT pin level
		0 BOOT(PC11) pin is low
		1 BOOT(PC11) pin is high

4.5.24 MCCR1 Miscellaneous Clock Control Register 1

The AC33Mx064T can drive the clock from internal MCLK clock with dedicated post divider. This register is 32-bit register.

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
																					STCSEL			SYSTICKDIV							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	000			0x00							
																					RW			RW							

10 8	STCSEL	SYSTIC Clock source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
7 0	STDIV	SYSTIC Clock N divider When STDIV is "0", the systick clock is stopped.

4.5.25 MCCR2 Miscellaneous Clock Control Register 2

The AC33Mx064T can drive the clock from internal MCLK clock with dedicated post divider. This register is 32-bit register.

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
																					PWM0CSEL			PWM0DIV							
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	000			0x00							
																					RW			RW							

10 8	PWM0CSEL	PWM0 Clock source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
7 0	PWM0DIV	PWM0 Clock N divider When PWM0DIV is "0", the PWM clock is stopped.

4.5.26 MCCR3 Miscellaneous Clock Control Register 3

The AC33Mx064T can drive the clock from internal MCLK clock with dedicated post divider. This register is 32-bit register.

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
					TIMERCSEL			TIMERDIV														WDTCSEL			WDTDIV							
0	0	0	0	0	000			0x01								0	0	0	0	0	000			0x01							
					RW			RW													RW			RW							

26 24	TIMERCSEL	Timer Clock source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
23 16	TIMERDIV	Timer Clock N divider When TIMERDIV is "0", the TIMER clock is stopped
10 8	WDTCSEL	WDT Clock source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
7 0	WDTDIV	WDT Clock N divider When WDTDIV is "0", the WDT clock is stopped

4.5.27 MCCR4 Miscellaneous Clock Control Register 4

The AC33Mx064T can drive the clock from internal MCLK clock with dedicated post divider. This register is 32-bit register.

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
					PBDCSEL			PBDDIV														PADCSEL			PADDIV							
0	0	0	0	0	000			0x01								0	0	0	0	0	000			0x01							
					RW			RW													RW			RW							

26 24	PBDCSEL	Debounce Clock for Port B source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
23 16	PBDDIV	PORT B Debounce Clock N divider When PBDDIV is "0", the PORT B Debounce clock is stopped
10 8	PADCSEL	Debounce Clock for Port A source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
7 0	PADDIV	PORT A Debounce Clock N divider When PADDIV is "0", the PORT A Debounce clock is stopped

4.5.28 MCCR5 Miscellaneous Clock Control Register 5

The AC33Mx064T can drive the clock from internal MCLK clock with dedicated post divider. This register is 32-bit register.

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
					PDDCSEL			PDDDIV														PCDCSEL			PCDDIV							
0	0	0	0	0	000			0x01								0	0	0	0	0	000			0x01							
					RW			RW													RW			RW							

26 24	PDDCSEL	Debounce Clock for PORT D source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
23 16	PDDDIV	PORT D Debounce Clock N divider When PDDDIV is "0", the PORT D Debounce clock is stopped
10 8	PCDCSEL	Debounce Clock for PORT C source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
7 0	PCDDIV	PORT C Debounce Clock N divider When PCDDIV is "0", the PORT C Debounce clock is stopped

4.5.29 MCCR7 Miscellaneous Clock Control Register 7

The AC33Mx064T can drive the clock from internal MCLK clock with dedicated post divider. .

MCCR7=0x4000_00A8

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
					ADCCSEL			ADCCDIV																								
0	0	0	0	0	000			0x01																							
					RW			RW																								

26 24	ADCCSEL	ADC clock source select bit 0xx RING OSC 1Mhz 100 MCLK (bus clock) 101 Reserved 110 External Main OSC (XTAL) 111 Reserved
23 16	ADCCDIV	ADC Clock N divider When ADCCDIV is "0", the ADC clock is stopped

5 Port Control Unit (PCU)

AC33Mx064T MCU's Port Control Unit (PCU) block controls the external input and output (I/O) ports.

The PCU configures and controls external I/Os as listed in the following ways:

- Set external signal directions of each pins
- Set interrupt trigger mode for each pins
- Set internal pull-up register control and open drain control

Table 11 are assigned for PCU blocks.

Table 11. PCU Pins

Pin name	Type	Description
PA	IO	PA0 to PA15
PB	IO	PB0 to PB7
PC	IO	PC0 to PC15
PD	IO	PD0 to PD3

5.1 PCU block diagram

Figure 18 describes PCU in block diagram.

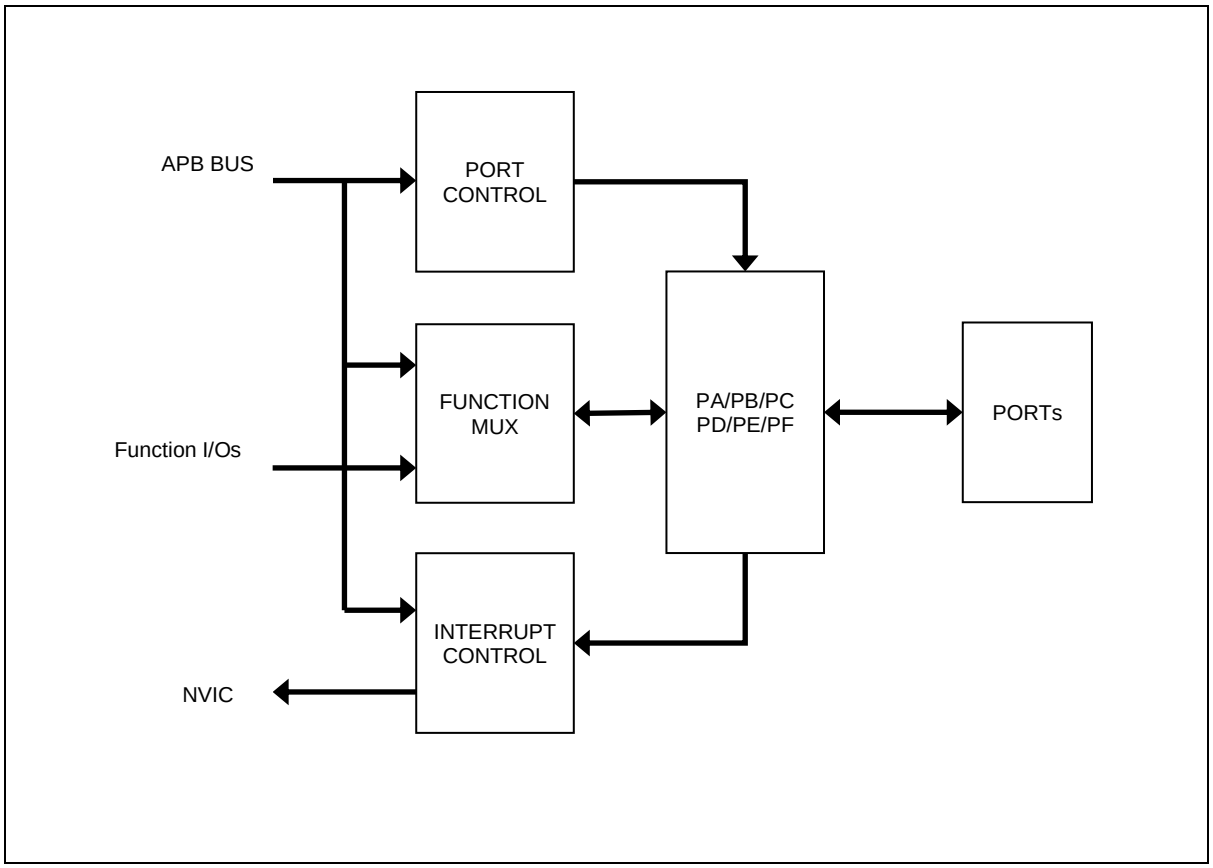


Figure 18. PCU Block Diagram

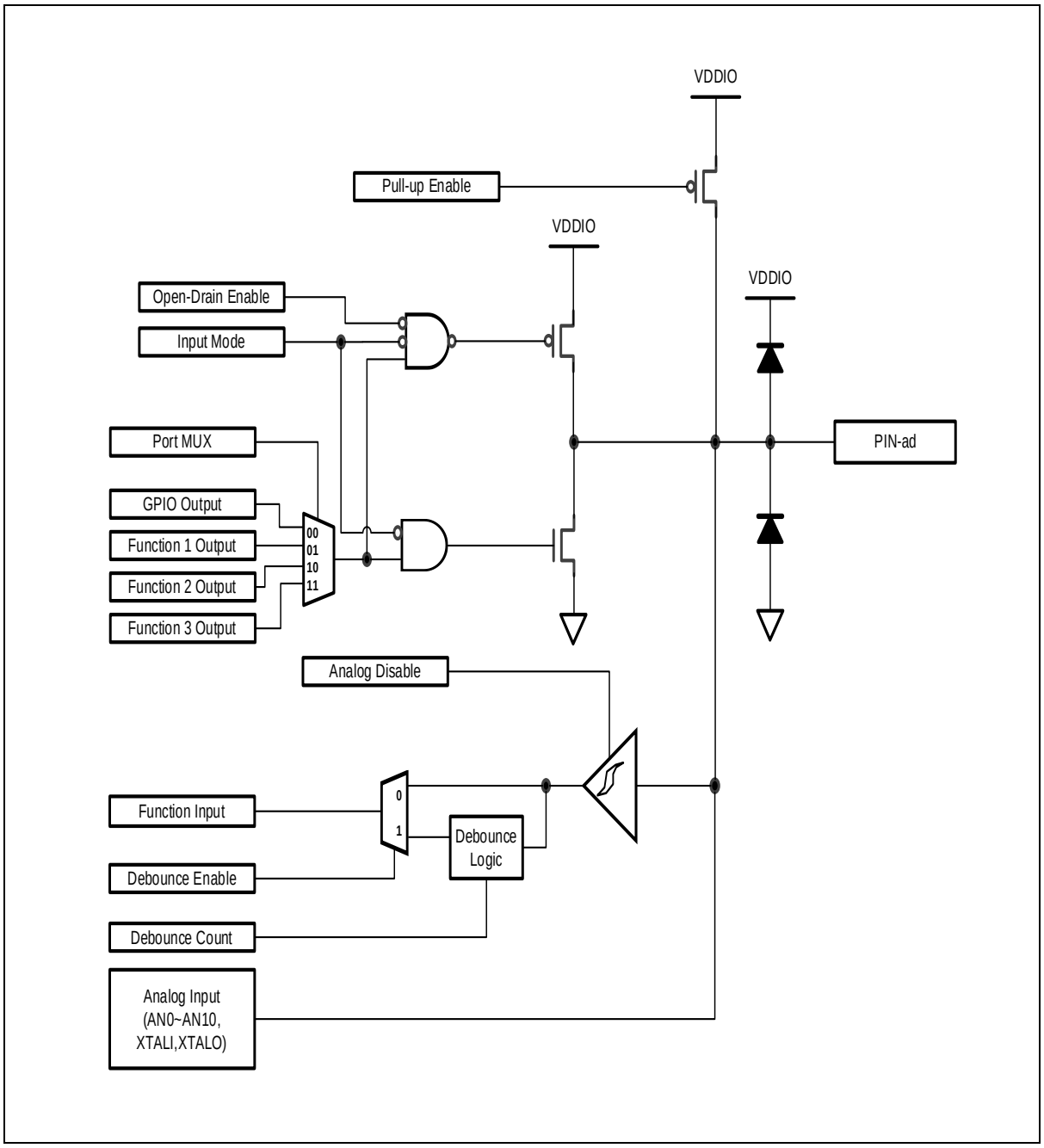


Figure 19. I/O Port Block Diagram (ADC and External Oscillator Pins)

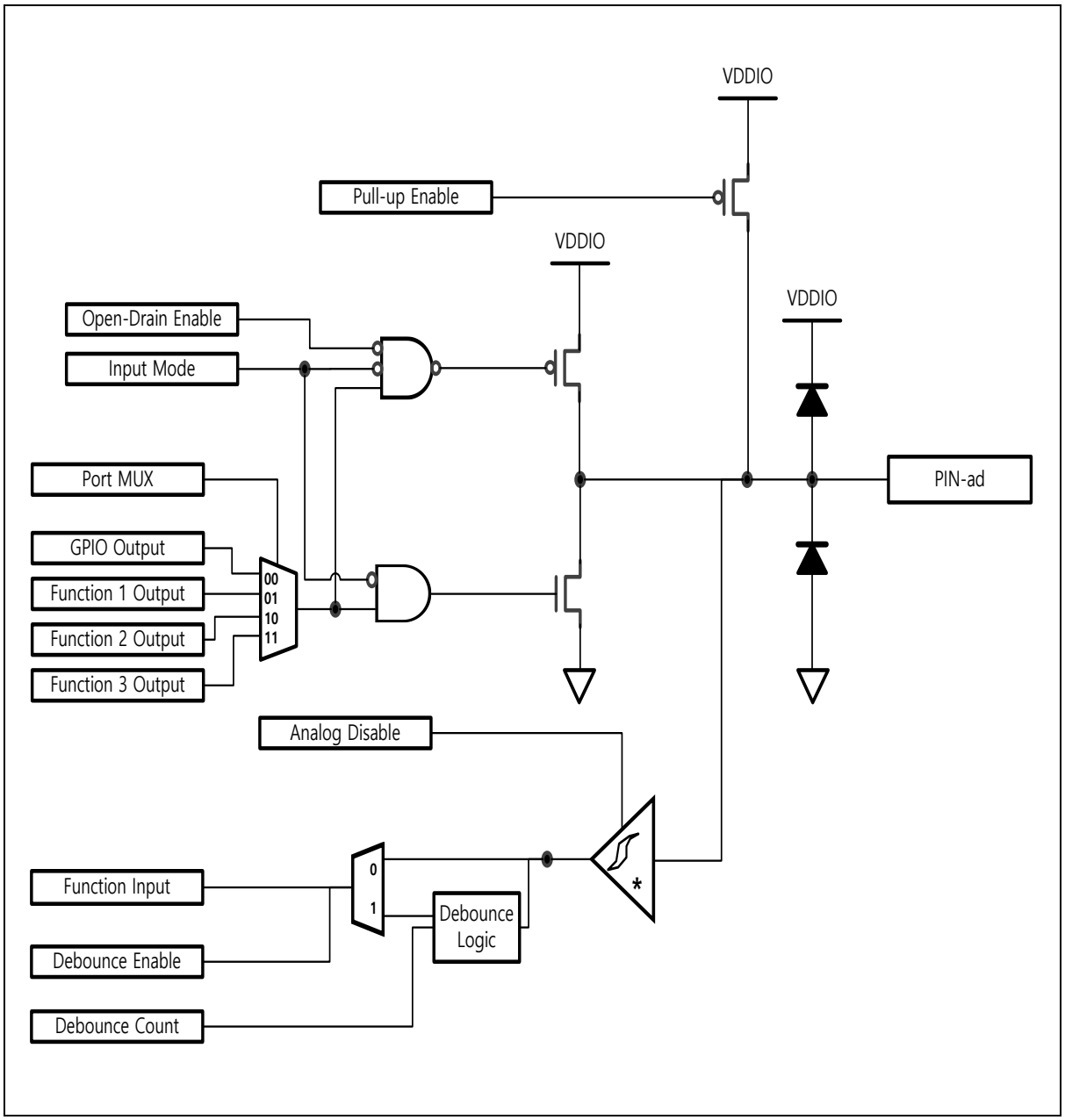


Figure 20. I/O Port Block Diagram (General I/O Pins)

5.2 Pin multiplexing

GPIO pins have alternative function pins. Table 12 shows pin multiplexing information.

Table 12. GPIO Alternative Function

Pin name	Alternative function			
	00	01	10	11
PA0	PA0*			AN0
PA1	PA1*			AN1
PA2	PA2*		WDTO	AN2
PA3	PA3*		STBO	AN3
PA4	PA4*	SS1		AN4
PA5	PA5*	SS2		AN5
PA6	PA6*	T0IO	T2IO	AN6
PA7	PA7*	T1IO	T3IO	AN7
PA8	PA8*	T2IO	T0IO	AN8
PA9	PA9*	T3IO	T1IO	AN9
PA10	PA10*	SS3		AN10
PA11	PA11*			
PA12	PA12*	T0IO		
PA13	PA13*	T1IO		
PA14	PA14*	T2IO		
PA15	PA15*	T3IO		
PB0	PB0*	MP0UH		
PB1	PB1*	MP0UL		
PB2	PB2*	MP0VH		
PB3	PB3*	MP0VL		
PB4	PB4*	MP0WH		
PB5	PB5*	MP0WL		
PB6	PB6*	PRTIN0	T0IO	
PB7	PB7*	OVIN0	T1IO	
PB8				
PB9				
PB10				
PB11				
PB12				
PB13				
PB14				

Table 12. GPIO Alternative Function (continued)

Pin name	Alternative function			
	00	01	10	11
PB15				
PC0	PC0	TCK/SWCLK*	RXD1	
PC1	PC1	TMS/SWDIO*	TXD1	
PC2	PC2	TDO/SWO*	T8IO	
PC3	PC3	TDI*	T9IO	
PC4	PC4	nTRST*	T0IO	
PC5*	PC5*	RXD1	T1IO	
PC6*	PC6*	TXD1	T2IO	
PC7*	PC7*	SCL0	T3IO	
PC8*	PC8*	SDA0		
PC9*	PC9*	CLKO		
PC10	PC10	nRESET*	T8IO	
PC11/BOOT*	PC11/BOOT*		T9IO	
PC12*	PC12*			XIN
PC13*	PC13*			XOUT
PC14*	PC14*	RXD0	MOSI0(2)	
PC15*	PC15*	TXD0	MISO0(2)	
PD0*	PD0*	SS0	T8IO	
PD1*	PD1*	SCK0	T9IO	
PD2*	PD2*	MOSI0	SCL0	
PD3*	PD3*	MISO0	SDA0	

NOTES:

1. (*) mark indicates default pin setting.
2. (2) mark indicates secondary port

5.3 Registers

Base address of PCU is introduced in the followings:

Table 13. Base Address of PCU

Name	Base address	Description
PCA	0x4000_1000	General Port A
PCB	0x4000_1100	General Port B
PCC	0x4000_1200	General Port C
PCD	0x4000_1300	General Port D

Table 14. PCU and GPIO Register Map

Name	Offset	Type	Description	Reference
PC*n*.MR	0x--00	RW	Port *n* pin mux select register	5.3.1
PC*n*.CR	0x--04	RW	Port *n* pin control register	5.3.5
PC*n*.PCR	0x--08	RW	Port *n* internal pull-up control register	5.3.7
PC*n*.DER	0x--0C	RW	Port *n* debounce control register	5.3.8
PC*n*.IER	0x--10	RW	Port *n* interrupt enable register	5.3.9
PC*n*.ISR	0x--14	RW	Port *n* interrupt status register	5.3.10
PC*n*.ICR	0x--18	RW	Port *n* interrupt control register	5.3.11
	0x--1C		Reserved	
PORTEN	0x1FF0	RW	Port Access enable	5.3.12

5.3.1 PCA.MR PORT A Pin MUX Register

PA port mode select register. This register must be set properly before use the port. Otherwise the port can't guarantee its functionality.

PCA.MR=0x4000_1000

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
PA15	PA14	PA13	PA12	PA11	PA10	PA9	PA8	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
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

Table 15. PCA.MR Selection Bit

Port	Selection bit			
	00	**01**	**10**	**11**
PA0	PA0*			AN0
PA1	PA1*			AN1
PA2	PA2*		WDTO	AN2
PA3	PA3*		STBO	AN3
PA4	PA4*	SS1		AN4
PA5	PA5*	SS2		AN5
PA6	PA6*	T0IO	T2IO	AN6
PA7	PA7*	T1IO	T3IO	AN7
PA8	PA8*	T2IO	T0IO	AN8
PA9	PA9*	T3IO	T1IO	AN9
PA10	PA10*	SS3		AN10
PA11	PA11*			
PA12	PA12*	T0IO		
PA13	PA13*	T1IO		
PA14	PA14*	T2IO		
PA15	PA15*	T3IO		

5.3.2 PCB.MR PORT B Pin MUX Register

PB port mode select register. This register must be set properly before use the port. Otherwise the port can't guarantee its functionality.

PCB.MR=0x4000_1100

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
PB15	PB14	PB13	PB12	PB11	PB10	PB9	PB8	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
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

Table 16. PCB.MR Selection Bit

Port	Selection bit			
	00	**01**	**10**	**11**
PB0	PB0*	MP0UH		
PB1	PB1*	MP0UL		
PB2	PB2*	MP0VH		
PB3	PB3*	MP0VL		
PB4	PB4*	MP0WH		
PB5	PB5*	MP0WL		
PB6	PB6*	PRTIN0	T0IO	
PB7	PB7*	OVIN0	T1IO	

5.3.3 PCC.MR PORT C Pin MUX Register

PC port mode select register. This register must be set properly before use the port. Otherwise the port can't guarantee its functionality.

PCC.MR=0x4000_1200

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
PC15		PC14		PC13		PC12		PC11		PC10		PC9		PC8		PC7		PC6		PC5		PC4		PC3		PC2		PC1		PC0	
00		00		00		00		01		01		00		00		00		00		00		01		01		01		01		01	
RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW		RW	

Table 17. PCC.MR Selection Bit

Port	Selection bit			
	00	**01**	**10**	**11**
PC0	PC0	TCK/SWCLK*	RXD1	
PC1	PC1	TMS/SWDIO*	TXD1	
PC2	PC2	TDO/SWO*	T8IO	
PC3	PC3	TDI*	T9IO	
PC4	PC4	nTRST*	T0IO	
PC5	PC5*	RXD1	T1IO	
PC6	PC6*	TXD1	T2IO	
PC7	PC7*	SCL0	T3IO	
PC8	PC8*	SDA0		
PC9	PC9*	CLKO		
PC10	PC10	nRESET*	T8IO	
PC11	PC11	BOOT*	T9IO	
PC12	PC12*			XIN
PC13	PC13*			XOUT
PC14	PC14*	RXD0	MOSI0(2)	
PC15	PC15*	TXD0	MISO0(2)	

5.3.4 PCD.MR PORT D Pin MUX Register

PD port mode select register. This register must be set properly before use the port. Otherwise the port can't guarantee its functionality.

PCD.MR=0x4000_1300

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
PD15		PD14		PD13		PD12		PD11		PD10		PD9		PD8		PD7		PD6		PD5		PD4		PD3		PD2		PD1		PD0	
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	

Table 18. PCD.MR Selection Bit

Port	Selection Bit			
	00	**01**	**10**	**11**
PD0	PD0*	SS0	T8IO	
PD1	PD1*	SCK0	T9IO	
PD2	PD2*	MOSI0	SCL0	
PD3	PD3*	MISO0	SDA0	

5.3.5 PCn.CRPORT n Pin Control Register (Except for PCC.CR)

Input or output control of each port pin. Each pin can be configured as input pin, output pin or open-drain pin.

PCA.CR=0x4000_1004, PCB.CR=0x4000_1104, PCD.CR=0x4000_1304

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
P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

n	Pn	Port control
		00 Push-pull output
		01 Open-drain output
		10 Input
		11 Analog

5.3.6 PCC.CR PORT C Pin Control Register

Input or output control of each port pin. Each pin can be configured as input pin, output pin or open-drain pin.

PCC.CR=0x4000_1204

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
P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0
11	11	11	11	10	10	11	11	11	11	11	10	10	00	10	10
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

n	Pn	Port control
		00 Push-pull output
		01 Open-drain output
		10 Input
		11 Analog

5.3.7 PCn.PCR PORT n Pull-up Resistor Control Register

Every pin in the port has on-chip pull-up resistors which can be configured by PCn.PCR registers.

PCA.PCR=0x4000_1008, PCB.PCR=0x4000_1108
PCC.PCR=0x4000_1208, PCD.PCR=0x4000_1308

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUE15	PUE14	PUE13	PUE12	PUE11	PUE10	PUE9	PUE8	PUE7	PUE6	PUE5	PUE4	PUE3	PUE2	PUE1	PUE0
0000															
RW															

n	PUEn	Port pull-up control
		0 Disable pull-up resistor
		1 Enable pull-up resister

5.3.8 PCn.DER PORT n Debounce Enable Register

Every pin in the port has a digital debounce filter which can be configured by PCn.DER registers.

PCA.DER=0x4000_100C, PCB.DER=0x4000_110C
PCC.DER=0x4000_120C, PCD.DER=0x4000_130C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDE15	PDE14	PDE13	PDE12	PDE11	PDE10	PDE9	PDE8	PDE7	PDE6	PDE5	PDE4	PDE3	PDE2	PDE1	PDE0
0000															
RW															

n	PDEn	Pin debounce enable
		0 Disable debounce filter
		1 Enable debounce filter

5.3.9 PCn.IER PORT n Interrupt Enable Register

The entire pin can be an external interrupt source. Both of edge trigger interrupt and level trigger interrupt are supported. The interrupt mode can be configured by setting PCn.IER registers

PCA.IER=0x4000_1010, PCB.IER=0x4000_1110
PCC.IER=0x4000_1210, PCD.IER=0x4000_1310

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

n	PIEn	Pin interrupt enable
	00	Interrupt disabled
	01	Enable interrupt as level trigger mode
	10	Reserved
	11	Enable interrupt as edge trigger mode

5.3.10 PCn.ISR PORT n Interrupt Status Register

When an interrupt is delivered to the CPU, the interrupt status can be detected by reading PCn.ISR register. PCn.ISR register will report a source pin of interrupt and a type of interrupt.

PCA.ISR=0x4000_1014, PCB.ISR=0x4000_1114
PCC.ISR=0x4000_1214, PCD.ISR=0x4000_1314

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

n	PISn	Pin interrupt status
	00	No interrupt event
	01	Low level interrupt or Falling edge interrupt event is present
	10	High level interrupt or rising edge interrupt event is present
	11	Both of rising and falling edge interrupt event is present in edge trigger interrupt mode. Not available in level trigger interrupt mode

5.3.11 PCn.ICR PORT n Interrupt Control Register

Interrupt mode control register.

PCA.ICR=0x4000_1018, PCB.ICR=0x4000_1118
PCC.ICR=0x4000_1218, PCD.ICR=0x4000_1318

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	

n	PICn	Pin interrupt mode
		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 mode

5.3.12 PORTEN Port Access Enable

PORTEN enables register writing permition of all PCU registers.

PORTEN=0x4000_1FF0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								PORTEN							
0	0	0	0	0	0	0	0	--							
								RW							

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.

5.4 Functional description

When the input functions of I/O ports are used by Pin Control Register, the output function of I/O port is disabled.

Each port function is different according to the Pin Mux Register.

The Input Data Register captures the data presented on the I/O pin or Debounced input data at every GPIO clock cycle.

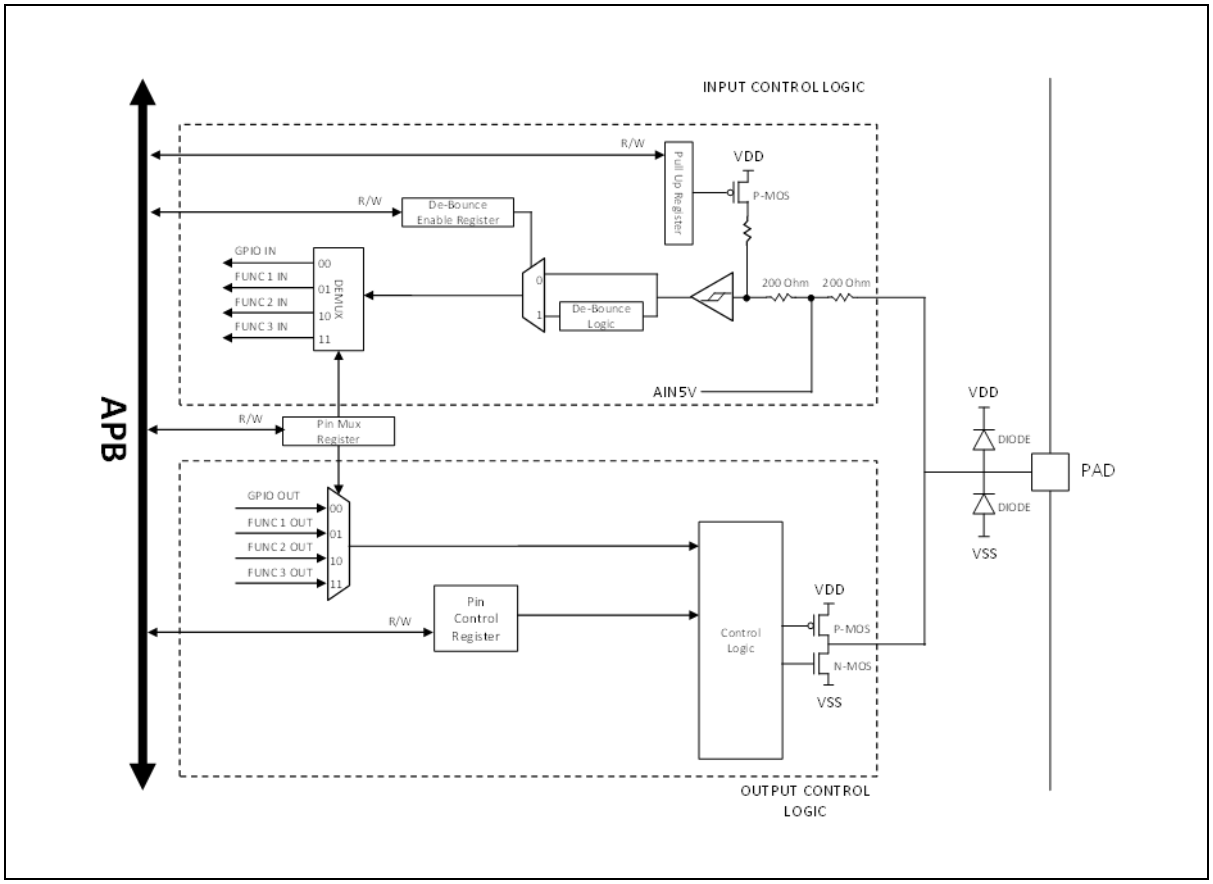


Figure 21. Port Diagram

When the De-Bounce function of Input Data is used by Debounce Enable Register, external input data is captured by Debounce CLK.

- If CNT Value is "01", Debounced Input Data is "1".
- If CNT Value is "10", Debounced Input Data is "0"

The Debounce CLK of each port group can be changed by configuring the MCCR4 and MCCR5 registers.

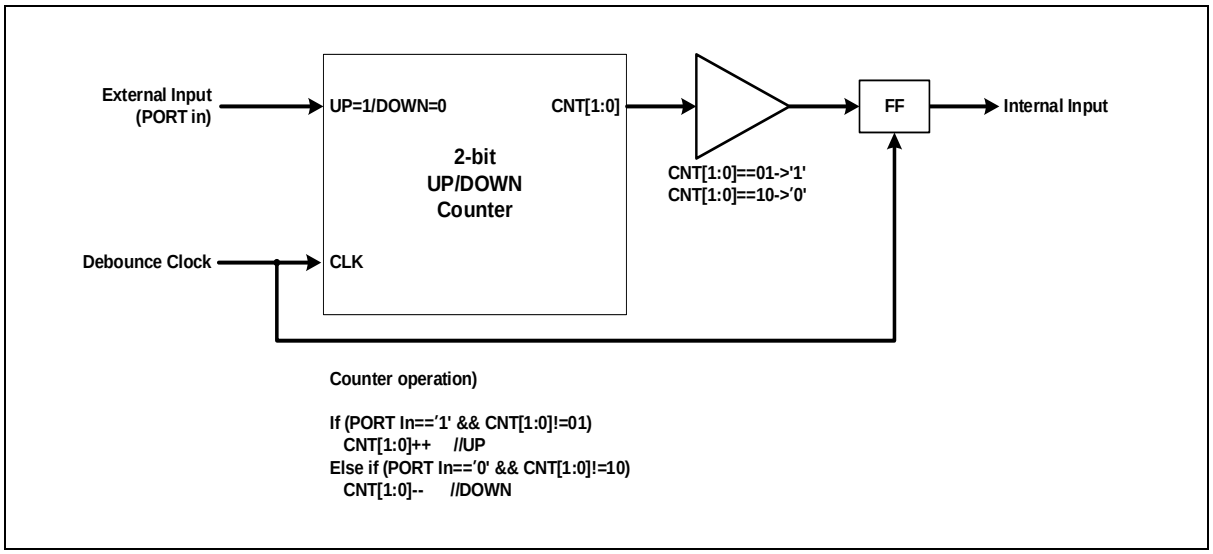


Figure 22. Debouncing Logic

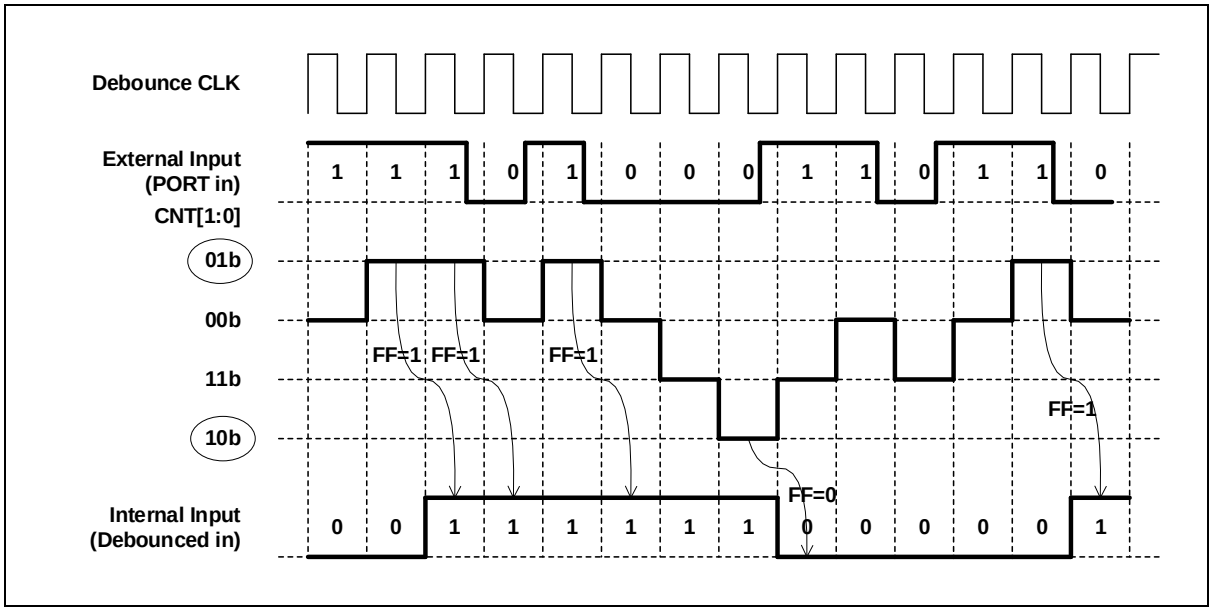


Figure 23. Port Debouncing Example

6 General Purpose I/O (GPIO)

Most of pins except dedicated function pins can be used general I/O ports. General input/output ports are controlled by GPIO block.

GPIO ports are controlled by GPIO block as listed in the followings:

- Output signal level (H/L) select
- External interrupt interface
- Pull up enable or disable

Table 19 are assigned for GPIO blocks.

Table 19. GPIO Pins

Pin name	Type	Description
PA	IO	PA0 to PA15
PB	IO	PB0 to PB7
PC	IO	PC0 to PC15
PD	IO	PD0 to PD3

6.1 GPIO block diagram

Figure 24 describes GPIO in block diagram.

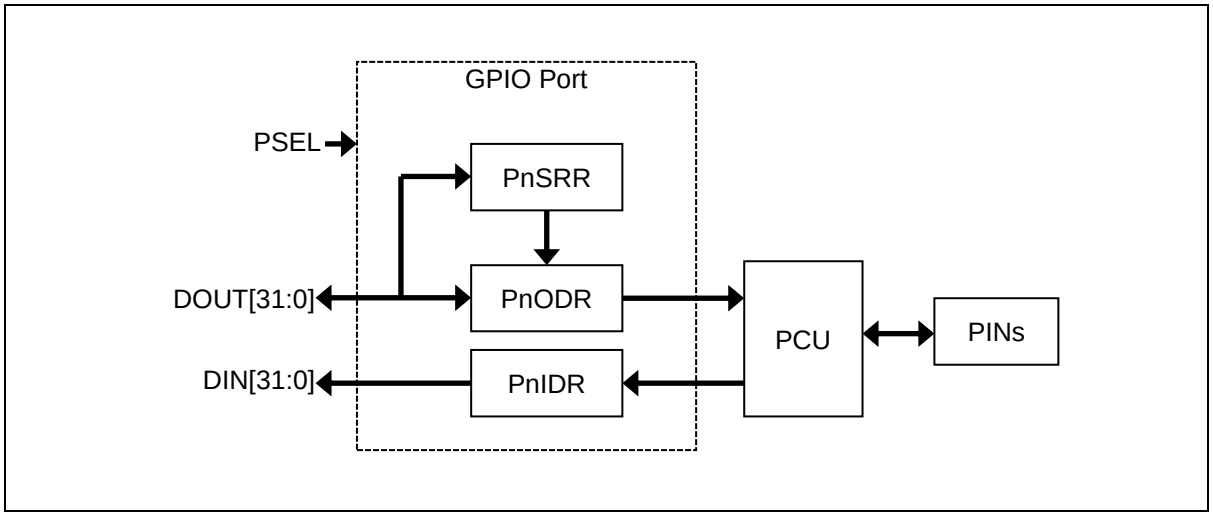


Figure 24. GPIO Block Diagram

6.2 Registers

Base address of PCU is introduced in the followings:

Table 20. Base Address of PCU

Name	Base address	Description
PA PORT	0x4000_2000	General Port A
PB PORT	0x4000_2100	General Port B
PC PORT	0x4000_2200	General Port C
PD PORT	0x4000_2300	General Port D

Table 21. PCU and GPIO Register Map

Name	Offset	Type	Description	Reset value	Reference
Pn.ODR	0x--00	RW	Port *n* Output data register	0x00000000	6.2.1
Pn.IDR	0x--04	RO	Port *n* Input data register	0x00000000	6.2.2
Pn.BSR	0x--08	WO	Port *n* Pin set register	0x00000000	6.2.3
Pn.BCR	0x—0C	WO	Port *n* Pin clear register	0x00000000	6.2.4

6.2.1 Pn.ODR PORT n Output Data Register

When the pin is set as output and GPIO mode, the pin output level is defined by Pn.ODR registers.

PA.ODR=0x4000_2000, PB.ODR=0x4000_2100
PC.ODR=0x4000_2200, PD.ODR=0x4000_2300

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODR															
0000															
RW															

ODR	Pin output level
0	Output low level
1	Output high level

6.2.2 Pn.IDR PORT n Input Data Register

Each pin level status can be read in the Pn.IDR register. Even if the pin is alternative mode except analog mode, the pin level can be detected in the PnIDR register.

PA.IDR=0x4000_2004, PB.IDR=0x4000_2104
PC.IDR=0x4000_2204, PD.IDR=0x4000_2304

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PnIDR															
0000															
RO															

IDR	Pin current level
0	The pin is low level
1	The pin is high level

6.2.3 Pn.BSRPORT n Bit Set Register

Pn.BSR is a register for control each bit of Pn.ODR register. Writing a '1' into the specific bit will set a corresponding bit of Pn.ODR to '1'. Writing '0' in this register has no effect.

PA.BSR=0x4000_2008, PB.BSR=0x4000_2108
PC.BSR=0x4000_2208, PD.BSR=0x4000_2308

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSR															
0000 WO															

BSR	Pin current level
0	Not effect
1	Set correspondent bit in PnODR register

6.2.4 Pn.BCRPORT n Bit Clear Register

Pn.BCR is a register for control each bit of Pn.ODR register. Writing a '1' into the specific bit will set a corresponding bit of Pn.ODR to '0'. Writing '0' in this register has no effect.

PA.BCR=0x4000_200C, PB.BCR=0x4000_210C
PC.BCR=0x4000_220C, PD.BCR=0x4000_230C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PnBCR															
0000 WO															

BCR	Pin current level
0	Not effect
1	Clear correspondent bit in PnODR register

6.3 Functional description

When configured as an output, the value written to the GPIO Ouput Data Register is output on the I/O Pin.

When setting the Bit Set Register, GPIO Output Data Register is set to the high.

When setting the Bit Clr Register, GPIO Output Data Register is set to the Low.

The Input Data Register captures the data presented on the I/O pin or Debounced input data at every GPIO clock cycle.

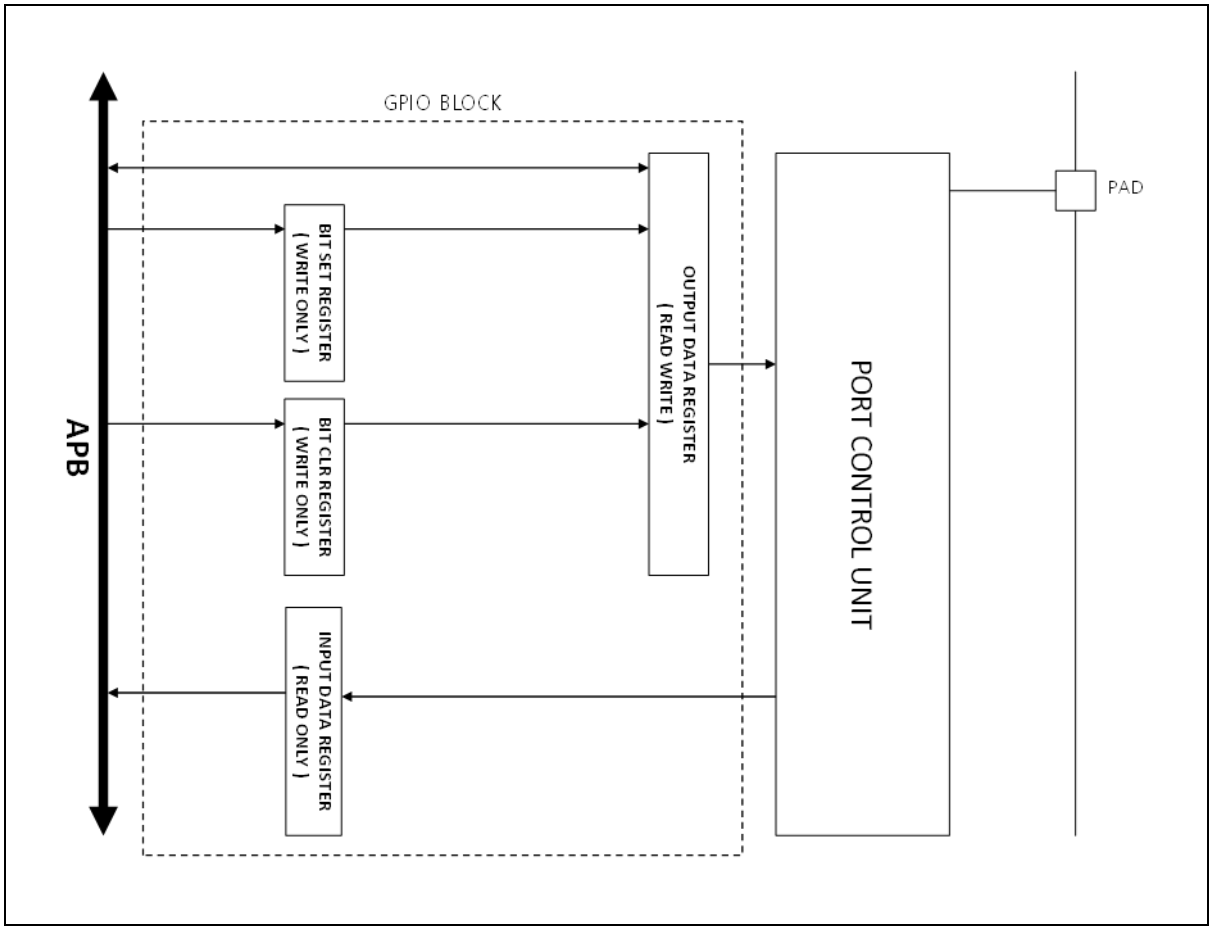


Figure 25. Functional Block Diagram

7 Flash Memory Controller (FMC)

The Flash Memory Controller (FMC) is an interface controller of internal flash memories:

- 64KB Flash code memory
- 32-bit read data bus width
- Code cache block for fast access mode
- 128-byte page size
- Support page erase and macro erase
- 128-byte unit program

Table 22. Internal flash specification

Item	Description
Size	64KB
Start Address	0x0000_0000
End Address	0x0000_FFFF
Page Size	128-byte
Total Page Count	512 pages
PGM Unit	128-byte
Erase Unit	128-byte

7.1 Flash memory map

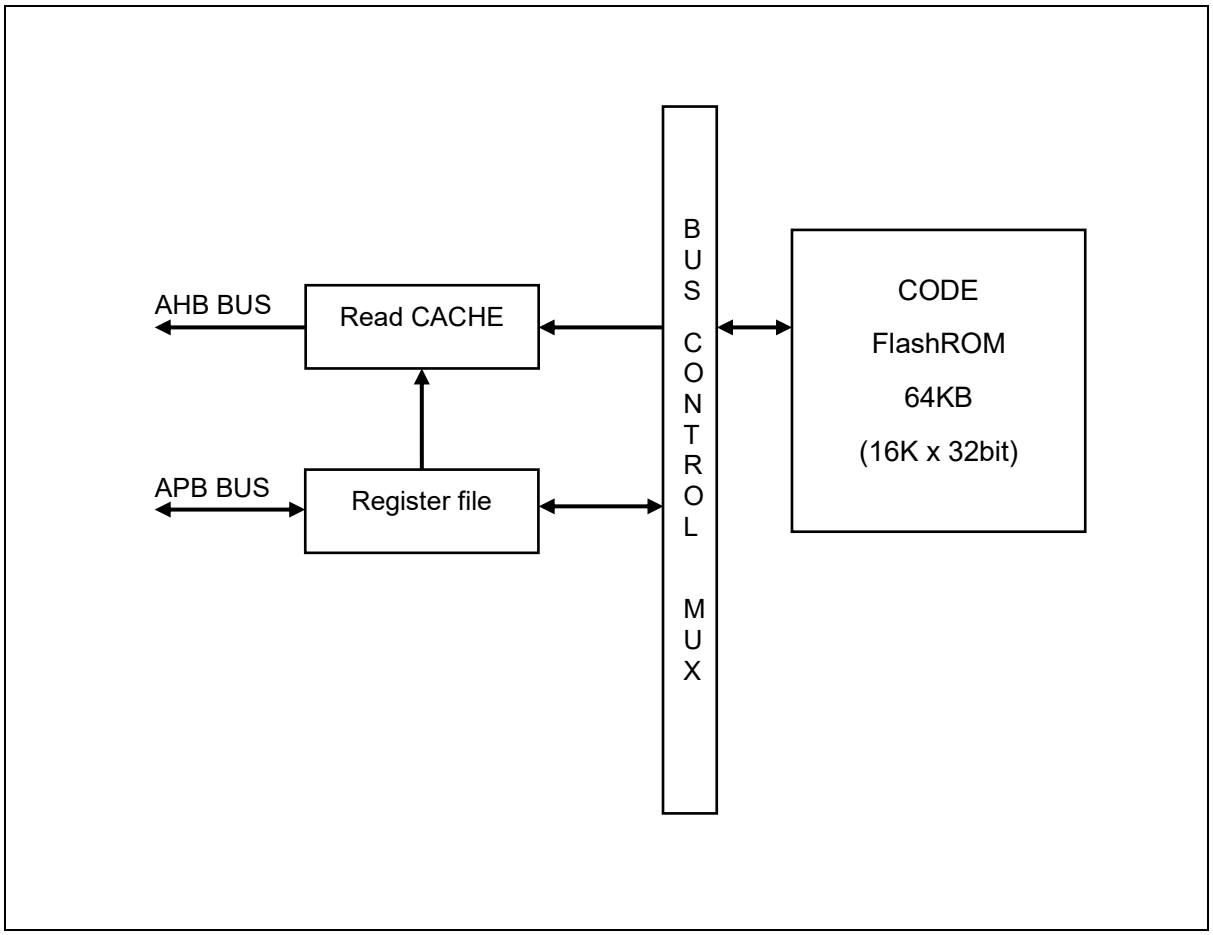


Figure 26. Block Diagram

7.2 Registers

Tables shown below introduce default address of the code flash memory controller (FMC):

Table 23. Base Address of Code Flash Memory Controller

Name	Base address
Flash Controller	0x4000_0100

Table 24. CFMC Register Map

Name	Offset	Type	Description	Reset value	Reference
FM.MR	0x0004	RW	Flash Memory Mode Select register	0x01000000	7.2.1
FM.CR	0x0008	RW	Flash Memory Control register	0x00000000	7.2.2
FM.AR	0x000C	RW	Flash Memory Address register	0x00000000	7.2.3
FM.DR	0x0010	RW	Flash Memory Data register	0x00000000	7.2.4
FM.TMR	0x0014	RW	Flash Memory Timer register	0x000000bb	7.2.5
FM.DRTY	0x0018	RW	Flash Memory Dirty bit		7.2.6
FM.TICK	0x001C	RO	Flash Memory Tick Timer	0x00000000	7.2.7
FM.CRC	0x0020	RO	Flash Memory Read CRC Value		7.2.8
FM.CFG	0x0030	RO	Flash Memory CONFIG value register		7.2.9
FM.BOOTCR	0x0074	RW	Boot ROM Remap Clear register	0x00000000	7.2.10
FM.PROT	0x0078	RW	Flash Page protection register	0x00000000	7.2.11
FM.JTAGEN	0x007C	RW	Jtag protection register	0x00000001	7.2.12

7.2.1 FM.MR Flash Memory Mode Register

FM.MR is internal flash memory mode register. This register is 32-bit register.

FM.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
BOOT							IDLE	VERIFY	AMBAEN					TRMEN	TRM							FEMOD	FMOD	ACODE							
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x00							
R							R	R	R					R	R							R	R	RW						

31	BOOT	0	
		1	Boot mode enable status(read only)
24	IDLE	0	Flash busy status(read only)
		1	Flash idle status(read only)
23	VERIFY	0	
		1	Flash Verify mode enable status(read only)
22	AMBAEN	0	AMBA mode disable status(read only)
		1	AMBA mode enable status(read only)
17	TRMEN	0	
		1	Trim mode entry status(read only)
16	TRM	0	
		1	Trim mode status(read only)
9	FEMOD	0	
		1	Flash mode entry status(read only)
8	FMOD	0	
		1	Flash mode status(read only)
7 0	ACODE	5A → A5	Flash mode
		A5 → 5A	Trim mode
		81 → 28	AMBA mode

7.2.2 FM.CR Flash Memory Control Register

FM.CR is internal flash memory control register

FM.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
											TIMER			TEST1	TEST0	VPPOUT	EVER	PVER		OTP2E	OTPE	PPGM	AE			PMOD	WEN	PBLD	PGM	ERS	PBR
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	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	R	RW	RW		RW	RW		RW			RW	RW	RW	RW	RW	RW

20	TIMER	0	Program/Erase timer enable
		1	(timer can be enable by PGM or ERS bit)
17	TEST[1:0]	00	Normal operation
16		01	(read) Row voltage mode
		01	(write) ODD Row program
		10	Even Row program
		11	All Row program
15	VPPOUT		Enable charge-pump Vpp output
14	EVER		Set erase verify mode
13	PVER		Set program verify mode
11	OTPBE		OTP area B enable
10	OTPAE		OTP area A enable
9	PPGM		Pre PGM enable Page buffer set automatically
8	AE		All erase enable
5	PMODE		PMODE enable(Address path changing)
4	WE		Write enable
3	PBLD		Page buffer load(WE should be set)
2	PGM		Program enable
1	ERS	0	Program mode enable
		1	Erase mode enable
0	PBR		Page buffer reset

7.2.3 FM.AR Flash Memory Address Register

FM.AR is internal flash memory program, erase address register

FM.AR=0x4000_010C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		FADDR													
0		0x0000													
		RW													

14 0	FADDR	16K words address (one word = 4 bytes)

7.2.4 FM.DR Flash Memory Data Register

FM.DR is internal flash memory program data register

FM.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																															
0x0000_0000																															
RW																															

31 0	FDATA	Flash PGM data (32-bit)

7.2.5 FM.TMR Flash Memory Timer Register

FM.TMR is internal flash memory Timer value register (16-bit), Erase/Program timer runs up to TMR[15:0]

FM.TMR=0x4000_0114

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR															
0x09C4															
RW															

7 0	TMR	Erase/PGM timer (default, 0x09C4) Timer counts up to TMR[15:0] by 1MHz int. OSC clock or External OSC clock. It can be selected in TMRCK bit.

7.2.6 FM.DRTY Flash Memory Dirty Bit Register

FM.DRTY is internal flash memory dirty bit clear register

FM.DRTY=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
FDRTY																															
-																															
Write Only																															

31 0	FDRTY	Write any value here, cache line fill flag will be cleared.

7.2.7 FM.TICK Flash Memory Tick Timer Register

FM_TICK is internal flash memory Burst Mode channel selection register

FM.TICK=0x4000_011C

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
																FTICK															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x00000															
								RW																							

17 0	FTICK	TICK goes to 0x3FFFF from written TICK value while TRM runs by PCLK clock

7.2.8 FM.CRC Flash Memory CRC Value Register

FM.CRC is the CRC value resulted from read accesses on internal flash memory.

FM.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
																CRC															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0xFFFF															
								RO																							

15 0	CRC	CRC16 value

7.2.9 FM.CFG Flash Memory CONFIG Value Register

FM.CFG is the FLASH configuration register.

FM.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
WRITE KEY																HRESPD			TMRCK		WAIT			CRCINIT	CRCEN			TRIM			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1			0		0	1	1	0	0			0			
																												RW			

31 15	WRITE KEY		KEY Value : 0x7858
15	HRESPD		Disable HRESP(error response function) of Data or System bus (HRESP is AMBA AHB signal)
12	TMRCK	0	PGM/ERASE timer source is 1MHz RING OSC PRM/ERASE timer source is External Clock
10 8	WAIT	000	No wait access for flash memory
		001	1-wait inserted for flash access
		010	2-wait inserted for flash access
		011	3-wait inserted for flash access
7	CRCINIT	0 1	CRC register will be initialized. It should be reset again before read flash to generate CRC16 calculation (Initial value of FMCRC is 0xFFFF)
6	CRCEN	0 1	CRC16 enable CRC value will be calculated at every flash read timing
3 0	TRIM		FLASH TRIM Value (trim_mode_entry)

NOTE: Before changing the system clock, flash access wait should be set to the maximum value. After the system clock is changed, you will need to set flash access wait that you want if necessary.

7.2.10 FM.BOOTCR Boot ROM Remap Clear Register

BOOTCR is Boot ROM remap clear register. This register is 8-bit register.

FM.BOOTCR=0x4000_0174

7	6	5	4	3	2	1	0
							BOOTROM
0	0	0	0	0	0	0	1
							R

0	BOOTROM	Boot Mode (only can be written in boot loader mode) This bit is used to clear boot loader mode at end of boot code (when BOOTROM low, external BOOT pin signal is masked)

7.2.11 FM.PROTECT Write Protection Control Register

FM.PROTECT is internal flash memory control register

FM.PROTECT=0x4000_0178

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
WRITE_KEY		WP15 WP14 WP13 WP12 WP11 WP10 WP9 WP8	WP7 WP6 WP5 WP4 WP3 WP2 WP1 WP0
0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 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

15	W1P5	0xF000 ~ 0xFFFF, write_key is 0x87
14	WP14	0xE000 ~ 0xEFFF, write_key is 0x87
13	WP13	0xD000 ~ 0xDFFF, write_key is 0x87
12	WP12	0xC000 ~ 0xCFFF, write_key is 0x87
11	WP11	0xB000 ~ 0xBFFF, write_key is 0x87
10	WP10	0xA000 ~ 0xAFFF, write_key is 0x87
9	WP9	0x9000 ~ 0x9FFF, write_key is 0x87
8	WP8	0x8000 ~ 0x8FFF, write_key is 0x87
7	WP7	0x7000 ~ 0x7FFF, write_key is 0x87
6	WP6	0x6000 ~ 0x6FFF, write_key is 0x87
5	WP5	0x5000 ~ 0x5FFF, write_key is 0x87
4	WP4	0x4000 ~ 0x4FFF, write_key is 0x87
3	WP3	0x3000 ~ 0x3FFF, write_key is 0x87
2	WP2	0x2000 ~ 0x2FFF, write_key is 0x87
1	WP1	0x1000 ~ 0x1FFF, write_key is 0x98
0	WP0	0x0000 ~ 0x0FFF, write_key is 0x98

7.2.12 FM.JTAGEN JTAG Protection Control Register

JTAGEN is Debug access control register

FM.JTAGEN=0x4000_017C

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
WRITE_KEY																															JTAGEN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
																															RW

0	JTAGEN	0 1	Debug access port is disabled, write access code is 0xC7 Debug access port is enabled

7.3 Functional description

Flash Memory Controller is an internal flash memory interface controller. It mainly controls the program flash memory operation and preparing read data for requesting from the bus.

7.3.1 Flash organization

The 64Kbytes code flash memory consists of 1,024 pages which has uniform 128 bytes page size. The flash controller allows to read or write a data of the flash memory. The read access can be done by 8, 16 and 32 bits wide

This memory is located at 0x0000_0000 address on system memory map. The system boot address is 0x0000_0000, so this flash memory is boot memory. The code data which is programmed in the flash memory will boot up the device after boot rom sequence is finished.

Flash Memory Controller is an internal flash memory interface controller. It mainly controls the program flash memory operation and preparing read data for requesting from the bus.

7.3.2 Flash read operation

The flash data read operation is requested from the bus. The flash controller will response to the request by itself. The wait time should be defined properly. Because normally the bus speed is faster than flash data access time.

The normal read operation is not available in FLASH MODE in ACODE.FM.MR field.

7.3.3 Flash program operation

The erase and program access of flash memory is available only in FLASH MODE in ACODE.FM.MR field. So, self-program is not supported. The flash program/erase operation should be performed by the execution program on the SRAM memory.

The flash program operation will write one page to the target address selected by FM.AR register. At first, user should write the program data into the page buffer. Page buffer write is performed by word write access to FM.DR register on FM.AR address. .

After fill the page buffer, user can start to flash write operation and should wait for the IDLE bit set.

Figure 27 shows page buffer loading operation.

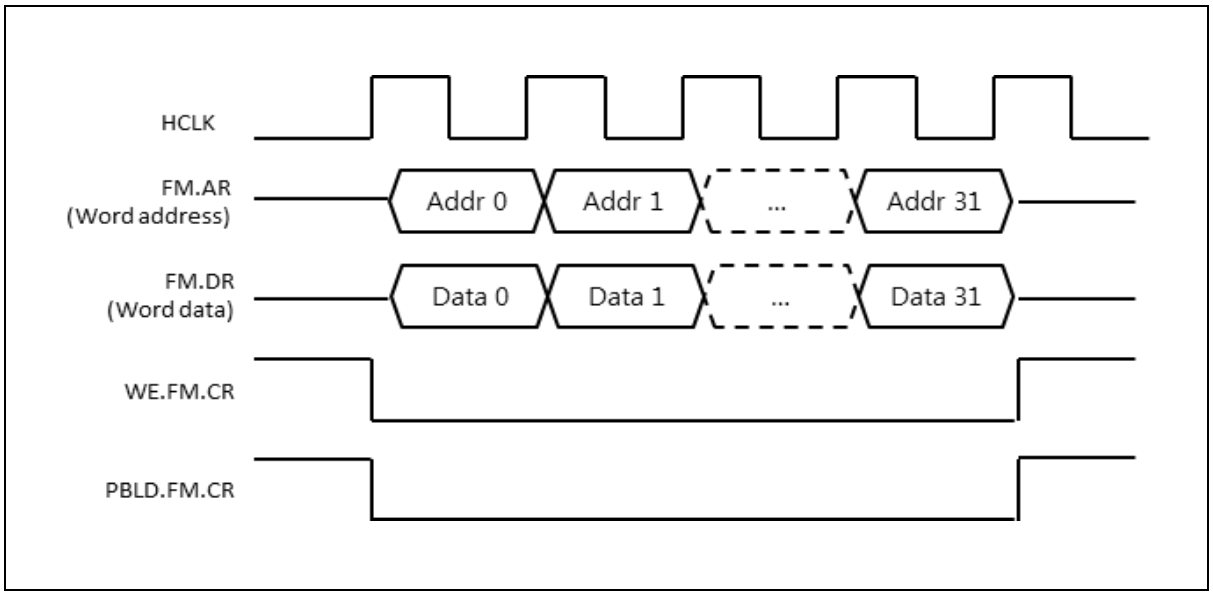


Figure 27. Page Buffer Load Timing Diagram

The flash write of page buffer data is done by PRGM.FM.CR command. The safe writing operation requires proper program time. The program time tPGM is defined by FM.TMR register. This timer will count the number of HCLK clock to the FM.TMR value. The timer count is start, the IDLE.FM.MR register is cleared. The timer count is done then the IDLE.FM.MR register is set.

In this page write operation, the target page address should be written in FM.AR register.

Figure 28 shows page write operation.

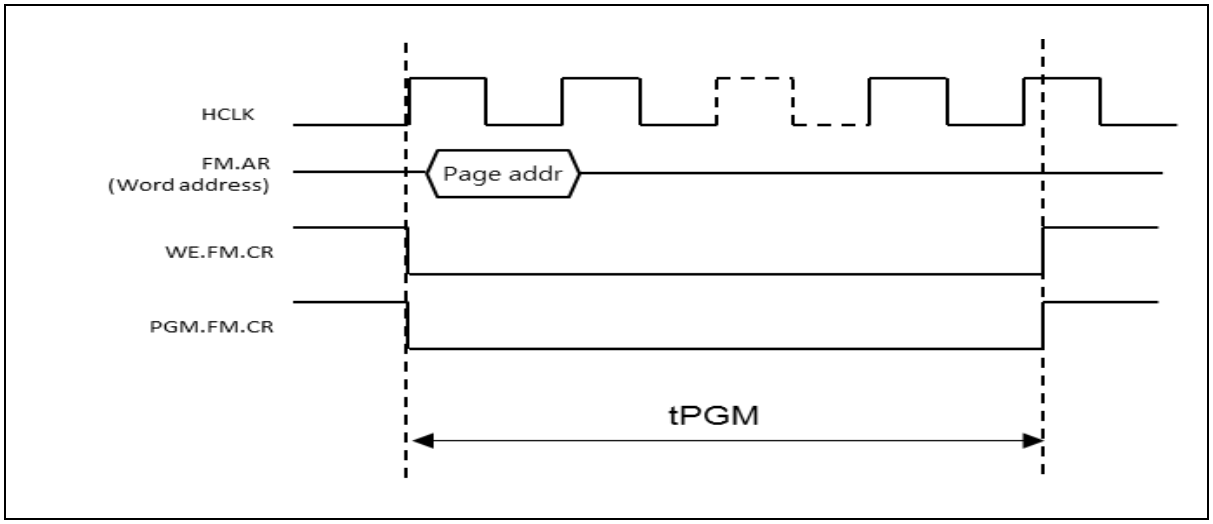


Figure 28. Page Write Timing Diagram

7.3.4 Flash erase operation

The erase and program access of flash memory is available only in FLASH MODE in ACODE.FM.MR field. So, self-program is not supported. The flash program/erase operation should be performed by the execution program on the SRAM memory.

2 kind of flash erase operations are supported. One is page erase and the other one is bulk erase. The page erase operation will erase one page to the target address selected by FM.AR register. .

User starts to flash write operation and should wait for the IDLE bit set.

The bulk erase operation will erase whole flash memory data and the FM.AR address will be ignored. The control is same as the page erase operation except AE.FM.CR bit keep set.

Figure 29 shows page erase operation.

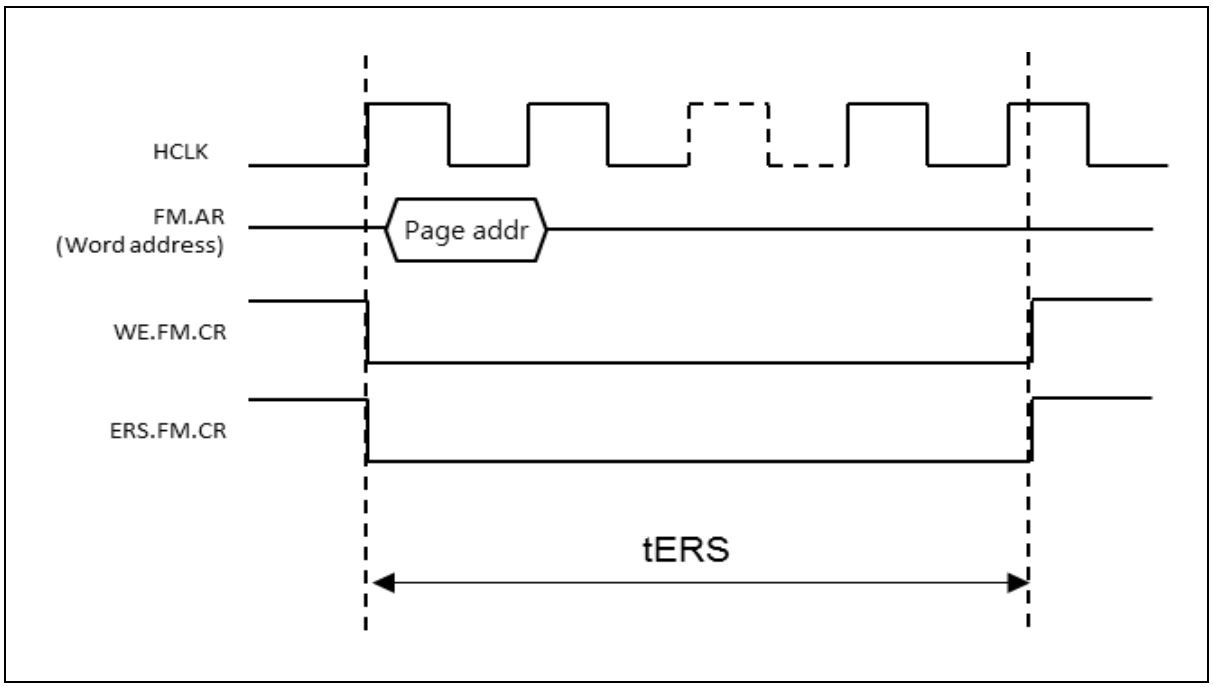


Figure 29. Page Erase Timing Diagram

The flash erase of page data is done by ERS.FM.CR command. The safe writing operation requires proper program time. The erase time tERS is defined by FM.TMR register. This timer will count the number of HCLK clock to the FM.TMR value. The timer count is start, the IDLE.FM.MR register is cleared. The timer count is done then the IDLE.FM.MR register is set.

Figure 30 shows bulk erase operation.

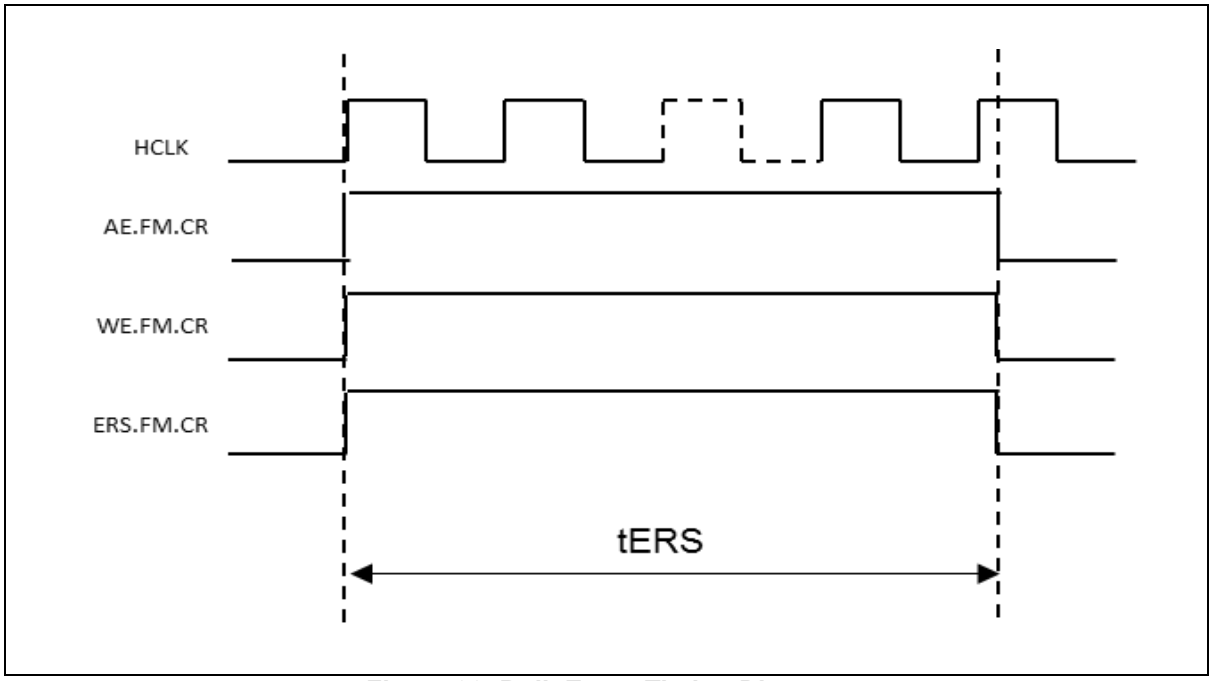


Figure 30. Bulk Erase Timing Diagram

8 Direct Memory Access Controller (DMAC)

The DMAC is direct memory access controller who can establish the data transfer between memory and peripherals without CPU operation.

DMAC of AC33Mx064T series features followings:

- 4 Channels
- Single transfer only
- Support 8/16/32-bit data size
- Support multiple buffer with same size
- Interrupt condition is transferred through peripheral interrupt

8.1 DMAC block diagram

The DMAC block diagram is introduced in Figure 31.

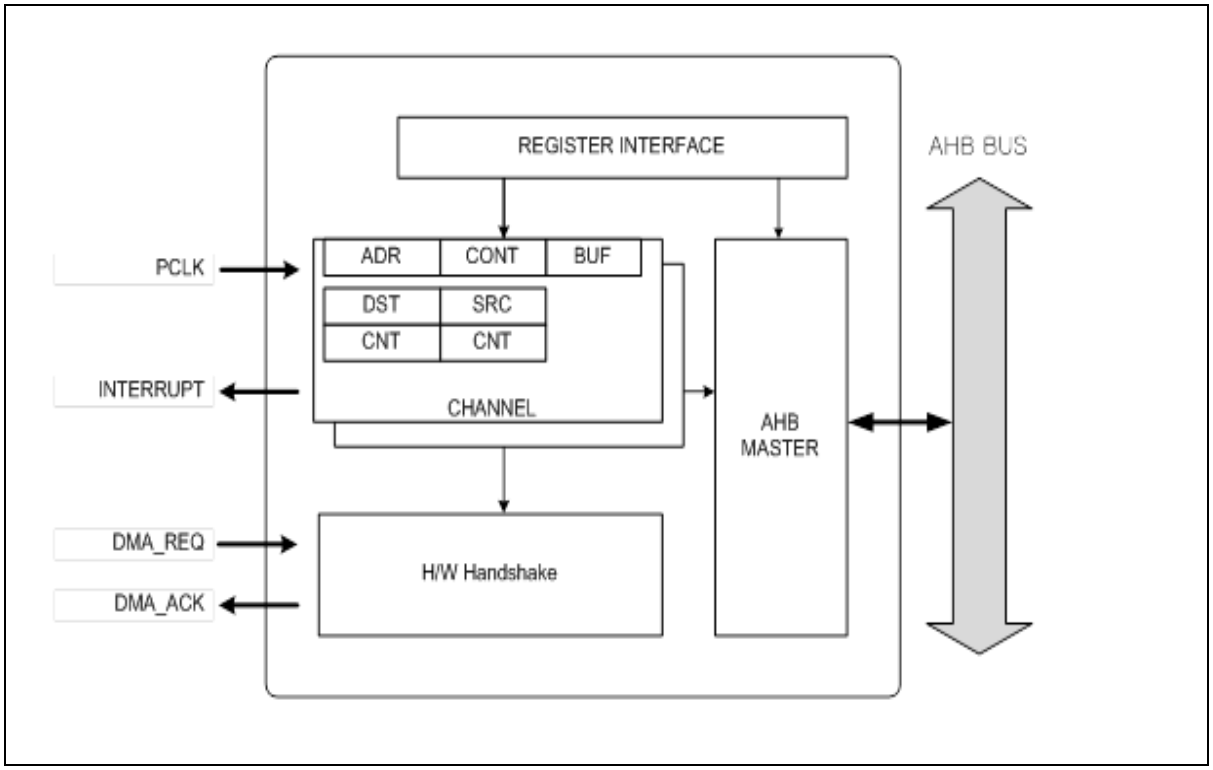


Figure 31. DMAC Block Diagram

8.2 Registers

Base address of DMAC is introduced in the followings:

Table 25. Base Address of DMAC

Name	Base address
DMACH0	0x4000_0400
DMACH1	0x4000_0410
DMACH2	0x4000_0420
DMACH3	0x4000_0430

Table 26. WDT Register Map

Name	Offset	Type	Description	Reset value	Reference
DC*n*.CR	0x0000	RW	DMA Channel n Control Register	0x0000_0000	8.2.1
DC*n*.SR	0x0004	RW	DMA Channel n Status Register	0x0000_0000	8.2.2
DC*n*.PAR	0x0008	R	DMA Channel n Peripheral Address	0x0000_0000	8.2.3
DC*n*.MAR	0x000C	RW	DMA Channel n Memory Address	0x2000_0000	8.2.4

8.2.1 DC*n*.CR DMA Controller Configuration Register

DMA operation control register is 32-bit register.

DC0.CR=0x4000_0400 , DC1.CR=0x4000_0410
DC2.CR=0x4000_0420 , DC3.CR=0x4000_0430

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
				TRANSCNT																PERISEL								SIZE		DIR	
0	0	0	0	0x000												0	0	0	0	0				0	0	0	0	00		0	0
RW																RW								RW							

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
11 8	PERISEL	Peripheral selection
		N Associated peripheral selection. Refer to DMA Peripheral connection table
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)

A DMA channel will be connected with selected peripheral. Table 27 shows peripheral selection numbers.
This PERISEL field should be set with proper number of peripheral which will be connected with DMA interface.

Table 27. DMAC PERISEL Selection

PERISEL[3:0]	Associate Peripheral
0	CHANNEL IDLE
1	UART0 RX
2	UART0 TX
3	UART1 RX
4	UART1 TX
5	SPI0 RX
6	SPI0 TX
7	ADC0
8	ADC1
9 - 15	N.A.

PERISEL cannot have same value in different channels. If same PERISEL value written in more than one channel, the proper operation is not guaranteed.
Not used channel should have CHANNEL IDLE value in PERISEL bit position.

8.2.2 DCn.SR DMA Controller Status Register

DMA Controller Status Register is 8-bit register.

This register represents the current status of DMA Controller and enables DMA function.

DC0.SR=0x4000_0404 , DC1.SR=0x4000_0414
DC2.SR=0x4000_0424 , DC3.SR=0x4000_0434

7	6	5	4	3	2	1	0
EOT							DMAEN
1	0	0	0	0	0	0	0
RO							RW

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
0	DMAEN	DMA Enable
		0 DMA is in stop or hold state
		1 DMA is running or enabled

8.2.3 DC*n*.PAR DMA Controller Peripheral Address Register

These registers represent the peripheral address.

DC0.PAR=0x4000_0408 , DC1.PAR=0x4000_0418
DC2.PAR=0x4000_0428 , DC3.PAR=0x4000_0438

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	

31 0	PAR	Target Peripheral address of transmit buffer or receive buffer. User must set exact target peripheral buffer address in this field. If DIR is "0" this address is destination address of data transfer. If DIR is "1", this address is source address of data transfer.

8.2.4 DC*n*.MAR DMA Controller Memory Address Register

These registers represent the memory address.

DC0.MAR=0x4000_040C , DC1.MAR=0x4000_041C
DC2.MAR=0x4000_042C , DC3.MAR=0x4000_043C

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	

31 0	MAR	Target memory address of data transfer. Address is automatically incremented according to SIZE bits when each transfer is done. If DIR is "0" this address is source address of data transfer. If DIR is "1", this address is destination address of data transfer.

8.3 Functional description

A DMAC controller of AC33Mx064T performs direct memory transfer by sharing the system bus with CPU core. The system bus is shared by 2 AHB 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 peripheral request. When a peripheral request the transfer to the DMA controller, related channel is activate and access the bus to transfer requested data from memory to peripheral data buffer or from peripheral data buffer to memory space.

- User set both of peripheral address and memory address
- User configure DMA operation mode and transfer count.
- User enable DMA channel
- DMA request is occurred from peripheral.
- DMA activate channel which was requested
- DMA read data from source address and save it internal buffer.
- DMA write the buffered data to destination address.
- Transfer count number is decreased by 1.
- When Transfer count is 0, EOT flag is set and notice to peripheral to issue the interrupt
- DMA does not have interrupt source, the interrupt related DMA status can be shown from assigned peripheral interrupt.

Figure 32 shows a functional block diagram of DMA controller.

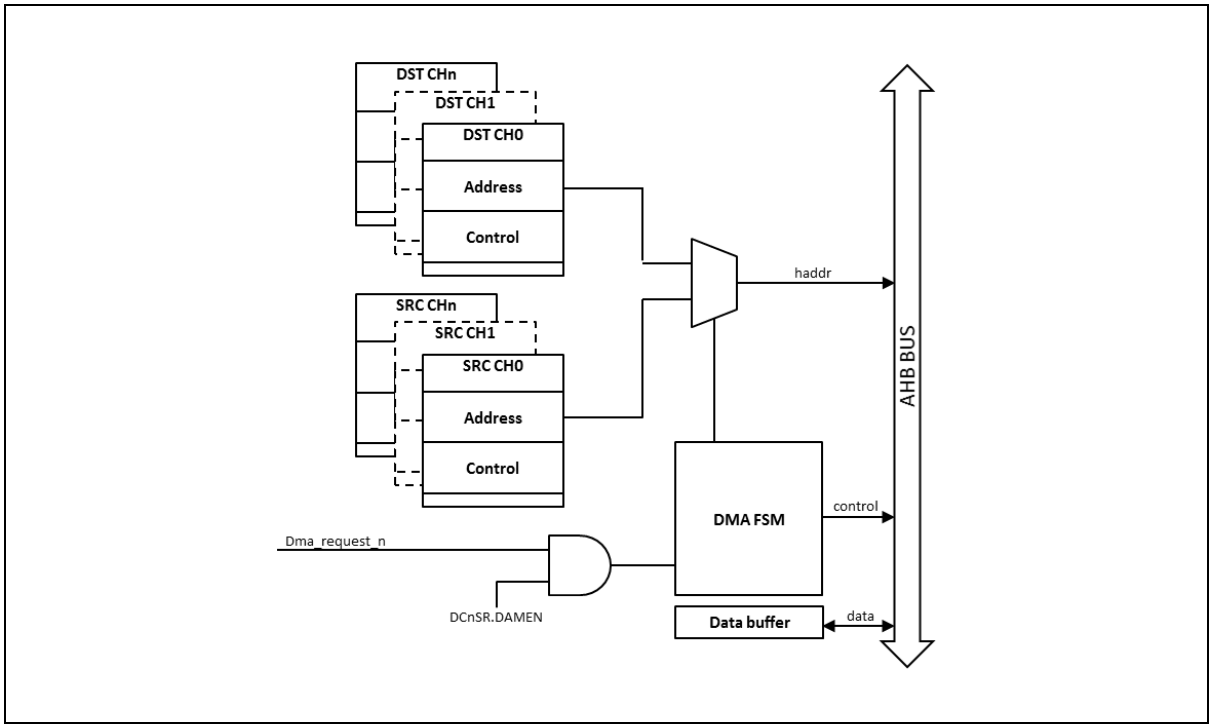


Figure 32. Block Diagram

Figure 33 shows a functional timing diagram of DMA controller. The transfer request from 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.

The timing diagram for a DMA transfer from peripheral to memory is shown in below figure. 4-clock cycle latency exists during accessing the peripheral. If the bus is occupied by different bus master, there are amount of bus waiting cycles.

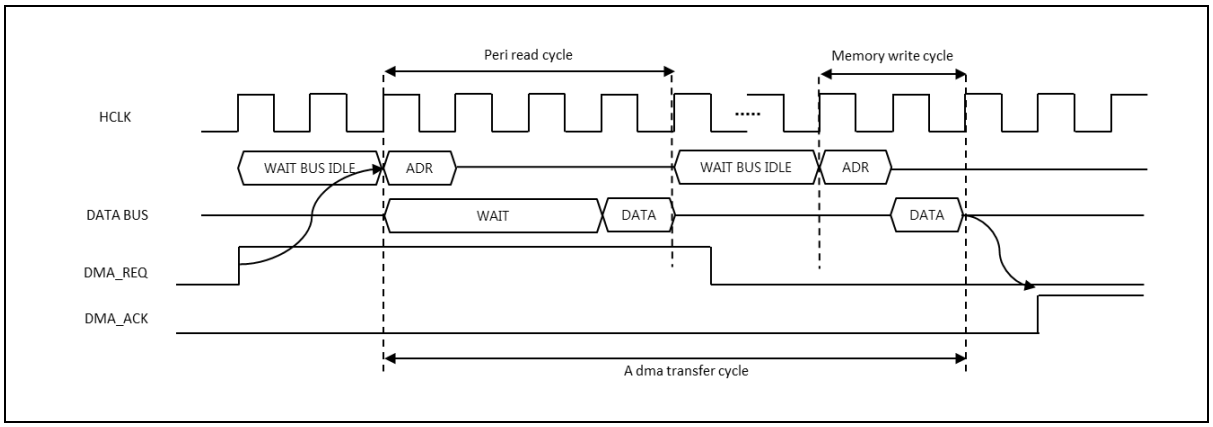


Figure 33. DMA Transfer from Peripheral to Memory

In Figure 34, the timing diagram for a DMA transfer from memory to peripheral is shown. 4-clock cycle latency exists during accessing the peripheral. If the bus is occupied by different bus master, there are amount of bus waiting cycles.

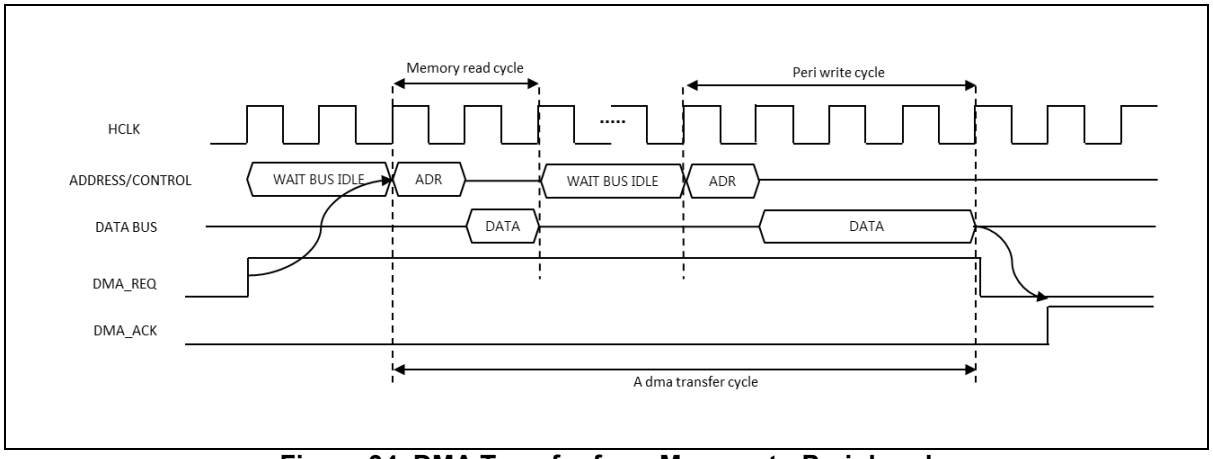


Figure 34. DMA Transfer from Memory to Peripheral

Figure 35 shows an example of N data transfers with the DMA. The DMA transfer is started when DCnSR.DMAEN is set and will be cleared when all the number of transfer is completed.

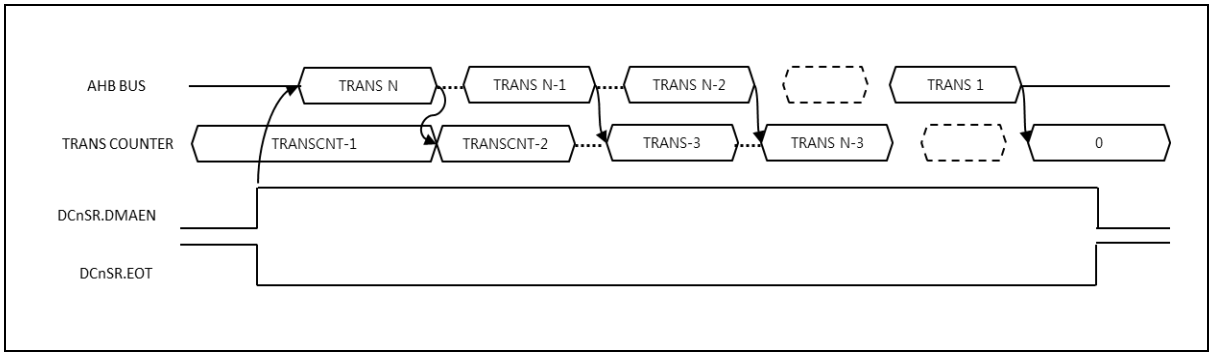


Figure 35. N DMA Transfer Example

9 Watchdog Timer (WDT)

Watchdog timer (WDT) monitors the operation of the MCU and is typically used to detect software errors. When the MCU becomes uncontrollable due to a malfunction, the WDT resets the MCU to recover it.

The AC33Mx064T series has one WDT module built in, which functions as a 32-bit down-counter. Once the WDT counts down to zero while set as a reset source, the MCU gets reset. When it is not used to monitor the MCU, it can be used as a cycle timer along with an interrupt.

WDT of AC33Mx064T series features followings:

- 32-bit down counter
- Select reset or periodic interrupt
- Count clock selection
- Dedicated pre-scaler
- Watchdog underflow output signal

9.1 WDT block diagram

In this section, WDT block diagram is introduced in Figure 36.

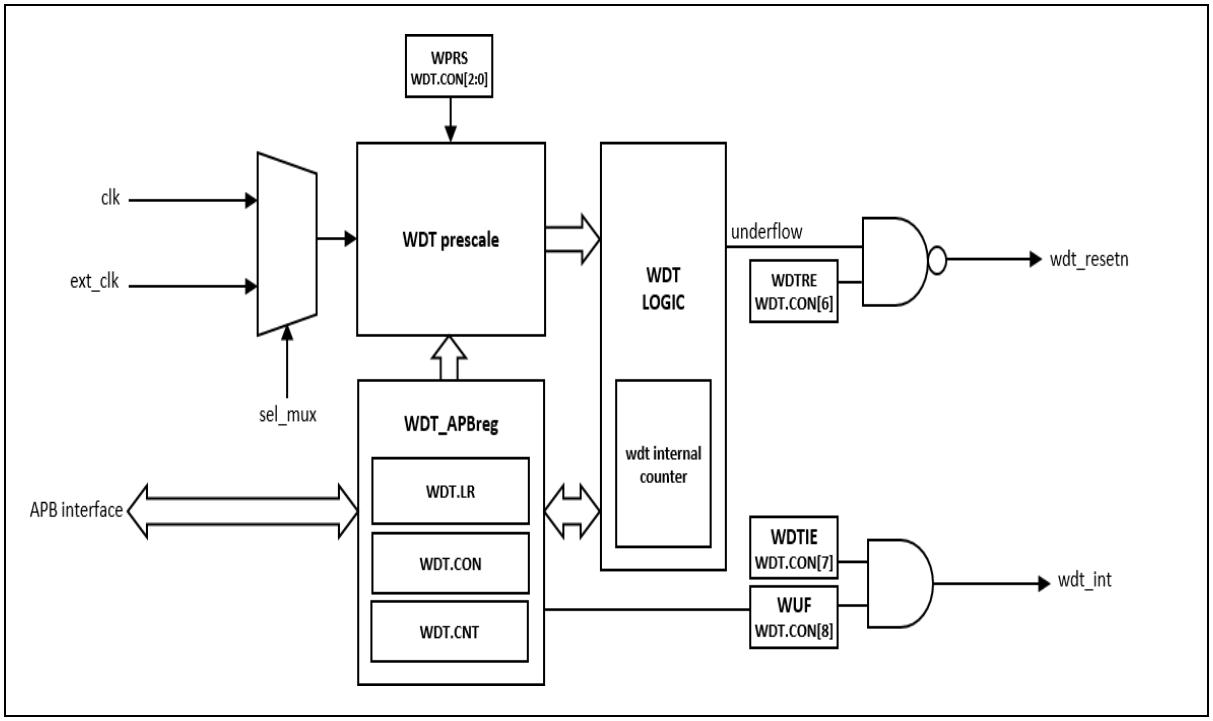


Figure 36. WDT Block Diagram

9.2 Registers

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

Table 28. Base Address of WDT

Name	Base address
WDT	0x4000_0200

Table 29. WDT Register Map

Name	Offset	Type	Description	Reset value	Reference
WDT.LR	0x0000	W	WDT Load register	0x00000000	9.2.1
WDT.CNT	0x0004	R	WDT Current counter register	0x0000FFFF	9.2.2
WDT.CON	0x0008	RW	WDT Control register	0x0000805C	9.2.3

9.2.1 WDT.LR Watchdog Timer Load Register

The WDTLR register is used to update WDTCNT register. To update WDTCNT register, the WDTEN bit of WDTCON should be set to '1' and write into WDTLR register with target value of WDTCNT. It needs at least 5 WDT counter clocks to update WDTLR to WDTCNT.

WDT.LR=0x4000_0200

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
WDTLR																															
0x0000_0000																															
W																															

31 0	WDTLR	Watchdog timer load value register Keeping WDTEN bit as '1', write WDTLR register will update WDTCNT value with written value

9.2.2 WDT.CNT Watchdog Timer Current Counter Register

The WDTCNT 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.

WDT.CNT=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
WDTCNT																															
0x0000_FFFF																															
R																															

31 0	WDTCNT	Watchdog timer current counter register 32-bit down counter will run from the written value.

9.2.3 WDT.CON Watchdog Timer Control Register

WDT module should be configured properly before running. WDT module can make reset event or assert interrupt signal to system. If user don't use both reset and interrupt functions, WDT can be used like a loadable down-counter. WUF flag will be set when WDT counts down to 0.

WDT.CON=0x4000_0208

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDBG							WUF	WDTIE	WDTRE		WDTEN	CKSEL	WPRS		
1	0	0	0	0	0	0	0	0	1	0	1	1	100		
RW							RW	RW	RW		RW	RW	RW		

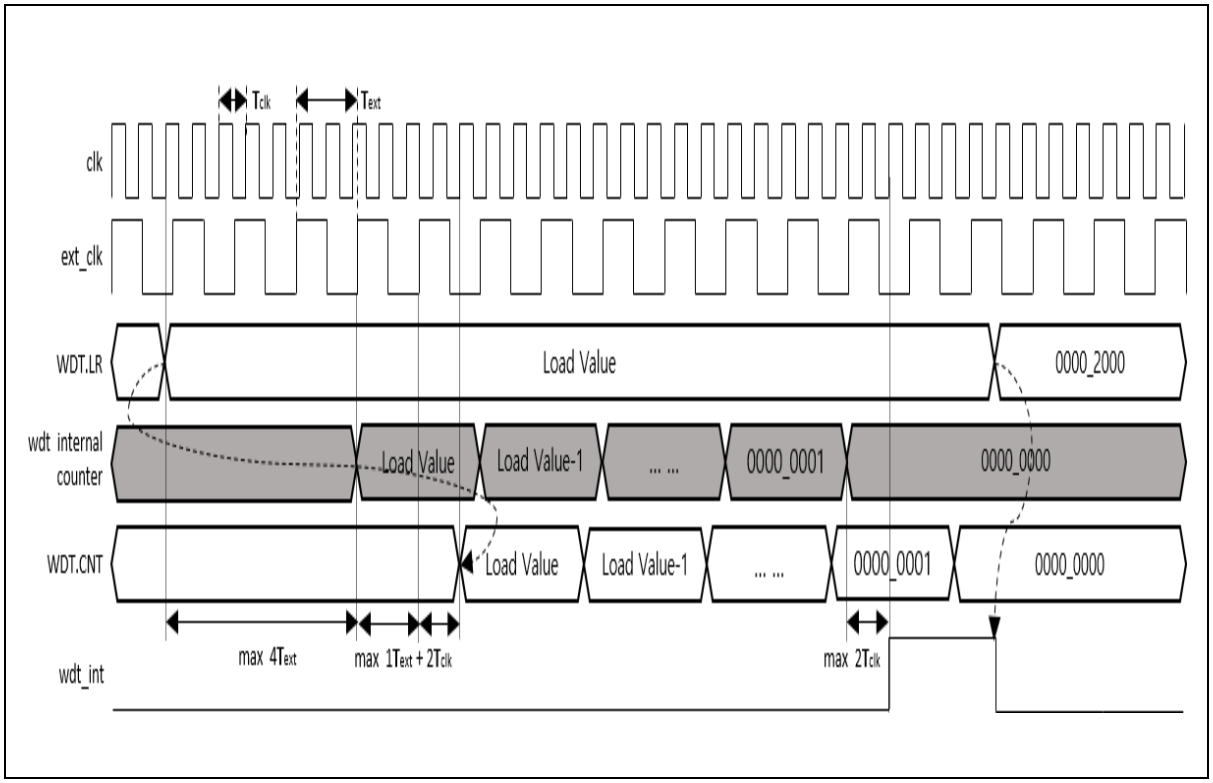
15	WDBG	Watchdog operation control in debug mode
	0	Watchdog counter running when debug mode
	1	Watchdog counter stopped when debug mode
8	WUF	Watchdog timer underflow flag (This bit is cleared when WDTLR is written)
	0	No underflow
	1	Underflow is pending
7	WDTIE	Watchdog timer counter underflow interrupt enable
	0	Disable interrupt
	1	Enable interrupt
6	WDTRE	Watchdog timer counter underflow interrupt enable
	0	Disable reset
	1	Enable reset
4	WDTEN	Watchdog Counter enable
	0	Watch dog counter disabled
	1	Watch dog counter enabled
3	CKSEL	WDTCLKIN clock source select
	0	PCLK
	1	External clock
2 0	WPRS[2:0]	Counter clock prescaler WDTCLK = WDTCLKIN/WPRS
	000	WDTCLKIN
	001	WDTCLKIN / 4
	010	WDTCLKIN / 8
	011	WDTCLKIN / 16
	100	WDTCLKIN / 32
	101	WDTCLKIN / 64
	110	WDTCLKIN / 128
	111	WDTCLKIN / 256

9.3 Functional description

The watchdog timer Count can be enabled by WDTEN (WDT.CON[4]) to 1. As watchdog timer is enabled, the down counter will start counting from the Load Value.

If WDTRE (WDT.CON[6]) is set as 1, WDT reset would be asserted when the WDT counter value reached to 0(underflow event) from WDTLR value. Before WDT counter down to 0, software can write a certain value to register WDTLR to reload WDT counter.

9.3.1 Timing diagram



In WDT interrupt mode, once WDT underflow occurred then a certain count value would be reloaded to prevent next WDT interrupt in short time period and this reloading action only be activated when the watchdog timer counter set to be Interrupt mode (set WDTIE of WDT.CON).

It takes up to 5 cycles from Load value to the CNT value. The WDT interrupt signal and CNT value data might be delayed maximum by 2 system bus clocks in synchronous logic.

9.3.2 Prescaler table

The WDT includes a 32-bit down counter with programmable pre-scaler to define different time-out intervals.

The clock sources of watchdog timer can be peripheral clock (PCLK) or one of 3 external clock sources. External clock source can be enabled by CKSEL (WDT.CON[3]) set to '1' and External clock source was chosen in MCCR3 register of SCU (System Control Unit) block.

To make WDT counter base clock, user can control 3-bit pre-scaler WPRS [2:0] in WDT.CON register and the maximum pre-scaled value is "clock source frequency/256". The pre-scaled WDT counter clock frequency values are listed in following table.

Selectable clock source (40 kHz ~ 16 MHz) and the time out interval when 1 count

Time out period =
{(Load Value) * (1/pre-scaled WDT counter clock frequency) + max $5T_{ext}$} + max $4T_{clk}$

*Time out period (time out period from load Value to interrupt set '1')

Table 30. Prescaled WDT Counter Clock Frequencies

Clock source	WDTCLKIN	WDTCLKIN /4	WDTCLKIN /8	WDTCLKIN /16	WDTCLKIN /32	WDTCLKIN /64	WDTCLKIN /128	WDTCLKIN /256
RING OSC	1Mhz	250kHz	125kHz	62.5kHz	31.25kHz	15.625kHz	7.8125kHz	3.90625kHz
MCLK	MCLK (BUS CLK)	MCLK/4	MCLK/8	MCLK/16	MCLK/32	MCLK/64	MCLK/128	MCLK/256
EOSC	XTAL	XTAL/4	XTAL/8	XTAL/16	XTAL/32	XTAL/64	XTAL/128	XTAL/256
Clock source	WDTCLKIN	WDTCLKIN /4	WDTCLKIN /8	WDTCLKIN /16	WDTCLKIN /32	WDTCLKIN /64	WDTCLKIN /128	WDTCLKIN /256

10 16-bit timer

A timer block of AC33Mx064T consists of 4 channels of 16 bit General purpose timers. They have independent 16 bit counter and dedicated prescaler feeds counting clock. They can support periodic timer, PWM pulse, one-shot timer and capture mode. They can be synchronized together.

One more optional free-run timer is provided. The main purpose of this timer is a periodical tick timer or a wake-up source.

16-bit timer of A33M1x series features the followings:

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

Table 31 introduces pins assigned for 16-bit timer.

Table 31. Pin Assignment of 16-bit Timer: External Pins

Pin name	Type	Description
TnIO	I/O	External clock/ capture input and PWM/ one-shot output

10.1 16-bit timer block diagram

In this section, 16-bit timer is described in a block diagram in Figure 37.

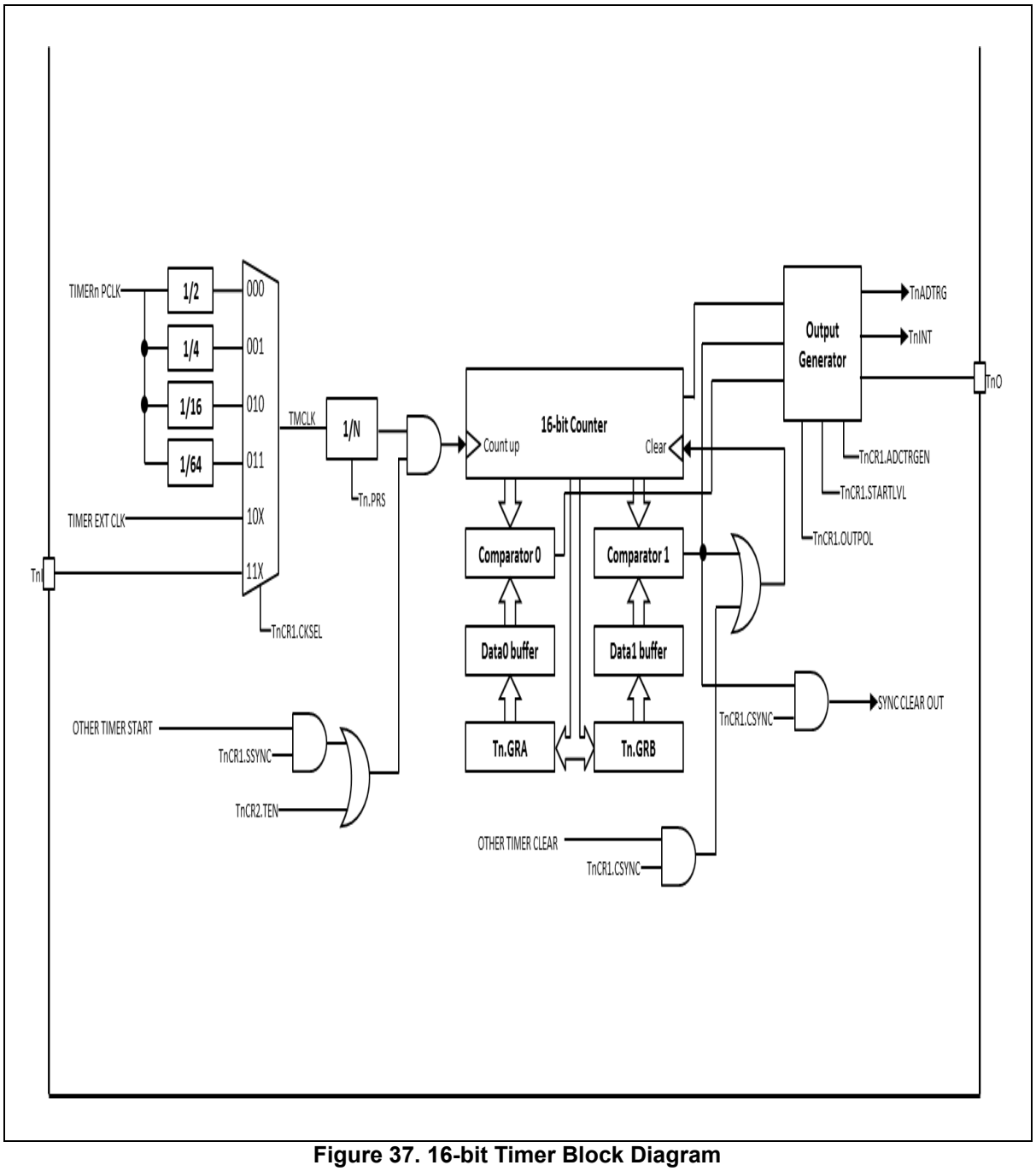


Figure 37. 16-bit Timer Block Diagram

10.2 Registers

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

Table 32. Base Address of 16-bit Timer

Name	Base address
T0	0x4000_3000
T1	0x4000_3020
T2	0x4000_3040
T3	0x4000_3060
T8	0x4000_3100
T9	0x4000_3120

Table 33. TIMER Register Map

Name	Offset	Type	Description	Reset value	Reference
Tn.CR1	0x--00	RW	Timer control register 1	0x00000000	10.2.1
Tn.CR2	0x--04	RW	Timer control register 2	0x00000000	10.2.2
Tn.PRS	0x--08	RW	Timer prescaler register	0x00000000	10.2.3
Tn.GRA	0x--0C	RW	Timer general data register A	0x00000000	10.2.4
Tn.GRB	0x--10	RW	Timer general data register B	0x00000000	10.2.5
Tn.CNT	0x--14	RW	Timer counter register	0x00000000	10.2.6
Tn.SR	0x--18	RW	Timer status register	0x00000000	10.2.7
Tn.IER	0x--1C	RW	Timer interrupt enable register	0x00000000	10.2.8

10.2.1 Tn.CR1 Timer n Control Register 1

Timer Control Register 1 is 16-bit register.

Timer module should be configured properly before running. When target purpose is defined, the timer can be configured in the Tn.CR1 register. After configuring this register, you can start or stop the timer function by Tn.CR2 register.

T0.CR1=0x4000_3000, T1.CR1=0x4000_3020
T2.CR1=0x4000_3040, T3.CR1=0x4000_3060
T8.CR1=0x4000_3100, T9.CR1=0x4000_3120

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSYNC	CSYNC	UAO	OUTPOL				ADCTRGEN	STARTLVL	CKSEL			CLRMOD		MODE	
0	0	0	0	0	0	0	0	0	000			00		00	
RW	RW	RW	RW				RW		RW			RW		RW	

15	SSYNC	Synchronize start counter with other synchronized timers	
		0	Single counter mode
		1	Synchronized counter start mode
14	CSYNC	Synchronize clear counter with other synchronized timers	
		0	Single counter mode
		1	Synchronized counter clear mode
13	UAO	Select GRA, GRB update mode	
		0	Writing GRA or GRB takes effect after current period
		1	Writing GRA or GRB takes effect in current period
12	OUTPOL	Timer output polarity	
		0	Normal output
		1	Negated output
8	ADCTRGEN	ADC Trigger enable control	
		0	Disable adc trigger
		1	Enable adc trigger
7	STARTLVL	Timer output polarity control	
		0	Default output level is HIGH
		1	Default output level is LOW
6 4	CKSEL[2:0]	Counter clock source select	
		000	PCLK/2
		001	PCLK/4
		010	PCLK/16
		011	PCLK/64
		10X	MCCR3 clock setting
		11X	TnIO pin input (TnIO pin must be set as input mode)
3 2	CLRMD	Clear select when capture mode	
		00	Rising edge clear mode
		01	Falling edge clear mode
		10	Both edge clear mode
		11	None clear mode
1 0	MODE[1:0]	Timer operation mode control	
		00	Normal periodic operation mode
		01	PWM mode
		10	One shot mode
		11	Capture mode

10.2.2 Tn.CR2 Timer n Control Register 2

Timer Control Register 2 is 8-bit register.

T0.CR2=0x4000_3004, T1.CR2=0x4000_3024
T2.CR2=0x4000_3044, T3.CR2=0x4000_3064
T8.CR2=0x4000_3104, T9.CR2=0x4000_3124

7	6	5	4	3	2	1	0
						TCLR	TEN
0	0	0	0	0	0	0	0
R	R	R	R	R	R	WO	RW

1	TCLR	Timer register clear
	0	Normal operation
	1	Clear count register. (This bit will be cleared after next timer clock)
0	TEN	Timer enable bit
	0	Stop timer counting
	1	Start timer counting
NOTE: It is recommended to start timer with TCLR bit setting to be '1'.		

10.2.3 Tn.PRS Timer n Prescaler Register

Timer Prescaler Register is 16-bit register in order to prescale the counter input clock.

T0.PRS=0x4000_3008, T1.PRS=0x4000_3028
T2.PRS =0x4000_3048, T3.PRS=0x4000_3068
T8.PRS=0x4000_3108, T9.PRS=0x4000_3128

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						PRS									
0	0	0	0	0	0	000									
						RW									

9	PRS	Pre-scale value of count clock
0		TCLK = CLOCK_IN/(PRS+1) (*CLOCK_IN* is a selected timer input clock)

10.2.4 Tn.GRATimer n General Register A

Timer General Register A is 16-bit register.

T0.GRA=0x4000_300C, T1.GRA=0x4000_302C
T2.GRA =0x4000_304C, T3.GRA=0x4000_306C
T8.GRA=0x4000_310C, T9.GRA=0x4000_312C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GRA															
0x0000															
RW															

15 0	GRA	General Register A (Duty/Interrupt Register) Periodic mode / PWM / One-shot mode - In PWM mode this register is used as duty value. - When the counter value is matched with this value, GRA Match interrupt is requested Capture mode - Falling edge of TnIO port will capture the count value when rising edge clear mode - Rising edge of TnIO port will capture the count value when falling edge clear mode

10.2.5 Tn.GRBTimer n General Register B

Timer General Register B is 16-bit register.

T0.GRB=0x4000_3010, T1.GRB=0x4000_3030
T2.GRB=0x4000_3050, T3.GRB=0x4000_3070
T8.GRB=0x4000_3110, T9.GRB=0x4000_3130

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GRB															
0x0000															
RW															

15 0	GRB	General Register B (Period Register) Periodic mode / PWM / One-shot mode - In periodic mode or PWM mode, this register is used as Period value. The counter will count up to (GRB-1) value. - When the counter value is matched with this value, GRB Match interrupt is requested only in PWM and one-shot modes. Capture mode - Rising edge of TnIO port will capture the count value when rising edge clear mode - Falling edge of TnIO port will capture the count value when falling edge clear mode

10.2.6 Tn.CNT Timer n Count Register.

Timer Count Register is 16-bit register.

T0.CNT=0x4000_3014, T1.CNT=0x4000_3034
T2.CNT=0x4000_3054, T3.CNT=0x4000_3074
T8.CNT=0x4000_3114, T9.CNT=0x4000_3134

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT															
0x0000 RW															

15 0	CNT	Timer count value register	
		R	Read current timer count value
		W	Set count value

10.2.7 Tn.SR Timer n Status Register

Timer Status Register is 8-bit register.

This register indicates the current status of timer module

T0.SR=0x4000_3018, T1.SR=0x4000_3038
T2.SR=0x4000_3058, T3.SR=0x4000_3078
T8.SR=0x4000_3118, T9.SR=0x4000_3138

7	6	5	4	3	2	1	0
					MFA	MFB	OVF
0	0	0	0	0	0 RW	0 RW	0 RW

2	MFA	GRA Match flag	
		0	No direction change
		1	Match flag with GRA
1	MFB	GRB Match flag	
		0	No direction change
		1	Match flag with GRB
0	OVF	Counter overflow flag	
		0	No direction change
		1	Counter overflow flag

10.2.8 Tn.IER Timer n Interrupt Enable Register

Timer Interrupt Enable Register is 8-bit register.

Each status flag of the timer block can issue the interrupt. To enable the interrupt, write "1" in correspondent bit in the TnIER register.

T0.IER=0x4000_301C, T1.IER=0x4000_303C
T2.IER=0x4000_305C, T3.IER=0x4000_307C
T8.IER=0x4000_311C, T9.IER=0x4000_313C

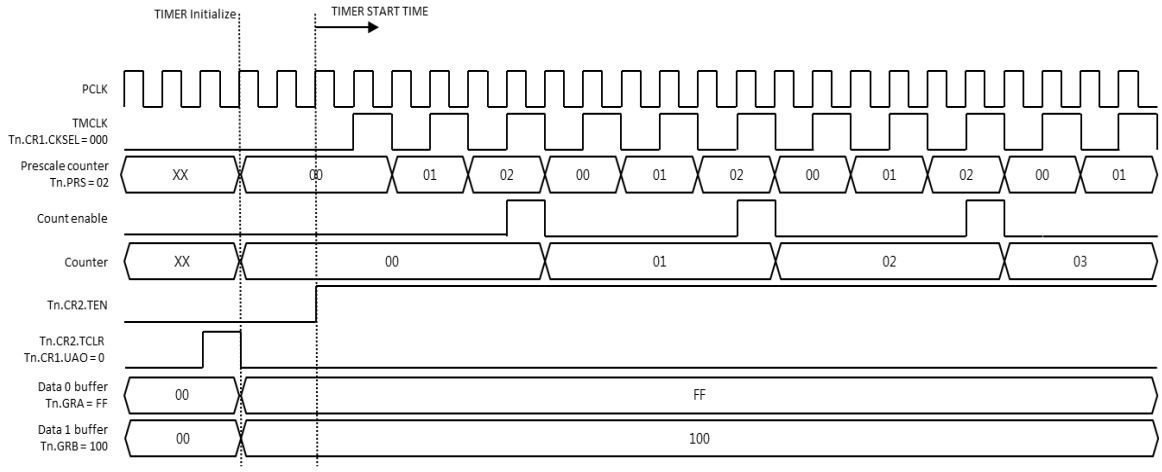
7	6	5	4	3	2	1	0
					MAIE	MBIE	OVIE
0	0	0	0	0	0	0	0
					W	RW	W

2	MAIE	GRA Match interrupt enable	
		0	Not effect
		1	Enable match register A interrupt
1	MBIE	GRB Match interrupt enable	
		0	Not effect
		1	Enable match register B interrupt
0	OVIE	Counter overflow interrupt enable	
		0	Not effect
		1	Enable counter overflow interrupt

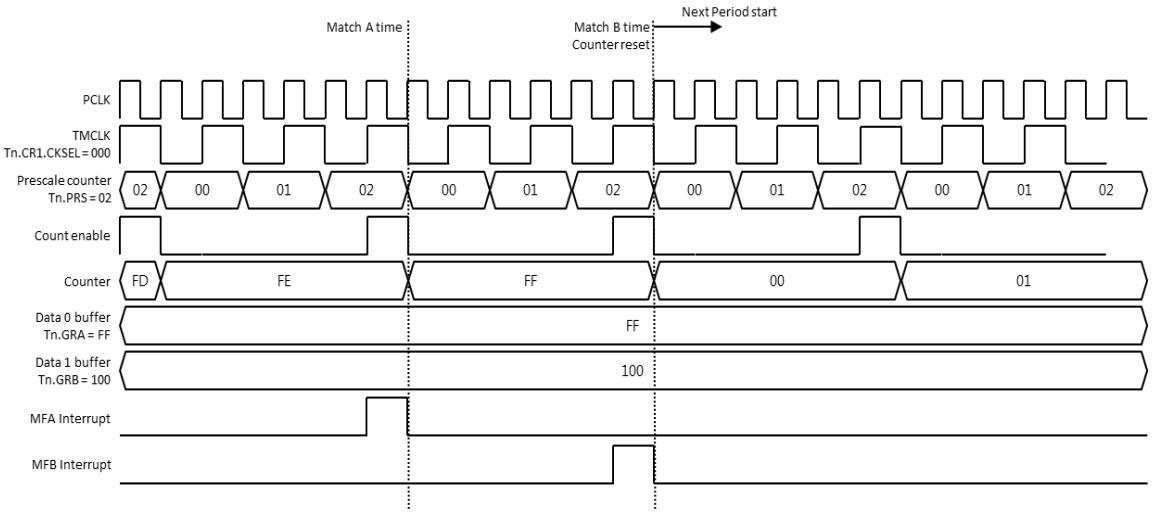
10.3 Functional description

10.3.1 Basic timer operations

The TMCLK shown in Figure 38 is reference clock for operation of the timer. This clock will be divided by prescaler setting and the counting clock will work. Below figures show the starting point of the counter and the ending of the period point of the counter in normal periodic mode.



(a) Timer initialization is done by TCLR command and timer will be started by TEN command



(b) Timer will be reset by match GRB timing and count again from 00

Figure 38. Basic Start and Match Operations

The following formula calculates the timer's count period:

The period = TMCLK Period * Tn.GRB value.

Match A interrupt time = TMCLK Period * Tn.GRA value.

If Tn.CR1.UAO bit is '0'. Tn.CR2.TCLR command will initialize all the registers in timer block and load the GRA and GRB value into Data0 and Data1 buffer. When you change the timer setting and restart the timer with new setting, it's recommended that you should write Tn.CR2.TCLR command before Tn.CR2.TEN command.

10.3.2 Normal periodic mode

Figure 39 shows the timing diagram for normal periodic mode. The GRA value determines the timer period. The GRB value has no effect on the timer in this mode.

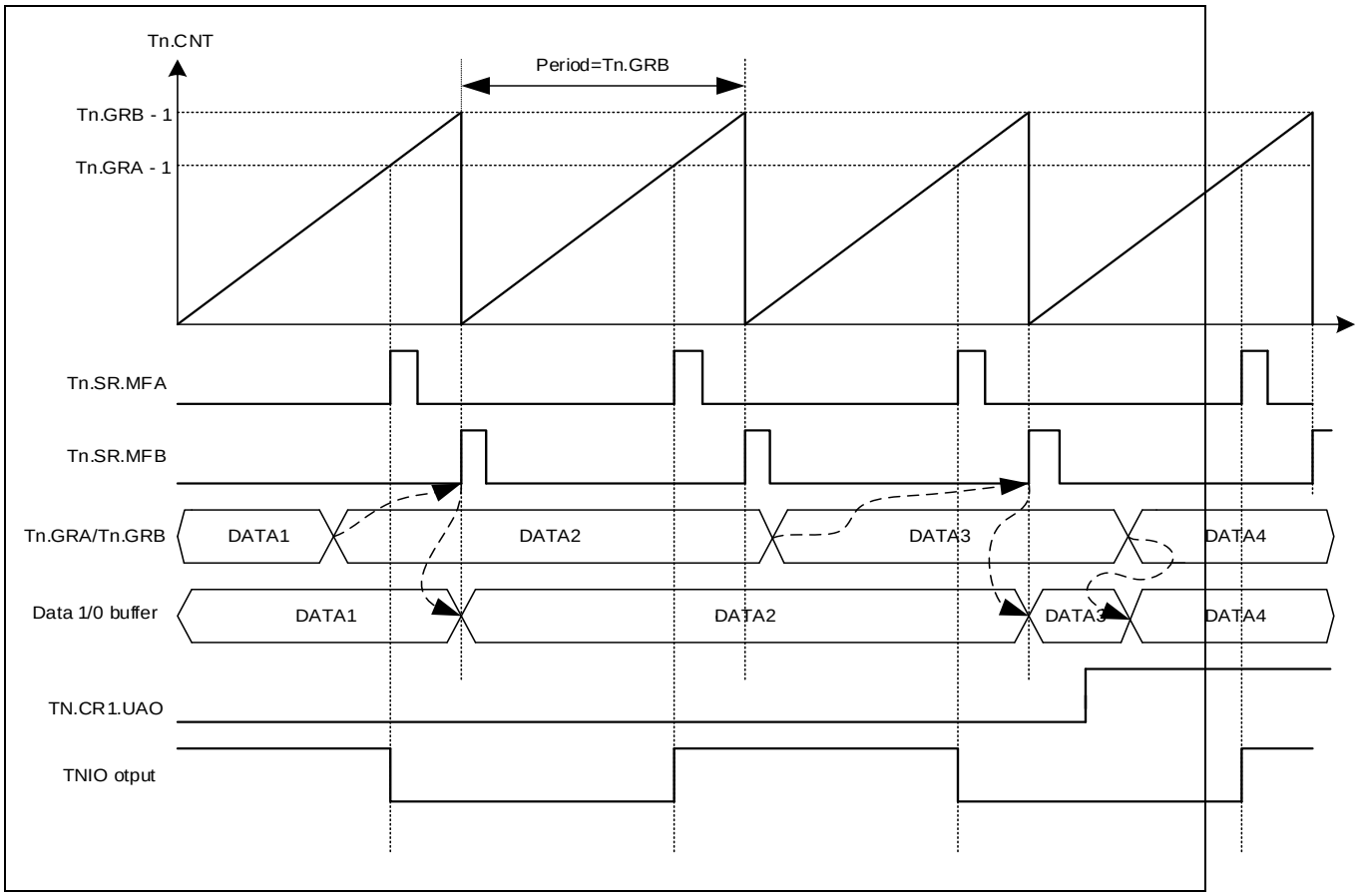


Figure 39. Normal Periodic Mode Operation

The following formula calculates the timer's count period:

Period = TMCLK period * GRA value

Match A interrupt time = TMCLK period * GRA value

If Tn.GRB = 0, the timer cannot be started even Tn.CR2.TEN is "1". That's because the period is "0".

The value in Tn.GRA and Tn.GRB is loaded into internal compare data buffer 0 and 1 when the loading condition is occurred. In this periodic mode with Tn.CR1.UAO =0, Tn.CR2.TCLR write operation will load the data buffer and the next GRB match event will load the data buffer.

When Tn.CR1.UAO is 1, the internal compare data buffer is updated whenever the Tn.GRA or Tn.GRB data is updated.

TnIO output signal will be toggled at every Match A condition time. If Tn.GRA is 0 value, the TnIO output is not change its previous level. If Tn.GRA is same as Tn.GRB, the TnIO output will toggle at same time as counter start time. The initial level of TnIO signal is decided by Tn.CR1.STARTLVL value.

10.3.3 One-shot mode

Figure 40 shows the timing diagram in one shot mode. Tn.GRB value decides the one shot period. One more compare point is provided with Tn.GRA register value.

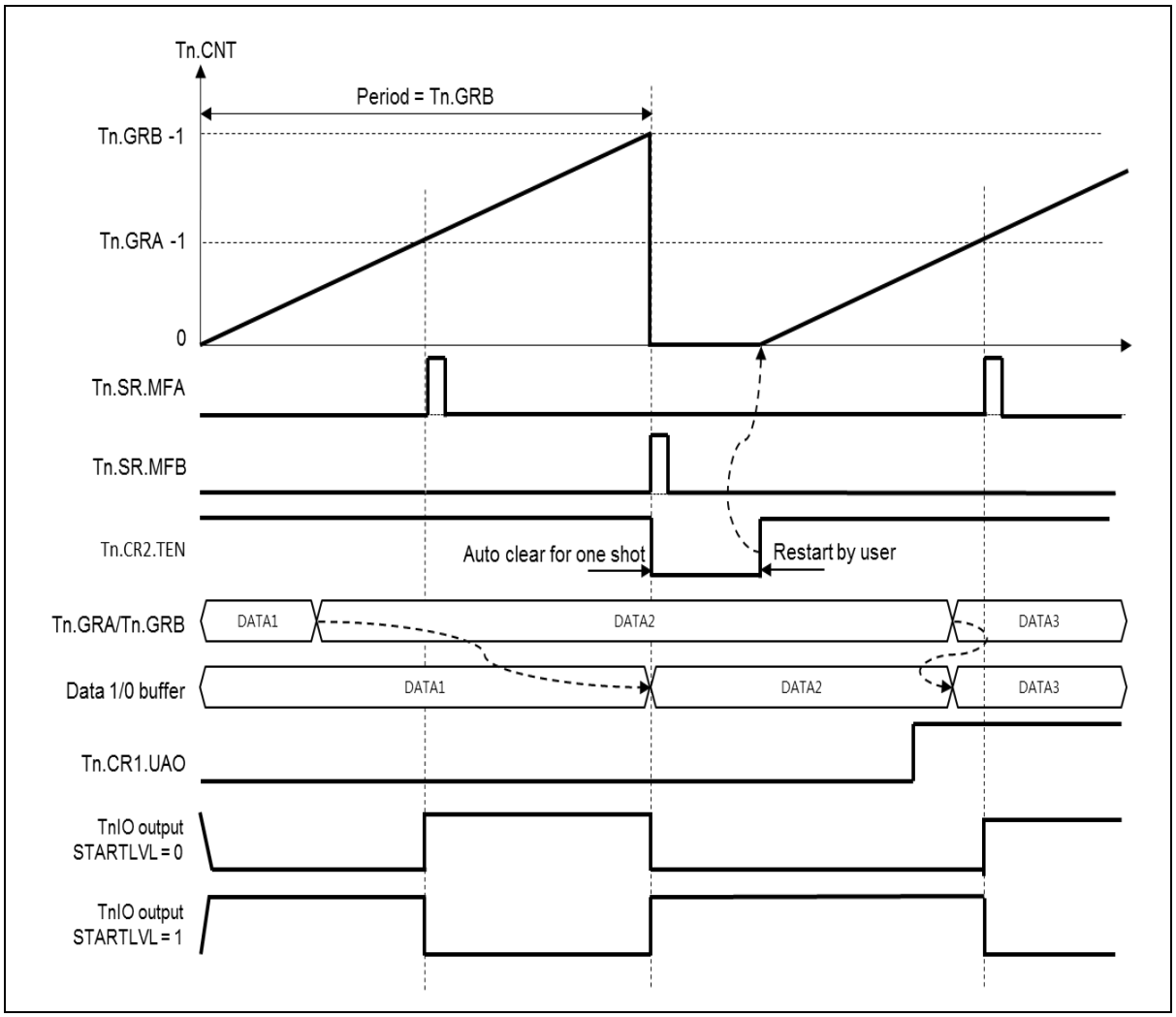


Figure 40. One-Shot Mode Operation

The period of one shot count can be calculated as below equation.

The period = TMCLK Period * Tn.GRB value.

Match A interrupt time = TMCLK Period * Tn.GRA value.

If Tn.GRB = 0, the timer cannot be started even Tn.CR2.TEN is "1". Because the period is "0".

The value in Tn.GRA and Tn.GRB is loaded into internal compare data buffer 0 and 1 when the loading condition is occurred. In this periodic mode with Tn.CR1.UAO =0, Tn.CR2.TCLR write operation will load the data buffer and the next GRB match event will load the data buffer.

When Tn.CR1.UAO is 1, the internal compare data buffer is updated whenever the Tn.GRA or Tn.GRB data is updated.

TnIO output signal format is same as PWM mode. Tn.GRB value defines the output pulse period and Tn.GRA value defines the pulse width of one shot pulse.

10.3.4 PWM timer output

Figure 41 shows the timing diagram PWM output mode. Tn.GRB value decides the PWM pulse period. One more compare point is provided with Tn.GRA register value which defines pulse width of PWM output.

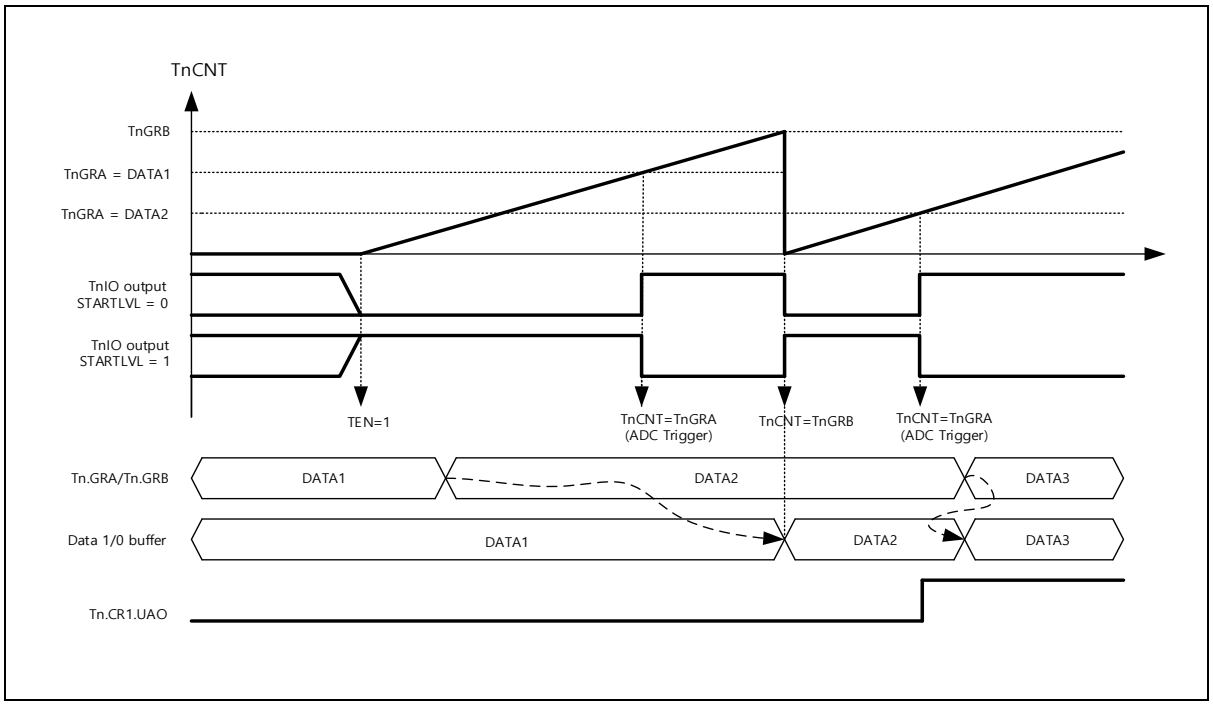


Figure 41. PWM Output Operation

The period of PWM pulse can be calculated as below equation.

The period = TMCLK Period * Tn.GRB value.

Match A interrupt time = TMCLK Period * Tn.GRA value.

If Tn.GRB = 0, the timer cannot be started even Tn.CR2.TEN is "1". Because the period is "0". The value in Tn.GRA and Tn.GRB is loaded into internal compare data buffer 0 and 1 when the loading condition is occurred. In this periodic mode with Tn.CR1.UAO =0, Tn.CR2.TCLR write operation will load the data buffer and the next GRB match event will load the data buffer. When Tn.CR1.UAO is 1, the internal compare data buffer is updated whenever the Tn.GRA or Tn.GRB data is updated.

TnIO output signal generates PWM pulse. Tn.GRB value defines the output pulse period and Tn.GRA value defines the pulse width of one shot pulse. The active level of PWM pulse can be control by Tn.CR1.STARTLVL bit value.

ADC Trigger generation is available at Match A interrupt time.

10.3.5 PWM synchronization function

2-PWM outputs are usually used as synchronous PWM signal control. This function is provided with synchronous start. Figure 42 shows synchronous PWM generation.

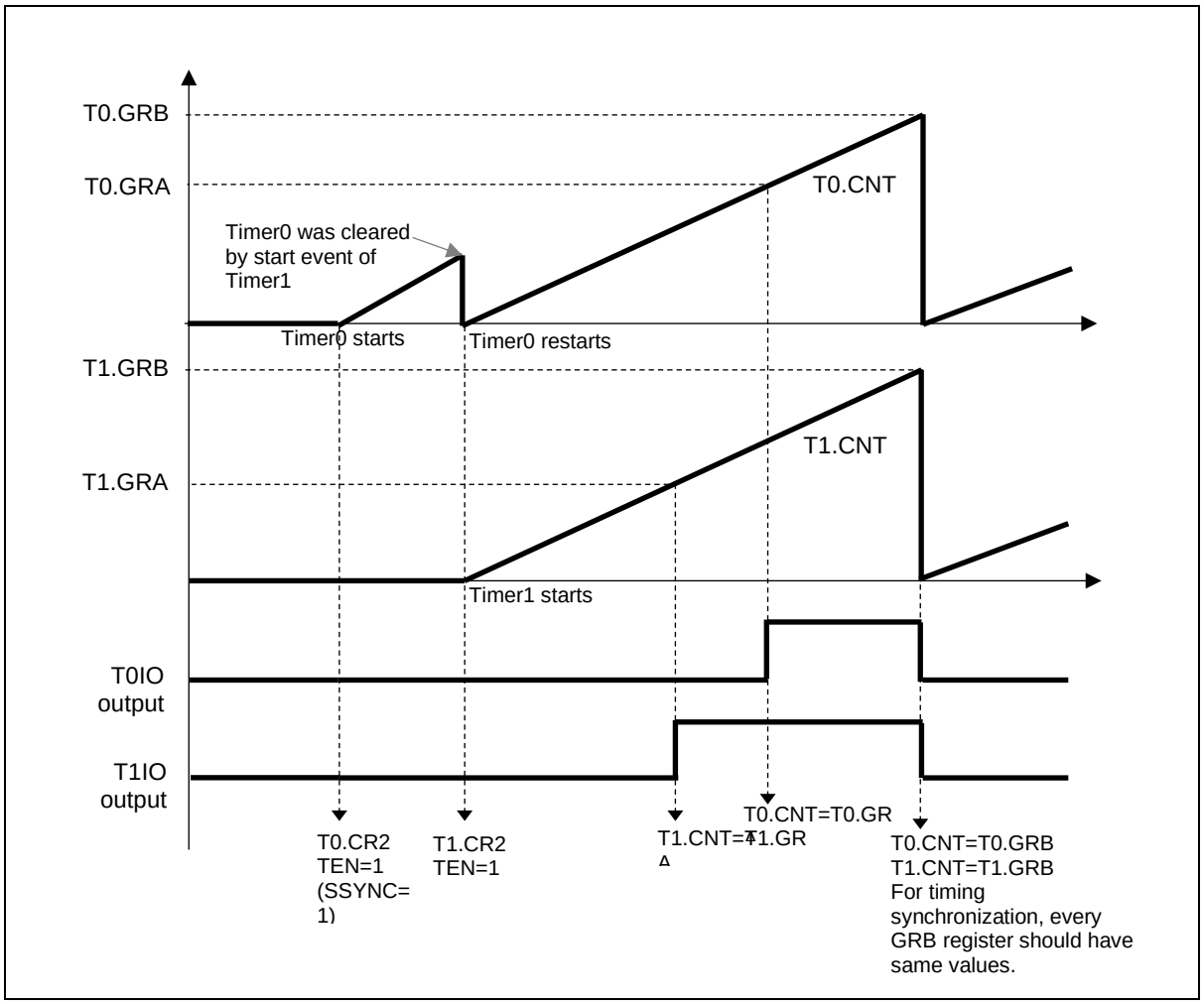


Figure 42. Timer Synchronization Example (When SSYNC = 1)

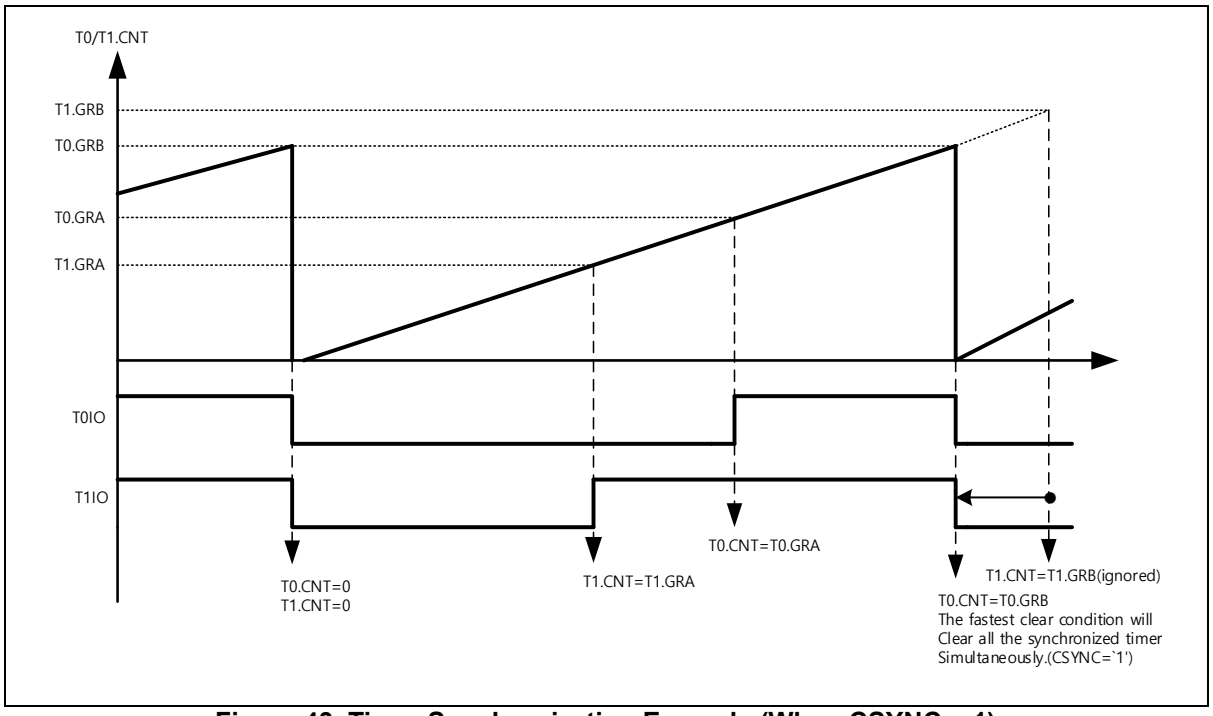


Figure 43. Timer Synchronization Example (When CSYNC = 1)

Tn.CR1.SSYNC bit controls start sync with other timer blocks. Tn.CR1.CSYCN bit controls clear sync with other timer blocks. These bit is effective at least 2 more timers are set these sync control bits.

For example, timer0 and timer1 set SSYNC and CCSYNC bit in each CR1 registers, both timers will started whenever one of them is enabled. And both timers will be cleared with short period match value. But others are not affected by these 2 timers, and they can be operated independently because their SYNC control bit is 0.

10.3.6 Capture mode

Figure 44 shows the timing diagram in the capture mode operation. The TnIO input signal is used for capturing the pulse. Rising and falling edges can capture the counter value in each capture condition.

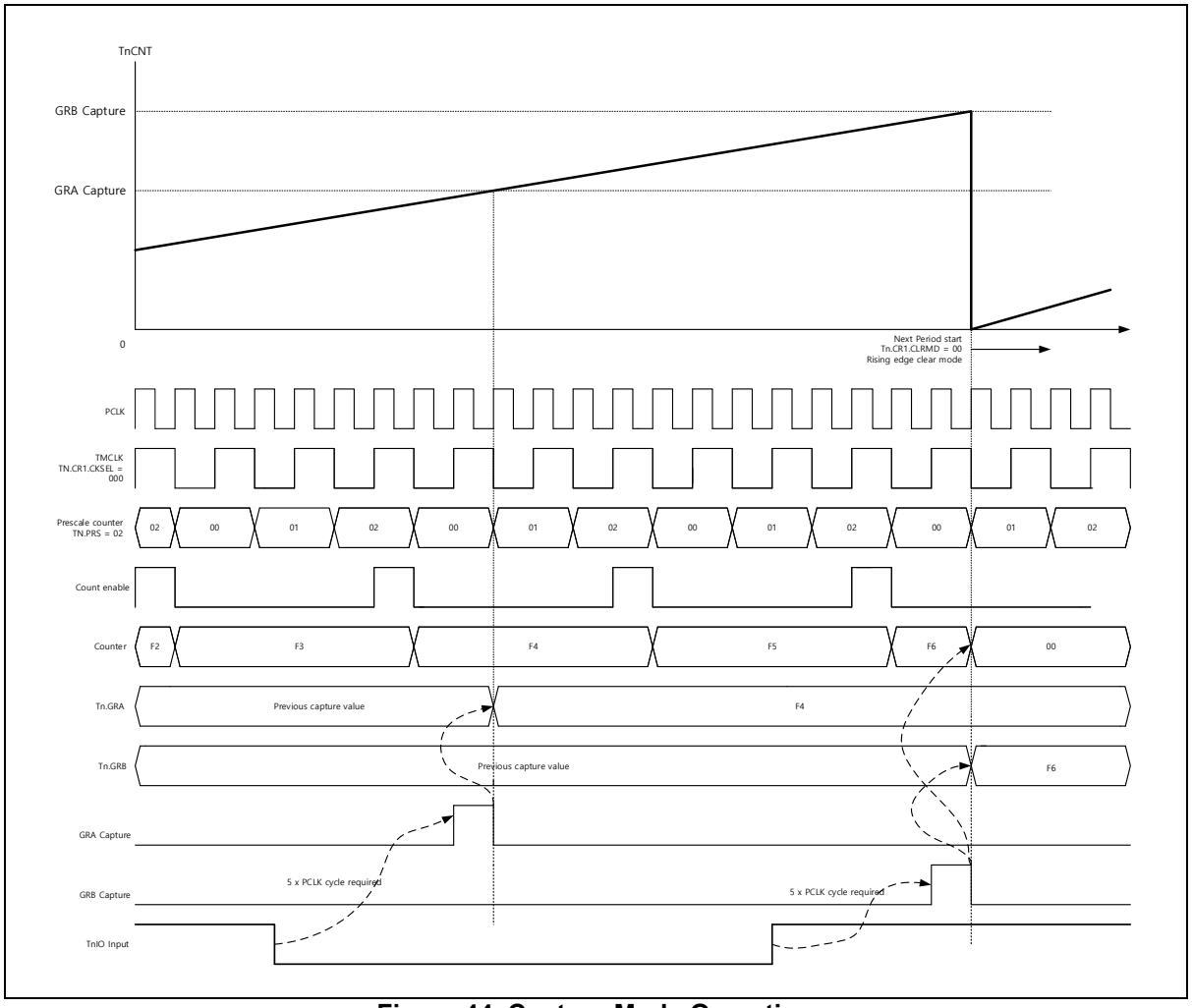


Figure 44. Capture Mode Operation

5 PCLK clock cycle is required internally. So actual capture point is after 5 PCLK clock cycles from rising or falling edge of TnIO input signal.

Internal counter can be cleared in various mode. Tn.CR1.CLRMD field controls the counter clear mode. Rising edge clear mode, falling edge clear mode, both edge clear mode and none clear mode are supported.

Figure 44 shows a case of rising edge clear mode.

10.3.7 ADC triggering

Timer module can generate ADC start trigger signals. One timer can be one trigger source of ADC block. Trigger source control is done by ADC control register (Figure 45 shows ADC trigger function).

The conversion rate must be shorter than timer period. If it is not a case, overrun situation can be happened. ADC acknowledge is not required, because trigger signal will be cleared automatically after 3 pclk clock pulses.

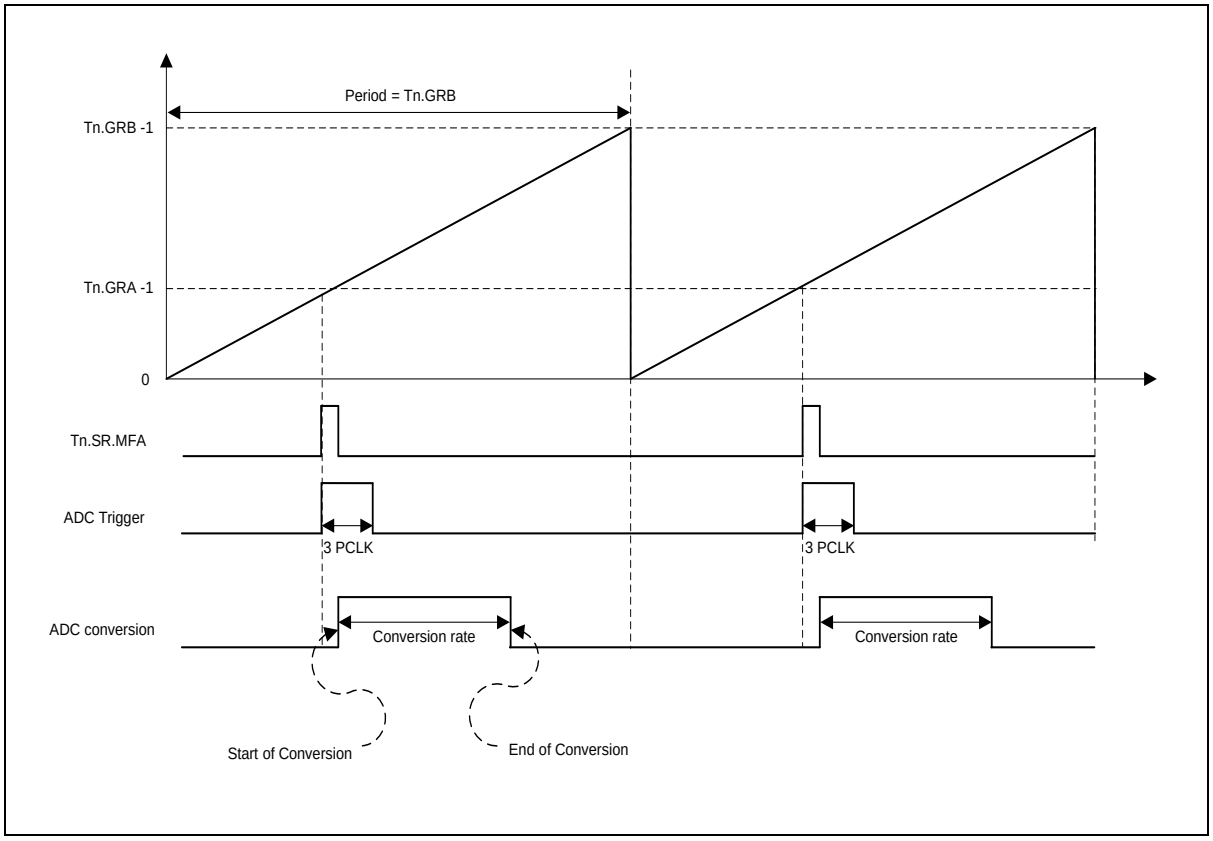


Figure 45. ADC Triggering Timing Diagram

11 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. The UART of AC33Mx064T series features the followings:

- Compatible with 16450
- Standard asynchronous control bit (start, stop, and parity) configurable
- Programmable 16-bit fractional baud generator
- Programmable serial communication

  5-, 6-, 7- or 8- bit data 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
- Loop-back control

Table 34 introduces pins assigned for the UART.

Table 34. Pin Assignment of UART: External Pins

Pin name	Type	Description
TXD0	O	UART Channel 0 transmit output
RXD0	I	UART Channel 0 receive input
TXD1	O	UART Channel 1 transmit output
RXD1	I	UART Channel 1 receive input

11.1 UART block diagram

In this section, UART is introduced in block diagrams.

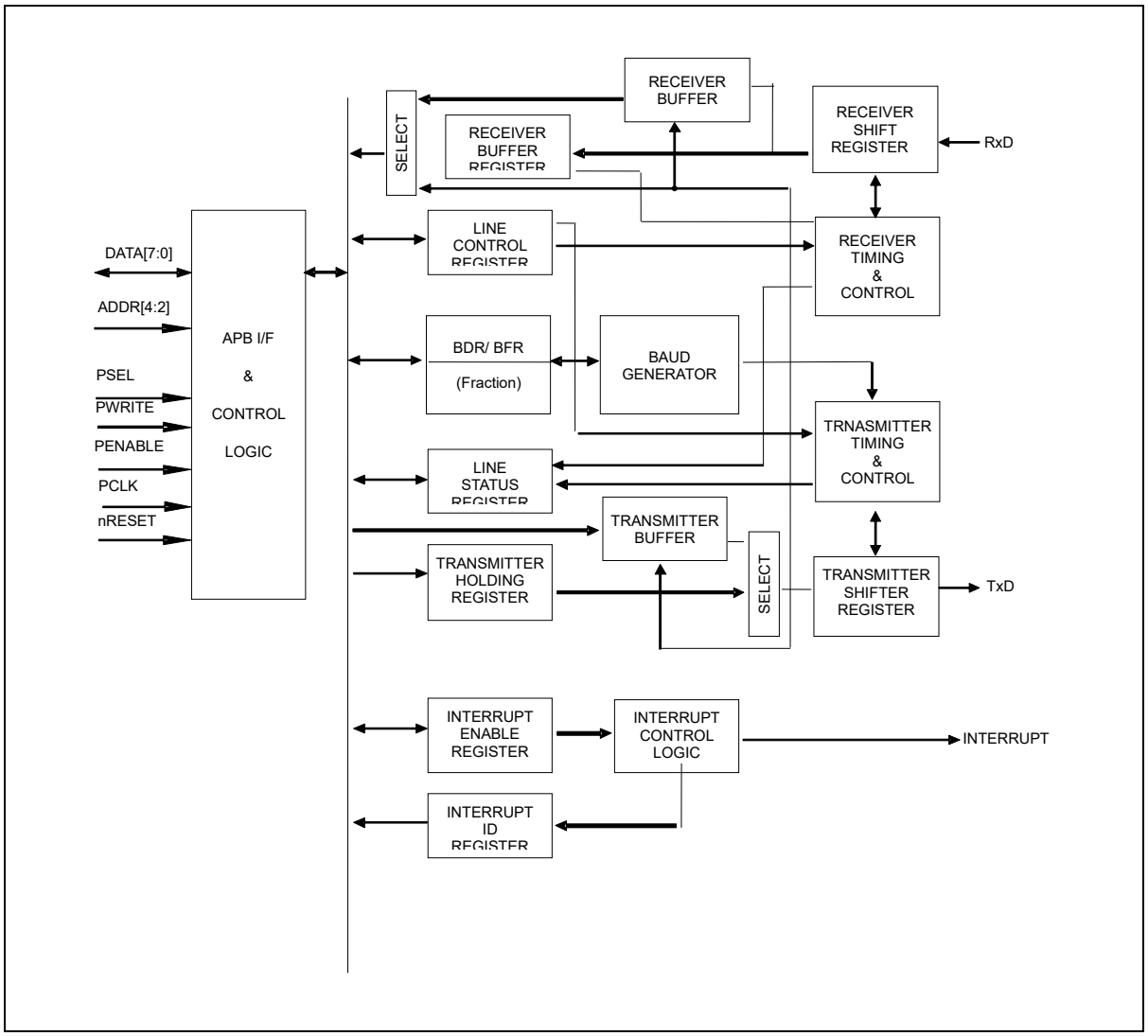


Figure 46. UART Block Diagram

11.2 Registers

Base address of UART is introduced in the followings:

Table 35. Base Address of UART

Name	Base address
UART 0	0x4000_8000
UART 1	0x4000_8100

Table 36. UART Register Map

Name	Offset	Type	Description	Reset value	Reference
Un.RBR	0x00	R	Receive data buffer register	0x00	11.2.1
Un.THR	0x00	W	Transmit data hold register	0x00	11.2.2
Un.IER	0x04	RW	Interrupt enable register	0x00	11.2.3
Un.IIR	0x08	R	Interrupt ID register	0x01	11.2.4
-	0x08	-	Reserved	-	
Un.LCR	0x0C	RW	Line control register	0x00	11.2.5
Un.DCR	0x10	RW	Data Control Register	0x00	11.2.6
Un.LSR	0x14	R	Line status register	0x00	11.2.7
-	0x18	-	reserved	-	
Un.BDR	0x20	RW	Baud rate Divisor Latch Register	0x0000	11.2.8
Un.BFR	0x24	RW	Baud rate Fractional Counter Value	0x00	11.2.9
Un.IDTR	0x30	RW	Inter-frame Delay Time Register	0x00	11.2.10

11.2.1 Un.RBR Receive Buffer Register

UART Receive Buffer Register is 8-bit Read-Only 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.

U0.RBR=0x4000_8000, U1.RBR=0x4000_8100

7	6	5	4	3	2	1	0
RBR[7:0]							
-							
RO							

7 0	RBR	Receive Buffer Register

11.2.2 Un.THR Transmit Data Hold Register

UART Transmit Data Hold Register is 8-bit Write-Only register. The data for transmit can be stored in this register. But the write data cannot be read from this register. The data which is written in UnTHR register, will be transferred into the transmit shifter register whenever the transmit shifter register is empty.

U0.THR=0x4000_8000, U1.THR=0x4000_8100

7	6	5	4	3	2	1	0
THR							
-							
WO							

7 0	THR	Transmit Data Hold Register

11.2.3 Un.IER UART Interrupt Enable Register

UART Interrupt Enable Register is 8-bit register.

U0.IER=0x4000_8004, U1.IER=0x4000_8104

7	6	5	4	3	2	1	0
-	-	DTXIEN	DRXIEN	TXEIE	RLSIE	THREIE	DRIE
0	0	0	0	0	0	0	0
		RW	RW	RW	RW	RW	RW

5	DTXIEN	DMA transmit done interrupt enable
		0 Receive line status interrupt is disabled
		1 Receive line status interrupt is enabled
4	DRXIEN	DMA receive done interrupt enable
		0 DMA receive done interrupt is disabled
		1 DMA receive done interrupt is enabled
3	TXEIE	Transmit register empty interrupt enable
		0 Transmit register empty interrupt is disabled
		1 Transmit register empty interrupt is enabled
2	RLSIE	Receiver line status interrupt enable
		0 Receive line status interrupt is disabled
		1 Receive line status interrupt is enabled
1	THREIE	Transmit holding register empty interrupt enable
		0 Transmit holding register empty interrupt is disabled
		1 Transmit holding register empty interrupt is enabled
0	DRIE	Data receive interrupt enable
		0 Data receive interrupt is disabled
		1 Data receive interrupt is enabled

11.2.4 n.IIR UART Interrupt ID Register

UART Interrupt ID Register is 8-bit register.

U0.IIR=0x4000_8008, U1.IIR=0x4000_8108

7	6	5	4	3	2	1	0
			TXE	IID			IPEN
0	0	0	0	000			0
			R	R			R

4	TXE	Interrupt source ID See interrupt source ID table
3 1	IID	Interrupt source ID See interrupt source ID table
0	IPEN	Interrupt pending bit 0 Interrupt is pending 1 No interrupt is pending.

The UART supports 3-priority interrupt generation and interrupt source ID register shows one interrupt source which has highest priority among pending interrupts. The priority is defined as below.

- Receive line status interrupt
- Receive data ready interrupt/ Character timeout interrupt
- Transmit hold register em
- pty interrupt
- Tx/Rx DMA complete interrupt

Table 37. Interrupt ID and Control

Priority	TXE	IID			IPEN	Interrupt sources		
	bit4	bit3	bit2	bit1	bit0	Interrupt	Interrupt condition	Interrupt clear
-	0	0	0	0	1	None	-	-
1	0	0	1	1	0	Receiver Line Status	Overrun, Parity, Framing or Break Error	Read LSR register
2	0	0	1	0	0	Receiver Data Available	Receive data is available.	Read receive register or read IIR register
3	0	0	0	1	0	Transmitter Holding Register Empty	Transmit buffer empty	Write transmit hold register or read IIR register
4	1	X	X	X	X	Transmitter Register Empty	Transmit register empty	Write transmit hold register or read IIR register
5	0	1	1	0	0	Rx DMA done	Rx DMA completed.	Read IIR register
6	0	1	0	1	0	Tx DMA done	Tx DMA completed.	Read IIR register
7	1	X	X	X	X	Transmitter register Empty and DMA done	Transmitter register Empty and Tx DMA completed.	Read IIR register

11.2.5 Un.LCRUART Line Control Register

UART Line Control Register is 8-bit register.

U0.LCR=0x4000_800C, U1.LCR=0x4000_810C

7	6	5	4	3	2	1	0
	BREAK	STICKP	PARITY	PEN	STOPBIT	DLEN	
0	0	0	0	0	0	0	0
	RW	RW	RW	RW	RW	RW	RW

6	BREAK	When this bit is set, TxD 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 and it will be effective when PEN bit is set.
	0	Parity stuck is disabled
	1	Parity stuck is enabled and parity always the bit of PARITY.
4	PARITY	Parity mode selection bit and stuck parity select bit
	0	Odd parity mode
	1	Even parity mode
3	PEN	Parity bit transfer enable
	0	The parity bit disabled
	1	The parity bit enabled
2	STOPBIT	The number of stop bit followed by data bits.
	0	1 stop bit
	1	1.5 / 2 stop bit In case of 5 bit data case, 1.5 stop bit is added. In case of 6,7 or 8 bit data, 2 stop bit is added
1 0	DLEN	The data length in one transfer word.
	00	5 bit data
	01	6 bit data
	10	7 bit data
	11	8 bit data

Parity bit will be generated according to bit 3, 4, 5 of UnLCR register. Table 38 shows the variation of parity bit generation.

Table 38. UART Parity Bit Configuration

STICKP	PARITY	PEN	Parity
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"

11.2.6 Un.DCR UART Data Control Register

UART Data Control Register is 8-bit register. The inversion function of Tx or Rx data line, is controlled by this Un.DCR register. When the corresponding bit is set to 1, the data line of Tx or Rx signal will be inverted.

U0.DCR=0x4000_8010, U1.DCR=0x4000_8110

7	6	5	4	3	2	1	0
				RXINV	TXINV		
0	0	0	0	0	0	0	0
				RW	RW		

3	RXINV	Rx Data Inversion Selection
		0 Normal RxData Input
		1 Inverted RxData Input
2	TXINV	Tx Data Inversion Selection
		0 Normal TxData Output
		1 Inverted TxData Output

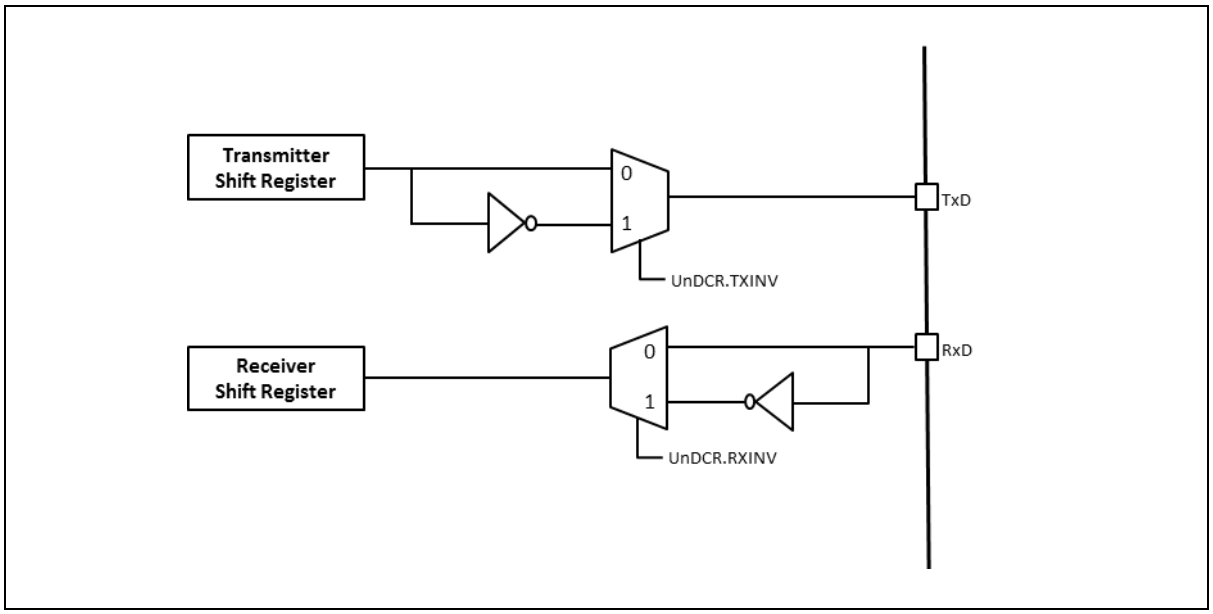


Figure 47. Data Inversion Control Diagram

11.2.7 Un.LSR UART Line Status Register

UART Line Status Register is 8-bit register.

U0.LSR=0x4000_8014, U1.LSR=0x4000_8114

7	6	5	4	3	2	1	0
-	TEMT	THRE	BI	FE	PE	OE	DR
0	1 R	1 R	0 R	0 R	0 R	0 R	0 R

6	TEMT	Transmit empty.
	0	Transmit register has the data is now transferring
	1	Transmit register is empty.
5	THRE	Transmit holding empty.
	0	Transmit holding register is not empty.
	1	Transmit holding register empty
4	BI	Break condition indication bit
	0	Normal status
	1	Break condition is detected
3	FE	Frame Error.
	0	No framing error.
	1	Framing error. The receive character did not have a valid stop bit
2	PE	Parity Error
	0	No parity error
	1	Parity error. The receive character does not have correct parity information.
1	OE	Overrun error
	0	No overrun error
	1	Overrun error. Additional data arrives while the RHR is full
0	DR	Data received
	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 UnIEN register is set. Other bits can generate its interrupt when its interrupt enable bit in UnIEN register is set.

11.2.8 Un.BDR Baud Rate Divisor Latch Register

UART Baud rate Divisor Latch Register is 16-bit register.

U0.BDR=0x4000_8020, U1.BDR=0x4000_8120

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDR															
0x0000 RW															

15 0	BDR	Baud rate Divider latch value

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 (UnBDR) should be written for expected baud rate.
Baud rate calculation formula is below.

$$\text{BDR} = \frac{PCLK/2}{16 \times BaudRate}$$

In case of 48 MHz UART_PCLK speed, the divider value and error rate is described in Table 39.

Table 39. Example of Baud Rate Calculation

PCLK=48 MHz		
Baud rate	**Divider (BDR)**	**Error (%)**
1200	1250	0.00%
2400	625	0.00%
4800	312	0.16%
9600	156	0.16%
19200	78	0.16%
38400	39	0.16%
57600	26	0.16%
115200	13	0.16%

11.2.9 Un.BFR Baud Rate Fraction Counter Register

Baud rate Fraction Counter Register is 8-bit register.

U0.BFR=0x4000_8024, U1.BFR=0x4000_8124

7	6	5	4	3	2	1	0
BFR							
0x00							
RW							

7 0	BFR	Fractions counter value.	
		0	Fraction counter is disabled
		N	Fraction counter enabled. Fraction compensation mode is operating. Fraction counter is incremented by FCNT.

Table 40. Example of Baud Rate Calculation with BFR

PCLK=48 MHz			
Baud rate	**Divider (BDR)**	**FCNT (BFR)**	**Error (%)**
1200	1250	0	0.0%
2400	625	0	0.0%
4800	312	128	0.0%
9600	156	64	0.0%
19200	78	32	0.0%
38400	39	16	0.0%
57600	26	10	0.01%
115200	13	5	0.01%

$$\text{FCNT} = \text{Float} * 256$$

8-bit fractional counter will count up by BFR 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.

Example) If baud rate is 9600 bps, following calculation leads to the BDR and BFR:

$$\frac{PCLK\ /\ 2}{16 \times\ BaudRate} = \frac{48000000\ /\ 2}{16 \times\ 9600} = 156.25 \quad Divider = 156, Float = 0.25$$

$$\text{FCNT} = \text{Float} * 256 \ = 0.25 * 256 = 64$$

BDR = 156, BFR = 64

11.2.10 Un.IDTR Inter-frame Delay Time Register

UART Inter-frame Time Register is 8-bit register.

Dummy delay can be inserted between two continuous transmits.

U0.IDTR=0x4000_8030, U1.IDTR=0x4000_8130

7	6	5	4	3	2	1	0
-					WAITVAL		UBCLKEN
0	0	0	0	0	00		0
					RW		W

2 1	WAITVAL	Wait time is decided by this value $\text{Wait Time} = \frac{\boldsymbol{WAITVAL} * \mathbf{2}}{\boldsymbol{BAUDRATE}}$
0	UBCLKEN	UART Baud clock enable. "1" is always written in this field whenever change WAITVAL value. Otherwise, UART operation is disabled.

11.3 Functional description

The UART module is compatible with 16450 UART. Additionally the dedicated DMA channels and fractional baud rate compensation logic are provided.

It doesn't have internal FIFO block. So data transfers will establish interactively or using DMA support. The DMA operation is described here.

Two DMA channels provided for each UART module, one channel is for TX transfer and the other one is for RX transfer. Each channel has a 32-bit memory address register and a 16-bit transfer counter register.

Before DMA operation, DMA memory address register and transfer count register should be configured. For the RX operation, the memory address will be destination memory address and for the TX operation, the memory address will be source memory address.

The transfer counter register will store the number of transfer data. Whenever a single transfer has been done, the counter will decremented by 1. When the counter reaches zero, the DMA done flag will delivered to UART control block. If the interrupt is enabled, this flag will generate the interrupt.

11.3.1 Receive data sampling timing

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 7th sample of 16 samples.

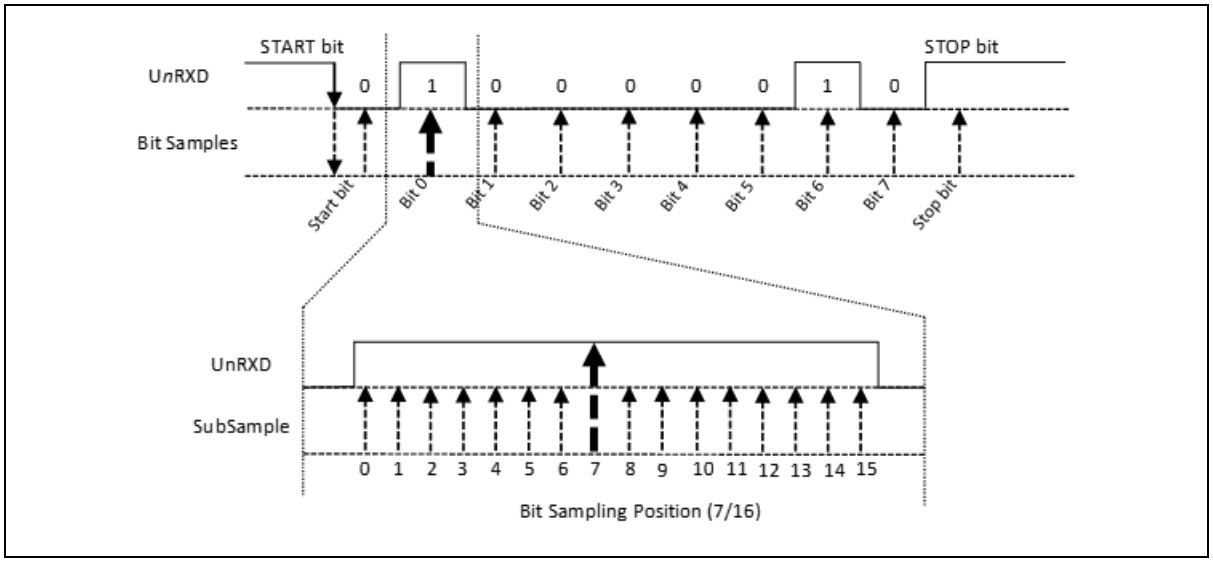


Figure 48. Sampling Timing of a UART Receiver

It is recommended to enable debounce settings in the PCU block to reinforce the immunity of external glitch noise.

11.3.2 Transmit data format

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 Un.LCR register.

The parity bit is set according to the PARITY and PEN bit filed in Un.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 Un.LCR register.

The example of transmit data format is below.

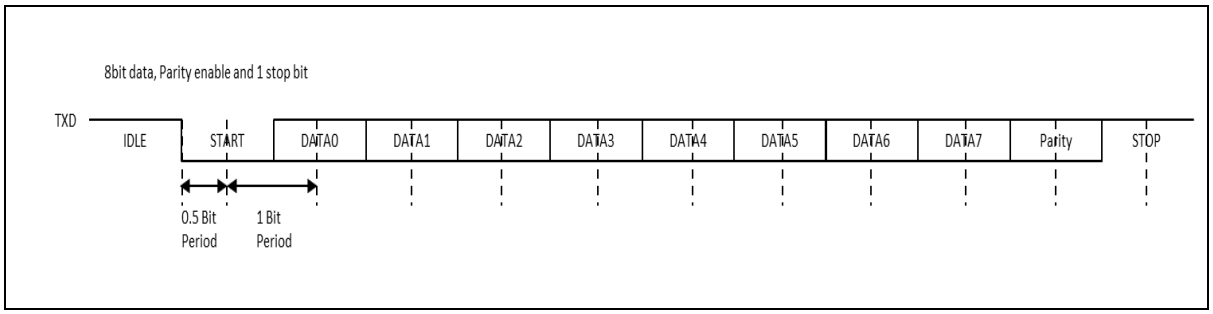


Figure 49. Transmit Data Format Example

11.3.3 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 Un.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.

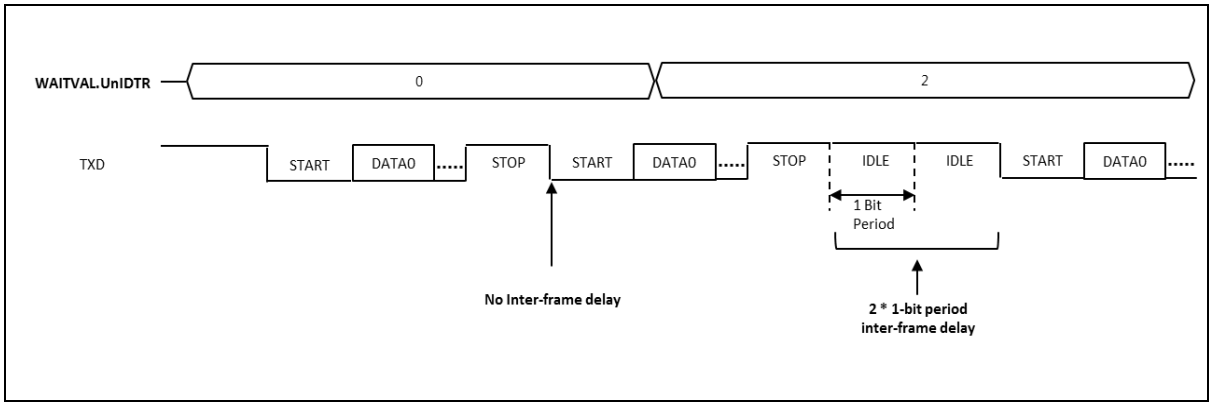


Figure 50. Inter-Frame Delay Timing Diagram

11.3.4 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.

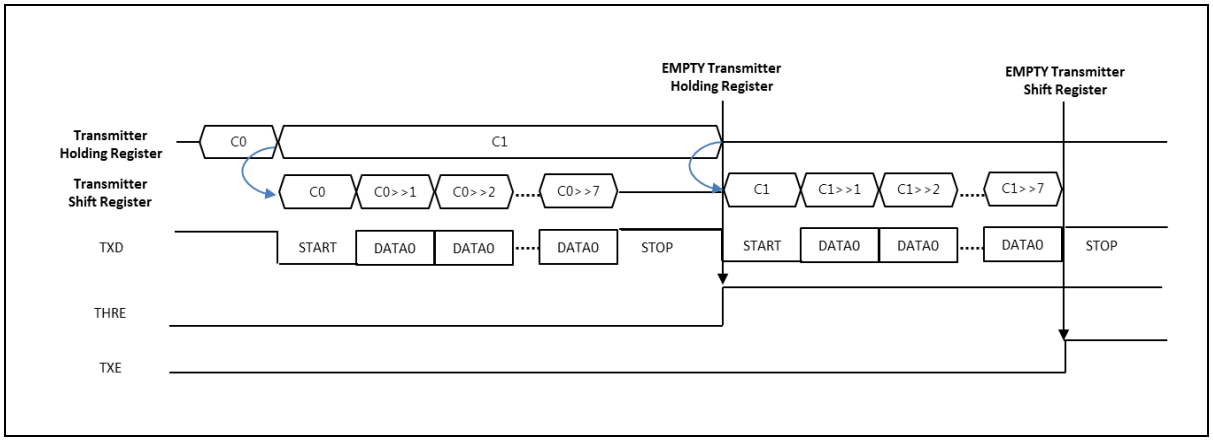


Figure 51. Transmit Interrupt Timing Diagram

11.3.5 DMA Transfers

The UART support DMA interface function. It is optionally provided depends on the device. The start memory address for transfer data and the length of transfer data are programmed in the registers in DMA block.

The end of transfer is notified related transfer done flag.

The transmit with DMA operation will invoke the DMA TX done flag DTX.UnIIR and will set DMA TX done interrupt ID when all the transmit data are written to transmit holding register. 2 transmit data are remained in registers in UART block after DMA transfer done interrupt.

The receive with DMA operation will invoke the DMA RX done flag RXT.UnIIR and will set DMA RX done interrupt ID when all the receive data are written to the destination memory. So, UART RXD signal is already IDLE state when the DMA RX done interrupt is issued.

12 Serial Peripheral Interface (SPI)

One Channel serial Interface is provided for synchronous serial communications with external peripherals. SPI block support both of master and slave mode. 4 signals will be used for SPI communication – SS, SCK, MOSI, and MISO.

- 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 41. Pin Assignment of SPI: External Pins

Pin name	Type	Description
SS	I/O	SPI Slave select input / output
SCK	I/O	SPI Serial clock input / output
MOSI	I/O	SPI Serial data (Master output, Slave input)
MISO	I/O	SPI Serial data (Master input, Slave output)

12.1 SPI block diagram

In this section, SPI is described in a block diagram in Figure 52.

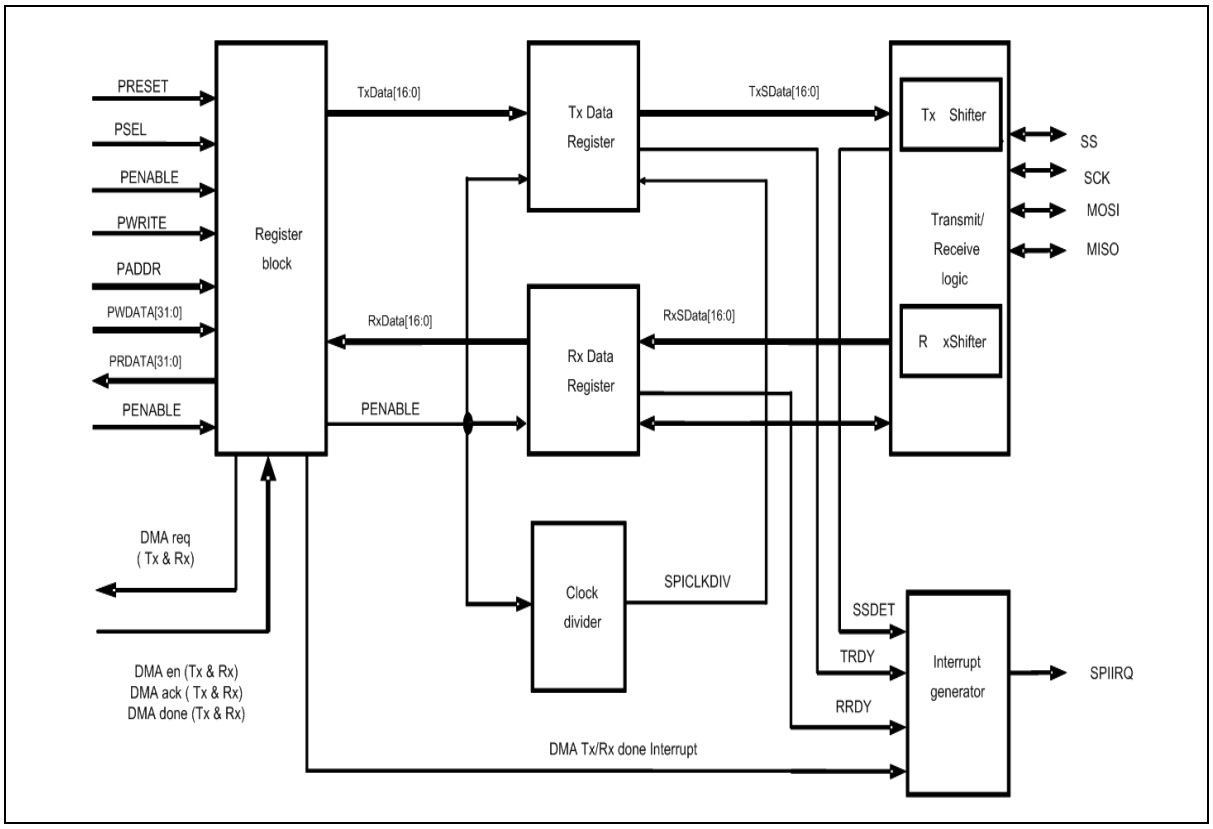


Figure 52. SPI Block Diagram

12.2 Registers

Base address of SPI is introduced in the followings:

Table 42. Base Address of SPI

Name	Base address
SPI	0x4000_9000

Table 43. SPI Register Map

Name	Offset	Type	Description	Reset value	Reference
SP0.TDR	0x00	W	SPI0 Transmit Data Register	-	12.2.1
SP0.RDR	0x00	R	SPI0 Receive Data Register	0x000000	12.2.2
SP0.CR	0x04	RW	SPI0 Control Register	0x001020	12.2.3
SP0.SR	0x08	RW	SPI0 Status Register	0x000006	12.2.4
SP0.BR	0x0C	RW	SPI0 Baud rate Register	0x0000FF	12.2.5
SP0.EN	0x10	RW	SPI0 Enable register	0x000000	12.2.6
SP0.LR	0x14	RW	SPI0 delay Length Register	0x010101	12.2.7

12.2.1 SP0.TDR SPI Transmit Data Register

SP0.TDR is a 17-bits read/write register. It contains serial transmit data.

SP0.TDR=0x4000_9000

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
																TDR															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x00000															
								RW																							

16	TDR	Transmit Data Register
0		

12.2.2 SP0.RDR SPI Receive Data Register

SP0.RDR is a 17-bits read/write register. It contains serial receive data.

SP0.RDR=0x4000_9000

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
																RDR															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x00000															
								RW																							

16	RDR	Receive Data Register
0		

12.2.3 SP0.CR SPI Control Register

SP0.CR is a 20-bits read/write register and can be set to configure SPI operation mode.

SP0.CR=0x4000_9004

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
											TXBC	RXBC	DTXIE	DRXIE	SSCIE	TXIE	RXIE	SSMOD	SSOUT	LBE	SSMARK	SS0MO	SS0POL			MS	MSBF	CPHA	CPOL	BITSZ	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	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		

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.
		0 SS output is not set by SSOUT (SPnCR[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.
7 6		Reserved
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	BITSZ	Transmit/Receive Data Bits select bit.	
0		0 0	8 bits
		0 1	9 bits
		1 0	16 bits
		1 1	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

12.2.4 SP0.SR SPI Status Register

SP0.SR is a 10-bits read/write register. It contains the status of SPI interface.

SP0.SR=0x4000_9008

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						TXDMAF	RXDMAF		SSDET	SSON	OVRF	UDRF	TXIDLE	TRDY	RRDY
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
						RC1	RC1		RC1	RC1	RC1	RC1	R	R	R

9	TXDMAF	DMA Transmit Operation Complete flag. (DMA to SPI)
	0	DMA Transmit Op is working or is disabled.
	1	DMA Transmit Op is done.
8	RXDMAF	DMA Receive Operation Complete flag. (SPI to DMA)
	0	DMA Receive Operation is working or is disabled.
	1	DMA Transmit Op is done.
7		Reserved
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".
5	SSON	SS signal Status flag.
	0	SS signal is inactive.
	1	SS signal is active.
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 SPnRDR.
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 SPnTDR.
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 SPnTDR.
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 SPnRDR.

12.2.5 SP0.BR SPI Baud Rate Register

SP0.BR is an16-bits read/write register. Baud rate can be set by writing the register.

SP0.BR=0x4000_900C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR															
0x00FF															
RW															

15 0	BR	Baud rate setting bits Baud Rate = PCLK / (BR + 1) (BR must be bigger than "0", BR >= 2)

12.2.6 SP0.EN SPI Enable register

SP0.EN is a bit read/write register. It contains SPI enable bit.

SP0.EN=0x4000_9010

7	6	5	4	3	2	1	0
							ENABLE
0	0	0	0	0	0	0	0
							RW

0	ENABLE	SPI Enable bit
		SPI is disabled. SPnSR is initialized by writing "0" to this bit but other registers aren't initialized.
		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 SPTDR before this bit is active.

12.2.7 SP0.LR SPI Delay Length Register

SP0.LR is a 24-bits read/write register. It contains start, burst, and stop length value.

SP0.LR=0x4000_9014

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
								SPL								BTL								STL							
0	0	0	0	0	0	0	0	0x01								0x01								0x01							
								RW								RW								RW							

23	SPL	StoPLength value
16		0x01 ~ 0xFF: 1 ~ 255 SCLKs. (SPL ≥ 1)
15	BTL	BursTLength value
8		0x01 ~ 0xFF: 1 ~ 255 SCLKs. (BTL ≥ 1)
7	STL	STart Length value
0		0x01 ~ 0xFF: 1 ~ 255 SCLKs. (STL ≥ 1)

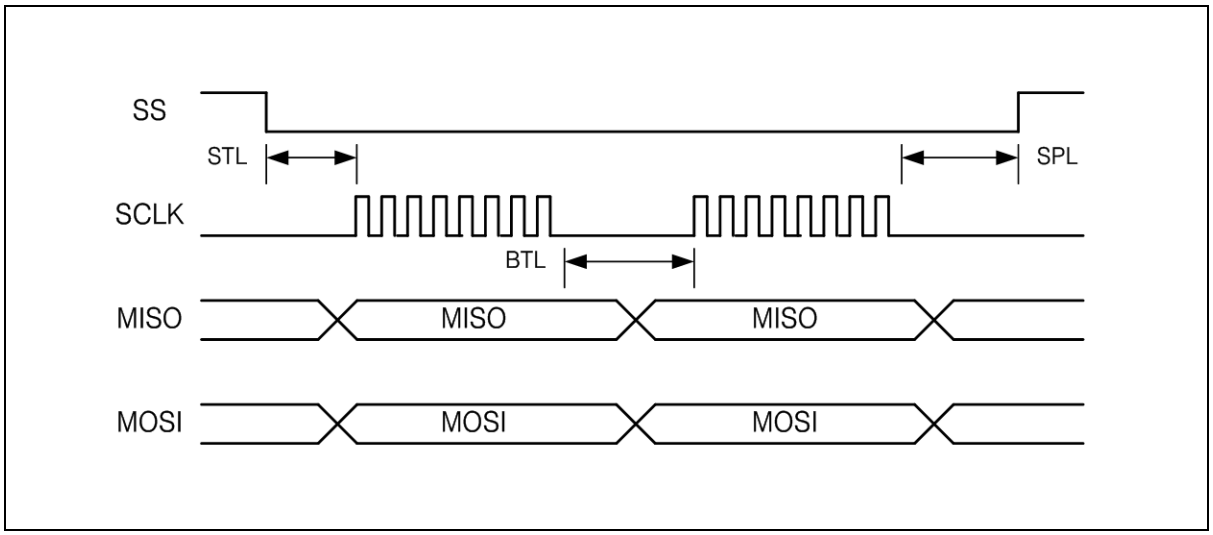


Figure 53. SPI Wave Form (STL, BTL and SPL)

12.3 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.

12.3.1 SPI timing

The 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 (SP.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 54 and Figure 55. 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 (SP.TDR) is output on the MISO line. The actual transfer is started by a software write to the SP.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.

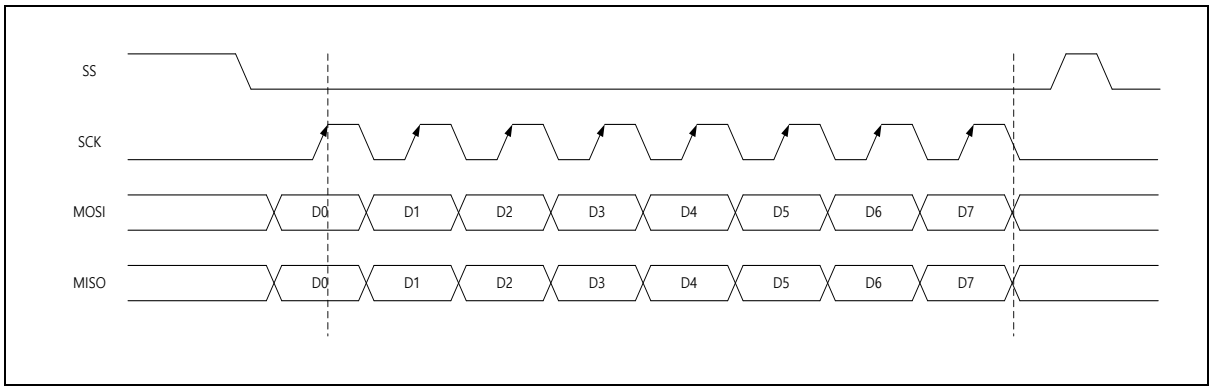


Figure 54. SPI Transfer Timing 1/4 (CPHA = 0, CPOL = 0, MSBF = 0)

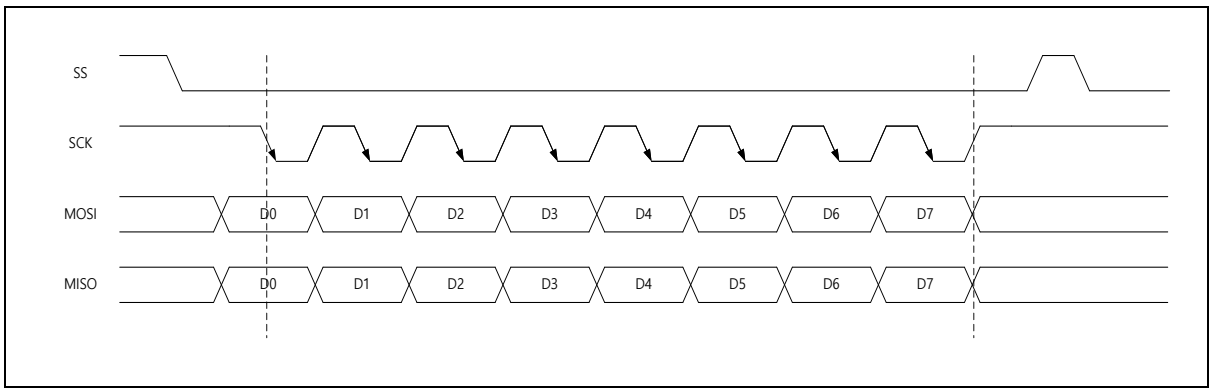


Figure 55. SPI Transfer Timing 2/4 (CPHA = 0, CPOL = 1, MSBF = 1)

The timing of a SPI transfer where CPHA is one is shown in Figure 56 and Figure 57.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 SP.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 SP.TDR.

As shown in Figure 56 and Figure 57, 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.

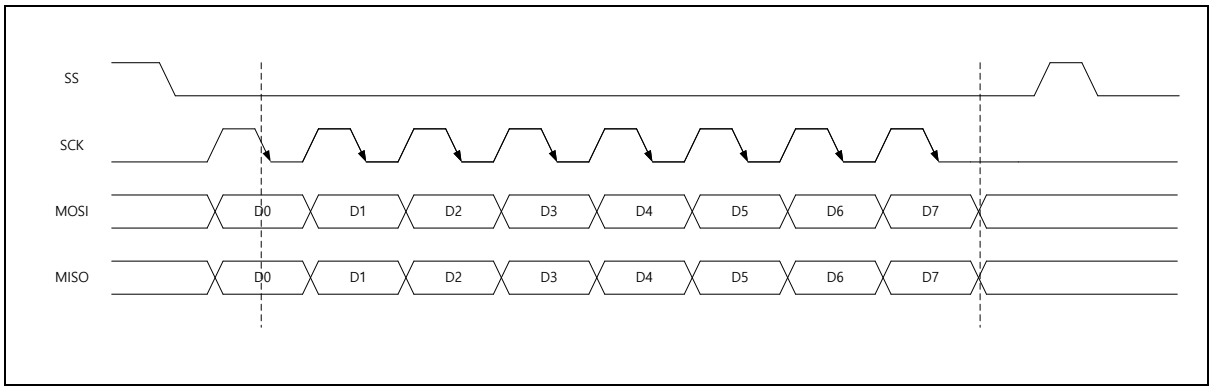


Figure 56. SPI Transfer Timing 3/4 (CPHA = 1, CPOL = 0, MSBF = 0)

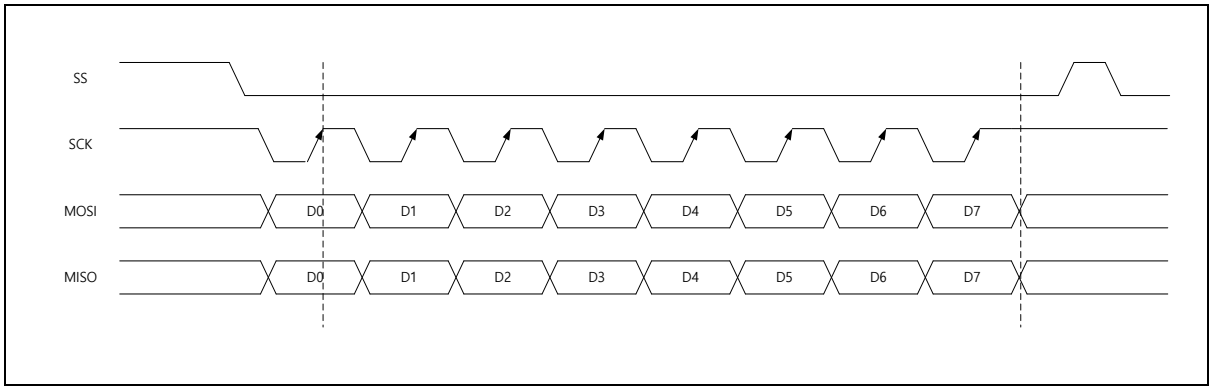


Figure 57. SPI Transfer Timing 4/4 (CPHA = 1, CPOL = 1, MSBF = 1)

12.3.2 DMA handshake

SPI supports DMA handshaking operation. In order to operate DMA handshake, DMA registers should be set first. (See Chapter 6. DMA Controller). 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 (SPnSR[8]) goes "1" and an interrupt is serviced when RXDIE (SPnCR[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 (SPnSR[9]) goes "1" and an interrupt is serviced when TXDIE (SPnCR[18]) is set.

Slave transmitter sends dummy data at the first transfer (8~17 SCLKs) in DMA handshake mode.

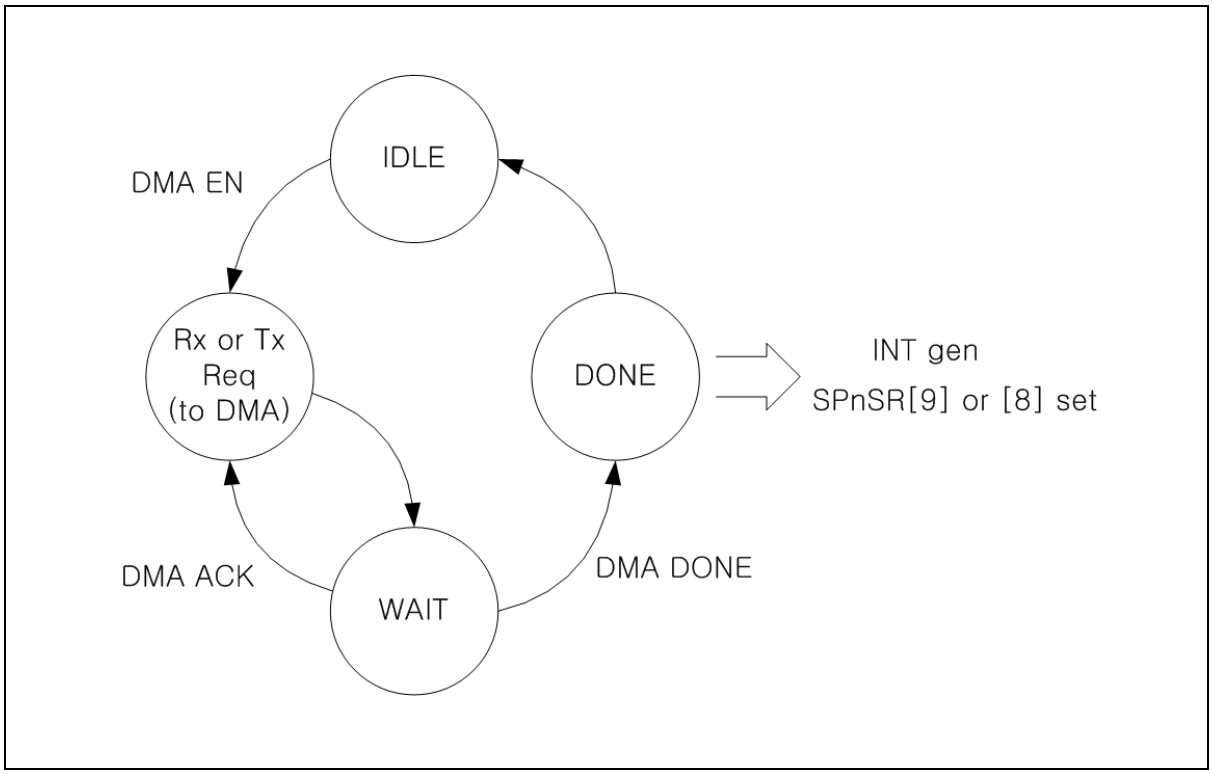


Figure 58. DMA Handshake Flow Chart

13 Inter Integrated Circuit (I2C)

Inter-Integrated Circuit (I2C) bus serves as an interface between the microcontroller and the serial I2C bus. It provides two wires, serial bus interface to a large number of popular devices and allows parallel-bus systems to communicate bidirectional with the I2C-bus.

I2C of AC33Mx064T features the followings:

- Master and slave operation
- Programmable communication speed
- Multi-master bus configuration
- 7-bit addressing mode
- Standard data rate of 100/400Kbps
- STOP signal generation and detection
- START signal generation
- ACK bit generation and detection

Table 44. Pin Assignment of I2C: External Pins

Pin name	Type	Description
SCL	I/O	I^2C channel Serial clock bus line (open-drain)
SDA	I/O	I^2C channel Serial data bus line (open-drain)

13.1 I2C block diagram

In this section, I2C interface block is described in a block diagram.

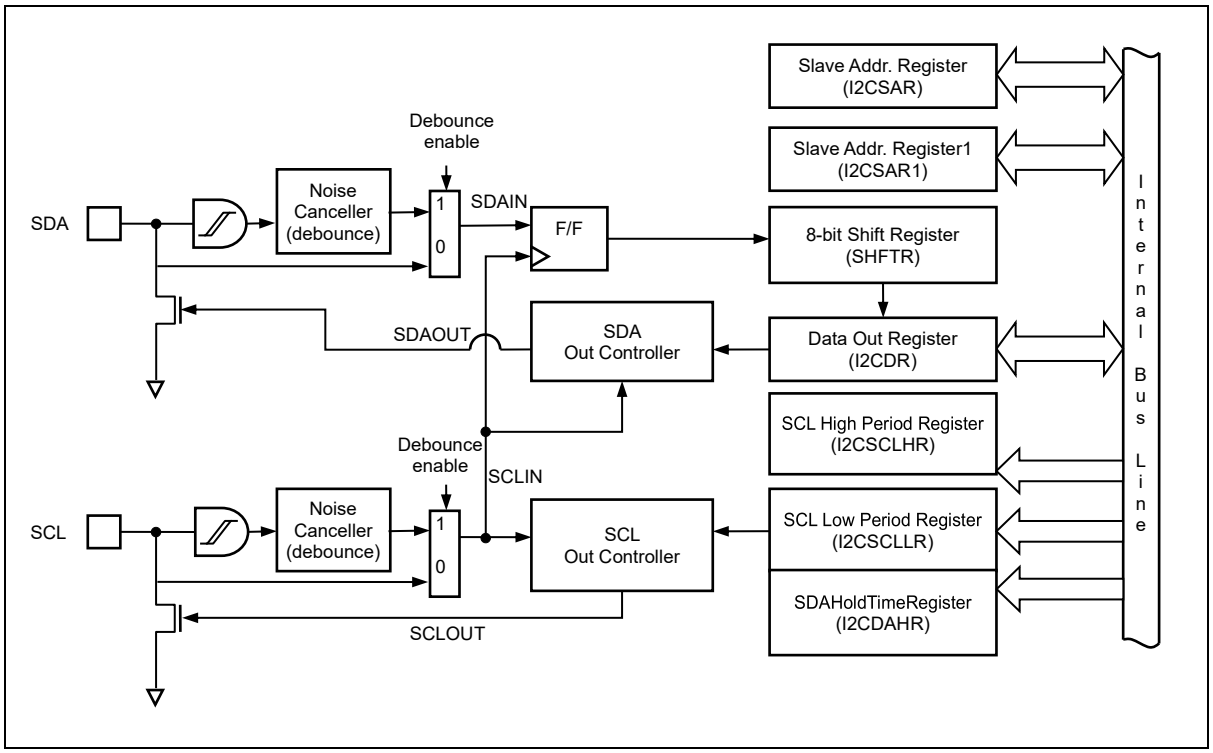


Figure 59. I2C Block Diagram

13.2 Registers

Base address of I2C is introduced in the followings:

Table 45. Base Address of I2C Interface

Name	Base address
I2C	0x4000_A000

Table 46. I2C Register Map

Name	Offset	Type	Description	Reset value	Reference
IC*n*.DR	0x00	RW	I²C0 Data Register	0xFF	13.2.1
IC*n*.SR	0x08	R, RW	I²C0 Status Register	0x00	13.2.2
IC*n*.SAR	0x0C	RW	I²C0 Slave Address Register	0x00	13.2.3
IC*n*.CR	0x14	RW	I²C0 Control Register	0x00	13.2.4
IC*n*.SCLL	0x18	RW	I²C0 SCL LOW duration Register	0xFFFF	13.2.5
IC*n*.SCLH	0x1C	RW	I²C0 SCL HIGH duration Register	0xFFFF	13.2.6
IC*n*.SDH	0x20	RW	I²C0 SDA Hold Register	0x7F	13.2.7

13.2.1 ICn.DR I²C Data Register

ICn.DR is an 8-bits read/write register. It contains a byte of serial data to be transmitted or a byte which has just been received.

IC0.DR=0x4000_A000

7	6	5	4	3	2	1	0
ICDR							
0xFF							
RW							

7 0	ICDR	The most recently received data or data to be transmitted.

13.2.2 ICn.SR I²C Status Register

ICn.SR is an 8-bit read/write register. It contains the status of I²C bus interface. Writing to the register clears the status bits.

IC0.SR=0x4000_A008

7	6	5	4	3	2	1	0
GCALL	TEND	STOP	SSEL	MLOST	BUSY	TMOD	RXACK
0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW

7	GCALL	General call flag	
		0	General call is not detected.
		1	General call detected or slave address (ID byte) was sent.
6	TEND	1 Byte transmission complete flag	
		0	The transmission is working or not completed.
		1	The transmission is completed.
5	STOP	STOP flag	
		0	STOP is not detected.
		1	STOP is detected.
4	SSEL	Slave flag	
		0	Slave is not selected.
		1	Slave is selected.
3	MLOST	Mastership lost flag	
		0	Mastership is not lost.
		1	Mastership is lost.
2	BUSY	BUSY flag	
		0	I²C bus is in IDLE state.
		1	I²C bus is busy.
1	TMODE	Transmitter/Receiver mode flag	
		0	Receiver mode.
		1	Transmitter mode.
0	RXACK	Rx ACK flag	
		0	Rx ACK is not received.
		1	Rx ACK is received.

13.2.3 ICn.SAR I²C Slave Address Register

ICn.SAR is an 8-bits read/write register. It shows the address in slave mode.

IC0.SAR=0x4000_A00C

7	6	5	4	3	2	1	0
SVAD							GCEN
0x00							0
RW							RW

7 1	SVAD	7-bit Slave Address
0	GCEN	General call enable bit
		0 General call is disabled.
		1 General call is enabled.

13.2.4 ICn.CR I²C Control Register

ICn.CR is an 8-bits read/write register. The register can be set to configure I2C operation mode and simultaneously allowed for I2C transactions to be kicked off.

IC0.CR=0x4000_A014

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						INTDEL		IIF		SOFTRST	INTEN	ACKEN		STOP	START
0	0	0	0	0	0	00		0	0	0	0	0	0	0	0
						RW		R		RW	RW	RW		RW	RW

9 8	INTDEL	Interval delay value between address and data transfer (or DATA and DATA)
		0 1 * ICnSCLL
		1 2 * ICnSCLL
		2 4 * ICnSCLL
		3 8 * ICnSCLL
7	IIF	Interrupt status bit
		0 Interrupt is inactive
		1 Interrupt is active
5	SOFTRST	Soft Reset enable bit.
		0 Soft Reset is disabled.
		1 Soft Reset is enabled.
4	INTEN	Interrupt enabled bit.
		0 Interrupt is disabled.
		1 Interrupt is enabled.
3	ACKEN	ACK enable bit in Receiver mode.
		0 ACK is not sent after receiving data.
		1 ACK is sent after receiving data.
1	STOP	Stop enable bit. When this bit is set as "1" in transmitter mode, next transmission will be stopped even though ACK signal has been received.
		0 Stop is disabled.
		1 Stop is enabled. When this bit is set, transmission will be stopped.
0	START	Transmission start bit in master mode.
		0 Waits in slave mode.
		1 Starts transmission in master mode.

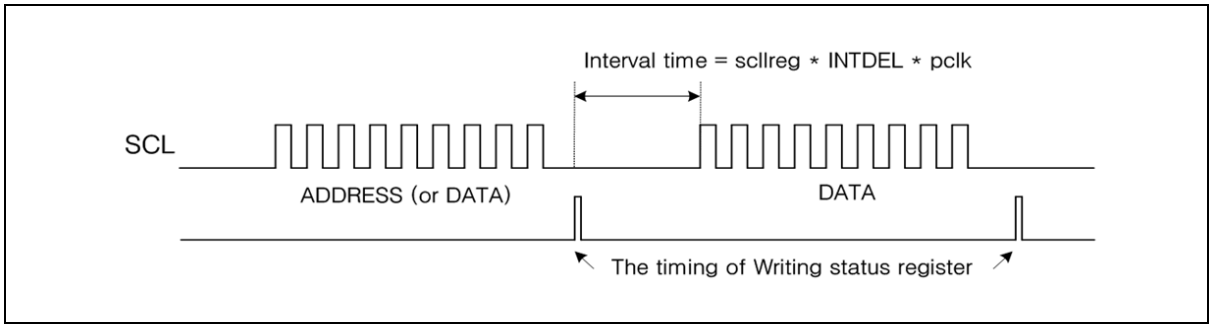


Figure 60. INTDEL in Master Mode

13.2.5 ICn.SCLL I²C SCL LOW Duration Register

ICnSCLL is a 16-bit read/write register. SCL LOW time can be set by writing this register in master mode.

IC0.SDLL=0x4000_A018

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCLL															
0xFFFF															
RW															

15	SCLL	SCL LOW duration value. SCLL = (PCLK * SCLL[15:0]) + 2*PCLKs
0		Default value is 0xFFFF.

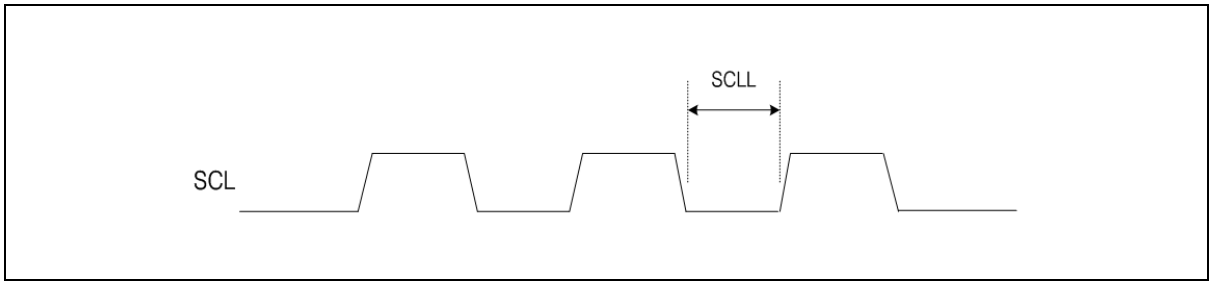


Figure 61. SCL LOW Timing

13.2.6 ICn.SCLH I²C SCL HIGH Duration Register

ICnSCLH is a 16-bit read/write register. SCL HIGH time will be set by writing this register in master mode.

IC0.SDLH=0x4000_A01C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCLH															
0xFFFF															
RW															

15 0	SCLH	SCL HIGH duration value. SCLH = (PCLK * SCLH[15:0]) + 3 PCLKs Default value is 0xFFFF.

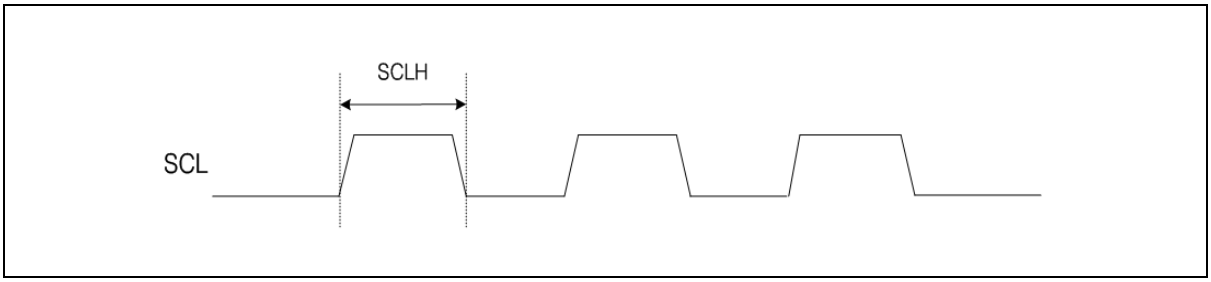


Figure 62. SCL LOW Timing

13.2.7 ICn.SDH SDA Hold Register

ICnSDH is a 15-bit read/write register. SDA HOLD time will be set by writing this register in master mode.

IC0.SDH=0x4000_A020

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDH														
0x3FFF															
RW															

14	SDH	SDA HOLD time setting value. SDH = (PCLK * SDH[14:0]) + 4 PCLKs
0		Default value is 0x3FFF.

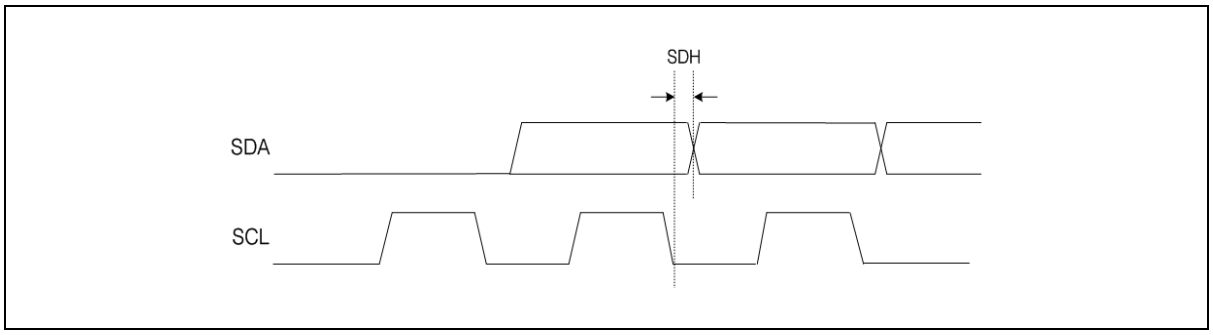


Figure 63. SDA HOLD Timing

13.3 Functional description

13.3.1 I2C bit transfer

The data on the SDA line must be stable during the "H" period of the clock. The "H" or "L" state of the data line can only change when the clock signal on the SCL line is "L" (see Fig 11.6).

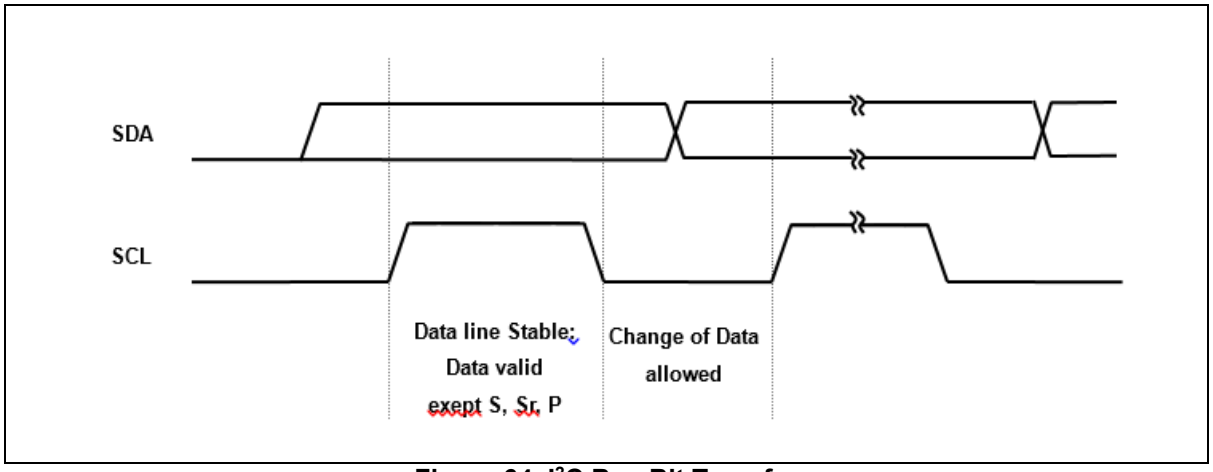


Figure 64. I²C Bus Bit Transfer

13.3.2 START/ repeated START/ STOP

Within the procedure of the I2C-bus, unique situations arise which are defined as START(S) and STOP(P) conditions (see Figure 65).

An "H" to "L" transition on the SDA line while SCL is "H" is one such unique case. This situation indicates a START condition.

A "L" to "H" transition on the SDA line while SCL is "H" defines a STOP condition.

START and STOP conditions are always generated by the master. The bus is considered to be busy after the START condition. The bus is considered to be free again a certain time after the STOP condition.

The bus is busy if a repeated START(Sr) is generated instead of a STOP condition. In this respect, the START(S) and repeated START(Sr) conditions are functionally identical. For the remainder of this document therefore, the S symbol will be used as a generic term to represent both the START and repeated START conditions, unless Sr is particularly relevant.

Detection of START and STOP conditions by devices connected to the bus is easy if they incorporate the necessary interfacing hardware. However, microcontrollers with no such interface have to sample the SDA line at least twice per clock period to sense the transition.

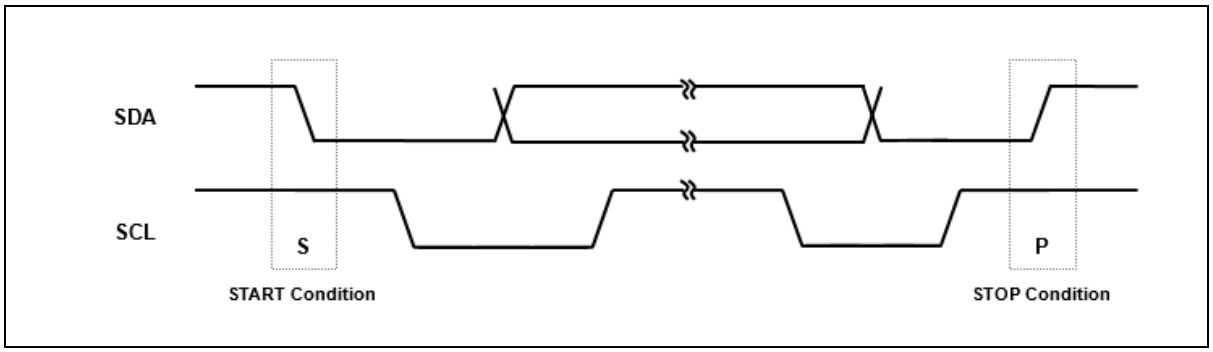


Figure 65. START and STOP Conditions

13.3.3 Data transfer

Every byte put on the SDA line must be 8-bits long. The number of bytes that can be transmitted per transfer is unrestricted. Each byte has to be followed by an acknowledge bit. Data is transferred with the most significant bit (MSB) first (see Figure 66).

If a slave can't receive or transmit another complete byte of data until it has performed some other function, for example servicing an internal interrupt, it can hold the clock line SCL "L" 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 SCL.

A message which starts with such an address can be terminated by generation of a STOP conditions, even during the transmission of a byte. In this case, no acknowledge is generated.

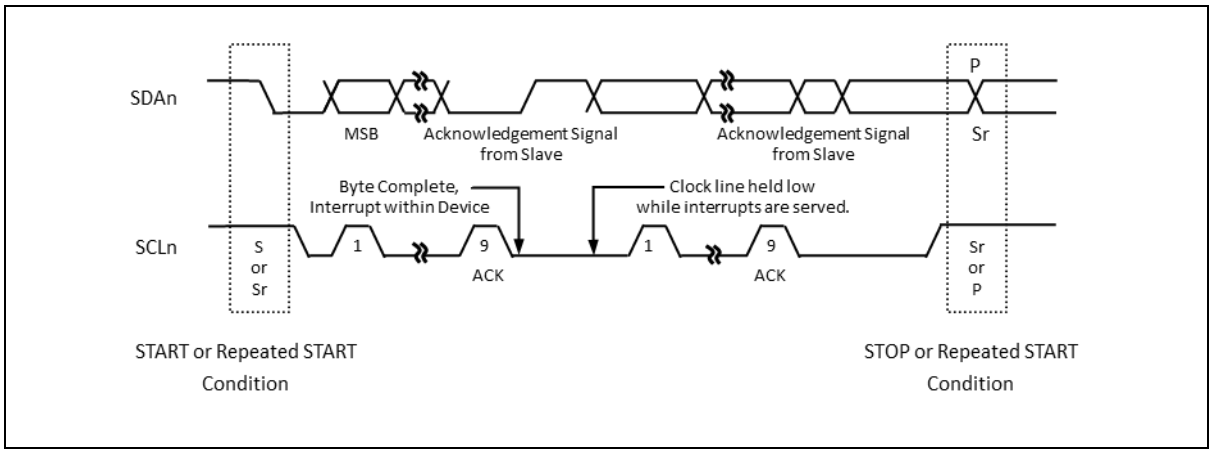


Figure 66. I2C Bus Data Transfer

13.3.4 Acknowledge

Data transfer with acknowledge is obligatory. The acknowledge-related clock pulse is generated by the master. The transmitter releases the SDA line (HIGH) during the acknowledge clock pulse.

The receiver must pull down the SDA line during the acknowledge clock pulse so that it remains stable "L" during the "H" period of this clock pulse (see Figure 67). Of course, set-up and hold times must also be taken into account.

When a slave doesn't acknowledge the slave address (for example, it's unable to receive or transmit because it's performing some real-time function), the data line must be left "H" by the slave. 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 slave-receiver does acknowledge the slave address but, sometime later in the transfer cannot receive any more data bytes, the master must again abort the transfer. This is indicated by the slave generating the not-acknowledge on the first byte to follow. The slave leaves the data line "H" and the master generates a STOP or a repeated START condition.

If a master-receiver is involved in a transfer, it must signal the end of data to the slave-transmitter by not generating 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.

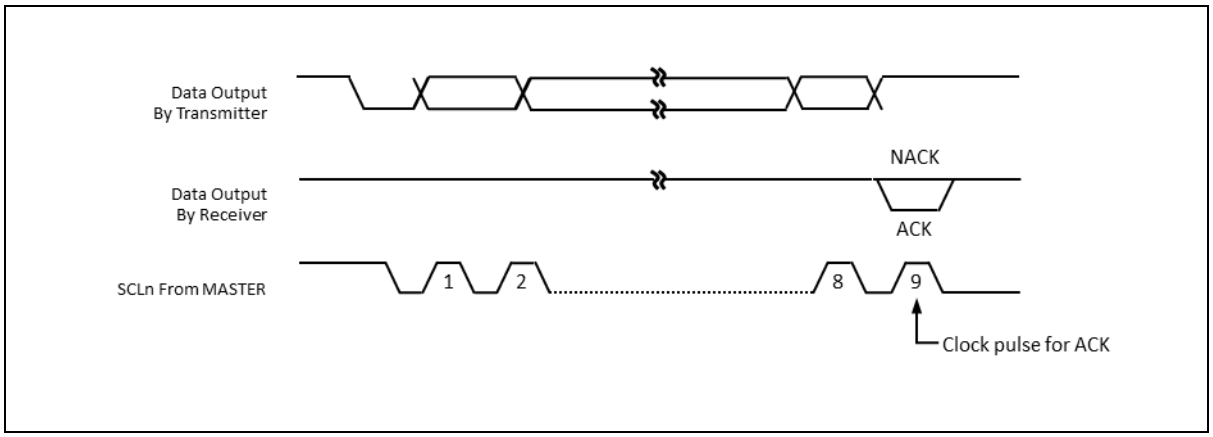


Figure 67. I2C Bus Response

13.3.5 Synchronization

All masters generate their own clock on the SCL line to transfer messages on the I2C-bus. Data is only valid during the "H" period of the clock. A defined clock is therefore needed for the bit-by-bit arbitration procedure to take place.

Clock synchronization is performed using the wired-AND connection of I2C interfaces to the SCL line. This means that an "H" to "L" transition on the SCL line will cause the devices concerned to start counting off their "L" period and, once a device clock has gone "L", it will hold the SCL line in that state until the clock "H" state is reached (see Figure 68).

However, the "L" to "H" transition of this clock may not change the state of the SCL line if another clock is still within its "L" by the device with the longest "L" period. Devices with shorter "L" periods enter an "H" wait-state during this time.

When all devices concerned have counted off their "L" period, the clock line will be released and go "H". There will then be no difference between the device clocks and the state of the SCL line, and the devices will start counting their "H" periods. The first device to complete its "H" period will again pull the SCL line "L".

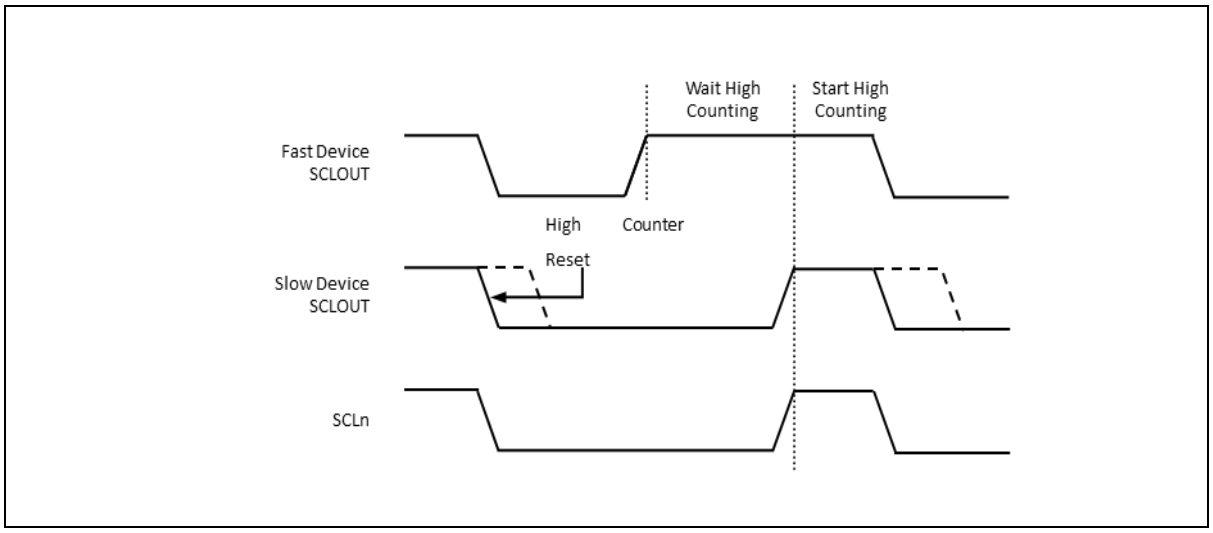


Figure 68. Clock Synchronization during Arbitration

13.3.6 Arbitration

A master may start a transfer only if the bus is free. Two or more masters may generate a START condition within the minimum hold time of the START condition which results in a defined START condition to the bus.

Arbitration takes place on the SDA line, while the SCL line is at the "H" level, in such a way that the master which transmits "H" level, while another master is transmitting "L" level will switch off its DATA output stage because the level on the bus doesn't correspond to its own level.

Arbitration can continue for many bits. Its first stage is comparison of the address bits. If the masters are each trying to address the same device, arbitration continues with comparison of the data-bits if they are master-transmitter or acknowledge-bits if they are master-receiver. Because address and data information on the I2C-bus is determined by the winning master, no information is lost during the arbitration process.

A master that loses the arbitration can generate clock pulses until the end of the byte in which it loses the arbitration.

If a master also incorporates a slave function and it loses arbitration during the addressing stage, it's possible that the winning master is trying to address it. The losing master must therefore switch over immediately to its slave mode.

Figure 69 shows the arbitration procedure for two masters. Of course, more may be involved (depending on how many masters are connected to the bus). The moment there is a difference between the internal data level of the master generating Device1 Data out and the actual level on the SDA line, its data output is switched off, which means that a "H" output level is then connected to the bus. This will not affect the data transfer initiated by the winning master.

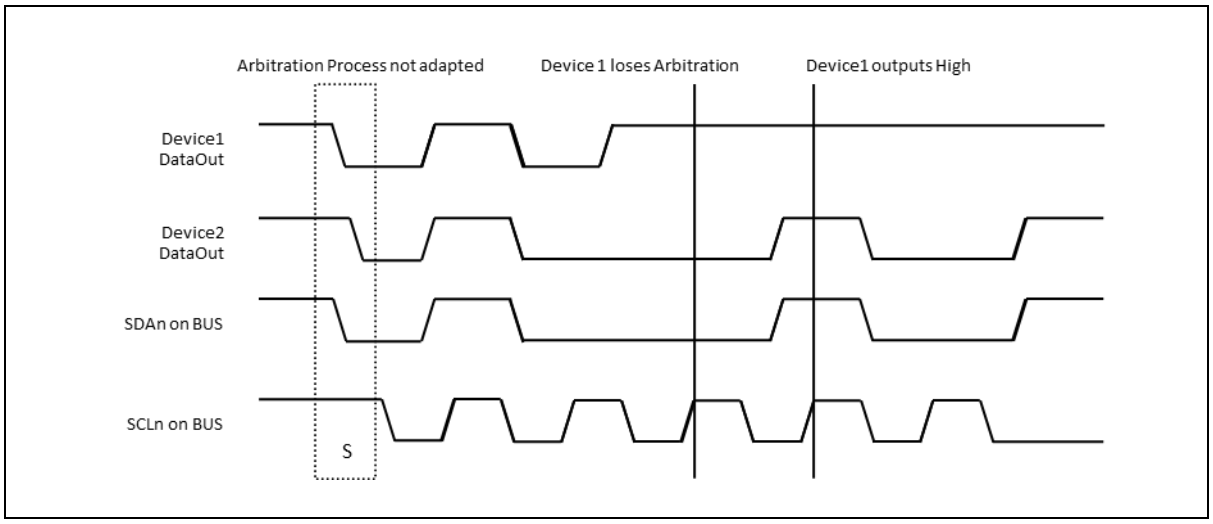


Figure 69. Arbitration Process between Two Masters

13.3.7 I2C operation

The I2C module uses interrupts. Once an interrupt is serviced, the CR register's 7th bit is flagged. The SR register shows I2C bus status information; the SCL line stays "L" until a certain value is written to the register. The status register can be cleared by being written to.

Master transmitter

Figure 70 shows a flowchart of the transmitter in master mode.

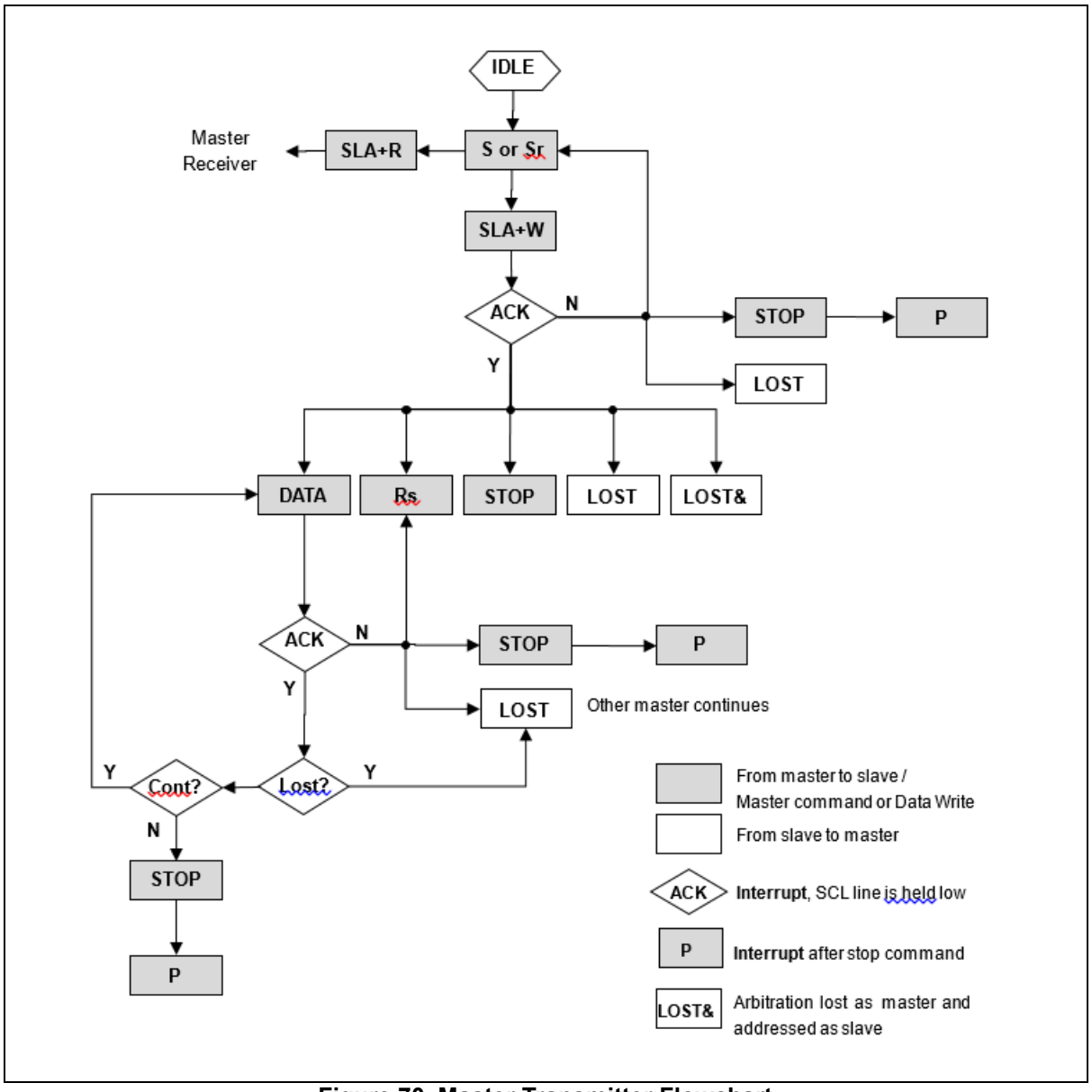


Figure 70. Master Transmitter Flowchart

13.3.8 Master receiver

Figure 71 shows a flowchart of the receiver in master mode.

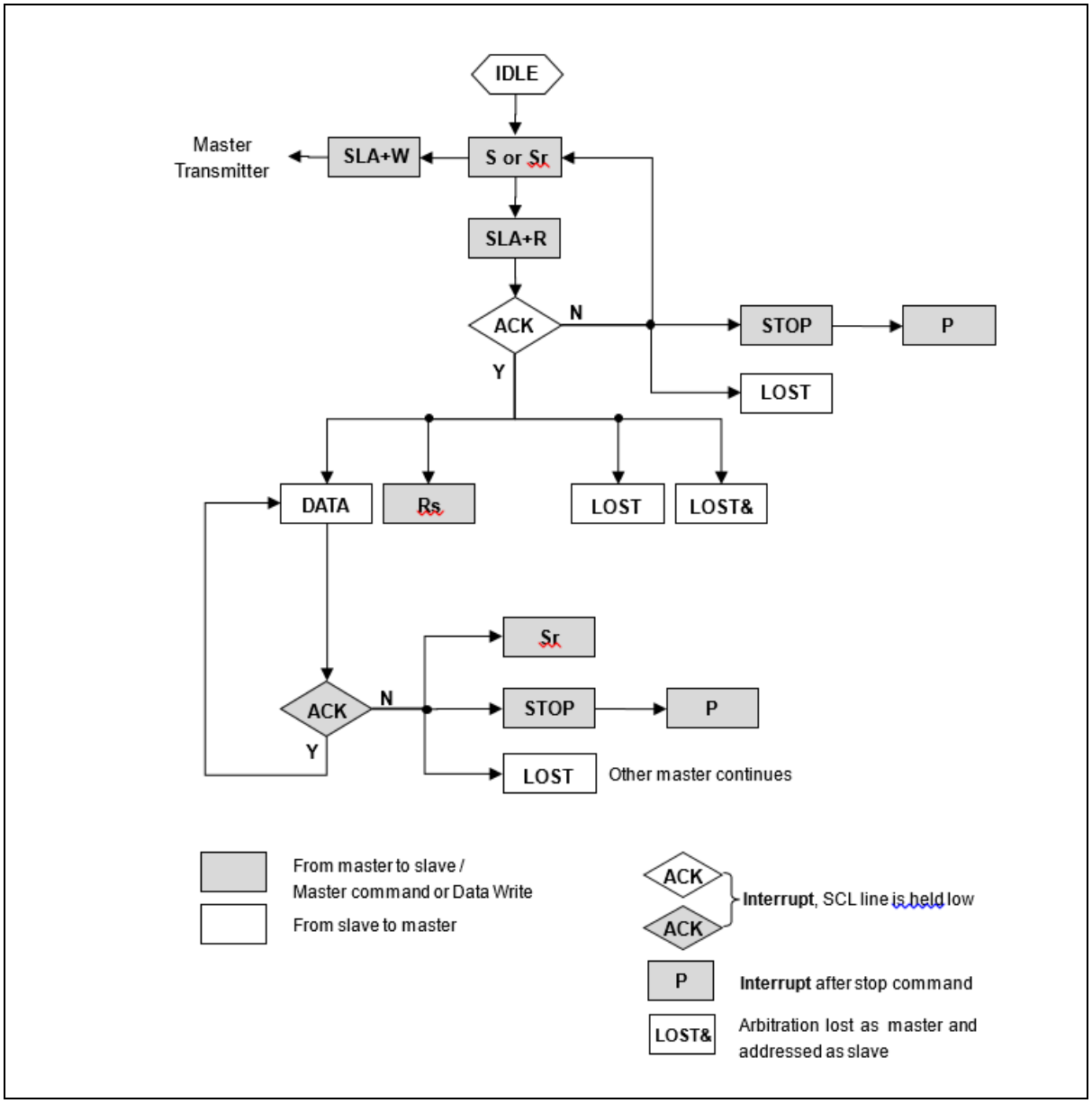


Figure 71. Master Receiver Flowchart

13.3.9 Slave transmitter

Figure 72 shows a flowchart of the transmitter in slave mode.

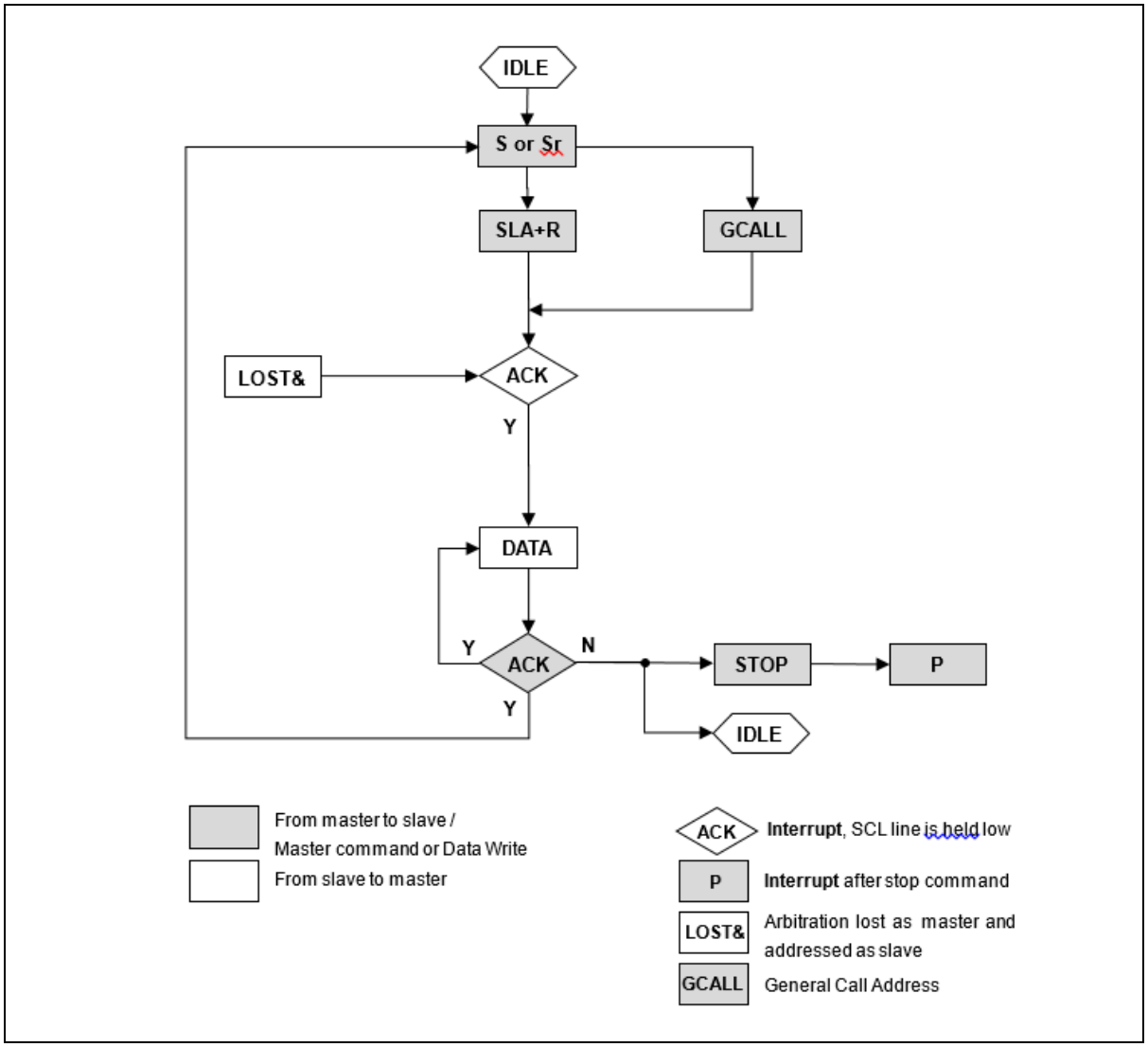


Figure 72. Slave Transmitter Flowchart

13.3.10 Slave receiver

Figure 73 shows a flowchart of the Receiver in slave mode.

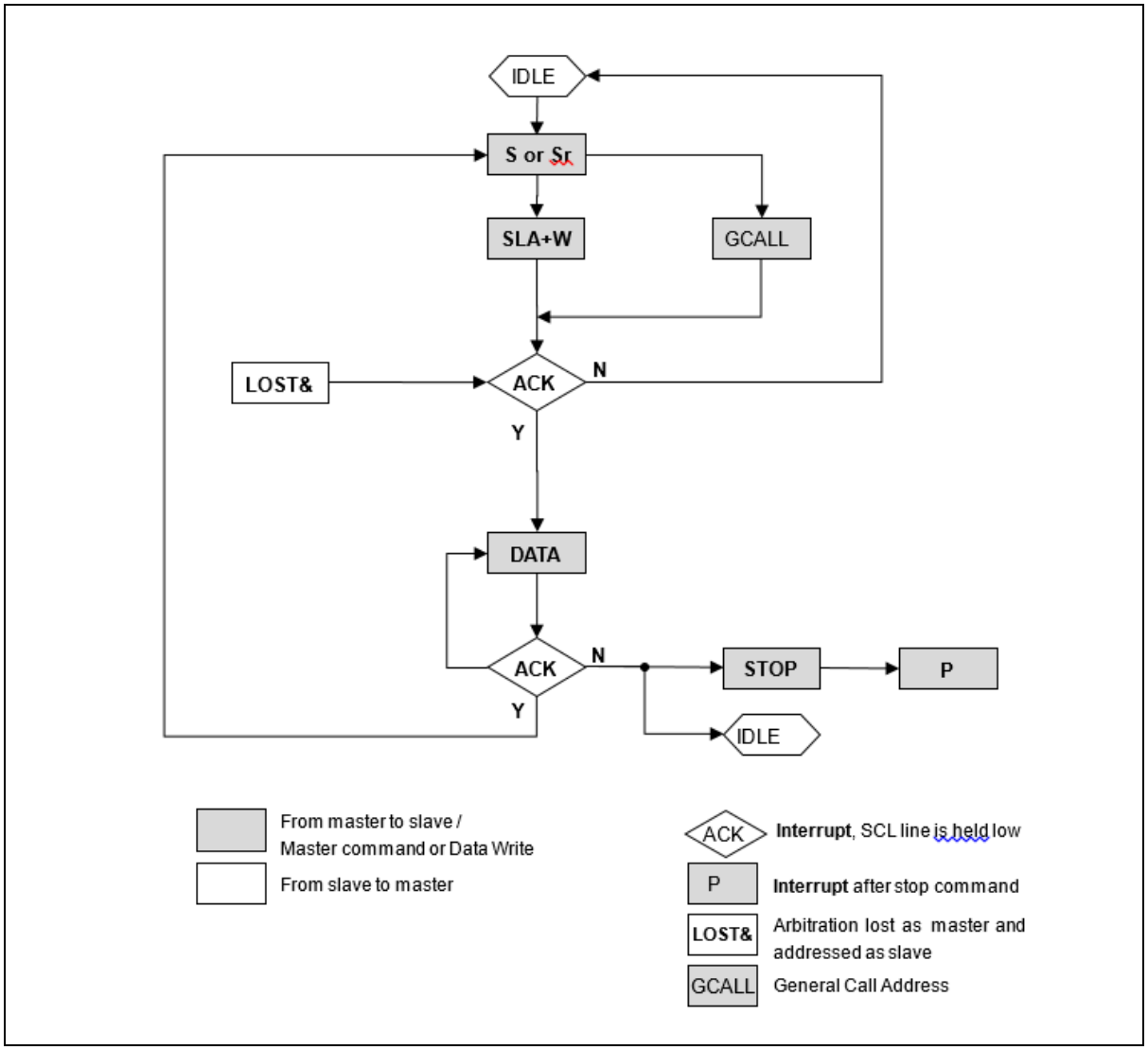


Figure 73. Slave Receiver Flowchart

14 Motor Pulse Width Modulation (MPWM)

The MPWM is Programmable Motor controller which is optimized for 3-phase AC and DC motor control application. It can be used in many other application that need timing, counting and comparison.

The MPWM includes 3 channels, each of which controls a pair of outputs that is turn can control a motor.

MPWM normal mode of AC33Mx064T series features the followings:

- 16-bit Counter
- 6-channel outputs for motor control
- Dead-time supports
- Protection event and over voltage event handling
- 6 ADC trigger outputs
- Interval interrupt mode (period interrupt only)
- Up-down count mode

The MPWM clock source which is MPWM counter clock source will be provided from SCU block. The MPWM resolution and period will be defined by this MPWM clock configuration. The default MPWM clock is same as RING OSC clock. Before enable MPWM module, the proper MPWM clock selection should be required.

Table 47 introduces pins assigned for MPWM.

Table 47. Pin Assignment of MPWM: External Pins

Pin name	Type	Description
MP0UH	O	MPWM 0 Phase-U H-side output
MP0UL	O	MPWM 0 Phase-U L-side output
MP0VH	O	MPWM 0 Phase-V H-side output
MP0VL	O	MPWM 0 Phase-V L-side output
MP0WH	O	MPWM 0 Phase-W H-side output
MP0WL	O	MPWM 0 Phase-W L-side output
PRTIN0	I	MPWM 0 Protection Input
OVIN0	I	MPWM 0 Over-voltage Input

14.1 MPWM block diagram

Figure 74 describes normal mode of MPWM in block diagram.

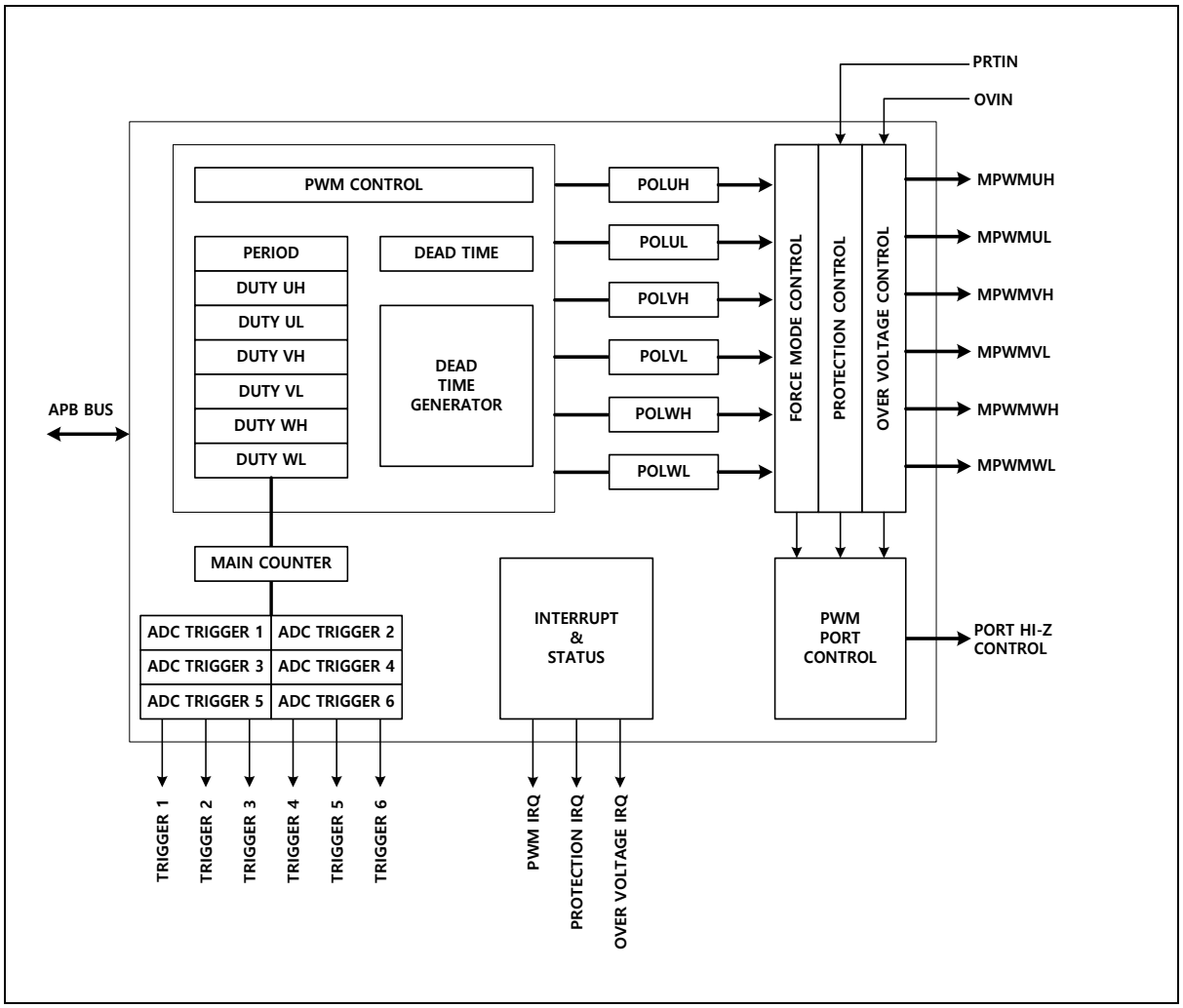


Figure 74. MPWM Block Diagram

14.2 Registers

Base address of MPWM is introduced in the followings:

Table 48. Base Address of MPWM Interface

Name	Base address
MPWM	0x4000_4000

Table 49. MPWM Register Map

Name	Offset	Type	Description	Reset value	Reference
MP0.MR	0x0000	RW	MPWM Mode register	0x0000_0000	14.2.1
MP0.OLR	0x0004	RW	MPWM Output Level register	0x0000_0000	14.2.2
MP0.FOLR	0x0008	RW	MPWM Force Output register	0x0000_0000	14.2.3
MP0.PRD	0x000C	RW	MPWM Period register	0x0000_0002	14.2.6
MP0.DUH	0x0010	RW	MPWM Duty UH register	0x0000_0001	0
MP0.DVH	0x0014	RW	MPWM Duty VH register	0x0000_0001	0
MP0.DWH	0x0018	RW	MPWM Duty WH register	0x0000_0001	14.2.9
MP0.DUL	0x001C	RW	MPWM Duty UL register	0x0000_0001	14.2.10
MP0.DVL	0x0020	RW	MPWM Duty VL register	0x0000_0001	14.2.11
MP0.DWL	0x0024	RW	MPWM Duty WL register	0x0000_0001	14.2.12
MP0.CR1	0x0028	RW	MPWM Control register 1	0x0000_0000	14.2.4
MP0.CR2	0x002C	RW	MPWM Control register 2	0x0000_0000	14.2.5
MP0.SR	0x0030	R	MPWM Status register	0x0000_0000	14.2.14
MP0.IER	0x0034	RW	MPWM Interrupt Enable	0x0000_0000	14.2.15
MP0.CNT	0x0038	R	MPWM counter register	0x0000_0001	14.2.16
MP0.DTR	0x003C	RW	MPWM dead time control	0x0000_0000	14.2.17
MP0.PCR0	0x0040	RW	MPWM protection 0 control register	0x0000_0000	14.2.18
MP0.PSR0	0x0044	RW	MPWM protection 0 status register	0x0000_0080	14.2.18
MP0.PCR1	0x0048	RW	MPWM protection 1 control register	0x0000_0000	14.2.18
MP0.PSR1	0x004C	RW	MPWM protection 1 status register	0x0000_0000	14.2.18
-	0x0054	-	Reserved	-	
MP0.ATR1	0x0058	RW	MPWM ADC Trigger reg1	0x0000_0000	14.2.19
MP0.ATR2	0x005C	RW	MPWM ADC Trigger reg2	0x0000_0000	14.2.19
MP0.ATR3	0x0060	RW	MPWM ADC Trigger reg3	0x0000_0000	14.2.19

Table 49. MPWM Register Map (continued)

Name	Offset	Type	Description	Reset value	Reference
MP0.ATR4	0x0064	RW	MPWM ADC Trigger reg4	0x0000_0000	14.2.19
MP0.ATR5	0x0068	RW	MPWM ADC Trigger reg5	0x0000_0000	14.2.19
MP0.ATR6	0x006C	RW	MPWM ADC Trigger reg6	0x0000_0000	14.2.19

14.2.1 MP0.MR MPWM Mode Register

PWM operation Mode register is 16-bit register.

MP0.MR=0x4000_4000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOTORB								UAO		TUP	BUP		MCHMOD		UPDOWN
0								0		0	0		00		0
RW								RW		RW	RW		RW		RW

15	MOTORB	0	Motor mode
		1	Normal mode
7	UAO	0	Update will be executed at designated timing.
		1	Update all duty, period register at once. When UPDATE set, Duty and Period registers are updated after two PWM clocks
5	TUP	0	Period, duty values are not updated at every period match.
		1	Period, duty values are updated at every period match.
4	BUP	0	Period, duty values are not updated at every bottom match
		1	Period, duty values are updated at every bottom match
2 1	MCHMOD	00	2 channels symmetric mode Duty H decides toggle high/low time of H-ch Duty L decides toggle high/low time of L-ch
		01	1 channel asymmetric mode Duty H decides toggle high time of H-ch Duty L decides toggle low time of H-ch L channel become the inversion of H channel
		10	1 channel symmetric mode Duty H decides toggle high/low time of H-ch L channel become the inversion of H channel
		11	Not valid (same with 00)
0	UPDOWN	0	PWM Up count mode (only available when MOTORB='1')
		1	PWM Up/Down count mode (This bit should be '1' if MOTORB='0')

After initial PWM period and duty setting is completed, the UAO bit should be set once for updating the setting value into internal operating registers. This action will help to transfer the setting data from user interface register to internal operating register. The UAO bit should be stay at set state at least 2-PWM clock period. Otherwise, the update command can be missed and internal registers will keep the previous data.

The MCHMOD in MP0.MR field is only effective when MOTORB in MP0.MR is clear "0". Otherwise the MCHMOD field value will be ignored internally and will keep "00" value.

The UPDOWN in MP0.MR field is only effective when MOTORB in MP0.MR is set "1". Otherwise the UPDOWN field value will be ignored internally and will keep "1" value. In the motor mode, the counter is always updown count operation.

14.2.2 MP0.OLR MPWM Output Level Register

PWM Port Mode register is 16-bit register. This register will control the active level of each PWM output port. The default active level is negated when the corresponding bit is set.

The normal level is defined in each operating mode.

MP0.OLR=0x4000_4004

7	6	5	4	3	2	1	0
		WHL	VHL	UHL	WLL	VLL	ULL
0	0	0	0	0	0	0	0
		RW	RW	RW	RW	RW	RW

5	WHL	0	Default Output Level
		1	Inversion Output Level
4	VHL	0	Default Output Level
		1	Inversion Output Level
3	UHL	0	Default Output Level
		1	Inversion Output Level
2	WLL	0	Default Output Level
		1	Inversion Output Level
1	VLL	0	Default Output Level
		1	Inversion Output Level
0	ULL	0	Default Output Level
		1	Inversion Output Level

The normal level is defined in each operating mode as below table.

Table 50. MPWM Register Map (MP.OLR = 0x00)

PWM Output	Level	NORMAL mode		MOTOR mode
		Up mode	Updown mode	
WH	Default level	LOW	HIGH	LOW
	Active level	HIGH	LOW	HIGH
WL	Default level	LOW	LOW	HIGH
	Active level	HIGH	HIGH	LOW
VH	Default level	LOW	HIGH	LOW
	Active level	HIGH	LOW	HIGH
VL	Default level	LOW	LOW	HIGH
	Active level	HIGH	HIGH	LOW
UH	Default level	LOW	HIGH	LOW
	Active level	HIGH	LOW	HIGH
UL	Default level	LOW	LOW	HIGH
	Active level	HIGH	HIGH	LOW

Polarity control block is consisted as below figure. This is for WH signal polarity control example.

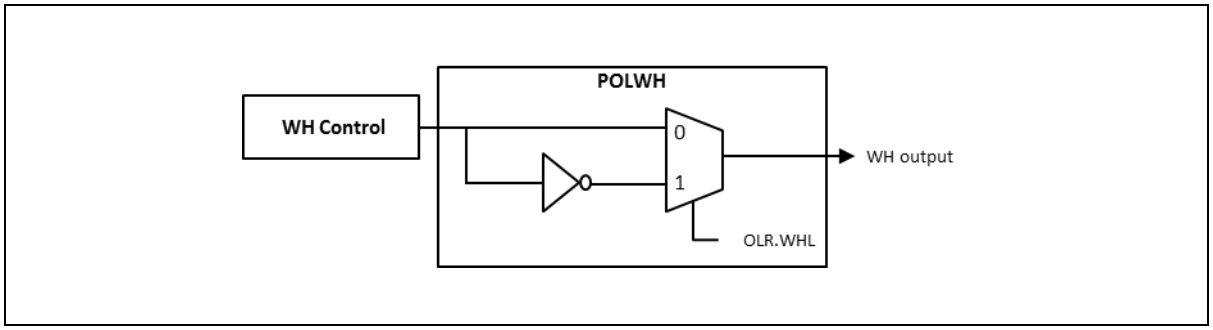


Figure 75. Polarity Control Block

14.2.3 MP0.FOLR MPWM Force Output Level Register

PWM force output register is 8-bit register. The PWM output level can be forced by an abnormal event from externally or user intended condition. When the forced condition is occurred, each PWM output level which is programmed in FOLR register will be forced.

MP0.FOLR=0x4000_4008

7	6	5	4	3	2	1	0
		WHFL	VHFL	UHFL	WLFL	VLFL	ULFL
0	0	0	0	0	0	0	0
		RW	RW	RW	RW	RW	RW

5	WHFL	Select WH Output Force Level
		0 Output Force Level is 'L'
		1 Output Force Level is 'H'
4	VHFL	Select VH Output Force Level
		0 Output Force Level is 'L'
		1 Output Force Level is 'H'
3	UHFL	Select UH Output Force Level
		0 Output Force Level is 'L'
		1 Output Force Level is 'H'
2	WLFL	Select WL Output Force Level
		0 Output Force Level is 'L'
		1 Output Force Level is 'H'
1	VLFL	Select VL Output Force Level
		0 Output Force Level is 'L'
		1 Output Force Level is 'H'
0	ULFL	Select UL Output Force Level
		0 Output Force Level is 'L'
		1 Output Force Level is 'H'

14.2.4 MP0.CR1 MPWM Control Register 1

PWM Control Register 1 is 16-bit register.

MP0.CR1=0x4000_4028

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					IRQN										PWMEN
						000			0	0	0	0	0	0	0
						RW									RW

10 8	IRQN	IRQ interval number (Every 1~8th PRDIRQ,BOTIRQ,ATRn)
0	PWMEN	PWM enable When this bit set 0, the PWM block stay in reset state but user interface can be accessed. To operate the PWM block, this bit should be set 1.

Basically, PRDIRQ and BOTIRQ are generated every period. But the interrupt interval can be controlled from 0 to 8 periods. When IRQN.CR1 = 0, the interrupt is requested every period, otherwise the interrupt is requested every (IRQN+1) times of period.

14.2.5 MP0.CR2 MPWM Control Register 2

PWM Control Register 2 is 8-bit register.

MP0.CR2=0x4000_402C

7	6	5	4	3	2	1	0
HALT							PSTART
0	0	0	0	0	0	0	0
RW							RW

7	HALT	PWM HALT (PWM counter stop but not reset) PWM outputs keep previous state	
0	PSTART	0	PWM counter stop and clear
		1	PWM counter start (will be resynced @PWM clock twice) PWMEN should be "1" to start PWM counter

14.2.6 MP0.PRD MPWM Period Register

PWM Period Register is 16-bit register.

MP0.PRD=0x4000_400C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERIOD															
0x0002															
RW															

15 0	PERIOD	16-bit PWM period. It should be larger than 0x0010

14.2.7 MP0.DUH MPWM Duty UH Register

PWM U channel duty register is 16-bit register.

MP0.DUH=0x4000_4010

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DUTY UH															
0x0001															
RW															

15 0	DUTY UH	16-bit PWM Duty for UH output.

14.2.8 MP0.DVH MPWM Duty VH Register

PWM V channel duty register is 16-bit register.

MP0.DVH=0x4000_4014

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DUTY VH															
0x0001															
RW															

15 0	DUTY VH	16-bit PWM Duty for VH output.

14.2.9 MP0.DWH MPWM Duty WH Register

PWM W channel duty register is 16-bit register.

MP0.DWH=0x4000_4018

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DUTY WH															
0x0001															
RW															

15 0	DUTY WH	16-bit PWM Duty for WH output.

14.2.10 MP0.DUL MPWM Duty UL Register

PWM U channel duty register is 16-bit register.

MP0.DUL=0x4000_401C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DUTY UL															
0x0001															
RW															

15 0	DUTY UL	16-bit PWM Duty for UL output.

14.2.11 MP0.DVL MPWM Duty VL Register

PWM V channel duty register is 16-bit register.

MP0.DVL=0x4000_4020

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DUTY VL															
0x0001															
RW															

15 0	DUTY VL	16-bit PWM Duty for VL output.

14.2.12 MP0.DWL MPWM Duty WL Register

PWM W channel duty register is 16-bit register.

MP0.DWL=0x4000_4024

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DUTY WL															
0x0001															
RW															

15 0	DUTY WL	16-bit PWM Duty for WL output.

14.2.13 MP0.IER MPWM Interrupt Enable Register

PWM Interrupt Enable Register is 8-bit register.

MP0.IER=0x4000_4034

7	6	5	4	3	2	1	0
PRDIEN	BOTIEN	WHIE	VHIE	UHIE	WLIE	VLIE	ULIE
0	0	0	0	0	0	0	0
RW	RW	RW	RW	RW	RW	RW	RW

7	PRDIEN	PWM Counter Period Interrupt enable 0 interrupt disable 1 interrupt enable
6	BOTIEN	PWM Counter Bottom Interrupt enable 0 interrupt disable 1 interrupt enable
5	WHIE ATR6IE	WH Duty or ATR6 Match Interrupt enable 0 interrupt disable 1 interrupt enable
4	VHIE ATR5IE	VH Duty or ATR5 Match Interrupt enable 0 interrupt disable 1 interrupt enable
3	UHIE ATR4IE	UH Duty or ATR4 Match Interrupt enable 0 interrupt disable 1 interrupt enable
2	WLIE ATR3IE	WL Duty or ATR3 Match Interrupt enable 0 interrupt disable 1 interrupt enable
1	VLIE ATR2IE	VL Duty or ATR2 Match Interrupt enable 0 interrupt disable 1 interrupt enable
0	ULIE ATR1IE	UL Duty or ATR1 Match Interrupt enable 0 interrupt disable 1 interrupt enable

MP0.IER[5:0] control bits are shared by duty match interrupt event and adc trigger match interrupt event. When ADC trigger mode is disabled, the interrupt is generated by duty match condition.

In other case, the interrupt is generated by ADC trigger counter match condition. The ADC trigger mode is selected by ATMOD bit field in ATRm register.

14.2.14 MP0.SRMPWM Status Register

PWM Status Register is 16-bit register.

MP0.SR=0x4000_4030

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOWN	IRQCNT							PRDIF	BOTIF	DWHIF ATR6F	DVHIF ATR5F	DUHIF ATR4F	DWLIF ATR3F	DVLIF ATR2F	DULIF ATR1F
0	000			0	0	0	0	0	0	0	0	0	0	0	0
RO	RO							RW	RW	RW	RW	RW	RW	RW	RW

15	DOWN	0	Current PWM Count mode is Up
		1	Current PWM Count mode is Down
14 12	IRQCNT[2:0]		Interrupt count number of period match (Interval PRDIRQ mode)
7	PRDIF		PWM Period Interrupt flag(write "1" to clear flag)
		0	No interrupt occurred
		1	Interrupt occurred
6	BOTIF		PWM Bottom Interrupt flag(write "1" to clear flag)
		0	No interrupt occurred
		1	Interrupt occurred
5	DWHIF ATR6F		PWM duty WH interrupt flag(write "1" to clear flag) (Duty interrupt is enabled if ATR6 was disabled)
		0	No interrupt occurred
		1	Interrupt occurred
4	DVHIF ATR5F		PWM duty VH interrupt flag(write "1" to clear flag) (Duty interrupt is enabled if ATR5 was disabled)
		0	No interrupt occurred
		1	Interrupt occurred
3	DUHIF ATR4F		PWM duty UH interrupt flag(write "1" to clear flag) (Duty interrupt is enabled if ATR4 was disabled)
		0	No interrupt occurred
		1	Interrupt occurred
2	DWLIF ATR3F		PWM duty WL interrupt flag(write "1" to clear flag) (Duty interrupt is enabled if ATR3 was disabled)
		0	No interrupt occurred
		1	Interrupt occurred
1	DVLIF ATR2F		PWM duty VL interrupt flag(write "1" to clear flag) (Duty interrupt is enabled if ATR2 was disabled)
		0	No interrupt occurred
		1	Interrupt occurred
0	DULIF ATR1F		PWM duty UL interrupt flag(write "1" to clear flag) (Duty interrupt is enabled if ATR1 was disabled)
		0	No interrupt occurred
		1	Interrupt occurred

MP0.SR[5:0] status bits are shared by duty match interrupt event and adc trigger match interrupt event. When ADC trigger mode is disabled, the interrupt is generated by duty match condition.

In other case, the interrupt is generated by ADC trigger counter match condition. The ADC trigger mode is selected by ATMOD bit field in ATRm register.

14.2.15 MP0.CNT MPWM Counter Register

PWM Counter Register is 16-bit Read-Only register.

MP0.CNT=0x4000_4038

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MPnCNT															
0x0000															
RW															

	MPnCNT	PWM Counter Value

14.2.16 MP0.DTR MPWM Dead Time Register

PWM Dead Time Register is 16-bit register.

MP0.DTR=0x4000_403C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DTEN	PSHRT						DTCLK	DT							
0	0	0	0	0	0	0	0	0x00							
RW							RW	RW							

15	DTEN	Dead-time function enable 2-channel symmetric mode does not support dead time function. It should be disabled in 2 channel symmetric mode.
		0 Disable Dead-time function 1 Enable Dead-time function
14	PSHRT	Protect short condition This function is effective only for 2 channel symmetric mode. For 1 channel mode, never activated on both H-side and L-side at same time. L-side is always opposite of H-side.
		0 Enable output short protection function. (Turn off both output when both H-side and L-side are active.) 1 Disable output short protection function.
8	DTCLK	Dead-time prescaler
		0 Dead time counter uses PWM CLK/4 1 Dead time counter uses PWM CLK/16
7	DT	Dead Time value (Dead time setting makes output delay of 'low to high transition' in normal polarity)
0		0x01 ~0xFF : Dead time

Protect short condition is for only internal pwm level not for external pwm level.

When internal signal of H-side and L-side are same high level, the protection short function is work to force both of H-side and L-side to low level.

14.2.17 MP0.PCRn MPWM Protection 0, 1 Control Register

PWM Protection Control Register is 16-bit register.

MP0.PCR0=0x4000_4040,MP0.PCR1=0x4000_4048

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PROTEN	PROTPOL				PROTD			PROTIE		WHPROTM	VHPROTM	UHPROTM	WLPROTM	VLPROTM	ULPROTM
0	0					000		0		0	0	0	0	0	0
RW	RW					RW		RW		RW	RW	RW	RW	RW	RW

15	PROT0EN	Enable Protection Input 0
14	PROT0POL	Select Protection Input Polarity 0: Low-Active 1: High-Active
10 8	PROTD	Protection Input debounce 0 – no debounce 1~7 – debounce by (MPWMCLK * PROTD[2:0])
7	PROTIE	Protection Interrupt enable 0 Disable protection interrupt 1 Enable protection interrupt
5	WHPROTM	Activate W-phase H-side protection output 0 Disable Protection Output 1 Enable Protection Output with FOR value
4	VHPROTM	Activate V-phase H-side protection output 0 Disable Protection Output 1 Enable Protection Output with FOR value
3	UHPROTM	Activate U-phase H-side protection output 0 Disable Protection Output 1 Enable Protection Output with FOR value
2	WLPROTM	Activate W-phase L-side protection output 0 Disable Protection Output 1 Enable Protection Output with FOR value
1	VLPROTM	Activate V-phase L-side protection output 0 Disable Protection Output 1 Enable Protection Output with FOR value
0	ULPROTM	Activate U-phase L-side protection output 0 Disable Protection Output 1 Enable Protection Output with FOR value
NOTE: MP.PCR0 is related to PRTIN pin and MP.PCR1 is related on OVIN.		

14.2.18 MP0.PSRn MPWM Protection 0, 1 Status Register

PWM Protection Status Register is 16-bit register. This register indicates which outputs are disabled. And User can set the output masks manually. Without writing PROTKEY when writing any value, the written values are ignored.

MP0.PSR0=0x4000_4044,MP0.PSR1=0x4000_404C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PROTKEY								PROTIF		WHPROTF	VHPROTF	UHPROTF	WLPROTF	VLPROTF	ULPROTF
-								0		0	0	0	0	0	0
WO								RC		RW	RW	RW	RW	RW	RW

15 8	PROTKEY	Protection Clear Access Key To clear flags, write the key with protection flag (PSR0 key is 0xCA and PSR1 key is 0xAC) Writing without PROTKEY prohibited.
7	PROTIF	Protection Interrupt status 0 No Protection Interrupt 1 Protection Interrupt occurred
5	WHPROT	Activate W-phase H-side protection flag 0 Protection not occurred. 1 Protection occurred or protection output enabled
4	VHPROT	Activate V-phase H-side protection flag 0 Protection not occurred. 1 Protection occurred or protection output enabled
3	UHPROT	Activate U-phase H-side protection flag 0 Protection not occurred. 1 Protection occurred or protection output enabled
2	WLPROT	Activate W-phase L-side protection flag 0 Protection not occurred. 1 Protection occurred or protection output enabled
1	VLPROT	Activate V-phase L-side protection flag 0 Protection not occurred. 1 Protection occurred or protection output enabled
0	ULPROT	Activate U-phase L-side protection flag 0 Protection not occurred. 1 Protection occurred or protection output enabled
NOTE: MP.PCR0 is related to PRTIN pin and MP.PCR1 is related on OVIN.		

If PROTEN bit in MP0.PCRn register is enabled, on any asserting signal on the external protection pins, the PWM output will be prohibited with output values are defined in MP0.FOLR register.

In addition, a user can prohibit the output manually by writing the designated value into the MP0.PSRn register.

14.2.19 MP0.ATRn MPWMn ADC Trigger Counter n Register (n = 1 to 6)

MP0.ATRn is a 32-bit size PWM ADC Trigger Counter Register.

MP0.ATR1=0x4000_4058
MP0.ATR2=0x4000_405C
MP0.ATR3=0x4000_4060
MP0.ATR4=0x4000_4064
MP0.ATR5=0x4000_4068
MP0.ATR6=0x4000_406C

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
												ATUDT		ATMOD		ATCNT																	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0x0000															
												RW		RW		RW																	

19	ATUDT	Trigger register update mode
		0 ADC trigger value applied at period match event (at the same time with period and duty registers update)
		1 Trigger register update mode When this bit set, written Trigger register values are sent to trigger compare block after two PWM clocks (through synchronization logic)
17	ATMOD	ADC trigger Mode register
16		00 ADC trigger Disable
		01 Trigger out when up count match
		10 Trigger out when down count match
		00 Trigger out when up-down count match
15	ATCNT	ADC Trigger counter
0		(it should be less than PWM period)

14.3 Functional Description

The MPWM includes 3 channels, each of which controls a pair of outputs that in turn can control something off-chip component. In normal pwm mode, each channel is running independently. 6 PWM output can be generated.

Each pwm output is build up with various setting. The picture shows the diagram for generating pwm output signal.

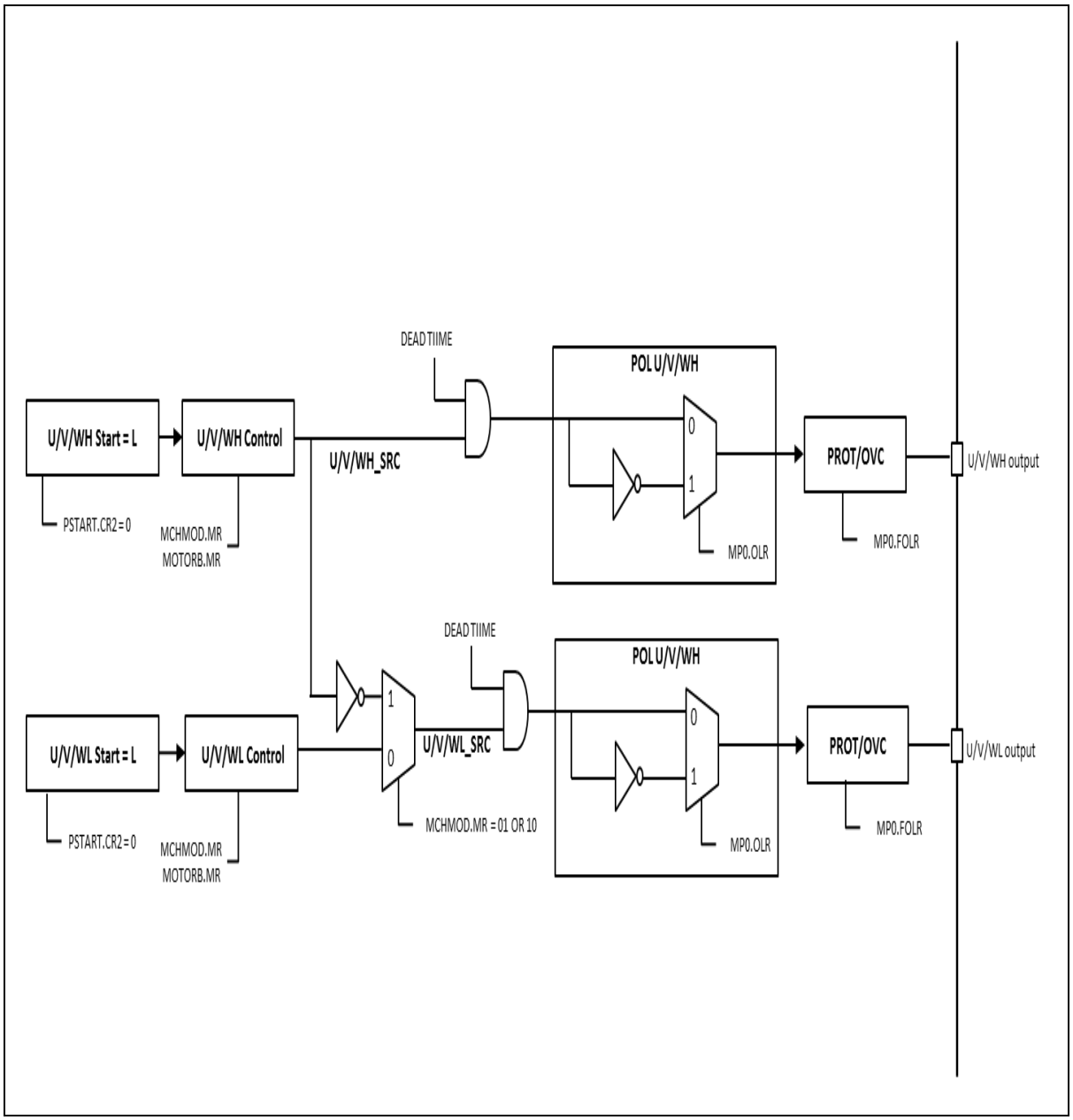


Figure 76. PWM Output Generation Chain

14.3.1 Normal PWM Up count mode timing

In normal pwm mode, each channel is running independently. 6 PWM output can be generated. The example waveform is below figure. Before PSTART is activated, the PWM output will stay default value L. When PSTART is enabled, the period counter starts up count until MP.PRD count value. First period, the MPWM does not generate PWM pulse.

The PWM pulse will generated from 2nd period. The active level is driven at start of the counter value during duty value time.

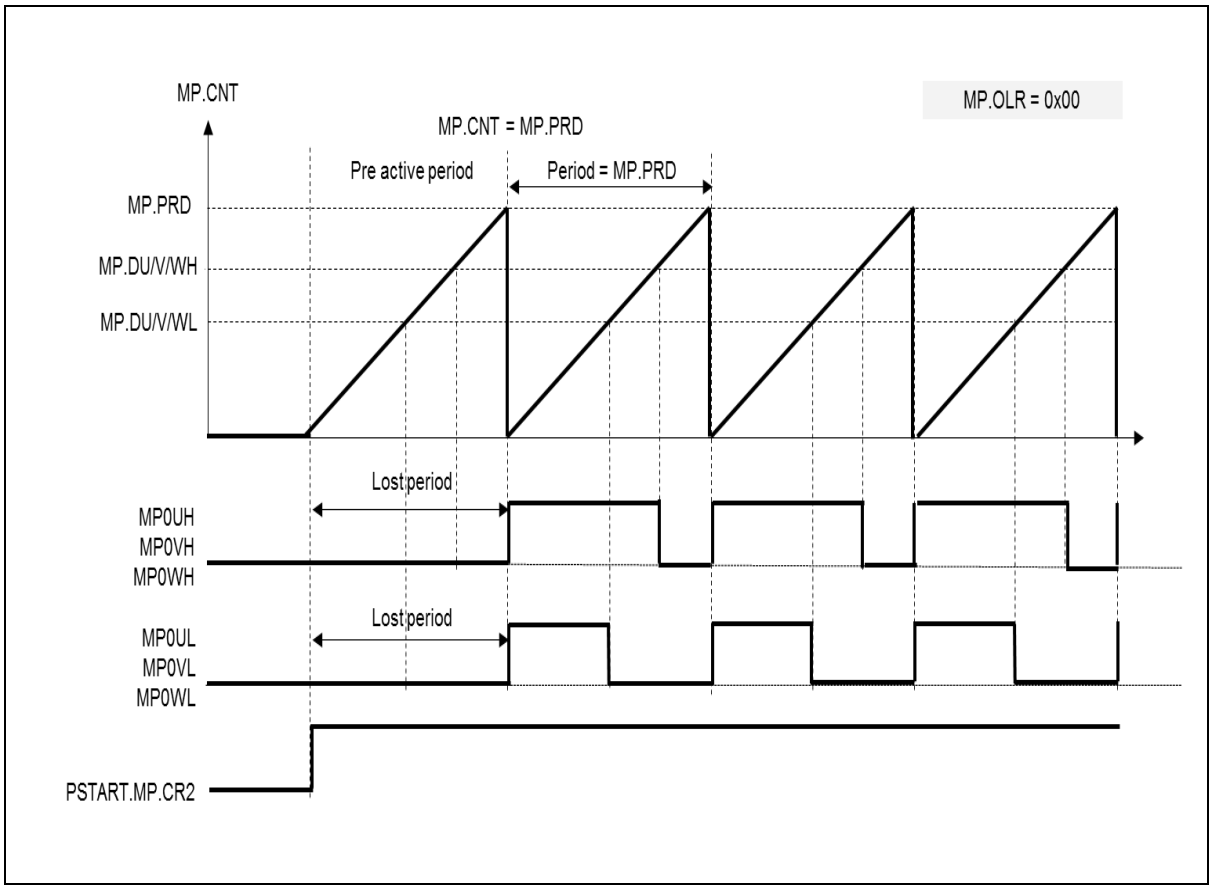


Figure 77. Up Count Mode Wave Form (MOTORB=1, UPDOWN=0)

14.3.2 Normal PWM Up/Down count mode timing

The basic operation of UP/DOWN count mode is same as UP count mode except the one period is twice than UP count mode. Default active level is opposite in a pair pwm output. This output polarity can be controlled by MP.OLR register.

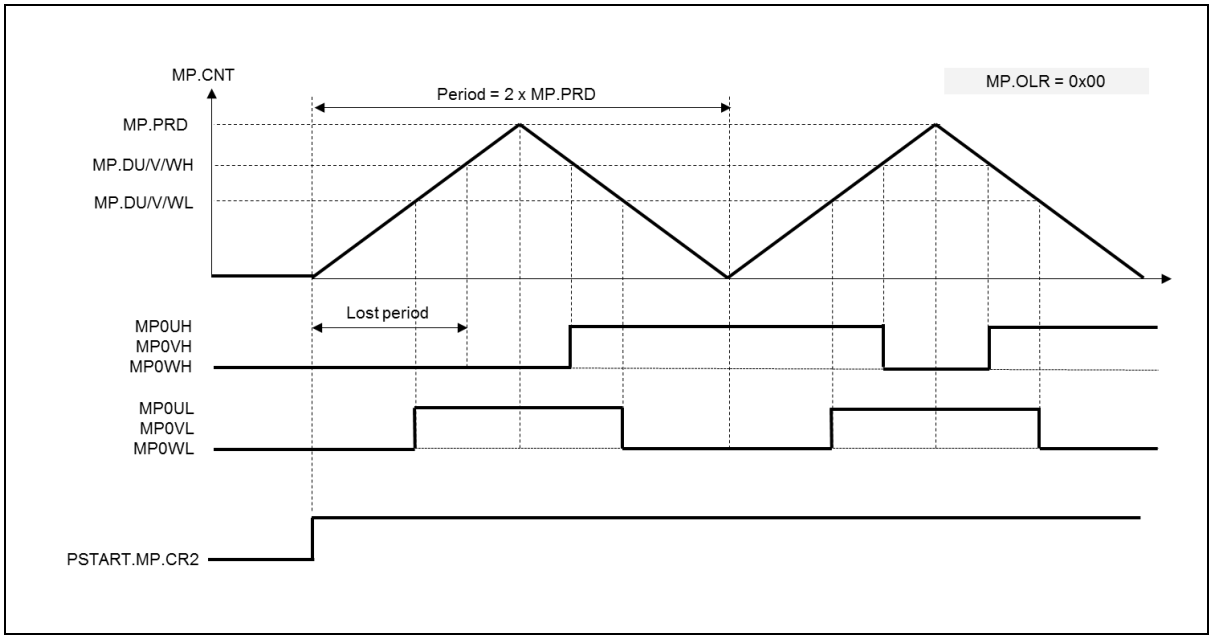


Figure 78. Up/Down Count Mode Wave Form (MOTORB=0, MCHMOD=0, UPDOWN=1)

14.3.3 Motor PWM 2-chennel symmetric mode timing

The motor pwm operation has 3 kind of operating mode. 2-Channel Symmetric mode, 1-Channel Symmetric mode and 1-Channel Asymmetric mode.

Figure 79 shows the 2 channel symmetric mode waveform.

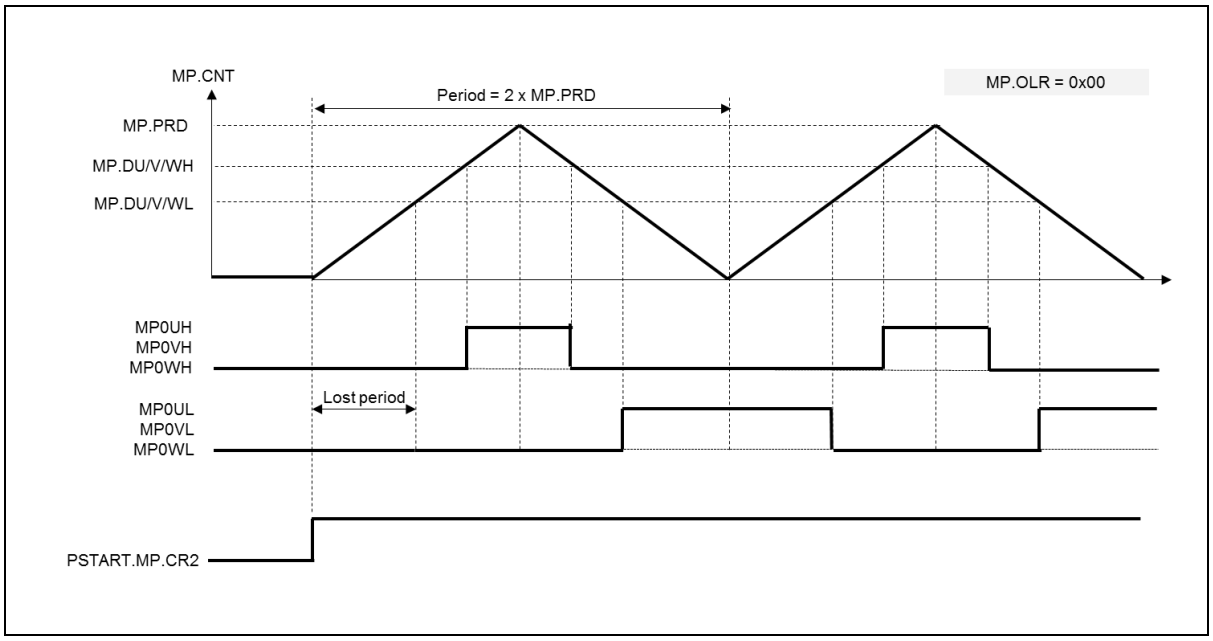


Figure 79. 2-Channel Symmetric Mode Wave Form (MOTORB=0, MCHMOD=00)

The default start level of both H-side and L-side is low. For the H-side, pwm output level is changed to active level when the duty level is matched in up count period and is returned to default level when the duty level is matched in down count period.

The symmetrical feature is appeared in each channel which is controlled by corresponding DUTY register value.

14.3.4 Motor PWM 1-chennel asymmetric mode timing

The 1 channel asymmetric mode makes asymmetric duration pulses which are defined by H-side and L-side DUTY register. So L-side signal is always negate signal of H-side. During up count period, the H-side DUTY register matching condition makes the active level pulse and during down count period, the L-side DUTY register matching condition makes the default level pulse.

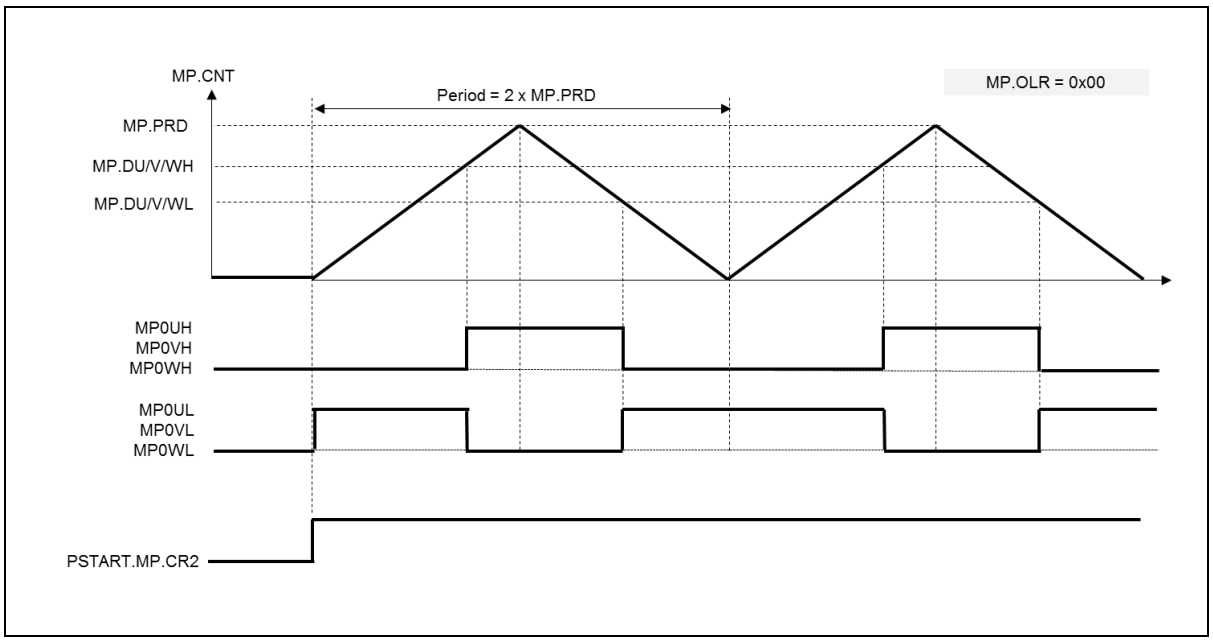


Figure 80. 1-Channel Asymmetric Mode Wave Form (MOTORB=0, MCHMOD=01)

The default start level of both H-side and L-side is low. For the H-side, pwm output level is changed to active level when the H-side duty level is matched in up count period and is returned to default level when the L-side duty level is matched in down count period.

When the PSTART is set, the L-side pwm output is changed to the active level then the L-side pwm output is inverse output of H-side output.

14.3.5 Motor PWM 1-chennel symmetric mode timing

The 1-channel symmetric mode makes symmetric duration pulse which are defined by H-side DUTY register. So L-side signal is always negate signal of H-side. During up count period, the H-side DUTY register matching condition makes the active level pulse and during down count period, the H-side DUTY register matching condition also makes the default level pulse.

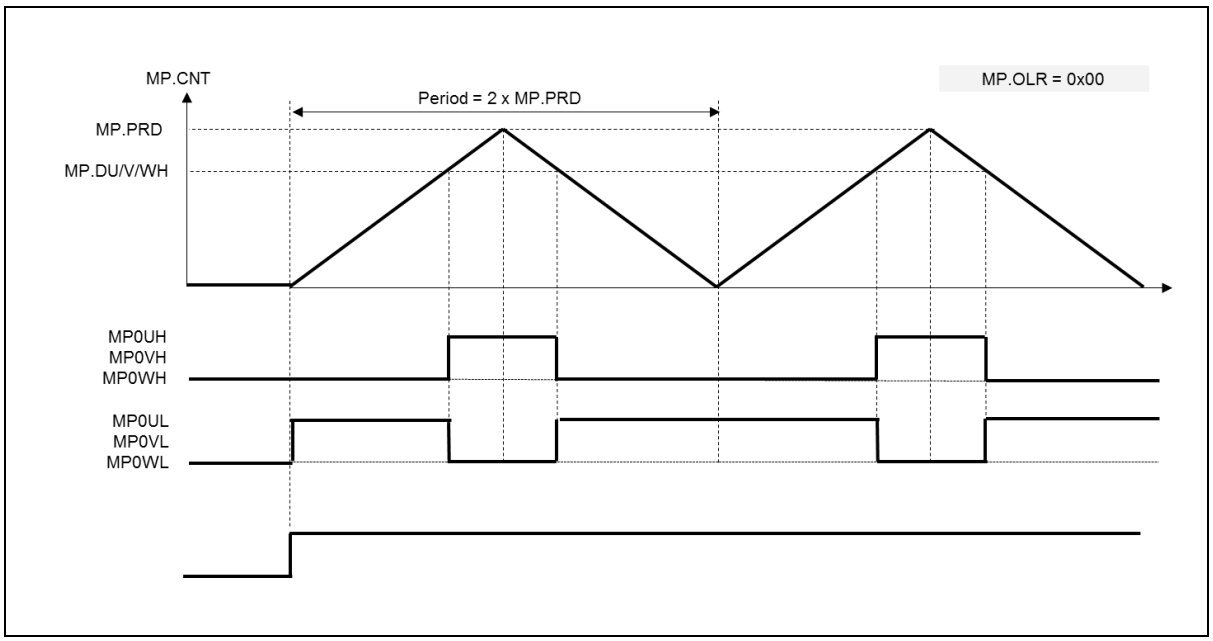


Figure 81. 1-Channel Symmetric Mode Wave Form (MOTORB=0, MCHMOD=10)

The default start level of both H-side and L-side is low. For the H-side, pwm output level is changed to active level when the H-side duty level is matched in up count period and is returned to default level when the H-side duty level is matched again in down count period.

When the PSTART is set, the L-side pwm output is changed to the active level then the L-side pwm output is inverse output of H-side output.

14.3.6 PWM dead-time operation

To prevent external short condition, the MPWM provide dead time function. This function is only available for motor pwm mode. When one of H-side or L-side output changes to active level, amount of dead time will be inserted if DTEN.MP.DTR bit is enabled.

The duration of dead time is decided a value in DT.MP.DTR[7:0] field as introduced below:

- When DTCLK = 0, the dead time duration = DT[7:0] * (PWM clock period * 4)
- When DTCLK = 1, the dead time duration = DT[7:0] * (PWM clock period * 16)

The pwm counter reached at duty value, the pwm output is masked and dead time counter starts to run. When dead time counter reached the value in DT[7:0] register, the output mask is disabled.

Figure 82 shows an example of dead time operation in 1 channel symmetric mode.

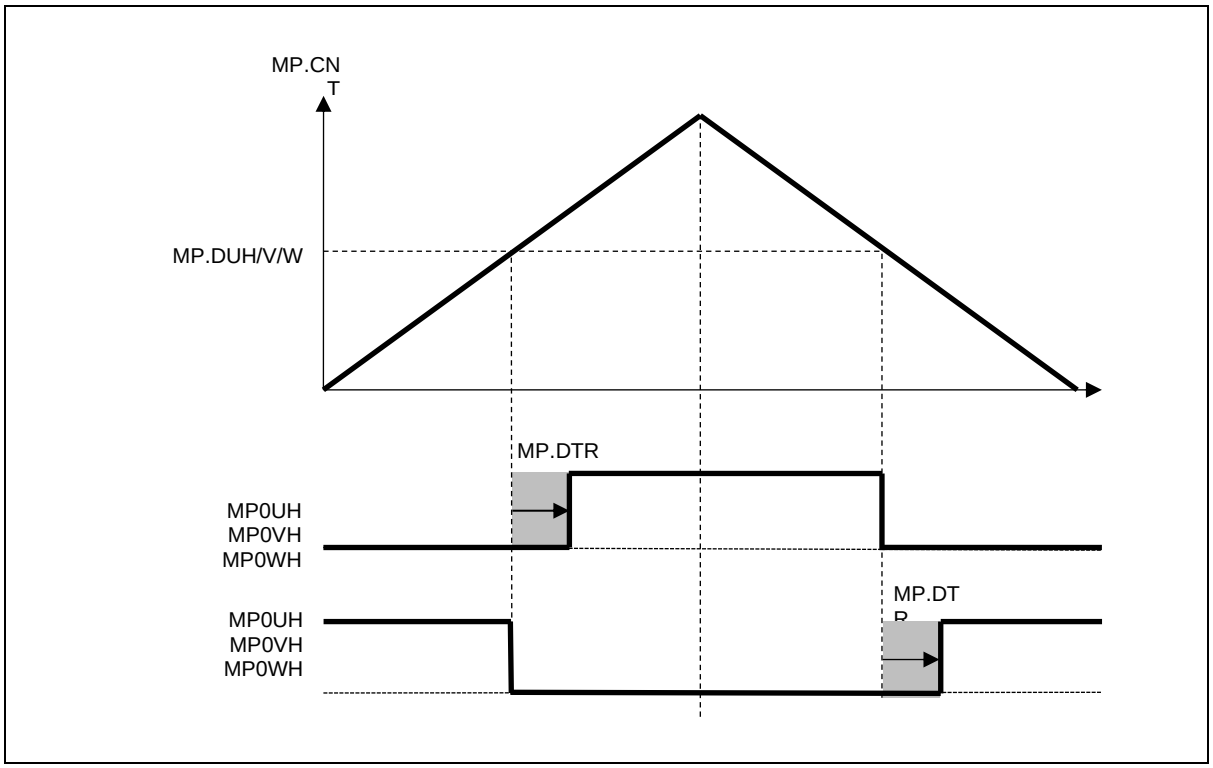


Figure 82. PWM Dead-Time Operation Timing Diagram (Symmetric mode)

Figure 83 shows a case of 1-channel asymmetric mode operation.

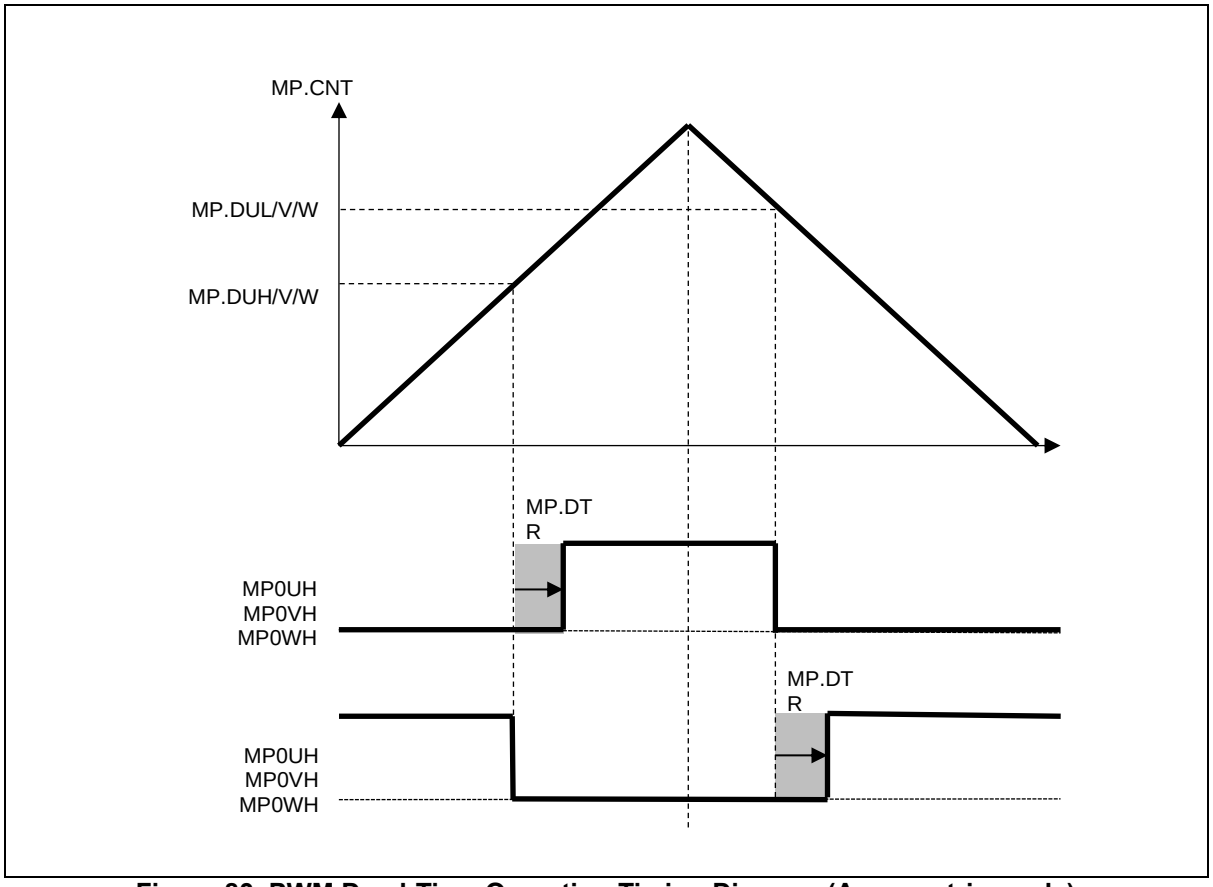


Figure 83. PWM Dead-Time Operation Timing Diagram (Asymmetric mode)

In case of 2-channel symmetric mode, the dead time function is no available. So the dead condition is generated by each channel duty control.

14.3.7 MPWM dead-time timing examples in special case

The following pictures show how the dead-time operate:

Norma dead time case is explained. The dead time masking is activated at duty match time and the dead time counter runs. When the dead time counter is reached to dead time value, the mask is disabled.

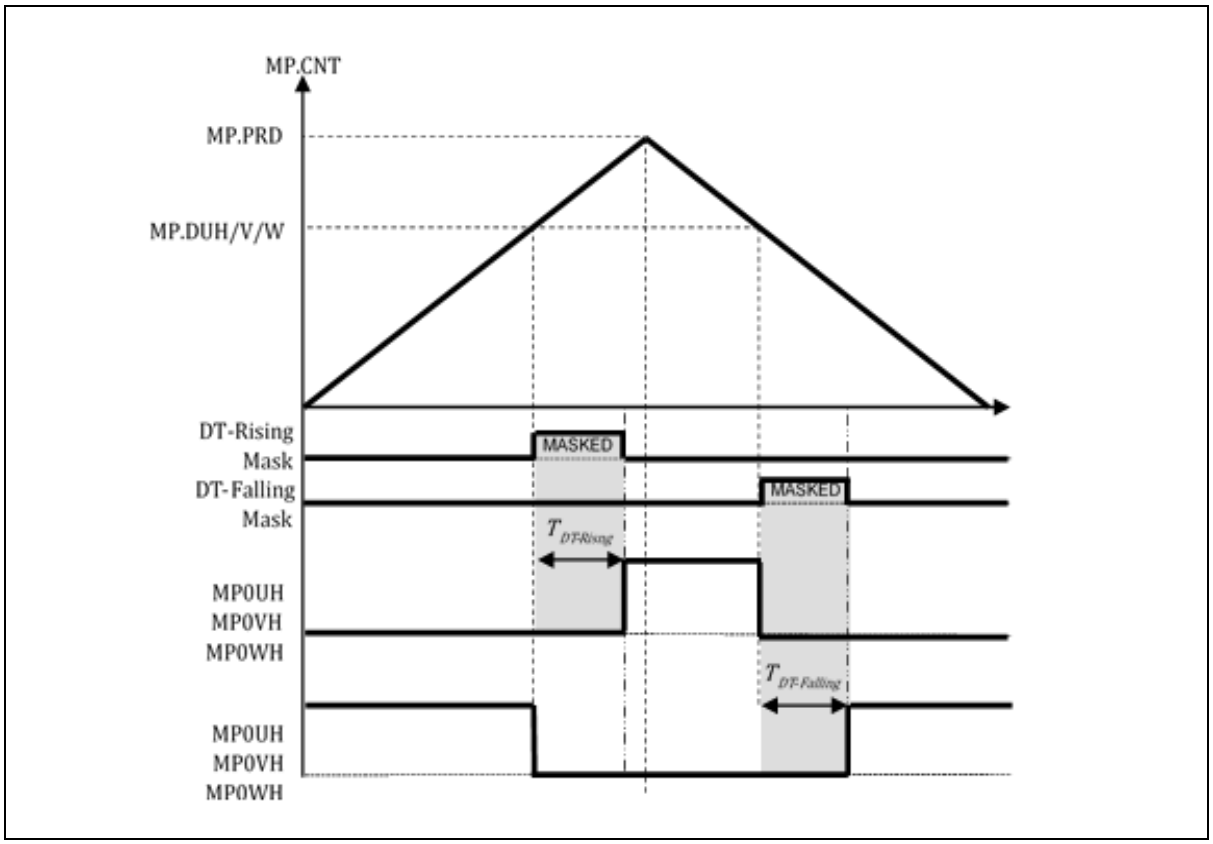


Figure 84. Normal Dead-Time Operation (TDUTY>TDT)

A couple of figures in below show special case of dead time configurations.

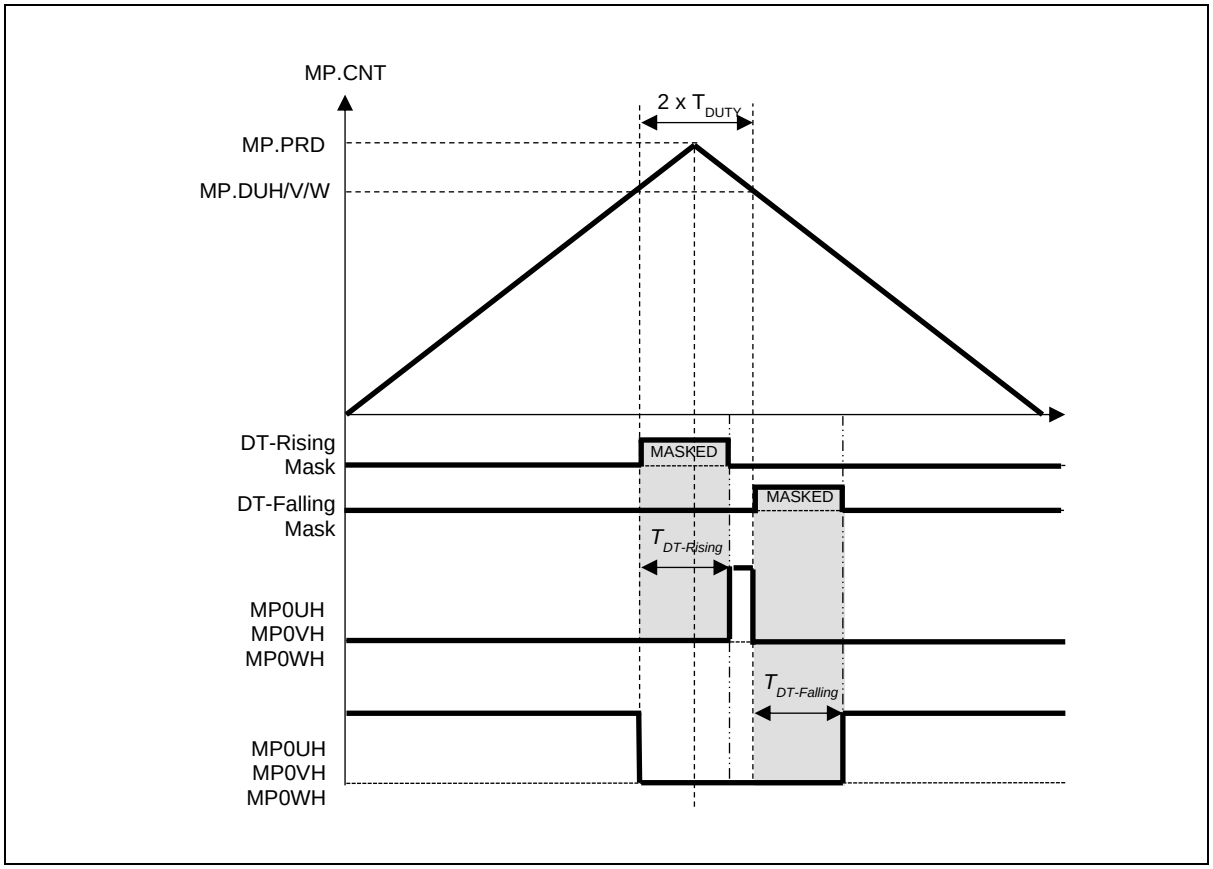


Figure 85. Minimum H-Side Pulse Timing (T_{DUTY}<T_{DT}<2xT_{DUTY})

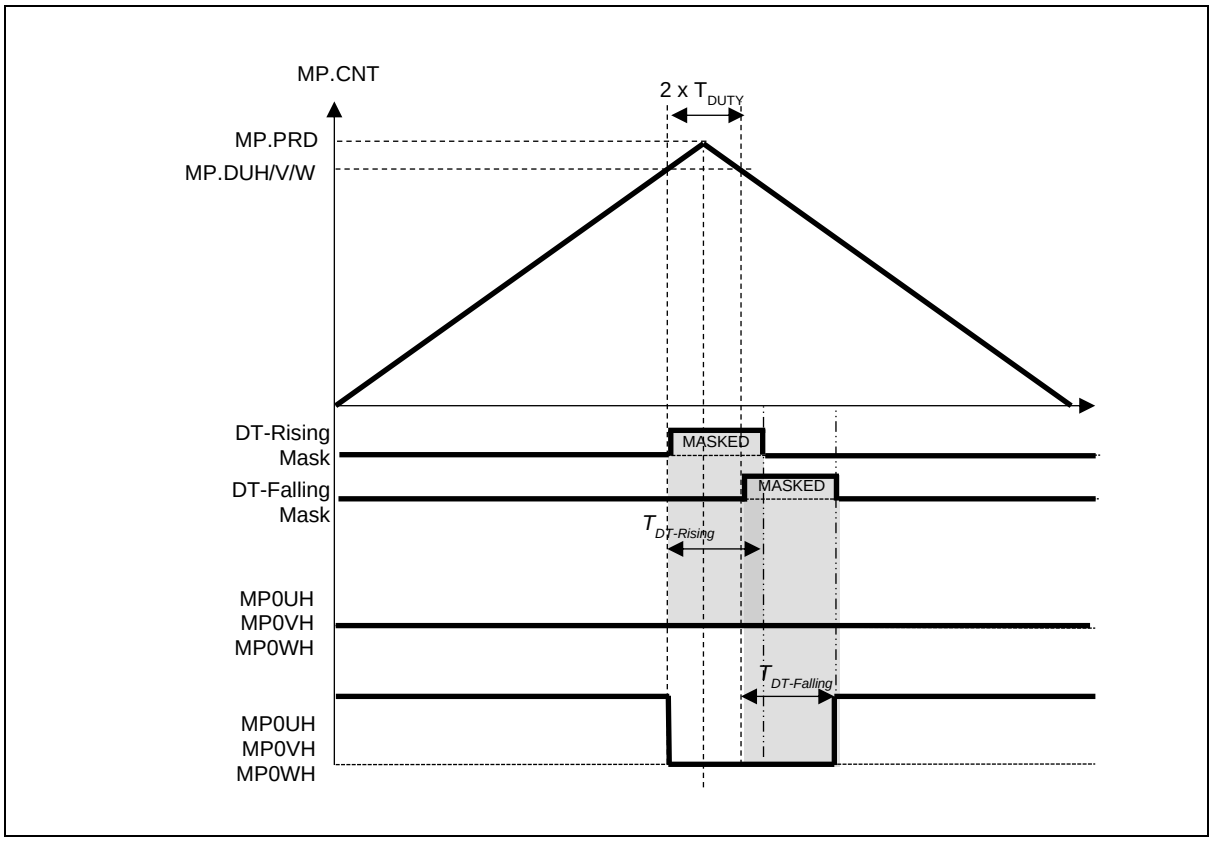


Figure 86. Zero H-Side Pulse Timing (TDT>2xTDUTY)

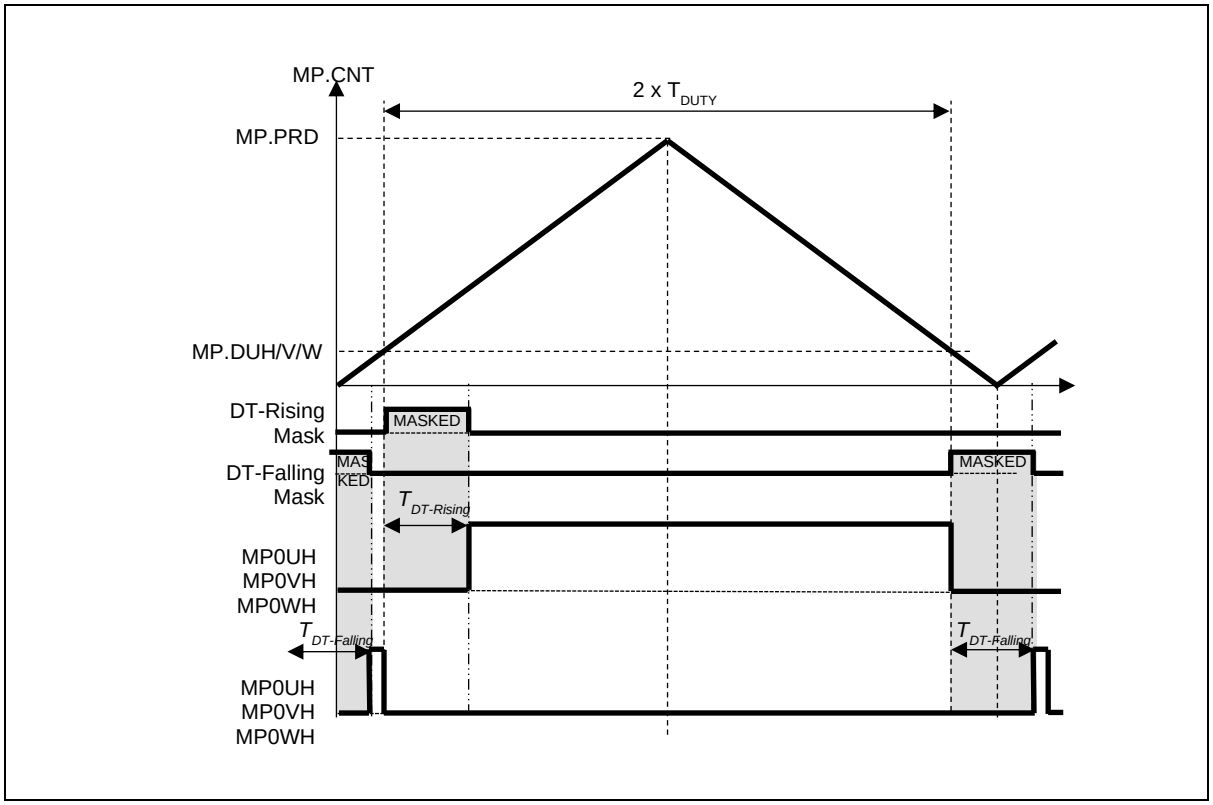


Figure 87. Minimum L-Side Pulse Timing (T_{DT}<Period-T_{DUTY})

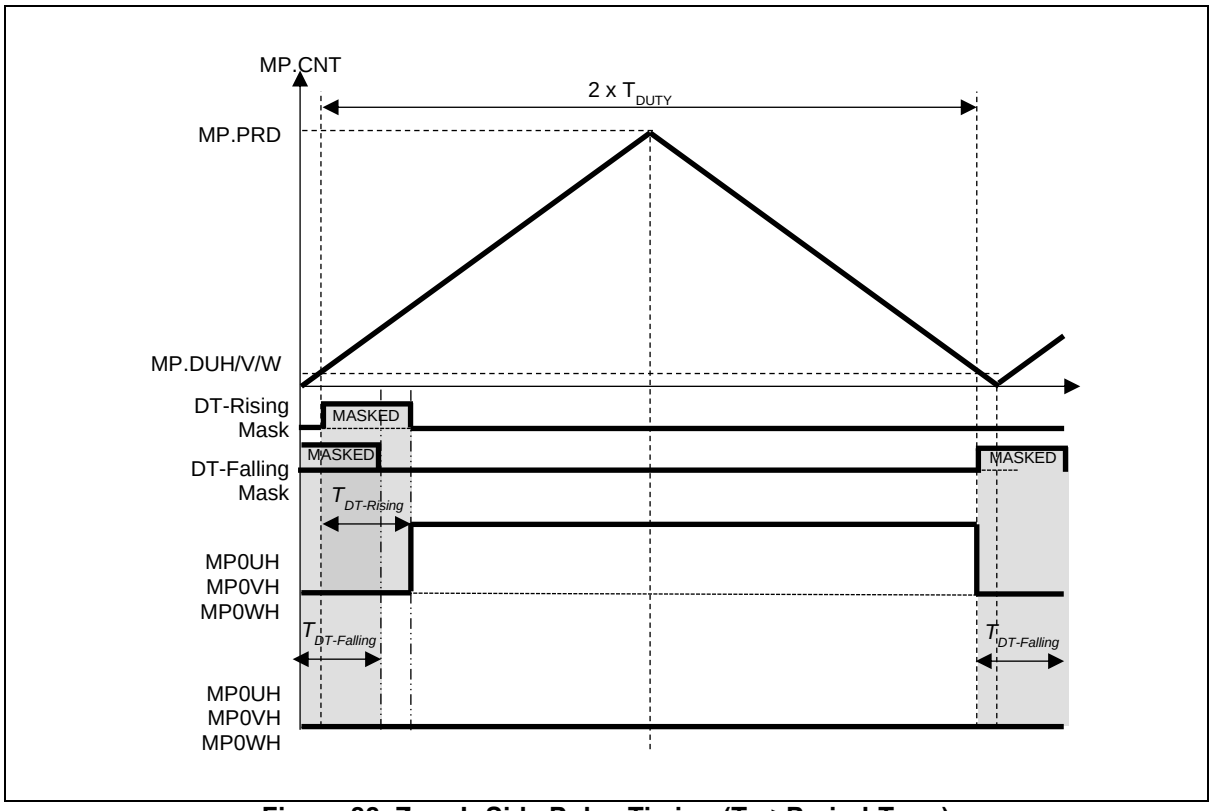


Figure 88. Zero L-Side Pulse Timing (T_{DT}>Period-T_{DUTY})

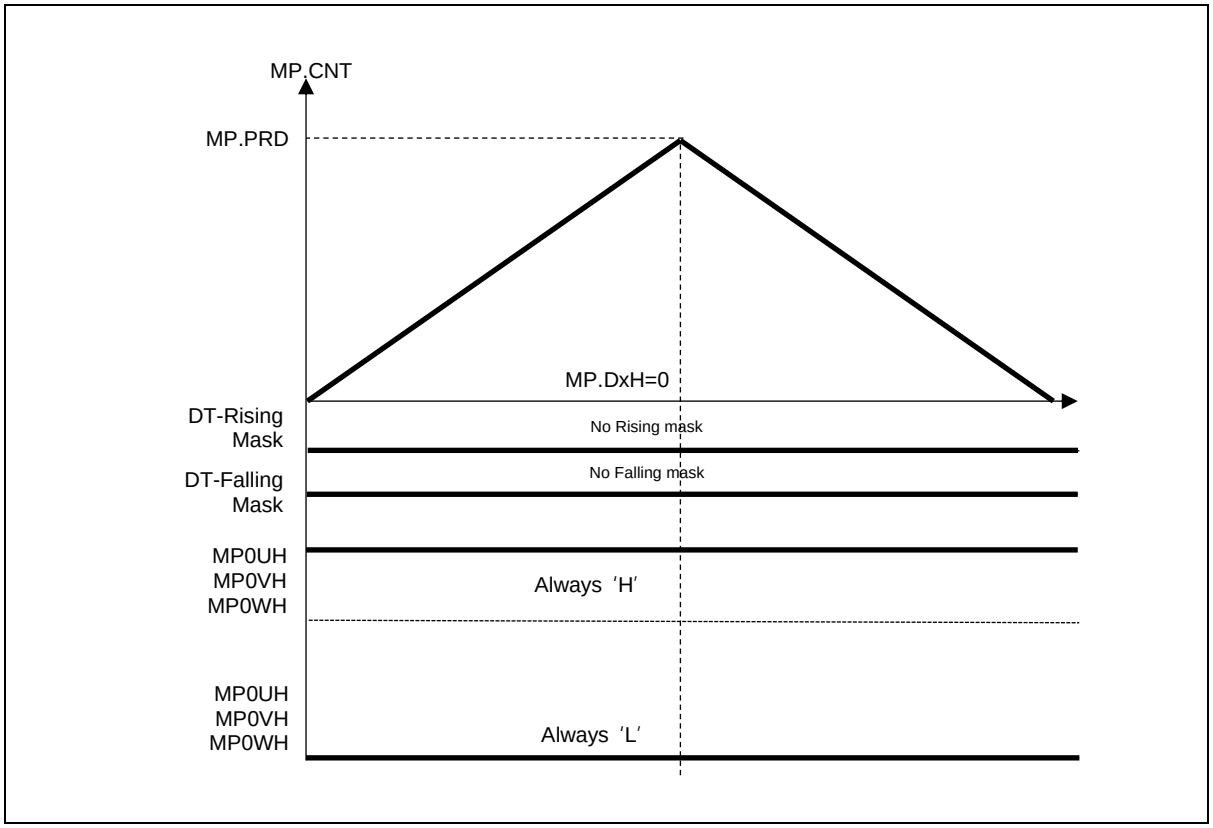


Figure 89. H-Side Always On (T_{DUTY}=Period: dead-time disabled)

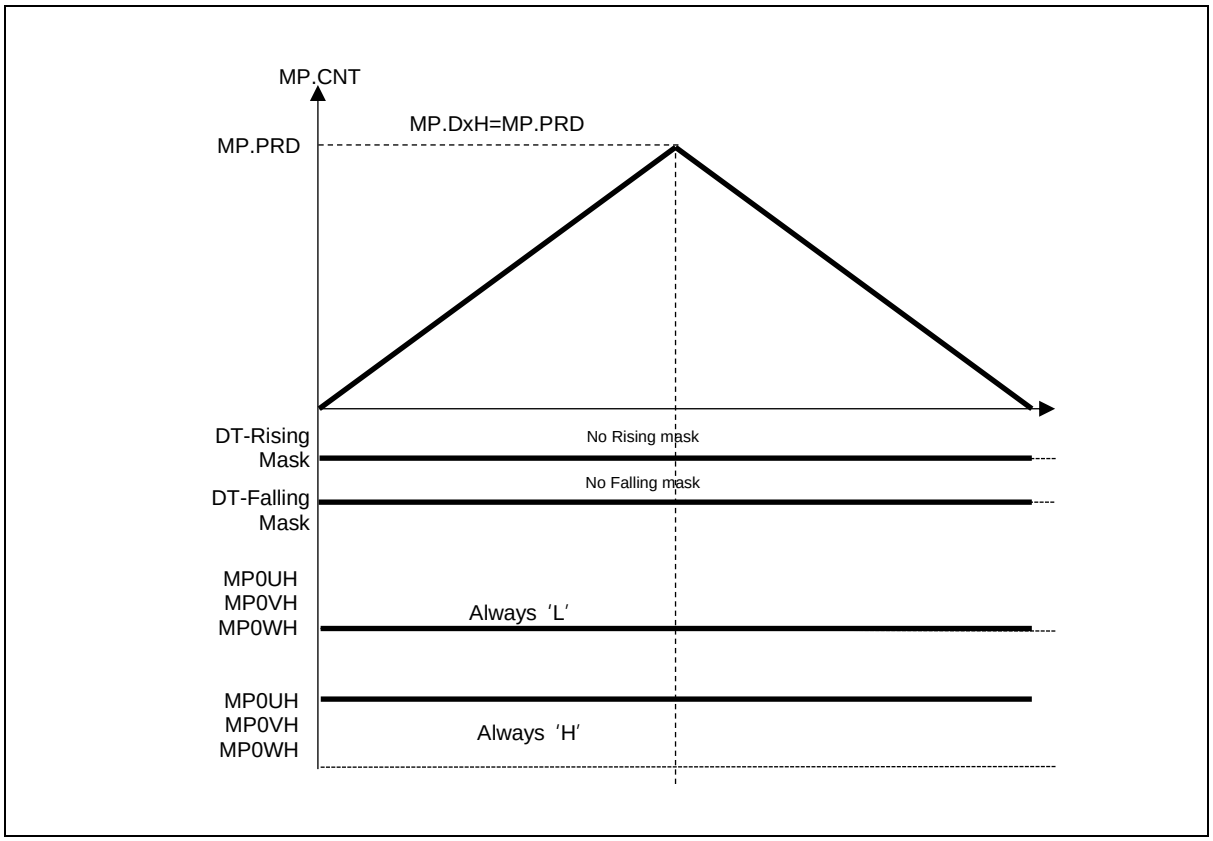


Figure 90. L-Side Always On (TDUTY='0': dead-time disabled)

14.3.8 Symmetrical mode vs asymmetrical mode

In symmetrical mode, the wave form is symmetric between and after the counter value being as period. The duty compare is performed twice in both up and down count period.

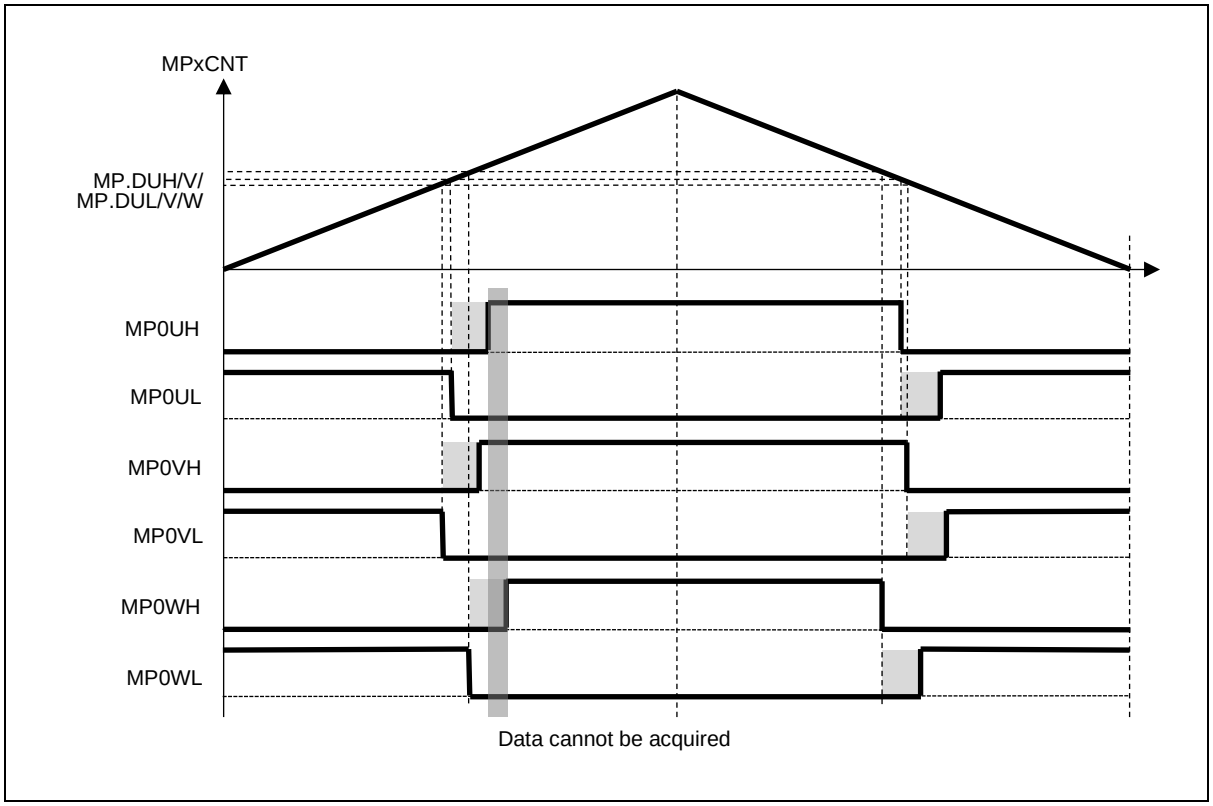


Figure 91. Symmetrical PWM Timing

In asymmetrical mode, the wave from is not symmetric between and after the counter value being as period. The duty compare of H-side is performed in both up count period. The duty compare of L-side is performed in both down count period.

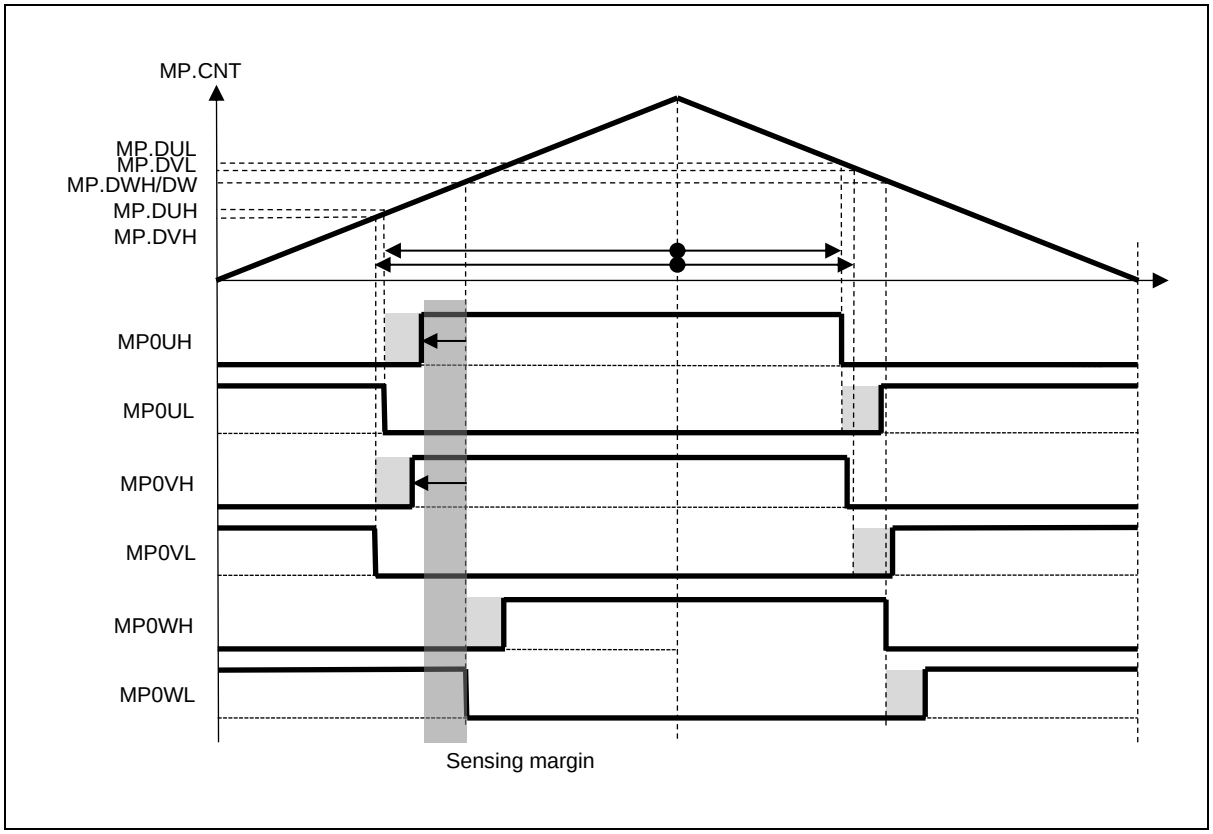


Figure 92. Asymmetrical PWM Timing and Sensing Margin

14.3.9 Description of ADC triggering function

Total 6 ADC trigger timing registers are provided. This dedicated register will make a trigger signal to start ADC conversion. The conversion channel of ADC will be defined in ADC control register.

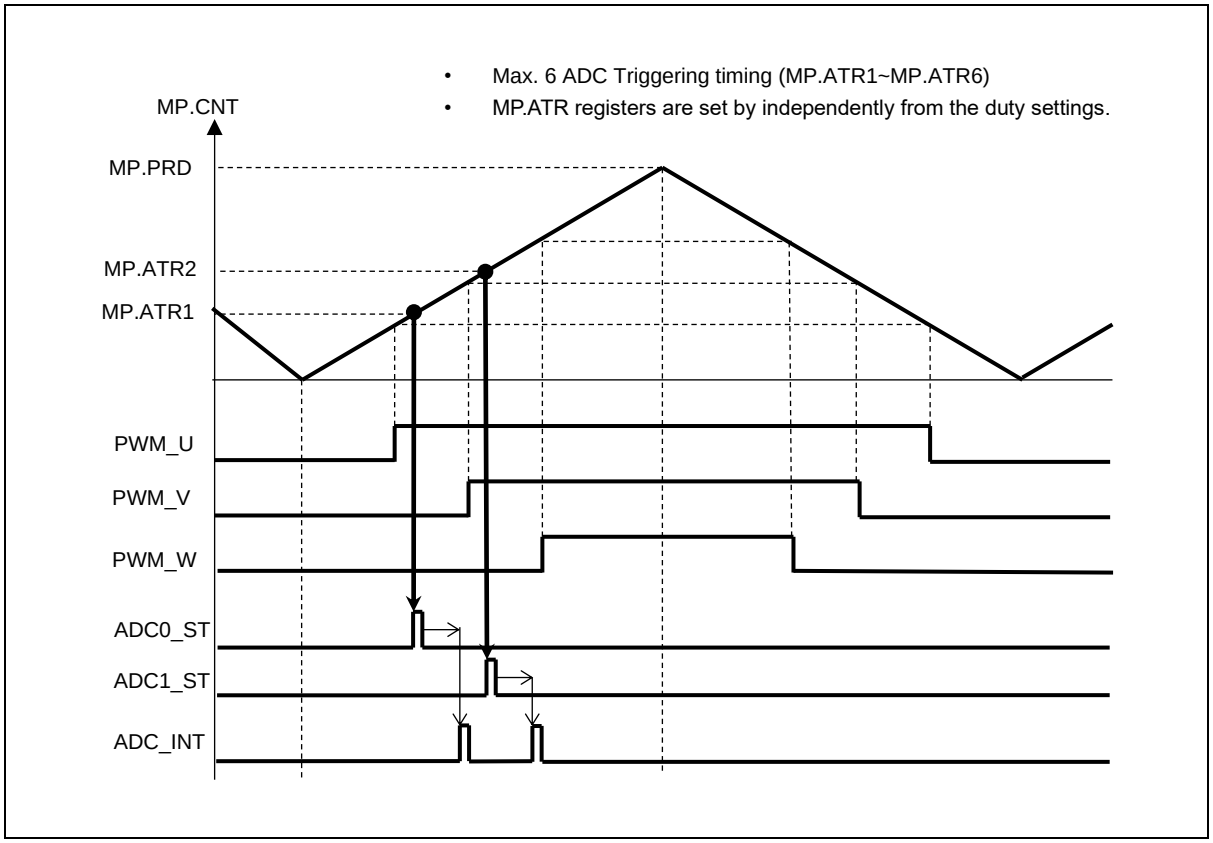


Figure 93. ADC Triggering Function Timing Diagram

Figure 94 shows an example of ADC Data acquisition.

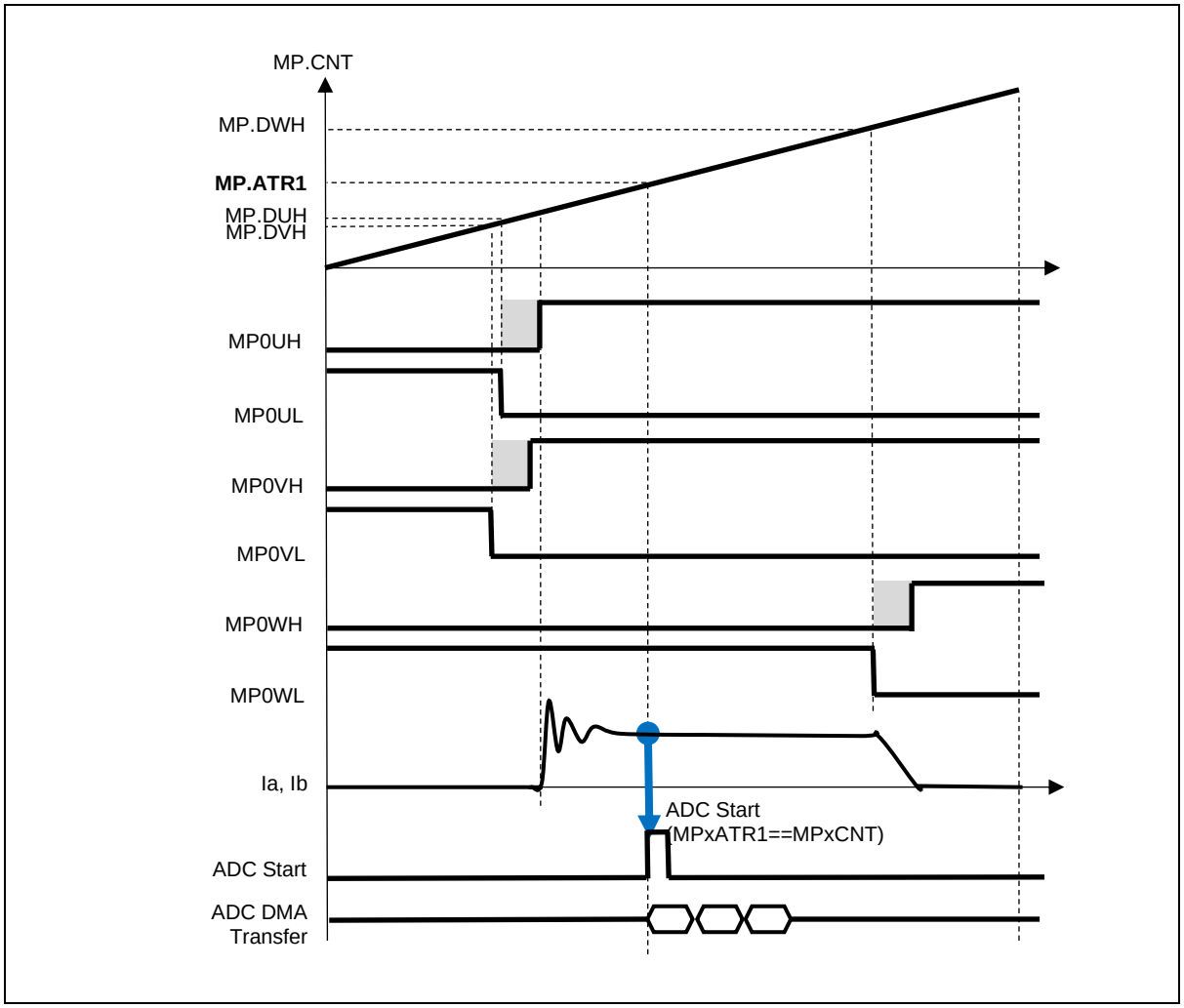


Figure 94. An Example of ADC Acquisition Timing by Event from MPWM

14.3.10 Interrupt generation timing

Each timing event can make interrupt request to the CPU as shown in Figure 95.

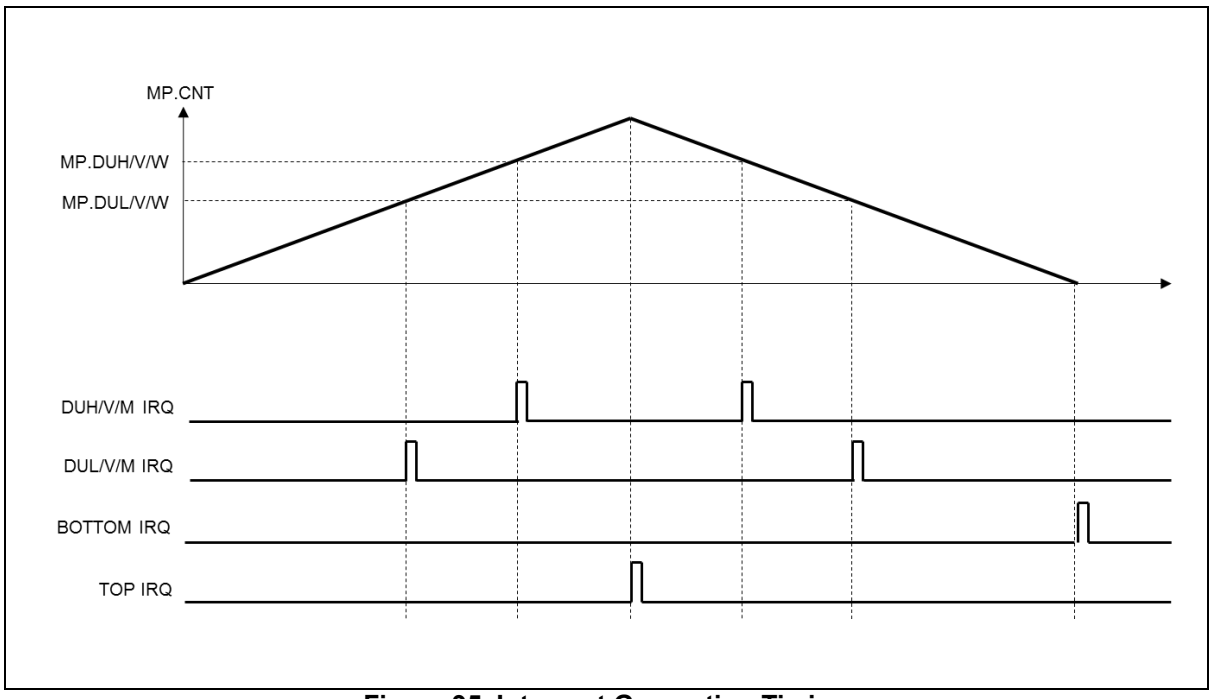


Figure 95. Interrupt Generation Timing

15 12-bit Analog-to-Digital Converter (ADC)

ADC block of AC33Mx064T series consists of an independent ADC unit featuring the followings:

- 11 channels of analog inputs (each ADC has 8 input channels)
- Single and Continuous conversion mode
- Up to 8 times sequential conversion supports
- Software trigger supports
- 8 internal trigger sources supports (PWMs, timers)
- Adjustable sample and hold time

Table 51 introduces pins assigned for ADC.

Table 51. Pin Assignment of ADC: External Pins

Pin name	Type	Description
VDD	P	Analog Power(3.0V~5V)
VSS	P	Analog GND
AN0	A	ADC Input 0
AN1	A	ADC Input 1
AN2	A	ADC Input 2
AN3	A	ADC Input 3
AN4	A	ADC Input 4
AN5	A	ADC Input 5
AN6	A	ADC Input 6
AN7	A	ADC Input 7
AN8	A	ADC Input 8
AN9	A	ADC Input 9
AN10	A	ADC Input 10

15.1 12-bit ADC block diagram

In this section, 12-bit ADC is described in a block diagram in Figure 96.

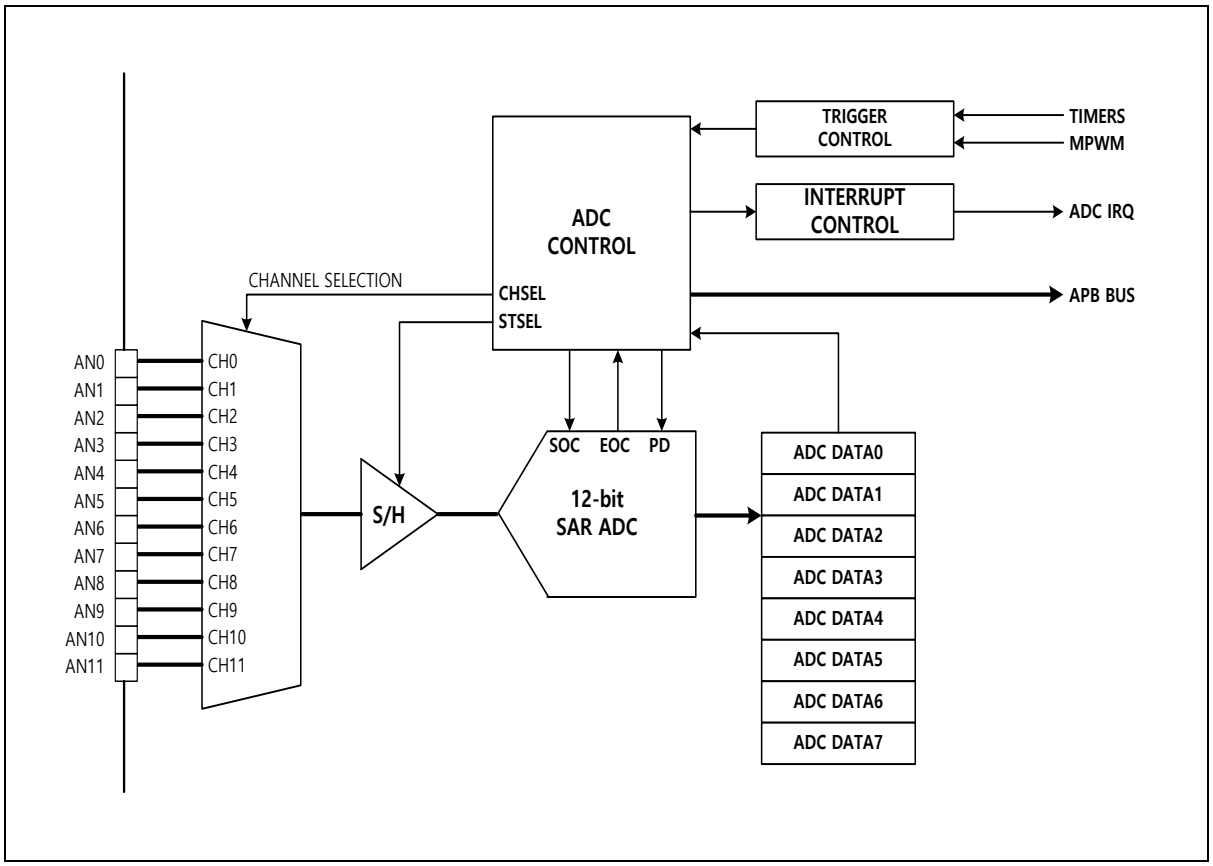


Figure 96. 12-bit ADC Block Diagram

15.2 Registers

Base address of ADC is introduced in the followings:

Table 52. Base Address of ADC

Name	Base address
ADC0	0x4000_B000
ADC1	0x4000_B100

Table 53. ADC Register Map

Name	Offset	Type	Description	Reset value	Reference
ADn.MR	0x0000	RW	ADC Mode register	0x00	15.2.1
ADn.CSCR	0x0004	RW	ADC Current Sequence/Channel register	0x00	15.2.2
ADn.CCR	0x0008	RW	ADC Clock Control register	0x80	15.2.3
ADn.TRG	0x000C	RW	ADC Trigger Selection register	0x00	15.2.4
-	0x0010	-	Reserved		
-	0x0014	-	Reserved		
ADn.SCSR	0x0018	RW	ADC Burst mode channel select	0x00	15.2.5
ADn.CR	0x0020	RW	ADC Control register	0x00	15.2.6
ADn.SR	0x0024	RW	ADC Status register	0x00	15.2.7
ADn.IER	0x0028	RW	ADC Interrupt Enable register	0x00	15.2.8
ADn.DDR	0x002C	R	ADCn DMA Data Register	0x00	15.2.9
ADn.DR0	0x0030	R	ADCn Sequence 0 Data register	0x00	15.2.9
ADn.DR1	0x0034	R	ADCn Sequence 1 Data register	0x00	15.2.9
ADn.DR2	0x0038	R	ADCn Sequence 2 Data register	0x00	15.2.9
ADn.DR3	0x003C	R	ADCn Sequence 3 Data register	0x00	15.2.9
ADn.DR4	0x0040	R	ADCn Sequence 4 Data register	0x00	15.2.9
ADn.DR5	0x0044	R	ADCn Sequence 5 Data register	0x00	15.2.9
ADn.DR6	0x0048	R	ADCn Sequence 6 Data register	0x00	15.2.9
ADn.DR7	0x004C	R	ADCn Sequence 7 Data register	0x00	15.2.9

15.2.1 AD*n*.MR ADC*n* Mode Register

ADC Mode Registers are 32-bit registers. This register configures ADC operation Mode. This register should be written first before other registers.

AD0.MR=0x4000_B000, AD1.MR=0x4000_B100

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
														DMAEN	DMACH	STSEL					SEQCNT			ADEN	ARST	ADMOD				TRGSEL	
														0x0	0x0	0x0					0x0			0x0	0x0	0x0				0x0	
														RW	RW	RW					RW			RW	RW	RW				RW	

17	DMAEN	DMA enable bit When DMA function is enabled, DMA request at every end of conversion (also in sequential mode) and interrupt request only be generated when ADC receives DMA done from DMAC.			
16	DMACH	DMA channel option When DMACH is set, Channel information of DMA data will be located at ADn.DDR[3:0] for half word size transfer. Channel information is at ADn.DDR [19:16] in default.(DMACH is low)			
15 12	STSEL	Sampling Time Selection ADC Sample & Hold circuit sampling time become (2 + STSEL[3:0]) ADC CLK cycles Minimum sampling time is 2 ADC CLK cycles			
10 8	SEQCNT	Number of conversion in a sequence If ADMOD is 2'h0 and SEQCNT is not 3'h0, CSEQN will be increased up to SEQCNT by trigger event.			
		000	Single mode	100	5 sequence AD conversion
		001	2 sequence AD conversion	101	6 sequence AD conversion
		010	3 sequence AD conversion	110	7 sequence AD conversion
		011	4 sequence AD conversion	111	8 sequence AD conversion
7	ADEN	0	ADC disable		
		1	ADC enable		
6	ARST	0	Stop at the end of sequence. Should set ASTART as 1 to restart again		
		1	Restart at the end of sequence.		
5 4	ADMOD	00	Single conversion mode		
		01	Burst conversion mode		
		10	Reserved		
		11	Reserved		
1 0	TRGSEL	00	Event Trigger Disabled/Soft-Trigger Only		
		01	Timer Event Trigger		
		10	MPWM0 Event Trigger		
		11	Reserved		

If ADCMOD was set for Burst Mode, ADC channels are controlled by SEQ0CH ~ SEQ7CH.

Sequential mode always start from SEQ0CH (In 3 sequential mode, Analog inputs of channels which assigned at SEQ0CH, SEQ1CH and SEQ2CH are converted sequentially).

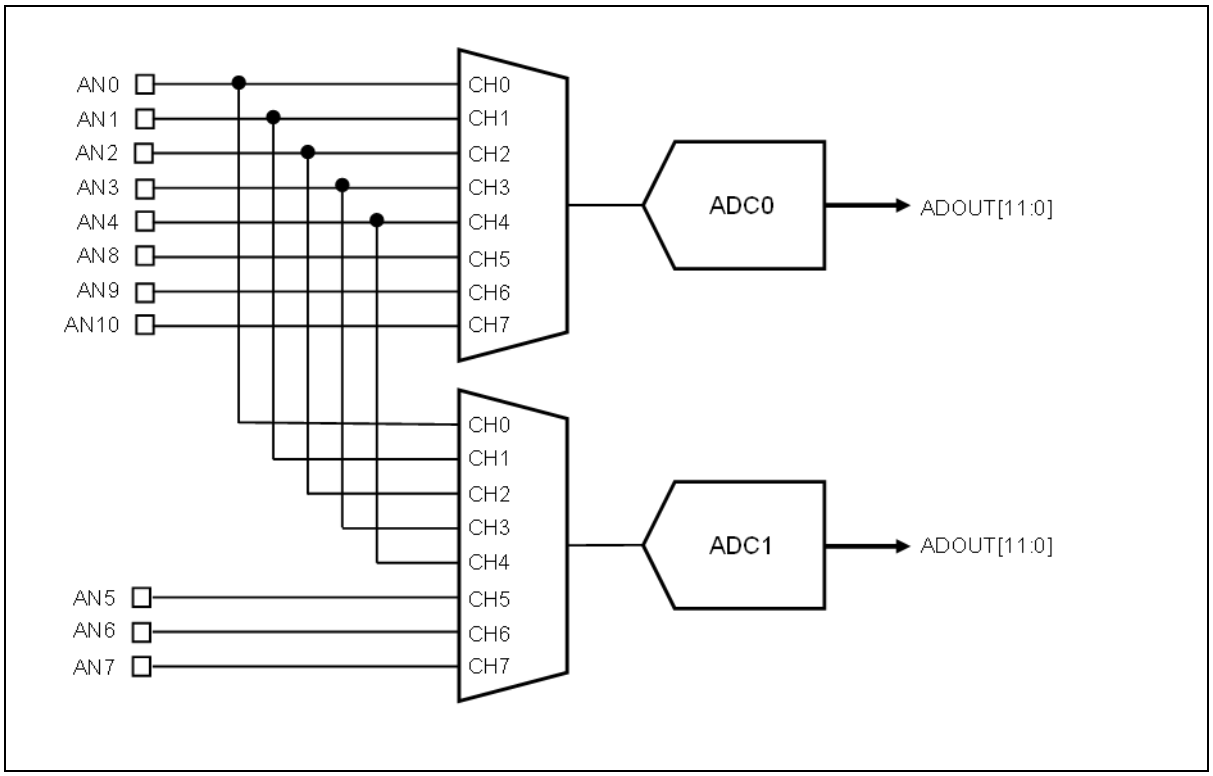


Figure 97. Analog Channel Block Diagram

15.2.2 AD*n*.CSCR ADC*n* Current Sequence/Channel Register

ADC Current Sequence/Channel Registers are 7-bit registers. This registers consist of Current Sequence Numbers and Current Active Channel values. CSEQN (Current Sequence Number) can be written to set current sequence number immediately. This register should be written first before ADn.SCSR.

AD0.SR=0x4000_B004, AD1.SR=0x4000_B104

7	6	5	4	3	2	1	0
	CSEQN			CACH			
	0x0			0x0			
	RW			RO			

7 4	CSEQN	Current Sequence Number, can write when not busy.	
		0000	Current Sequence is 0
		0001	Current Sequence is 1
		0010	Current Sequence is 2
		0011	Current Sequence is 3
		0100	Current Sequence is 4
		0101	Current Sequence is 5
		0110	Current Sequence is 6
		0111	Current Sequence is 7
3 0	CACH	Current Active Channel	
		0000	ADC channel 0 is active
		0001	ADC channel 1 is active
		0010	ADC channel 2 is active
		0011	ADC channel 3 is active
		0100	ADC channel 4 is active
		0101	ADC channel 5 is active
		0110	ADC channel 6 is active
		0111	ADC channel 7 is active
		1000	reserved
		1001	reserved
		1010	reserved
		1011	reserved
		1100	reserved
		1101	reserved
		1110	reserved
		1111	reserved

15.2.3 AD*n*.CCR ADC*n* Clock Control Register

ADC Control Registers are 16-bit registers.

ADC period register

AD0.CR1=0x4000_B008, AD1.CR1=0x4000_B108

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCPDA	CKDIV							ADCPD	EXTCLK	CLKINVT					
0	0x00							1	0	0					
RW	RW							RW	RW	RW					

15	ADCPDA	ADC R-DAC disable to save power Don't set "1" here(it's optional bit)
14 8	CLKDIV[6:0]	ADC clock divider when EXTCLK is '0'. ADC clock = system clock/CLKDIV CKDIV=0 : ADC clock=system clock CKDIV=1 : ADC clock=stop **Note) ADC clock ≤ 25MHz**
NOTE: In continuous conversion mode or burst conversion mode, the CLKDIV must be set to 3 or higher.		
7	ADCPD	ADC Power Down 0 – ADC normal mode 1 – ADC Power Down mode
6	EXTCLK	Select if ADC uses external clock. 0 – internal clock(CKDIV enabled) 1 – external clock(SCU clock)
NOTE: In continuous conversion mode or burst conversion mode, the EXTCLK must be set to 0(internal clock).		
5	CLKINVT	Divided clock inversion(optional bit) 0 – duty ratio of divided clock is larger than 50% 1 – duty ratio of divided clock is less than 50%

15.2.4 AD*n*.TRG ADC Trigger Selection Register

ADC Trigger registers are 32-bit registers.

ADC Trigger channel register.

In Single/Burst mode, all the bit fields are used.

In Burst conversion mode, Only BSTTRG bit field (bit3~bit0) is used.

AD0.TRG0=0x4000_B00C, AD1.TRG0=0x4000_B10C

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
0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
RW	RW	RW	RW	RW	RW	RW	RW

30 28	SEQTRG7	8th Sequence Trigger Source
26 24	SEQTRG6	7th Sequence Trigger Source
22 20	SEQTRG5	6th Sequence Trigger Source
18 16	SEQTRG4	5th Sequence Trigger Source
14 12	SEQTRG3	4th Sequence Trigger Source
10 8	SEQTRG2	3rd Sequence Trigger Source
6 4	SEQTRG1	2nd Sequence Trigger Source
2	SEQTRG0	1st Sequence Trigger Source
0	BSTTRG	Burst conversion Trigger Source

Table 54. ADn.TRG Trigger Selection Value

Value	Timer (TRGSEL '2'h1)	MPWM0 (TRGSEL '2'h2)
0	Timer 0	MP0ATR1
1	Timer 1	MP0ATR2
2	Timer 2	MP0ATR3
3	Timer 3	MP0ATR4
4	Timer 8	MP0ATR5
5	Timer 9	MP0ATR6
6	-	BOTTOM
7	-	PERIOD

15.2.5 AD*n*.SCSR ADC Sequence Channel Selection Register

ADC Burst Mode Channel Select Register is 32-bit register.

AD0.BCSR=0x4000_B018, AD1.BCSR=0x4000_B118

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
SEQ7CH	SEQ6CH	SEQ5CH	SEQ4CH	SEQ3CH	SEQ2CH	SEQ1CH	SEQ0CH
0x0	0x0	0x0	0x0	0x0	0x0	0x0	0x0
RW	RW	RW	RW	RW	RW	RW	RW

31 28	SEQ7CH	8th conversion sequence channel selection
27 24	SEQ6CH	7th conversion sequence channel selection
23 20	SEQ5CH	6th conversion sequence channel selection
19 16	SEQ4CH	5th conversion sequence channel selection
15 12	SEQ3CH	4th conversion sequence channel selection
11 8	SEQ2CH	3rd conversion sequence channel selection
7 4	SEQ1CH	2nd conversion sequence channel selection
3 0	SEQ0CH	1st conversion sequence channel selection This channel should be used for Single mode

15.2.6 AD*n*.CR ADCn Control Register

ADC start register. This register 2 is 8-bit register.

AD0.CR2=0x4000_B020, AD1.CR=0x4000_B120

7	6	5	4	3	2	1	0
ASTOP							ASTART
0 W							0 RW

7	ASTOP	0	No
		1	ADC conversion stop (will be clear next @ADC clock) If ASTOP set after conversion cycle start, present conversion would be completed.
0	ASTART	0	No ADC conversion
		1	ADC conversion start (will be clear next @ADC clock) ADCEN should be "1" to start ADC If ASTART is set as 1'h1 when ARST is 1'h0 in trigger event mode, ADC conversion will start once as SEQCNT set.

15.2.7 AD*n*.SRADC*n* Status Register

ADC Status Register is 32-bit register.

AD0.SR=0x4000_B024, AD1.SR=0x4000_B124

7	6	5	4	3	2	1	0
EOC	ABUSY	DOVRUN	DMAIRQ	TRGIRQ	EOSIRQ	-	EOCIRQ
0	0	0	0	0	0	-	0
RO	RO	RO	RO	RC	RC	-	RC

7	EOC	ADC End-of-Conversion flag (Start-of-Conversion made by ADC_CLK clears this bit , not ASTART)
6	ABUSY	ADC conversion busy flag
5	DOVRUN	DMA overrun flag (not interrupt) (DMA ACK didn't come until end of next conversion)
4	DMAIRQ	DMA done received (DMA transfer is completed)
3	TRGIRQ	ADC Trigger interrupt flag(Write "1" to clear flag) (0: no int / 1: int occurred)
2	EOSIRQ	This flag will be set upon final end of a sequence (Write "1" to clear flag)
		0 None.
		1 End-of-Sequence(burst) Interrupt occurred
0	EOCIRQ	This flag will be set upon each conversion in a sequence is occurred(Write "1" to clear flag)
		0 None.
		1 End-of-Conversion Interrupt occurred

15.2.8 AD*n*.IER Interrupt Enable Register

AD0.IER=0x4000_B028, AD1.IER=0x4000_B128

7	6	5	4	3	2	1	0
			DMAIRQE	TRGIRQE	EOSIRQE		EOCIRQE
			0	0	0		0
			RW	RW	RW		RW

4	DMAIRQE	DMA done interrupt enable 0: interrupt disable 1: interrupt enable
3	TRGIRQE	ADC trigger conversion interrupt enable
2	EOSIRQE	ADC sequence conversion interrupt enable
0	EOCIRQE	ADC single conversion interrupt enable

15.2.9 AD*n*.DDR ADC*n* DMA Data Register

ADC DMA Data Registers are 16-bit registers.

This register is a temporary register only for DMA transfer (A/D conversion data of just completed conversion).

AD0.DDR=0x4000_B02C, AD1.DDR=0x4000_B12C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC DMA Temporary Data												ADMACH			
0x000												0x0			
R												R			

15 4	ADDMAR	ADC conversion result data (12-bit)
3 0	ADMACH	ADC data channel indicator

15.2.10 AD*n*.DR ADC*n* Sequence Data Register 0~7

ADC Data Registers are 16-bit registers.

ADC conversion result register.

AD0.DR0=0x4000_B030, AD0.DR1=0x4000_B034, AD0.DR2=0x4000_B038, AD0.DR3=0x4000_B03C
AD0.DR4=0x4000_B040, AD0.DR5=0x4000_B044, AD0.DR6=0x4000_B048, AD0.DR7=0x4000_B04C
AD1.DR0=0x4000_B130, AD1.DR1=0x4000_B134, AD1.DR2=0x4000_B138, AD1.DR3=0x4000_B13C
AD1.DR4=0x4000_B140, AD1.DR5=0x4000_B144, AD1.DR6=0x4000_B148, AD1.DR7=0x4000_B14C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCDATA															
0x000															
R															

15 4	ADC DATA	ADC channel 0~7 data (12-bit)

15.3 Functional Description

15.3.1 ADC single mode timing diagram

ADC conversion will be started by ADCn.CR.ASTART written as '1' in single conversion mode. Once ADCnCR.ASTART is set, SOC (start of Conversion) will be activated in 3 ADC clocks and ADCn.SR.EOCIRQ will be set in 2 ADC clocks and 2 PCLKs after the End of Conversion.

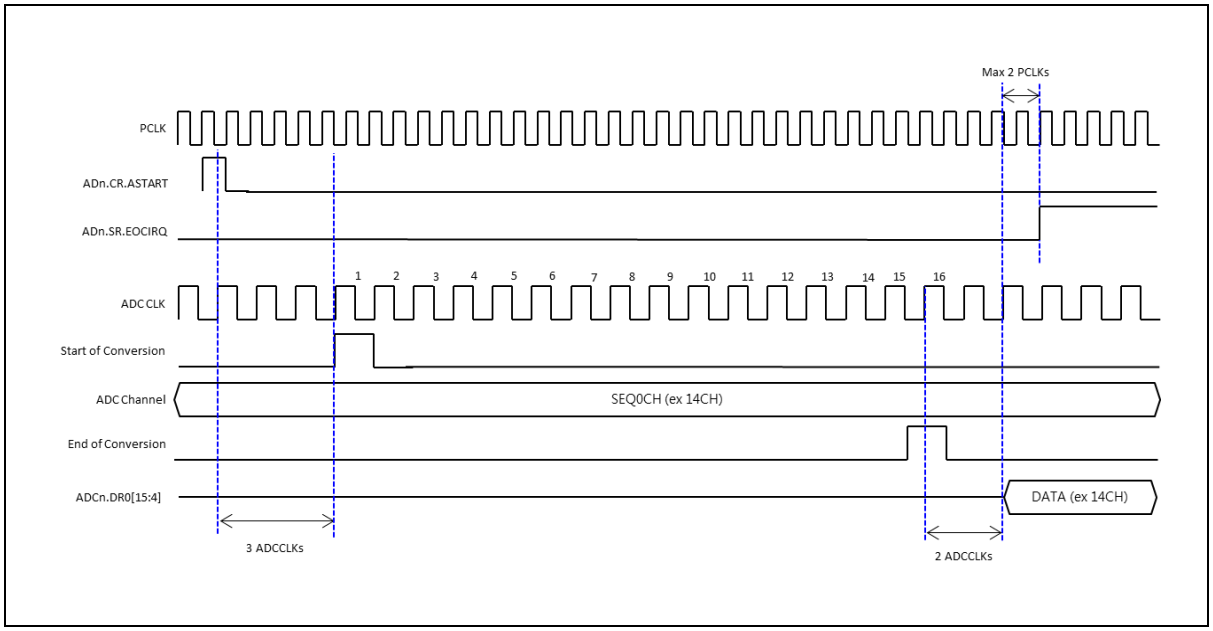


Figure 98. ADC Single Mode Timing (when ADCn.MR.AMOD = '0')

15.3.2 ADC sequential mode timing diagram

There are two sources to make SOC in burst mode. First is TRG event (TIMER and MPWM) and the other is ASTART. When TRGSEL is set as timer event trigger or MPWM event trigger, SOC will be made by the trigger of ADn.TRG.BSTTRG (And.TRG[3:0]). For example, ADC conversion will be started by the trigger of TIMER9 if ADn.TRG.BSTTRG is set as TIMER9. Once the BSTTRG's trigger events, ADC will covert ADC Channels as much as ADn.MR.SEQCNT set. See Figure 99.

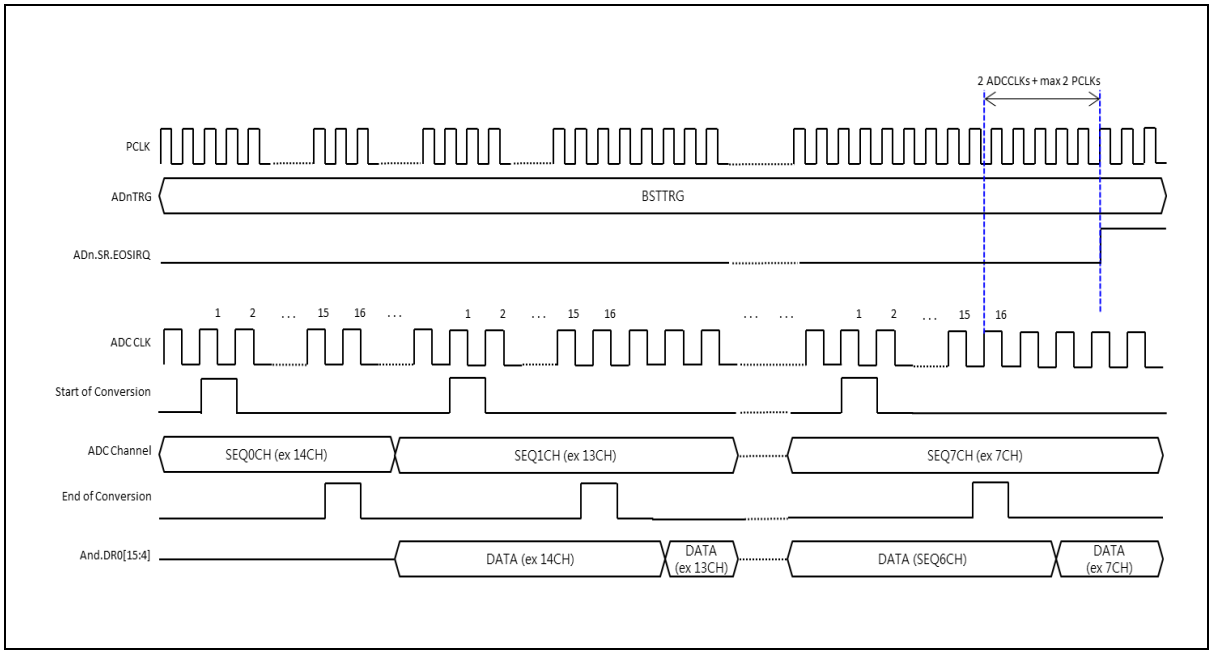


Figure 99. ADC Burst Mode Timing (when AD.MR.AMOD = '1')

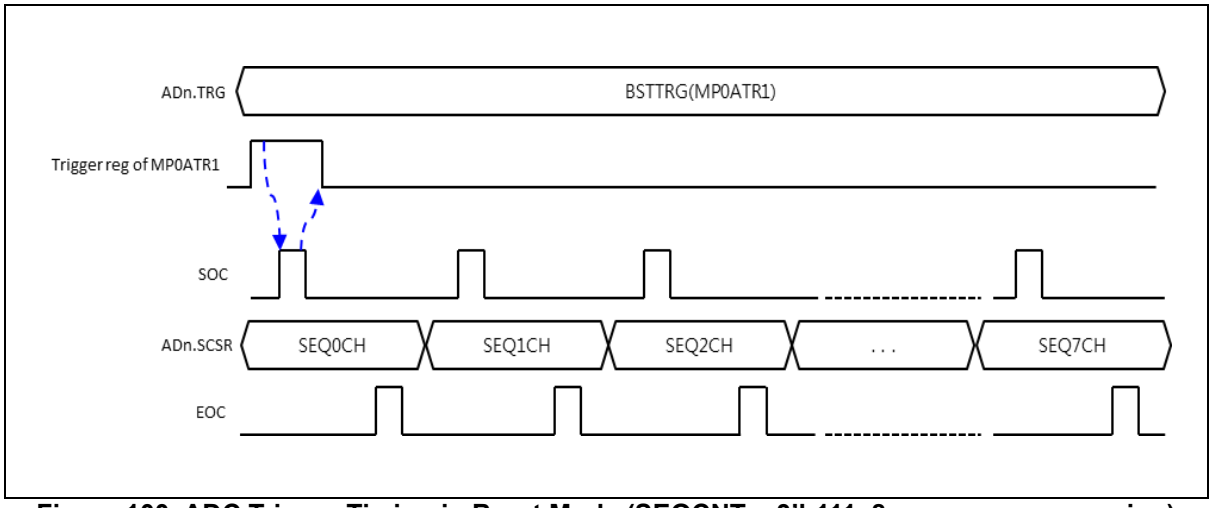


Figure 100. ADC Trigger Timing in Burst Mode (SEQCNT = 3'b111, 8 sequence conversion)

15.3.3 ADC sequential conversion mode timing diagram

To set sequential conversion mode, ADn.MR.AMOD is 2'b00 and ADn.MR.SEQCNT is not 2'b00.

The operation of sequential mode is the almost same as the burst mode. The difference is the source of SOC. Each SOC is made by the trigger of the SEQTRGx as each SEQCNT. See Figure 101.

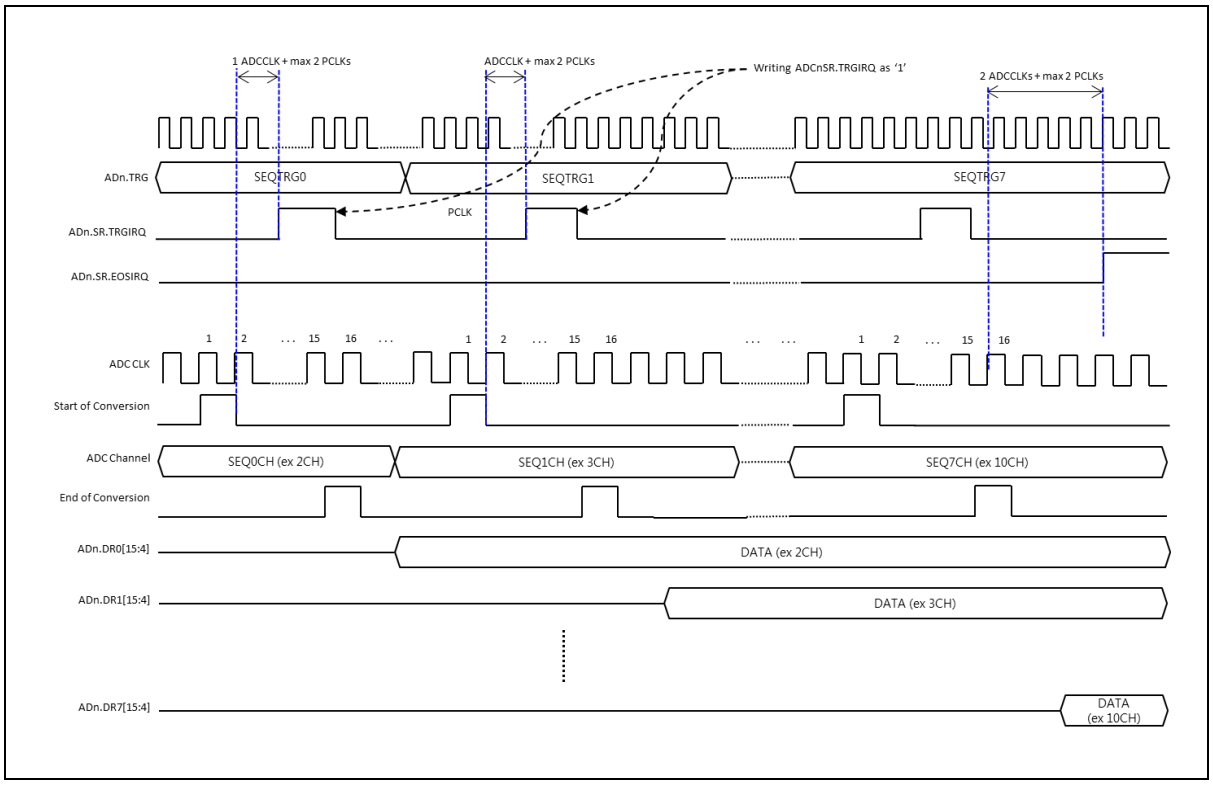


Figure 101. ADC Sequential Mode Timing (when AD.MR.AMOD = '0 and AD.MR.SEQCNT ≠ '0')

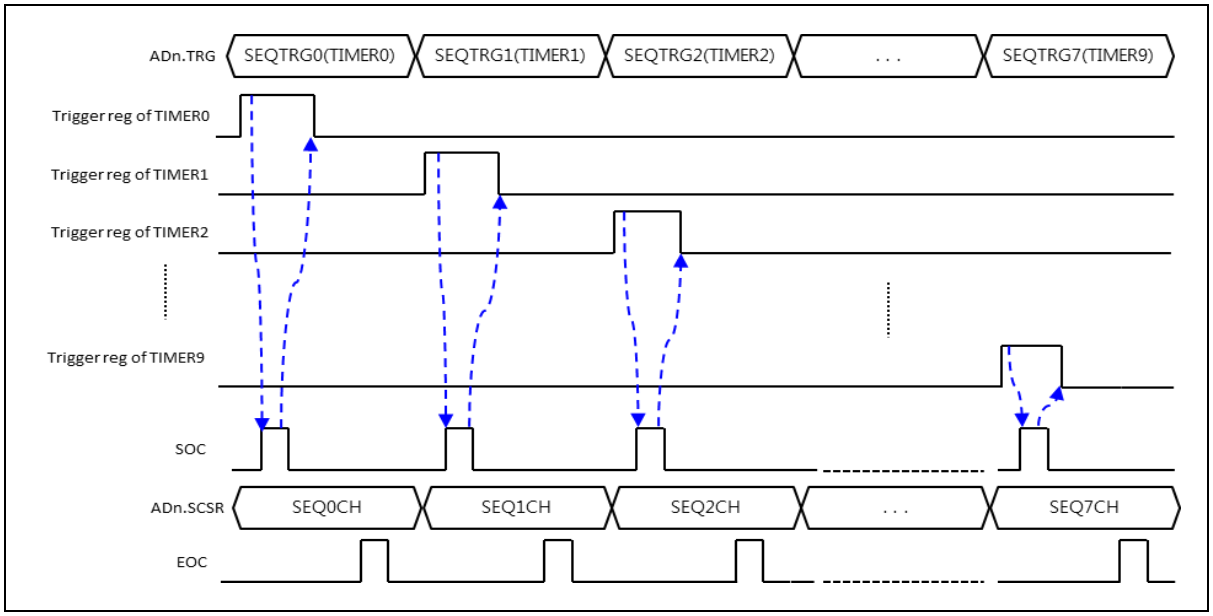


Figure 102. ADC Trigger Timing in Sequential Mode (SEQCNT = 3'b111, 8 sequence conversion)

16 Electrical characteristics

16.1 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 55. Absolute Maximum Rating

Parameter	Symbol	min	max	unit
Power supply (V_{DD})	V_{DD}	-0.5	+6	V
Analog power supply (AV_{DD})	AV_{DD}	-0.5	+6	V
VDC output voltage	V_{DD18}			V
Input high voltage		-	V_{DD}+0.5	V
Input low voltage		V_{SS} – 0.5	-	V
Output low current per pin	I_{OL}		2.5	mA
Output low current total	$\sum I_{OL}$		25	mA
Output high current per pin	I_{OH}		-2.5	mA
Output high current total	$\sum I_{OH}$		25	mA
Input main clock range		0.4	8	MHz
Operating frequency		-	48	MHz
Storage temperature	T_{ST}	-55	+125	℃
Operating temperature	T_{OP}	-40	+105	℃
Power supply (V_{DD})	V_{DD}	-0.5	+6	V

16.2 DC characteristics

Table 56. Recommended Operating Condition

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Supply voltage	V_{DD}		3.0		5.5	V
Supply voltage	AV_{DD}		3.0	5.0	5.5	V
Operating frequency	f	OSC_{MAIN}	4		8	MHz
		OSC_{INT}	0.5	1	1.5	MHz
		PLL	4		48	MHz
Operating temperature	T_{OP}	T_{OP}	-40		+105	℃

Table 57. DC Electrical Characteristics

(VDD = +5V, Ta = 25 ℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Input low voltage	V_{IL}	Schmitt input	-	-	$0.2V_{DD}$	V
Input high voltage	V_{IH}	Schmitt input	$0.8V_{DD}$	-	-	V
Output low voltage	V_{OL}	I_{OL} = 3mA	-	-	V_{SS}+1.0	V
Output high voltage	V_{OH}	I_{OH} = -3mA	V_{DD}–1.0	-	-	V
Input high leakage	I_{IH}				4	µA
Input low leakage	I_{IL}		-4			µA
Pull-up resister	R_{PU}	R_{MAX}:V_{DD}=3.0V R_{MIN}:V_{DD}=5V	30	-	70	kΩ

16.3 Current consumption

Table 58. Current Consumption in Each Mode

(Temperature: +25℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Normal operation	$I_{DD\ NORM}$	OSC_{RING}=RUN OSC_{MAIN}=8MHz HCLK=48MHz	-	20	-	mA
Sleep mode	$I_{DD\ SLEEP}$	OSC_{RING} =RUN OSC_{MAIN} =8MHz HCLK =48MHz	-	8.3	-	mA

NOTE: uart en, 1 port toggle @5V

16.4 POR electrical characteristics

Table 59. POR Electrical Characteristics

(Temperature: -40℃ to +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	V_{DD18}		1.6	1.8	2.0	V
Operating current	I_{DD}	Typ. <6μA If always on	-	60	-	nA
POR set level	V_{RISING}	V_{DD} rising (slow)	1.3	1.4	1.55	V
POR reset level	$V_{FALLING}$	V_{DD} falling (slow)	1.1	1.2	1.4	V

16.5 LVD electrical characteristics

Table 60. LVD Electrical Characteristics

(Temperature: -40℃ to +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	V_{DD}		1.7		5	V
Operating current	I_{DD}	Typ. <6μA when always on	-	1	-	mA
LVD set level 0	V_{LVD0}	V_{DD} falling (slow)	1.6	1.8	2.0	V
LVD set level 1	V_{LVD1}	V_{DD} falling (slow)	2.0	2.2	2.5	V
LVD set level 2	V_{LVD2}	V_{DD} falling (slow)	2.5	2.7	3.0	V
LVD set level 3	V_{LVD3}	V_{DD} falling (slow)	3.9	4.3	4.6	V

16.6 VDC electrical characteristics

Table 61. VDC Electrical Characteristics

(Temperature: -40℃ to +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	V_{DD}		3.0	-	5.5	V
VDC output voltage	V_{OUT}	@RUN	1.62	1.8	1.98	V
		@STOP	1.4	1.8	2.0	V
Regulation current	I_{OUT}				100	mA
Drop-out voltage	V_{DROP}	V_{DD}=3.0V I_{OUT}=100mA	-	-	200	mV
Current consumption	$I_{DD\ NORM}$	@RUN	-	100	150	μA
	$I_{DD\ STOP}$	@STOP	-	1	2	μA

16.7 External OSC characteristics

Table 62. External OSC Characteristics

(Temperature: -40℃ to +105℃)

Parameter	Symbol	Condition	Min	Typ	Max	unit
Operating voltage	V_{DD}		3.0	-	5.5	V
IDD	I_{DD}	@4MHz/5V	-	240		μA
Frequency	f_{osc}		4	8	10	MHz
Output voltage	V_{OUT}		1.2	2.4	-	V
Load capacitance	C_L		5	22	35	pF

16.8 PLL electrical characteristics

Table 63. PLL Electrical Characteristics

(Temperature: -40℃ to +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	V_{DD}		3.0		5.5	V
Output frequency	f_{OUT}		4		48	MHz
Operating current	I_{DD}	@50MHz		1.3		mA
Duty	f_{DUTY}		40	-	60	%
P-P jitter	JITTER	@Lock			500	Ps
VCO	VCO		20		80	MHz
Input frequency	f_{IN}		4		8	MHz
Locking time	t_{LOCK}				1	ms

16.9 ADC electrical characteristics

Table 64. ADC Electrical Characteristics

(Temperature: -40℃ ~to +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	AV_{DD}		3.0	5	5.5	V
Reference voltage	AV_{REF}		3.0	5	5.5	V
Resolution				12		Bit
Operating current	AI_{DD}				2.8	mA
Analog input range			0		AV_{DD}	V
Conversion rate				-	1.6	Msps
Operating Frequency	f_{ACLK}				25	MHz
DC Accuracy	INL			±2.5		LSB
	DNL			±1.0		LSB
Offset Error				±1.5		LSB
Full Scale Error				±1.5		LSB
SNDR	SNDR			68		dB
THD				-70		dB

NOTES:

1. DNL: Maximum deviation between actual steps and the ideal one.
2. INL: Integral Linearity Error: maximum deviation between any actual transition and the end point

17 Package information

17.1 48 LQFP package information

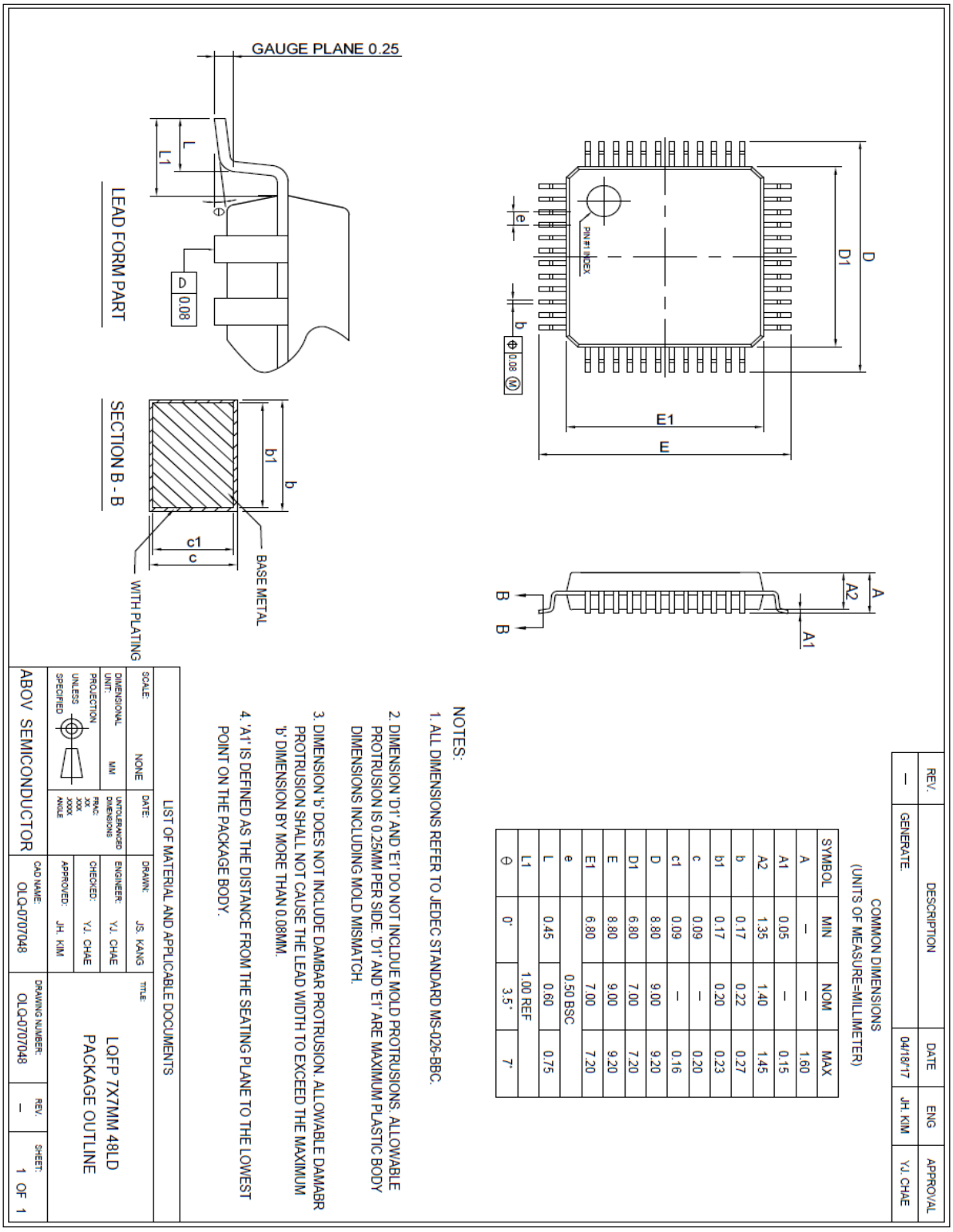


Figure 103. 48 LQFP Package Outline

17.2 32 LQFP package information

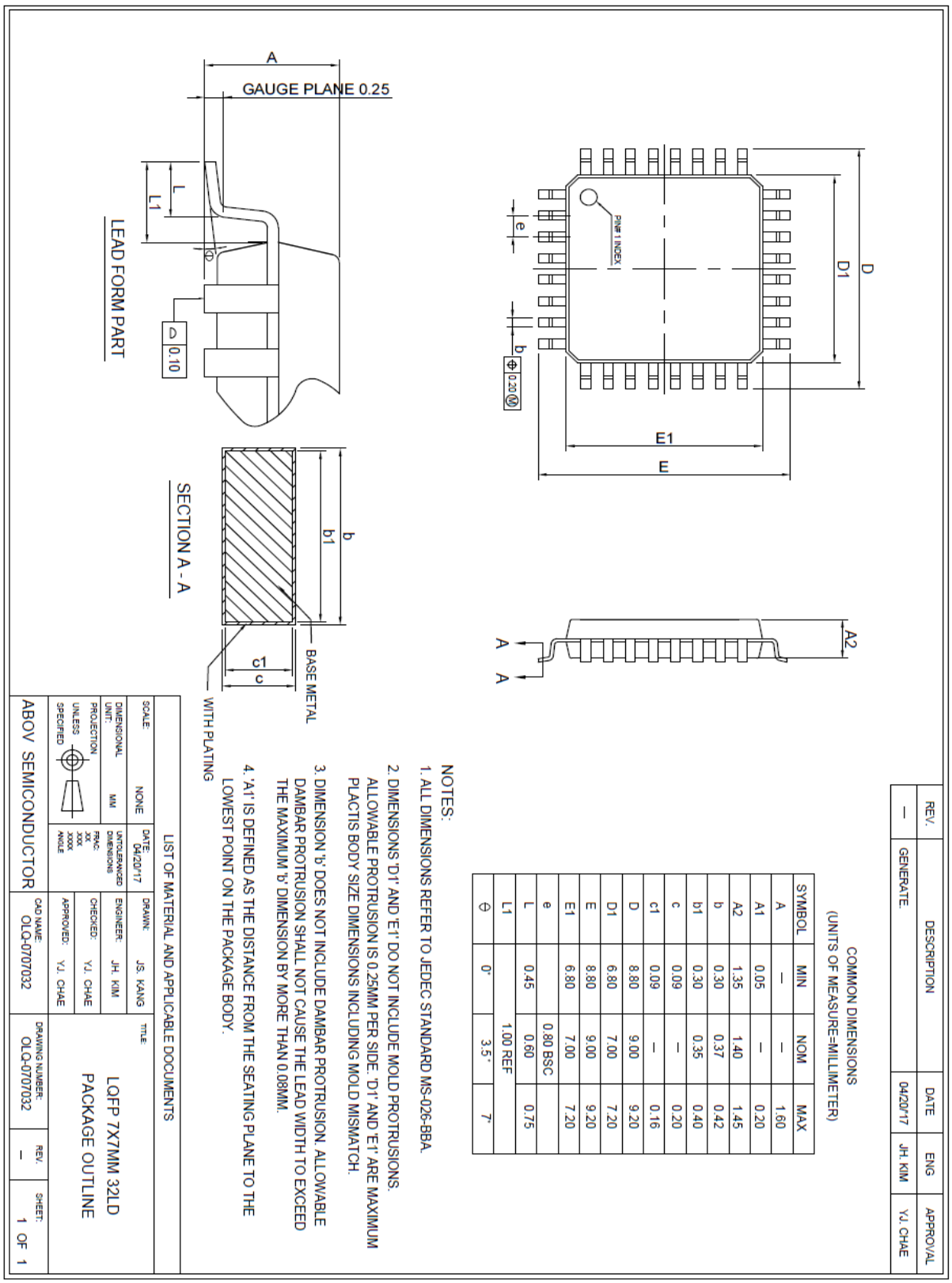


Figure 104. 32 LQFP Package Outline

18 Ordering information

Table 65. AC33Mx064T Series Device Ordering Information

Device name	Flash	SRAM	UART	SPI	I2C	MPWM	ADC	I/O ports	Package
AC33M4064T	64KB	8KB	2	1	1	1	2-unit 11 ch	44	LQFP-48
AC33M3064T*	64KB	8KB	2	1	1	1	2-unit 7 ch	28	LQFP-32

* For available options or further information on the device with a "*" mark, please contact the ABOV sales offices.

AC 33 M x 064 T L B
(1) (2) (3) (4) (5) (6) (7) (8)

(1)	(2)		(3)		(4)		(5)		(6)		(7)		(8)	
Head	**CPU type**		**Application**		**# of pins**		**Memory size**		**Operation. Temp.(℃)**		**PKG type**		**Material**	
AC	33	Cortex-M3	M	Motor	3	32pin	064	64KB	T	105	L	LQFP	B	Green product
					4	48pin								

Figure 105. Meaning of Product Code

19 Development tools

This chapter introduces wide range of development tools for AC33Mx064T. ABOV offers software tools, debuggers, and programmers to help a user in generating right results to match target applications. ABOV supports entire development ecosystem of the customers.

19.1 Compiler

ABOV semiconductor does not provide any compiler for AC33Mx064T. However, since AC33Mx064T have ARM's high-speed 32-bit Cortex-M3 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.

19.2 Debugger

The A-Link and A-Link Pro support ABOV Semiconductor's AC33Mx064T 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 106. 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.

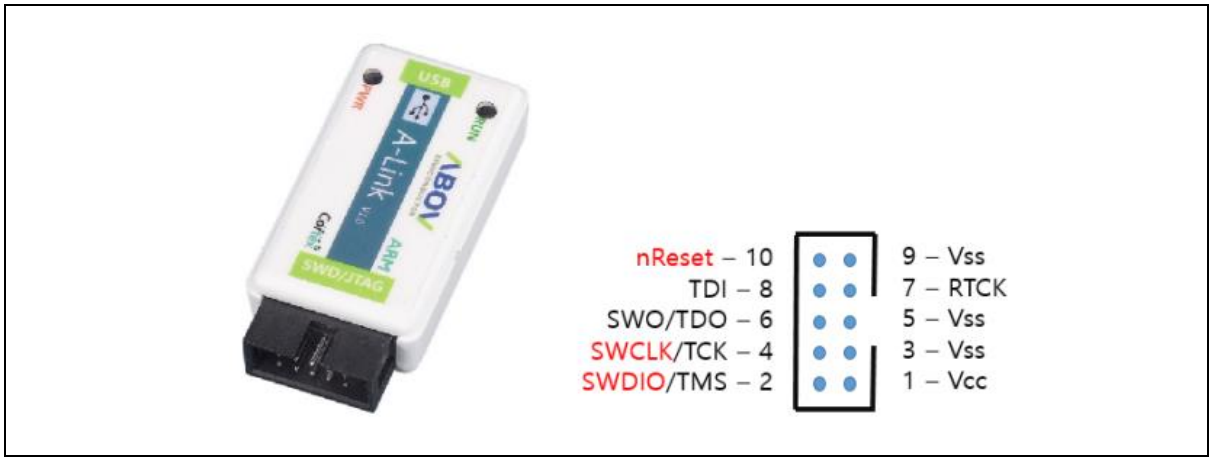


Figure 106. A-Link and Pin Descriptions

19.3 Programmer

19.3.1 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

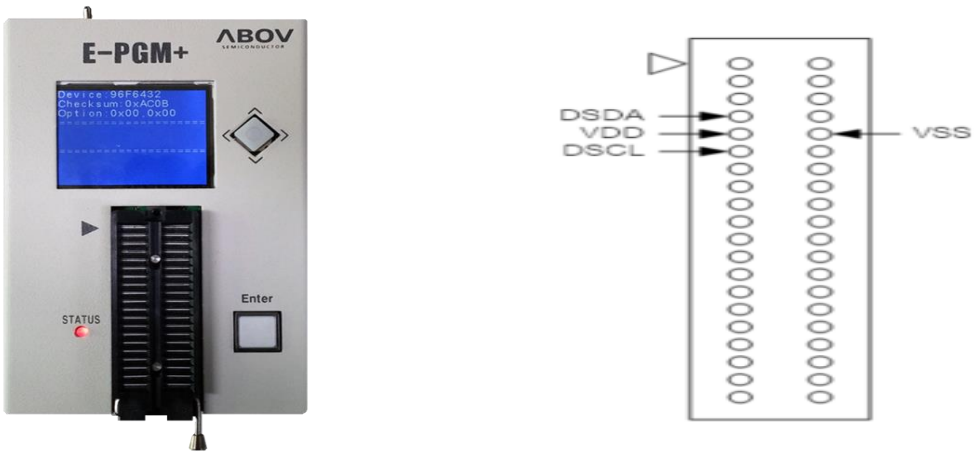


Figure 107. E-PGM+ (Single Writer) and Pin Descriptions

19.3.2 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 108. E-Gang4 and E-Gang6 (for Mass Production)

Revision history

Date	Version	Description
2014/10/06	0.1	File created
2014/10/24	0.1.1	MPWM and ADC revised
2014/10/24	0.1.2	Micellanous update
2014/10/24	0.1.4	DMA control register updated. Peripehral select table update
2015/1/25	0.1.6	Typo error correction
2015/2/2	0.1.7	Pin assign and pin map changed
2015/3/3	0.1.8	Typo error correction
2015/3/6	0.1.9	ADC: Input corrected (AN0~AN10) Timers: Timing diagrams are added. MPWM: Timing diagrams are revised.
2015/3/6	0.1.10	Correction of NMIR explanation, Addition of Revision ID(CHIPID2) Renumbering of Figures.
2015/3/9	0.1.11	The polarity of PSHRT bit in MP.DTR register was inverted. Description of MPWM wave form was revised. PROTKEY value in PSR register was corrected.
2015/6/4	0.1.12	ADC: Correct Input ch number (8-ch→7-ch) .
2015/6/23	0.1.13	LQFP-32 Package was added. The timing charts of MPWM deadtime were added.
2015/6/25	0.1.14	The explanation of SCU was modified. Clock, reset diagram and POR were added. Power down mode was removed.
2015/12/15	0.1.15	The explanations of FM.MR and MPWM duty were modified. The spec of VOL and VOH were changed.
2016/03/24	1.0	Clock configuration procedure was modified.
2016/07/19	1.1	Added debounce logic description in PCU Added MPWM note
2016/8/3	1.1.1	Typo error correction
2016/10/27	1.1.2	Modified MP.Duty, MP.SR, MP.OLR Explanation . Modified figure of MPWM functional description.
2017/4/6	1.1.3	Description of SMR contects was corrected PCC.MR's PC11(BOOT) reset value changed.
2017/8/8	1.2.0	Package figure changed
2017/8/22	1.2.1	Package figure Modified
2017/12/20	1.2.2	Modified description of STSEL in ADC.MR register. Added description of CLKDIV in ADC.CCR register
2018/1/11	1.2.3	Modified description of Un.IDTR register (UART) Modified description of DMAEN bit in And.MR register (ADC)
2018/2/21	1.2.4	Added meaning of product code
2020/06/10	1.00	1st creation (PMO)
2021/10/1	1.01	Modified description of PLL 80MHz -> 48MHz. Added meaning of Figure 11. Clock Tree Configuration
2023/1/16	1.02	Changed style and font
2023/2/03	1.03	Modified the Table 12 GPIO Alternative Function Added NOTE (ADn.CCR EXTCLK) Added NOTE (ADn.CCR CLKDIV [6:0])
2023/7/21	1.04	Modified the Table 6 RINGOSC 1MHz (±50%) Modified the Table 56. Recommended Operating Condition Added Min/Max OSC_{INT}

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