AC30M1x64, AC30M1x32
User's Manual

32-bit Cortex-M0 based
Programmable Motor Controller

User's Manual Version 1.01

Introduction

AC30M1x64/1x32 is a 32-bit microcontroller for applications that require large processing capabilities, including motor applications. The MCU is equipped with ARM's 32-bit Cortex-M0 core, and has a built-in peripherals for a variety of applications such as a motor control.

AC30M1x64/1x32 provides 3-phases PWM generator units suitable for an inverter motor drive system. The built-in 3-phases PWM generator controls one inverter motor. One 12-bit high speed ADC unit with 12-channel analog multiplexed inputs support to get feedback information from the motor. It can control up to one inverter motor.

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.

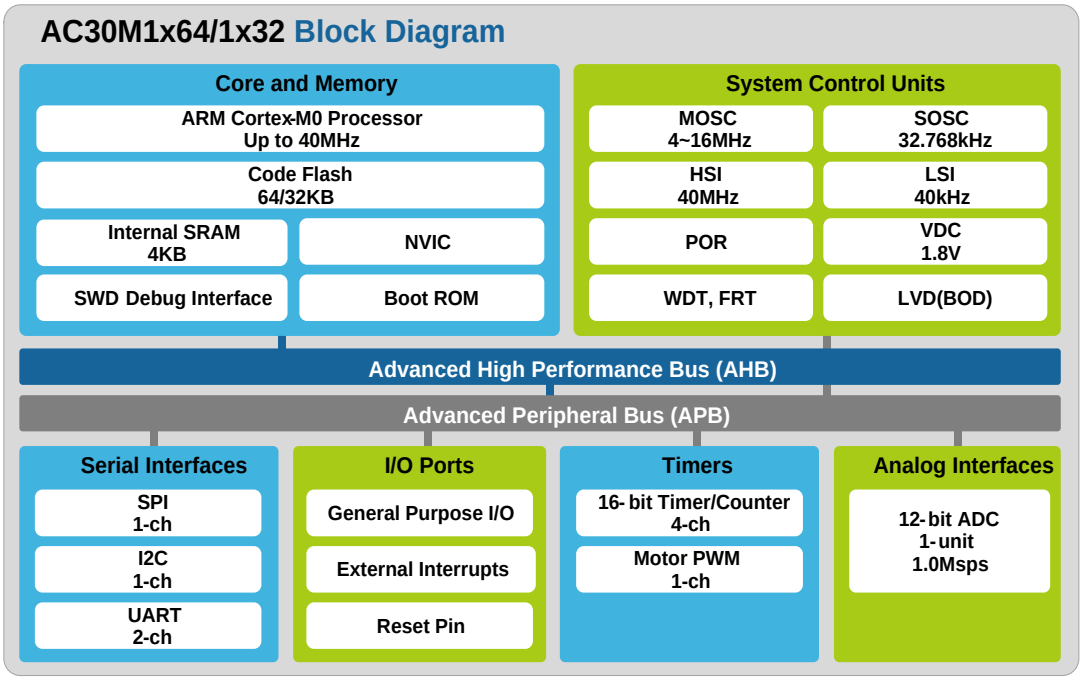


Figure 1. AC30M1x64/AC30M1x32

Reference document

- Document DDI0432C is provided by ARM, and provides information of Cortex-M0.
- AC30M1x64/32 Datasheet is provided by ABOV, and available at www.abovsemi.com.

Contents

List of figures

List of tables

1. Description

ARM powered Cortex-M0 Core is based on ARMv6M architecture, which is optimized for small size and low power system.

1.1 Device overview

In this section, features of AC30M1x64/AC30M1x32 series and peripheral counts are introduced.

Table 1. AC30M1x64/AC30M1x32 Series Features and Peripheral Counts

Peripherals		Description
Core	CPU	Maximum operating frequency: 40MHz 32-bit ARM Cortex-M0 CPU
	Interrupt	NVIC (Nested-Vectored Interrupt Controller)
Memory	Code flash	Capacity: 64Kbytes code flash memory 32Kbytes code flash memory Endurance: 10,000 cycles times at room temperature Retention: 10 years
	BOOT ROM	UART, SPI boot modes In-system programming
	SRM	4 KB
System Control Unit (SCU)	Operating frequency	40kHz ~ 40MHz External 32.768kHz crystal
	Clock	On-Chip RC-Oscillator HSI: 40MHz(±3% @-40 ~ +105℃) LSI: 40kHz(±20% @-40 ~ +105℃) XTAL OSC Fail monitoring Sub-Active mode System used external 32.768kHz crystal or system used internal 40kHz LSI
System Control Unit (SCU)	Clock monitoring	System Fail-Safe function by Clock Monitoring
	Operating mode	IDLE mode STOP1 mode STOP2 mode

Table 1. AC30M1x64/AC30M1x32 Series Features and Peripheral Counts (continued)

Peripherals		Description
System Control Unit (SCU)	Reset	nRESET pin reset Core reset Software reset POR (Power On Reset) LVR (Low Voltage Reset) WDTR (Watch Dog Timer Reset) Reset due to clock oscillating error
	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 (PA[15:0], PB[7:0], PC[15:0], PD[3:0]): 48-Pin 30Ports (PA[9:0], PB[7:0], PC[1:0], PC[8:7], PC[15:10], PD[3:2]): 32-Pin
TIMER	16-bit Timer	4 channels
	FRT	32-bit free-run timer 1 channels
	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
DIV64		3-Phase Hardware Divider (DIV64)
12-bit A/D Converter	ADC	3-Phase Motor PWM with ADC triggering function 12 analog input channels (48 PIN) 10 analog input channels (32 PIN)
Operating Voltage		2.2V to 5.5V
Operating temperature		Commercial grade (-40℃ to +105℃)
Package		Three types of package options 48-pin LQFP 32-pin LQFP 32-pin QFN

1.2 Block diagram

In this section, the AC30M1x64/AC30M1x32 series with peripherals is described in block diagram.

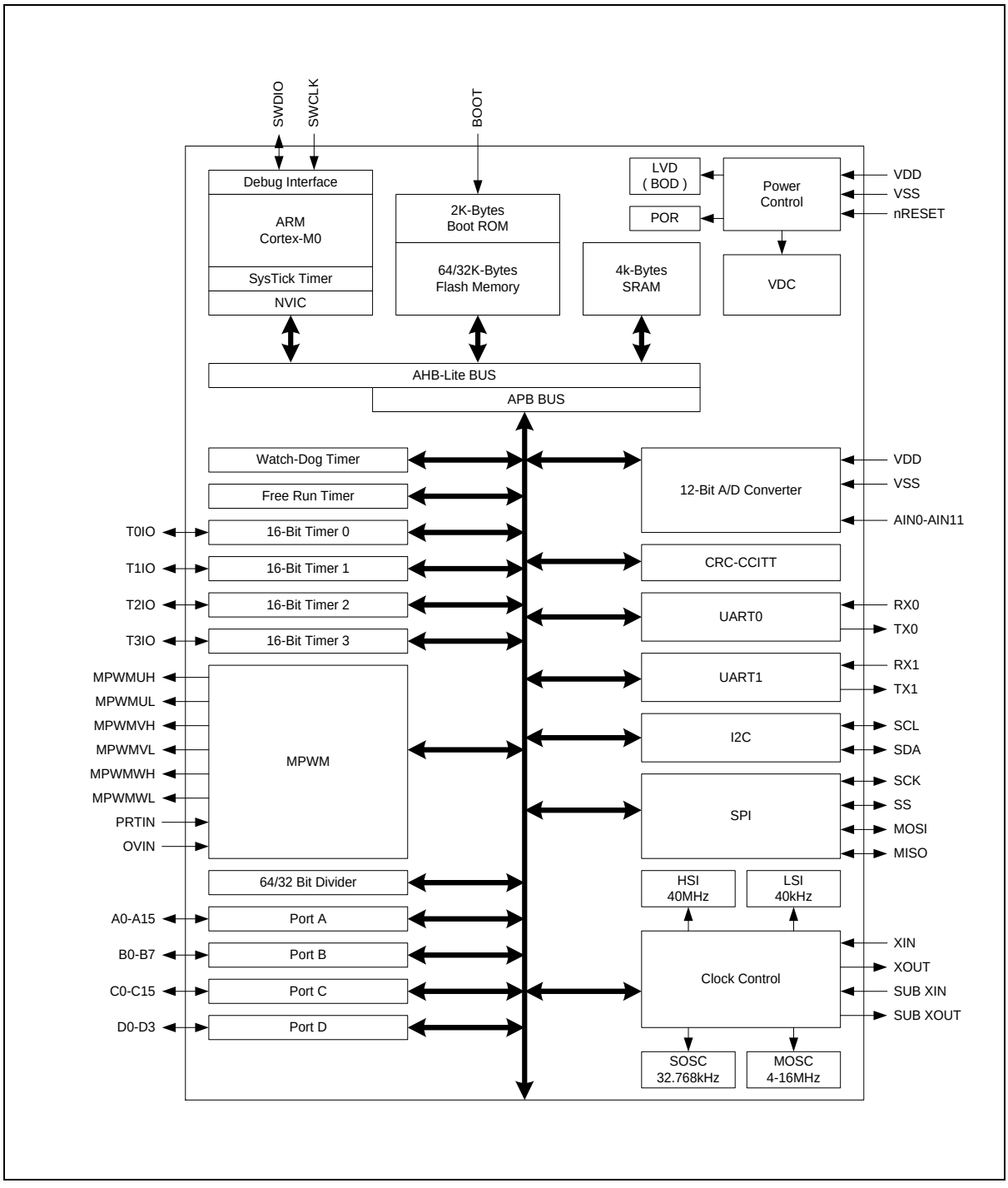


Figure 2. AC30M1x64/AC30M1x32 Block Diagram

2. Pinouts and pin descriptions

In this chapter, pinouts and pin descriptions of the AC30M1x64/AC30M1x32 series are introduced.

2.1 Pinouts

2.1.1 AC30M1464LBN (LQFP-48)

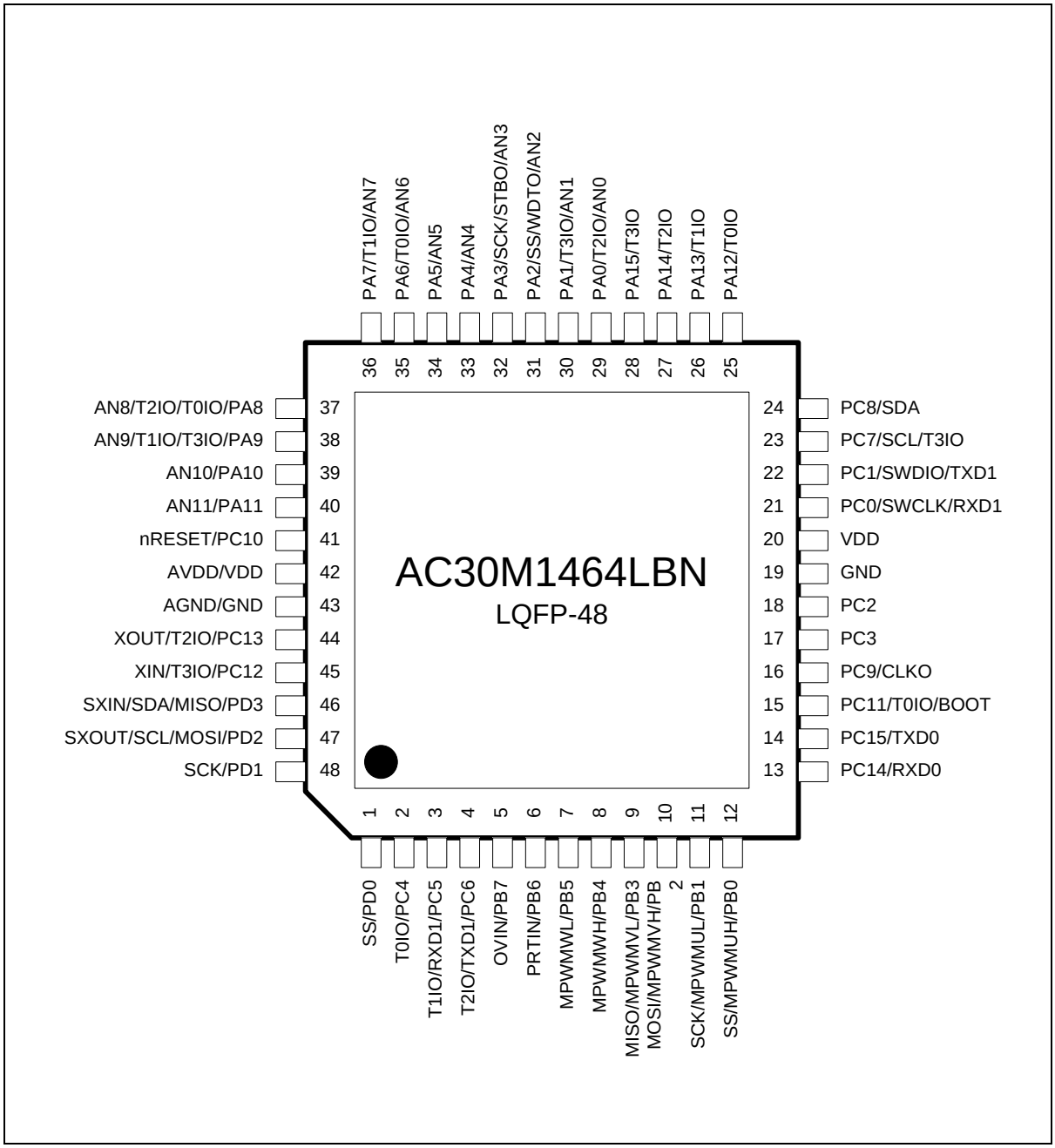


Figure 3. LQFP 48 Pinouts

2.1.2 AC30M1364LBN/AC30M1332LBN (LQFP-32)

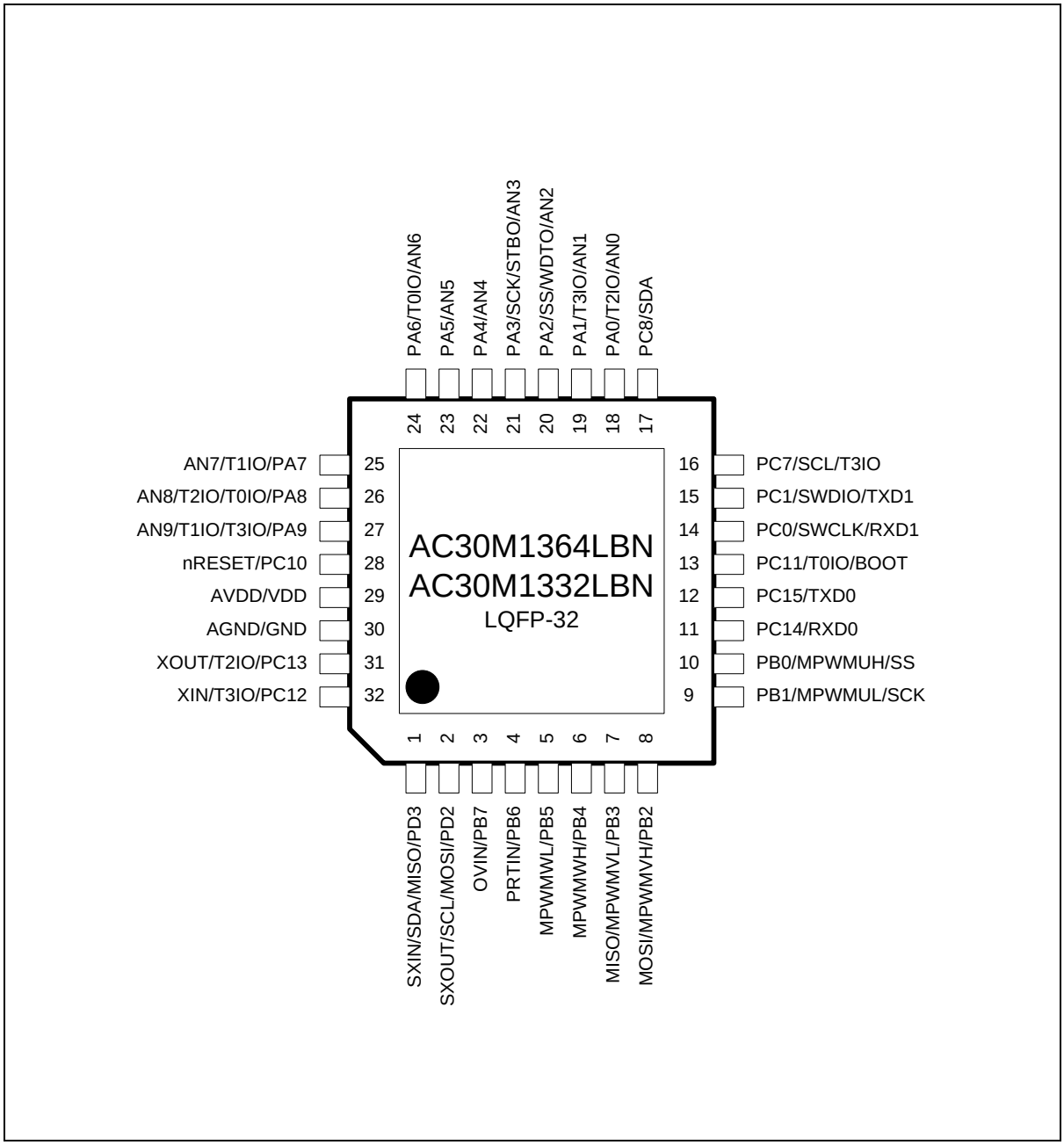


Figure 4. LQFP 32 Pinouts

2.1.3 AC30M1364UB/AC30M1332UB (QFN-32)

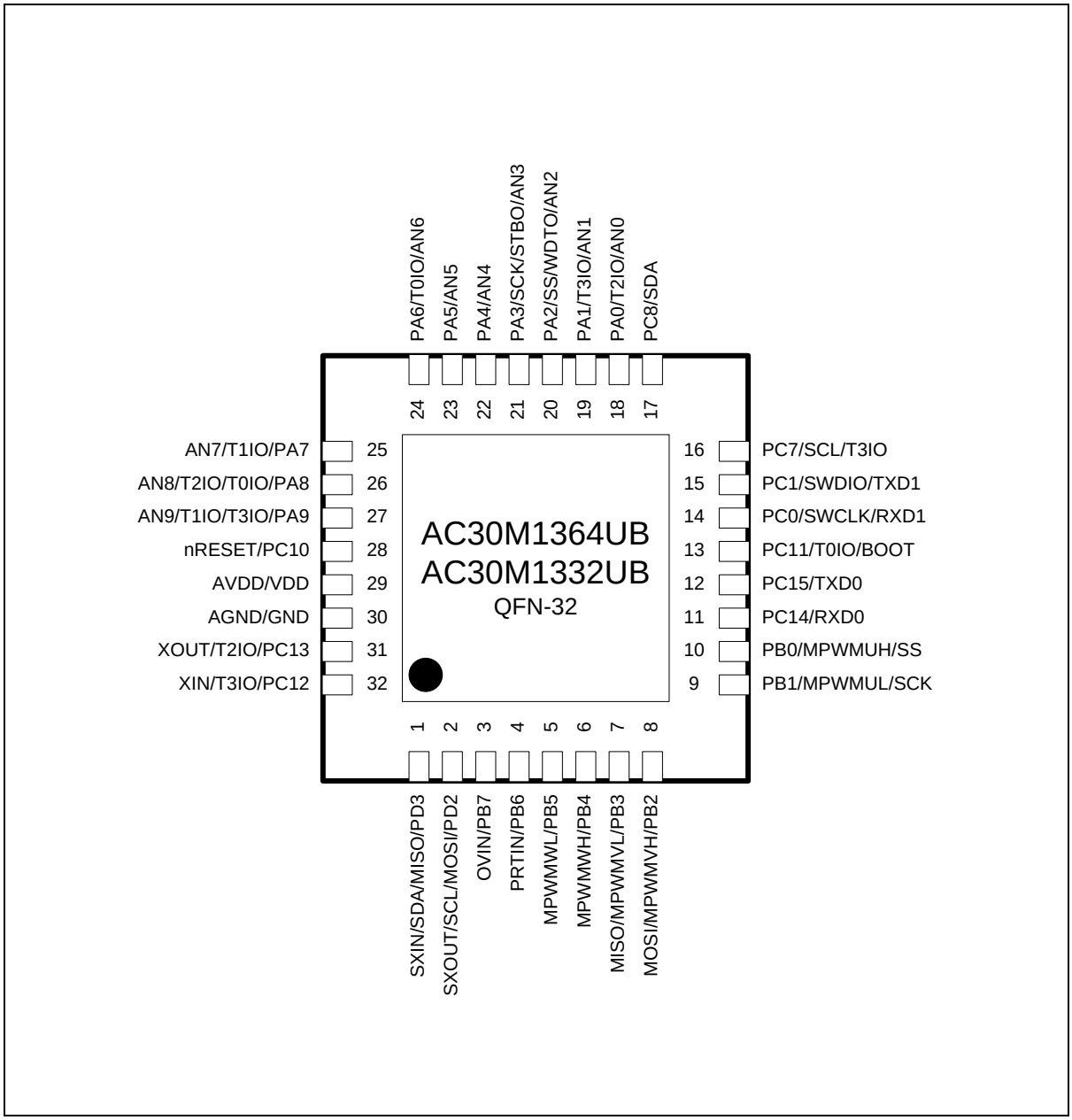


Figure 5. QFN 32 Pinouts

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 number		Pin name	Type	Description	Remark
LQFP48	LQFP32 QFN32				
1	-	PD0	IOUS	PORT D Bit 0 Input/Output	
		SS	I/O	SPI Channel Slave Select In/Out	
2	-	PC4	IOUS	PORT C Bit 4 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
3	-	PC5	IOUS	PORT C Bit 5 Input/Output	
		RXD1	I	Uart RXD1 Input	
		T1IO	I/O	Timer 1 Input/Output	
4	-	PC6	IOUS	PORT C Bit 6 Input/Output	
		TXD1	O	Uart TXD1 Output	
		T2IO	I/O	Timer 2 Input/Output	
5	3	PB7	IOUS	PORT B Bit 7 Input/Output	
		OVIN	I	PWM Over-voltage input signal	
6	4	PB6	IOUS	PORT B Bit 6 Input/Output	
		PRTIN	I	PWM Protection Input signal	
7	5	PB5	IOUS	PORT B Bit 5 Input/Output	
		MPWMWL	O	MPWM WL Output	
8	6	PB4	IOUS	PORT B Bit 4 Input/Output	
		MPWMWH	O	MPWM WH Output	
9	7	PB3	IOUS	PORT B Bit 3 Input/Output	
		MPWMVL	O	MPWM VL Output	
		MISO	I/O	SPI Channel Master In / Slave Out	
10	8	PB2	IOUS	PORT B Bit 2 Input/Output	
		MPWMVH	O	MPWM VH Output	
		MOSI	I/O	SPI Channel Master Out / Slave In	
11	9	PB1	IOUS	PORT B Bit 1 Input/Output	
		MPWMUL	O	MPWM UL Output	
		SCK	I/O	SPI Channel CLK In / Out	

Table 2. Pin Description (continued)

Pin number		Pin name	Type	Description	Remark
LQFP48	LQFP32 QFN32				
12	10	PB0	IOUS	PORT B Bit 0 Input/Output	
		MPWMUH	O	MPWM UH Output	
		SS	I/O	SPI Channel Slave Select In / Out	
13	11	PC14	IOUS	PORT C Bit 14 Input/Output	
		RXD0	I	Uart RXD0 Input	
14	12	PC15	IOUS	PORT C Bit 15 Input/Output	
		TXD0	O	Uart TXD0 Output	
15	13	PC11	IOUS	PORT C Bit 11 Input/Output	
		BOOT	IU	Boot mode Selection Input	Pull-up
		T0IO	I/O	Timer 0 Input/Output	
16	-	PC9	IOUS	PORT C Bit 9 Input/Output	
		CLKO	O	System Clock Output	
17	-	PC3	IOUS	PORT C Bit 3 Input/Output	
18	-	PC2	IOUS	PORT C Bit 2 Input/Output	
19	-	GND	P	GND	
20	-	VDD	P	VDD	
21	14	PC0	IOUS	PORT C Bit 0 Input/Output	
		SWCLK	I	SWD Clock Input	Pull-up
		RXD1	I	Uart1 RXD1 Input	
22	15	PC1	IOUS	PORT C Bit 1 Input/Output	
		SWDIO	I/O	SWD Data Input/Output	Pull-up
		TXD1	O	Uart1 TXD1 Output	
23	16	PC7	IOUS	PORT C Bit 7 Input/Output	
		SCL	I/O	I^2C Channel SCL In/Out	
		T3IO	I/O	Timer 3 Input/Output	
24	17	PC8	IOUS	PORT C Bit 8 Input/Output	
		SDA	I/O	I^2C Channel SDA In/Out	
25	-	PA12	IOUS	PORT A Bit 12 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
26	-	PA13	IOUS	PORT A Bit 13 Input/Output	
		T1IO	I/O	Timer 1 Input/Output	
27	-	PA14	IOUS	PORT A Bit 14 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	

Table 2. Pin Description (continued)

Pin number		Pin name	Type	Description	Remark
LQFP48	LQFP32 QFN32				
28	-	PA15	IOUS	PORT A Bit 15 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
29	18	PA0	IOUS	PORT A Bit 0 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	
		AIN0	IA	Analog Input 0	
30	19	PA1	IOUS	PORT A Bit 1 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
		AIN1	IA	Analog Input 1	
31	20	PA2	IOUS	PORT A Bit 2 Input/Output	
		SS	I/O	SPI Channel Slave Select In / Out	
		WDTO	O	Watchdog Timer Overflow Output	
		AIN2	IA	Analog Input 2	
32	21	PA3	IOUS	PORT A Bit 3 Input/Output	
		SCK	I/O	SPI Channel CLK In / Out	
		STBO	O	Power Down Mode Output	
		AIN3	IA	Analog Input 3	
33	22	PA4	IOUS	PORT A Bit 4 Input/Output	
		AIN4	IA	Analog Input 4	
34	23	PA5	IOUS	PORT A Bit 5 Input/Output	
		AIN5	IA	Analog Input 5	
35	24	PA6	IOUS	PORT A Bit 6 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
		AIN6	IA	Analog Input 6	
36	25	PA7	IOUS	PORT A Bit 7 Input/Output	
		T1IO	I/O	Timer 1 Input/Output	
		AIN7	IA	Analog Input 7	
37	26	PA8	IOUS	PORT A Bit 8 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	
		T0IO	I/O	Timer 0 Input/Output	
		AIN8	IA	Analog Input 8	
38	27	PA9	IOUS	PORT A Bit 9 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
		T1IO	I/O	Timer 1 Input/Output	

Table 2. Pin Description (continued)

Pin number		Pin name	Type	Description	Remark
LQFP48	LQFP32 QFN32				
		AIN9	IA	Analog Input 9	
39	-	PA10	IOUS	PORT A Bit 10 Input/Output	
		AIN10	IA	Analog Input 10	
40	-	PA11	IOUS	PORT A Bit 11 Input/Output	
		AIN11	IA	Analog Input 11	
41	28	PC10	IOUS	PORT C Bit 10 Input/Output	
		nRESET	IU	External Reset Input	Pull-up
42	29	VDD	P	VDD	
43	30	GND	P	GND	
44	31	PC13	IOUS	PORT C Bit 13 Input/Output	
		T2IO	I/O	Timer 2 Input/Output	
		XOUT	OA	External Crystal Oscillator Output	
45	32	PC12	IOUS	PORT C Bit 12 Input/Output	
		T3IO	I/O	Timer 3 Input/Output	
		XIN	IA	External Crystal Oscillator Input	
46	1	PD3	IOUS	PORT D Bit 3 Input/Output	
		MISO	I/O	SPI Channel Master In / Slave Out	
		SDA	I/O	I^2C Channel SDA In/Out	
		SXIN	I	External Crystal Sub Oscillator Input	
47	2	PD2	IOUS	PORT D Bit 2 Input/Output	
		MOSI	I/O	SPI Channel Master Out / Slave In	
		SCL	I/O	I^2C Channel SCL In/Out	
		SXOUT	OA	External Crystal Sub Oscillator Output	
48	-	PD1	IOUS	PORT D Bit 1 Input/Output	
		SCK	I/O	SPI Clock Input/Output	

NOTES:

1. I=Input, O=Output, U=Pull-up, D=Pull-down, S=Schmitt-Trigger Input Type, C=CMOS Input Type, A=Analog, P=Power
2. Pin order may be changed with revision notice

3. System and memory overview

3.1 System architecture

Main system of AC30M1x64/AC30M1x32 series consists of the following parts:

- ARM Cortex-M0 core
- Internal SRAM, Flash memory

3.1.1 Cortex-M0 core

The ARM® powered Cortex-M0 Core is based on ARMv6M 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. SWD debugging features are provided. Max 40MHz operating frequency has one 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 Handler
-14	0x0000_0008	NMI Handler
-13	0x0000_000C	Hard Fault Handler
-12	0x0000_0010	MemManage Handler
-11	0x0000_0014	BusFault Handler
-10	0x0000_0018	UsageFault 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
0	0x0000_0040	LVDFAIL

Table 3. Interrupt Vector Map (continued)

Priority	Vector address	Interrupt source
1	0x0000_0044	SYSCLKFAIL
2	0x0000_0048	MOSCFAIL
3	0x0000_004C	SOSCFAIL
4	0x0000_0050	WDT
5	0x0000_0054	TIMER0
6	0x0000_0058	TIMER1
7	0x0000_005C	TIMER2
8	0x0000_0060	TIMER3
9	0x0000_0064	FRT
10	0x0000_0068	GPIOAE
11	0x0000_006C	GPIOAO
12	0x0000_0070	GPIOBE
13	0x0000_0074	GPIOBO
14	0x0000_0078	GPIOCE
15	0x0000_007C	GPIOCO
16	0x0000_0080	GPIODE
17	0x0000_0084	GPIODO
18	0x0000_0088	MPWM
19	0x0000_008C	MPWMPROT
20	0x0000_0090	MPWMOVV
21	0x0000_0094	I2C
22	0x0000_0098	SPI
23	0x0000_009C	UART0
24	0x0000_00A0	UART1
25	0x0000_00A4	ADC
26	0x0000_00A8	Reserved
27	0x0000_00AC	
28	0x0000_00B0	
29	0x0000_00B4	
30	0x0000_00B8	
31	0x0000_00BC	

NOTES:

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

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 6 shows addressable memory space in memory map.

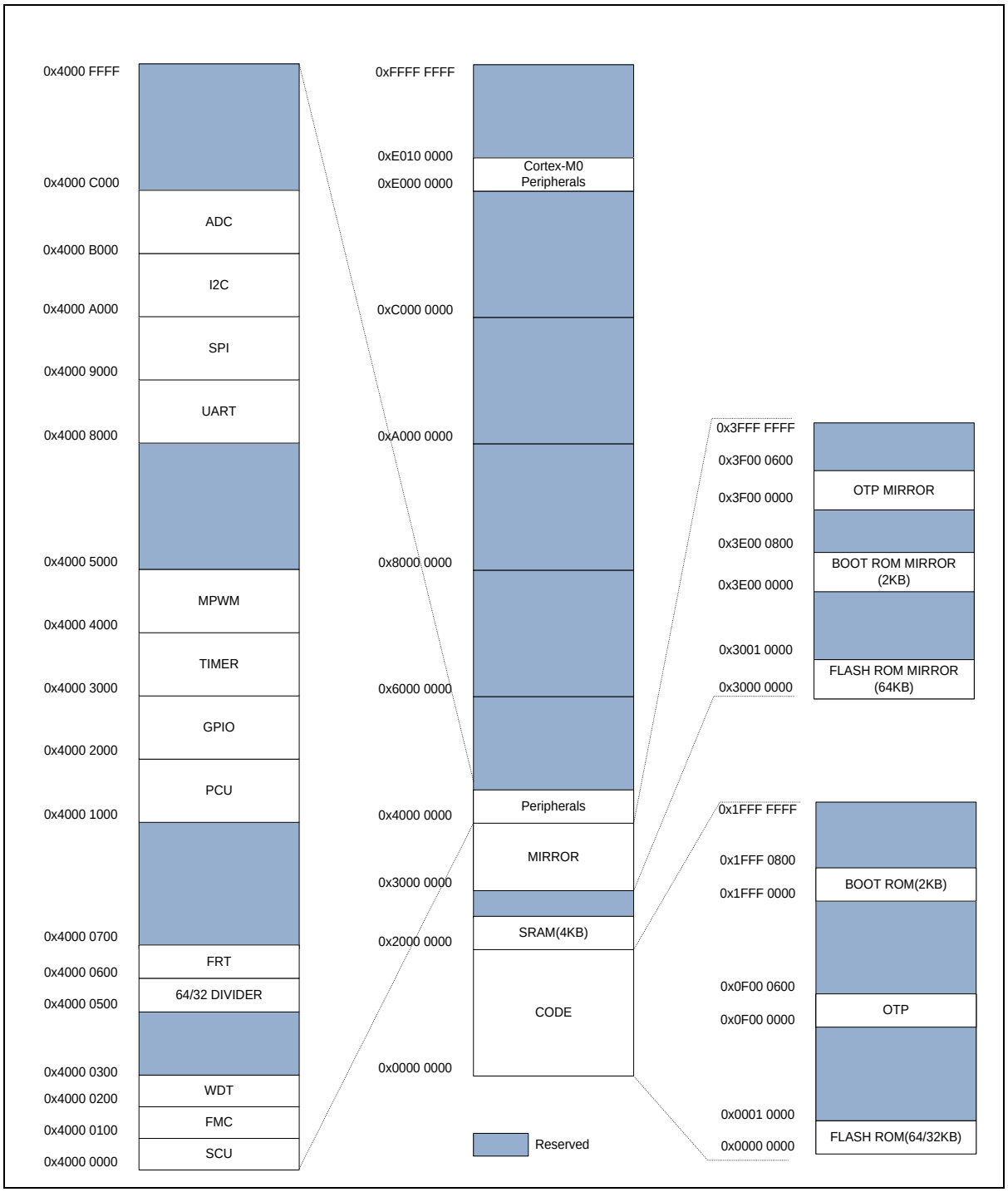


Figure 6. Memory Map

3.2.2 Embedded SRAM

On chip 4KB 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 AC30M1x64/1x32 provides internal 64/32KB 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 SWD programming is also supported in boot or debugging mode.

Instruction and data cache buffer are present and overcome the low bandwidth flash memory. The CPU can access flash memory with one wait state up to 40 MHz bus frequency.

3.3 Boot mode

3.3.1 Boot mode pins

The AC30M1x64/AC30M1x32 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
	nBOOT/PC11	I	Boot mode setting pin
UART0	RXD0/PC14	I	UART boot receive data
	TXD0/PC15	O	UART boot transmit data
SPI	SS/PA2	I	SPI Boot Slave Select
	SCK/PA3	I	SPI Boot Clock Input
	MOSI/PD2	I	SPI Boot Data Input
	MISO/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:

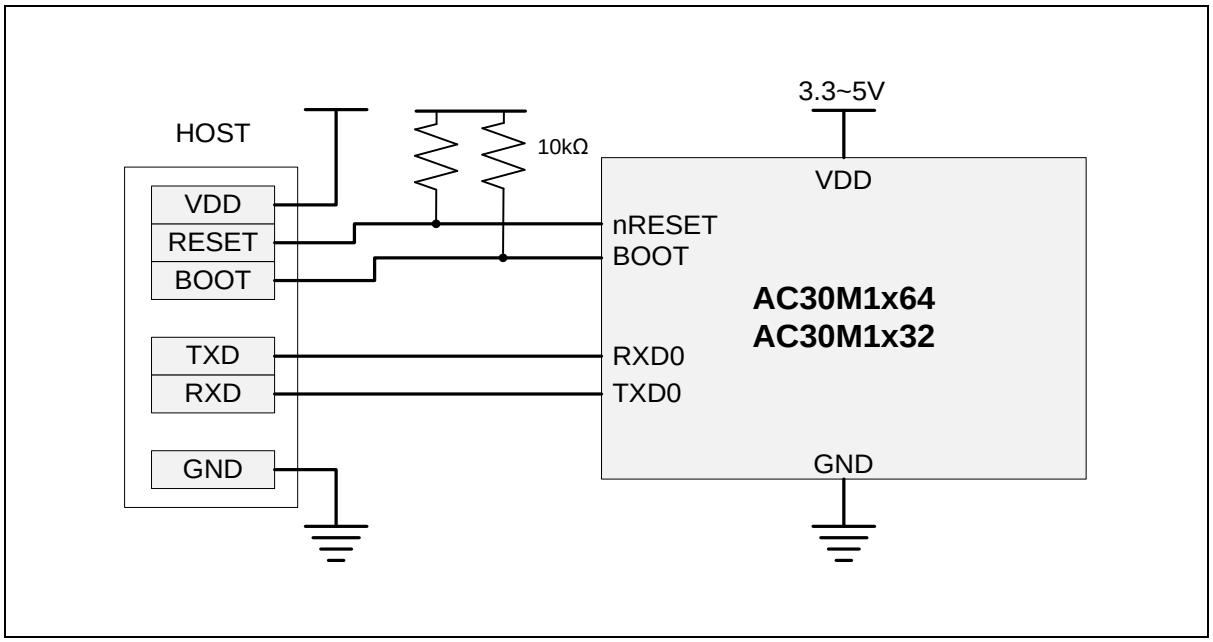


Figure 7. Connection Diagram of UART0 Boot

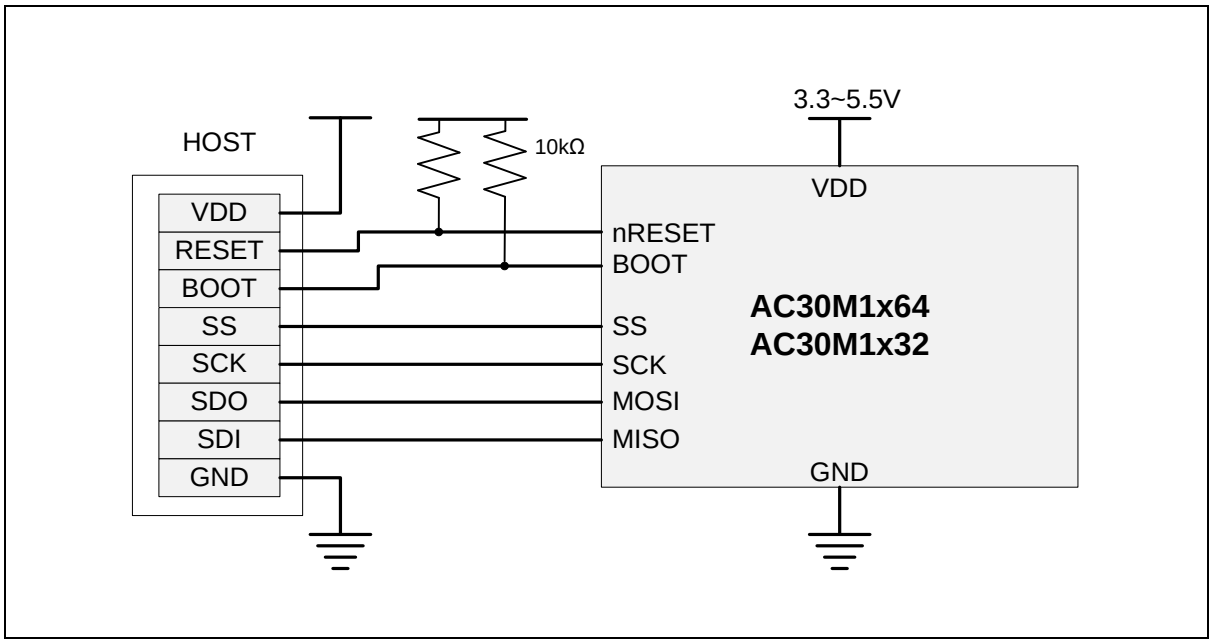


Figure 8. Connection Diagram of SPI Boot

3.3.3 SWD mode connections

Users can use SWD mode for writing with E-PGM+. This mode can be used for writing & debugging.

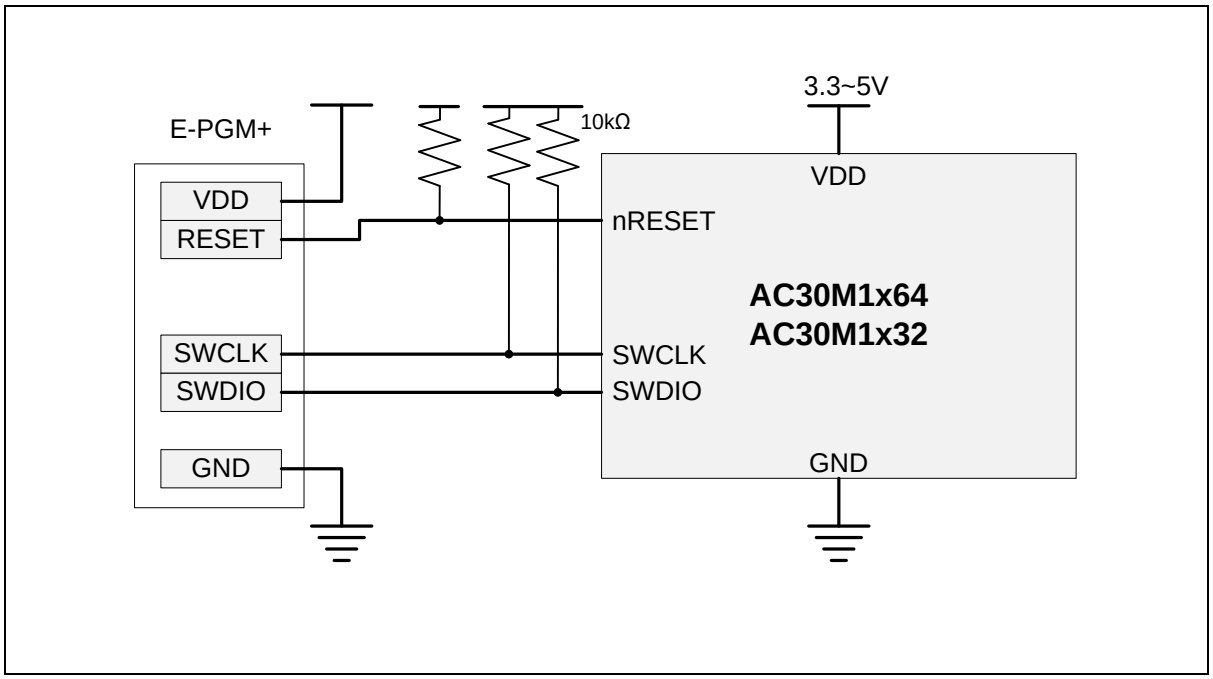


Figure 9. Connection Diagram of E-PGM+ and SWD Port

4. System Control Unit (SCU)

AC30M1x64/AC30M1x32 series has a built-in intelligent power control block which manages system analog blocks and operating modes. System control unit (SCU) block controls an internal reset and clock signals to maintain optimize system performance and power dissipation.

Four pins in Table 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
SXIN/SXOUT	OSC	External sub-Crystal Oscillator
STBO	O	Stand-by Output Signal
CLKO	O	Clock Output Monitoring Signal

4.1 SCU block diagram

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

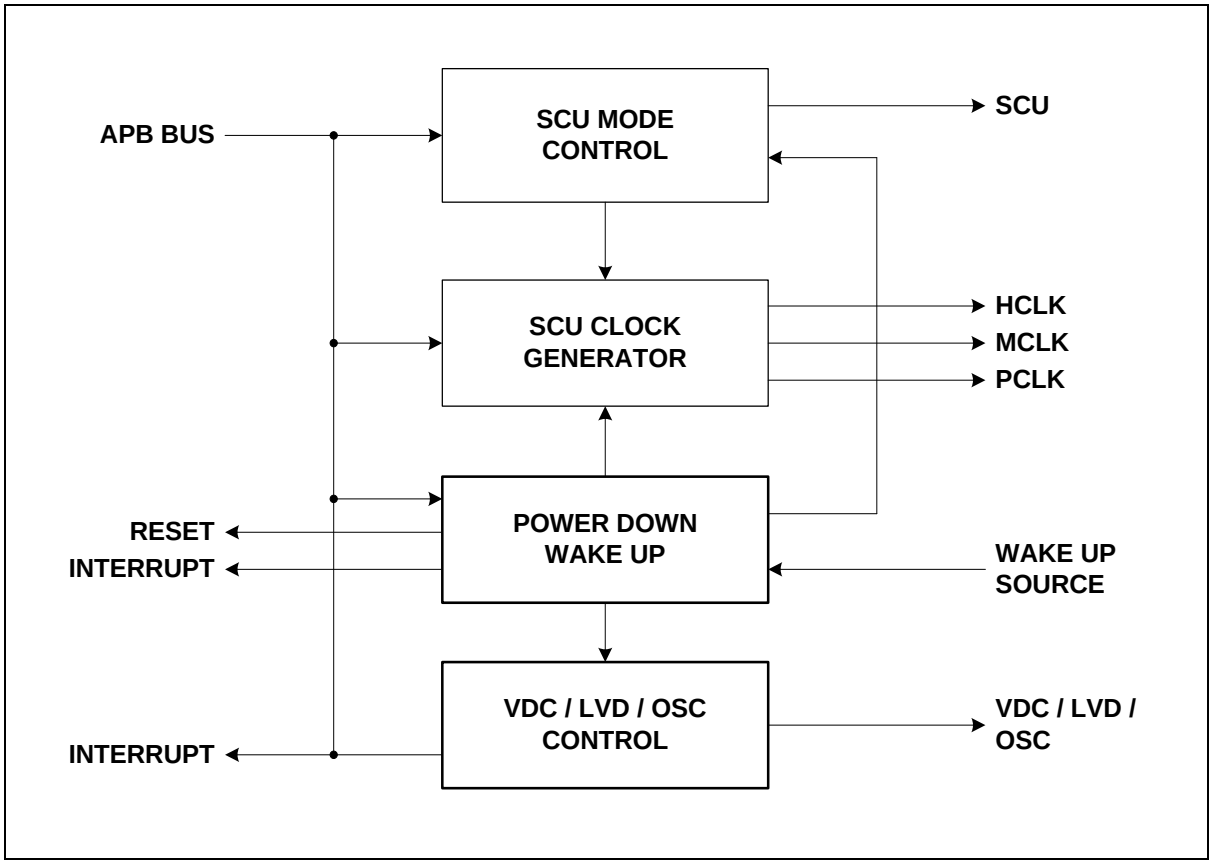


Figure 10. SCU Block Diagram

4.2 Clock system

AC30M1x64/AC30M1x32 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 learn about the clock system of AC30M1x64/AC30M1x32 devices and clock sources.

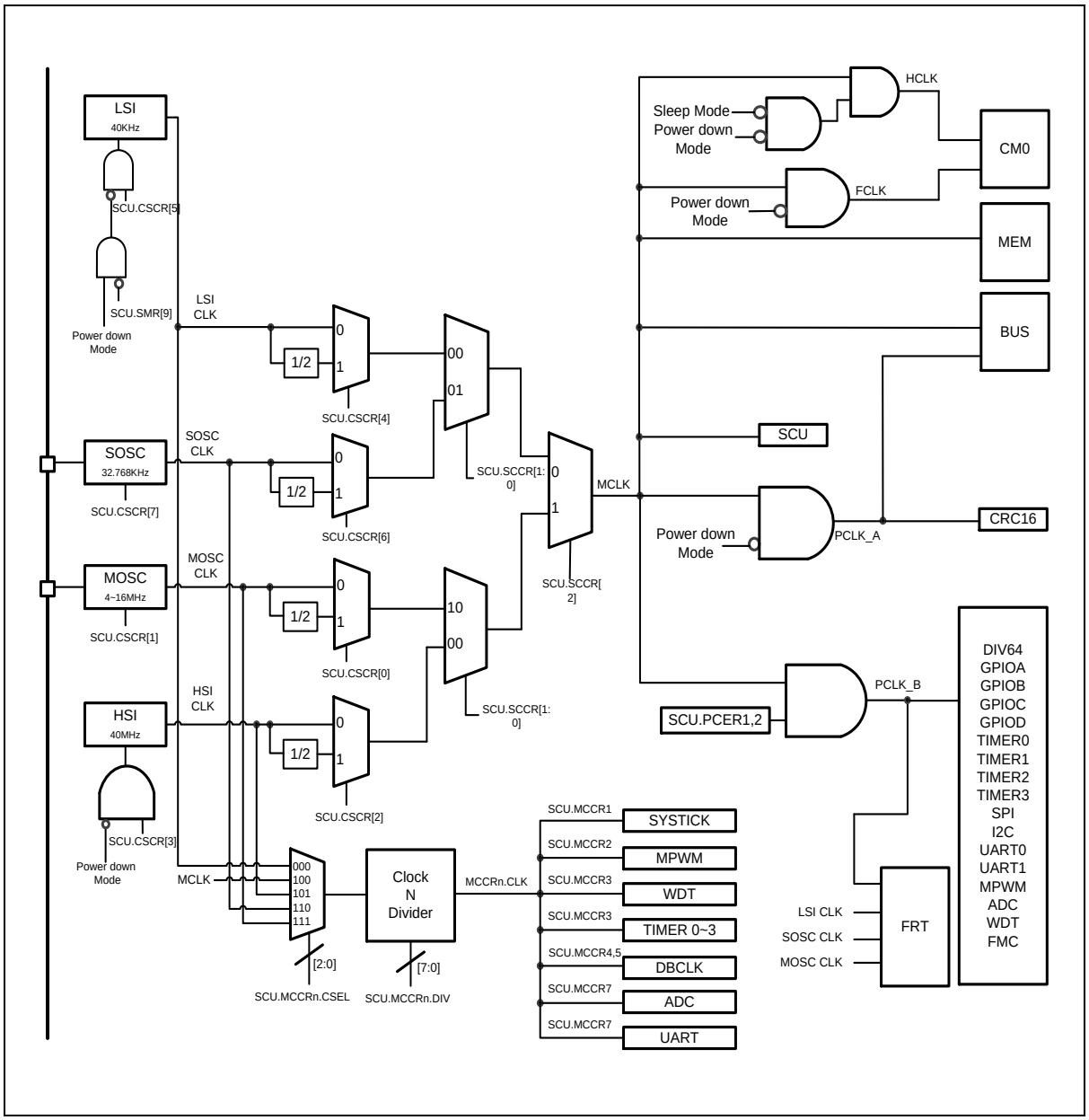


Figure 11. Clock Tree Configuration

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

Table 6. Clock Sources

Clock name	Frequency	Description
MOSC	4-16 MHz	External Crystal OSC
SOSC	32.768 kHz	External Sub Crystal OSC
HSI	40 MHz	High Speed Internal OSC
LSI	40 kHz	Low Speed Internal OSC

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-M0 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 and power down mode.

BUS system and memory systems operated by MCLK clock. Max bus operating clock speed is 40MHz.

4.2.3 PCLK clock domain

PCLK_B is master clock of all the peripheral. Each peripheral clocks enabled by SCU.PCER1 and SCU.PCER2 registers can be used by each peripheral. Before enabling the PCLK_B input clock of each block, it can't be accessible even reading its registers. In the case of FRT, various clocks can be used. But CRC16 uses PCLK_A. It can be stopped in power down mode.

4.2.4 Clock configuration procedure

After power up, the default system clock is feed by LSI (40kHz) clock. LSI is default enabled at power up sequence. The other clock sources will be enabled by user controls with the LSI system clock.

HSI (40MHz) clock can be enabled by SCU.CSCR register.

MOSC (4-16MHz) clock can be enabled by SCU.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 – PCC.MR and PCC.CR registers should be configured properly. After enabling the MOSC block, you must wait for more than 5msec time to ensure stable operation of crystal oscillation.

SOSC (32.768kHz) clock can be enabled by SCU.CSCR register. Before enable SOSC block, the pin mux configuration should be set for SXIN, SXOUT function. PD3 and PD2 pins are shared with SOSC's SXIN and SXOUT function – PCD.MR and PCD.CR registers should be configured properly. After enabling the SOSC block, you must wait for more than 10msec time to ensure stable operation of crystal oscillation.

You can change the MCLK by configuring the SCU.SCCR register. In addition, you can find an example flow chart configuring the system clock in Figure 12.

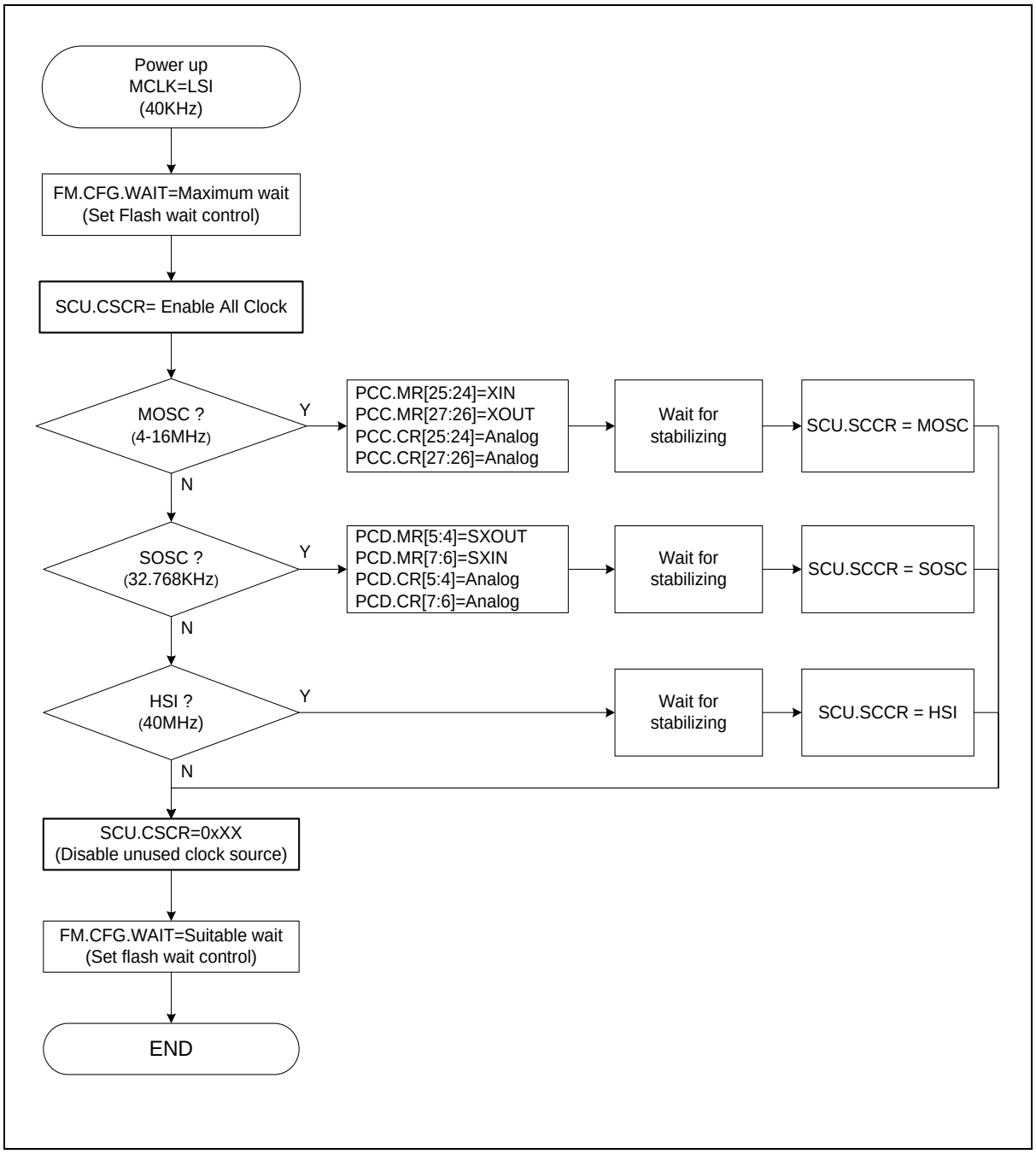


Figure 12. Clock Change Procedure

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

Table 7. Flash Wait Control Recommendation

FMCONF.WAIT	FLASH access wait	Available max. system clock frequency
00	0-clock wait	Up to 20MHz
01	1-clock wait	Up to 40MHz
11	2-clock wait	Up to 40MHz

4.3 Reset

The AC30M1x64/AC30M1x32 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 CPU Lockup 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 VDDEXT power is turn on. Internal POR trigger level is 1.4V of VDDEXT voltage out level. At this time, boot operation is started. The LSI clock is enabled and counts 4.25msec time for internal VDC level stabilizing. In this time, VDDEXT voltage level should be over than initial LVD level (1.65V). After 4.25msec counting, the cold reset is released and counts 0.4msec time for warm reset synchronizing. After released cold and warm reset, BOOTROM and CPU are running.

Figure 13 shows the power-up process and the initial reset waveforms.

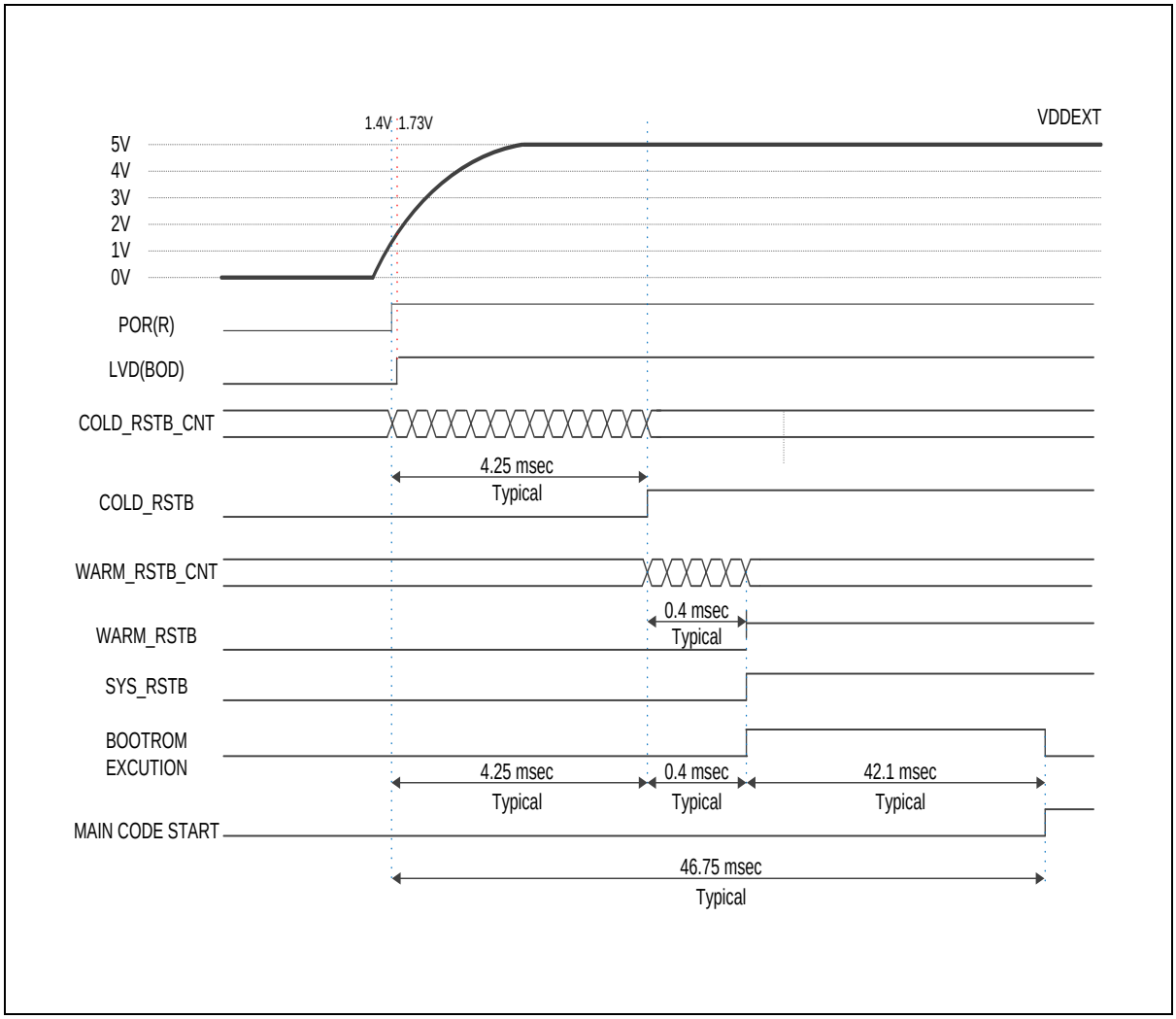


Figure 13. Power-up Procedure

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 SCU.RSER register and the status is appeared in SCU.RSSR register. The reset for each peripheral blocks is controlled by SCU.PRER register. The reset can be masked independently.

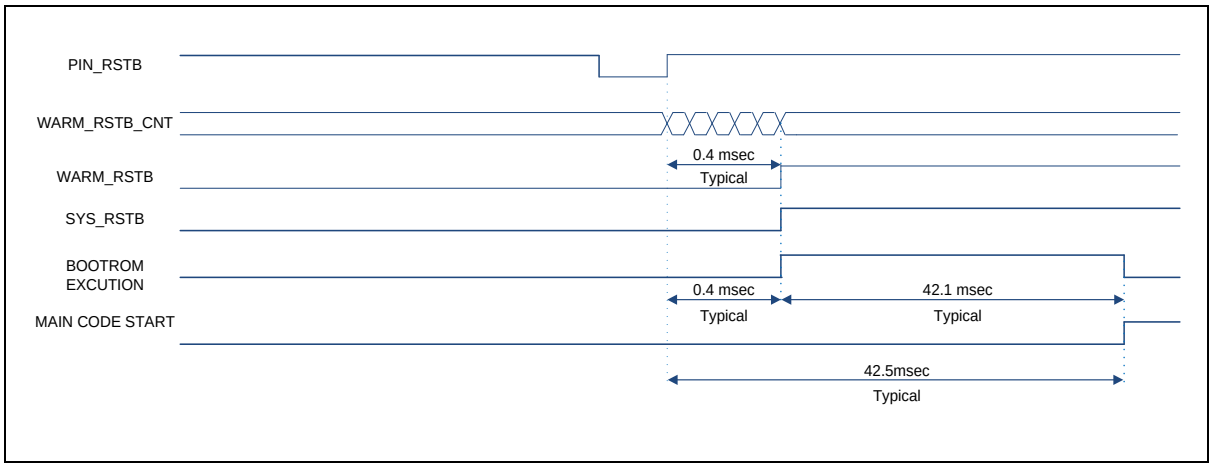


Figure 14. Warm Reset Diagram

4.3.3 LVR reset

An LVR event is triggered when the operating voltage drops below a certain level during the MCU's operation. A user can choose to set the MCU to perform a reset or an interrupt when an LVR event is triggered. This low voltage reset is a warm reset. See the description on warm resetting for details.

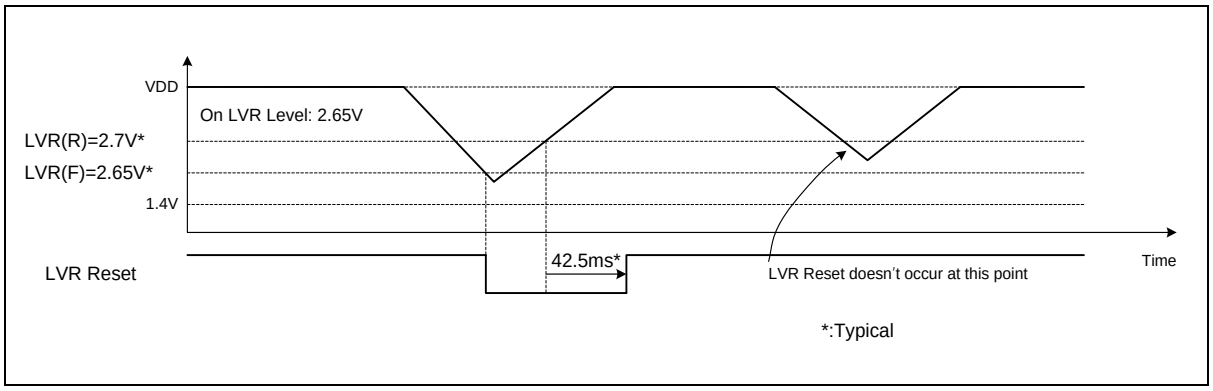


Figure 15. LVR Reset Timing Diagram

4.3.4 Reset tree

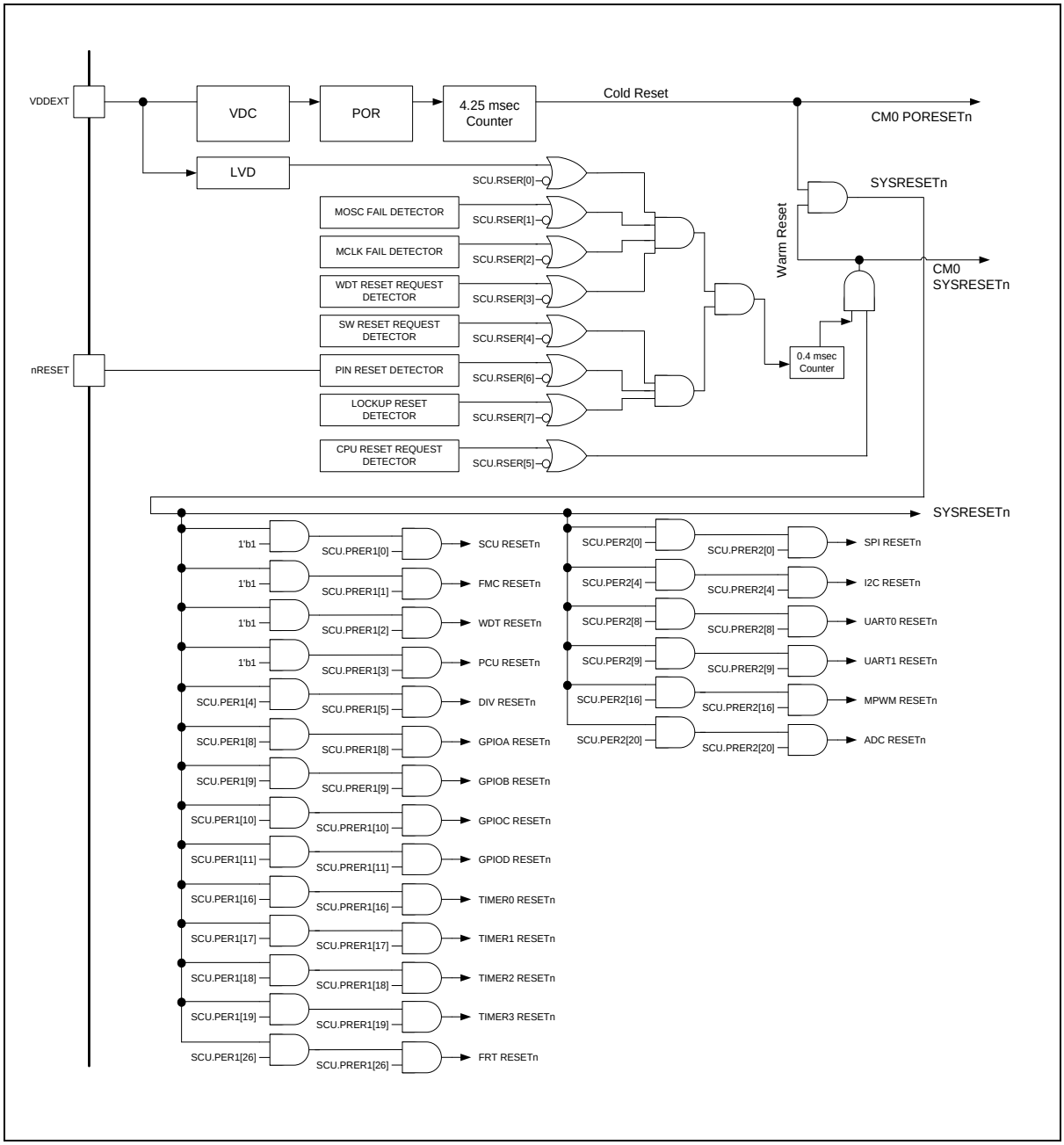
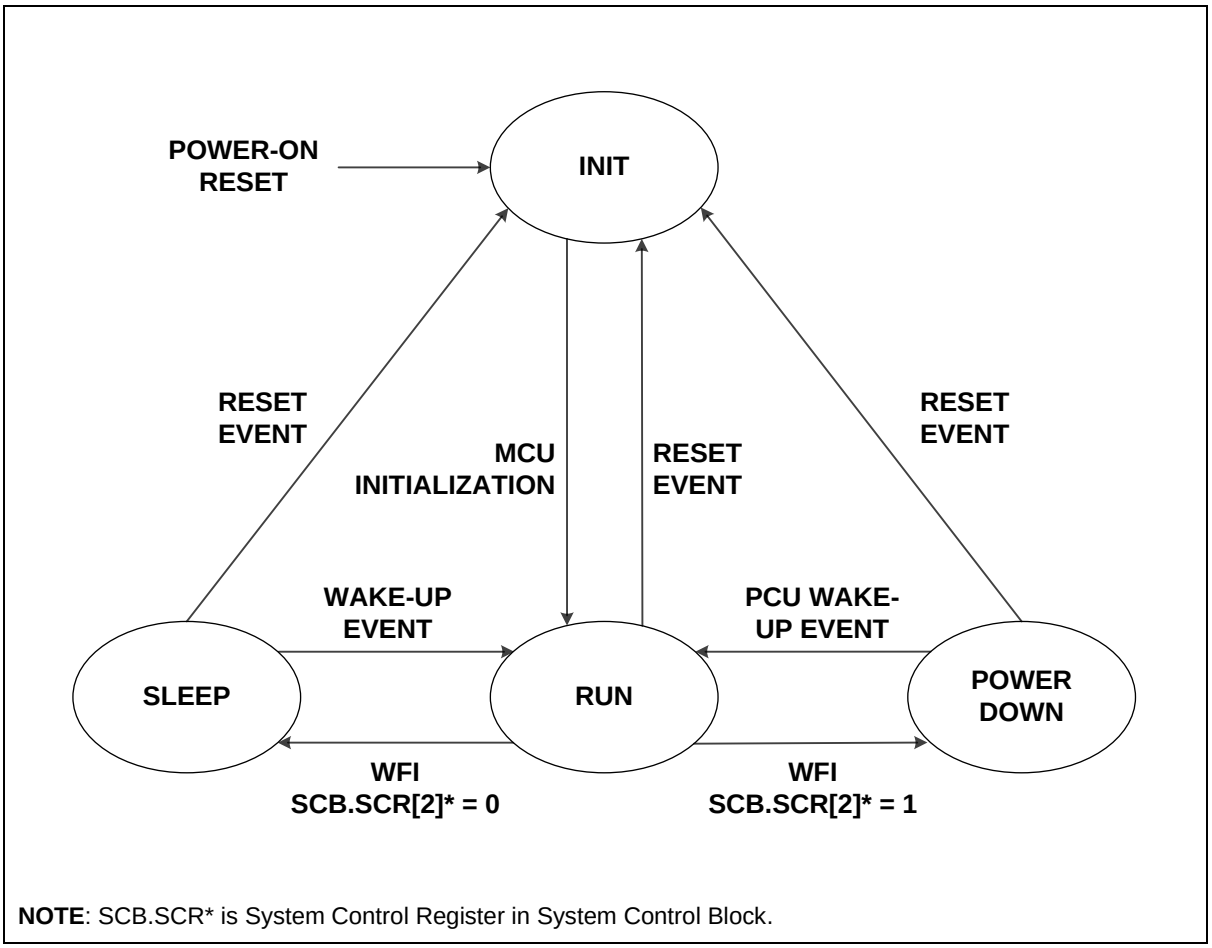


Figure 16. Reset Tree Configuration

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 and the PD mode can be used as the low power consumption mode. The low power consumption is achieved by halting processor core and unused peripherals.

Figure 17 shows the operation mode transition diagram.



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

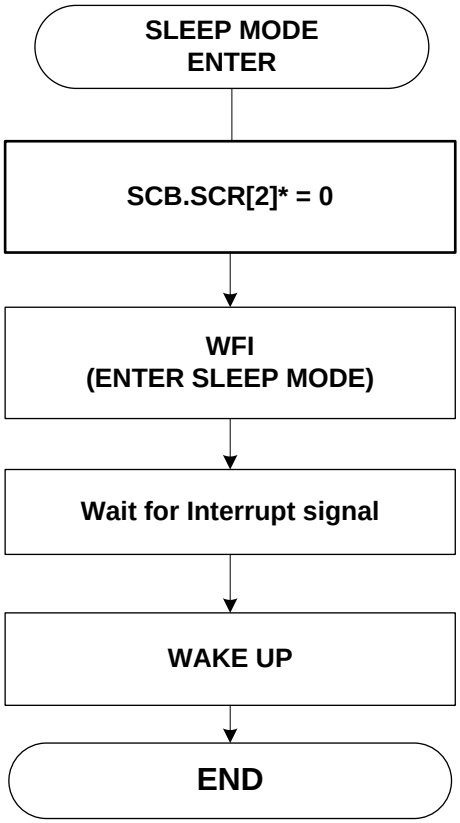
Figure 17. Transition between Operation Modes

4.4.1 RUN mode

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

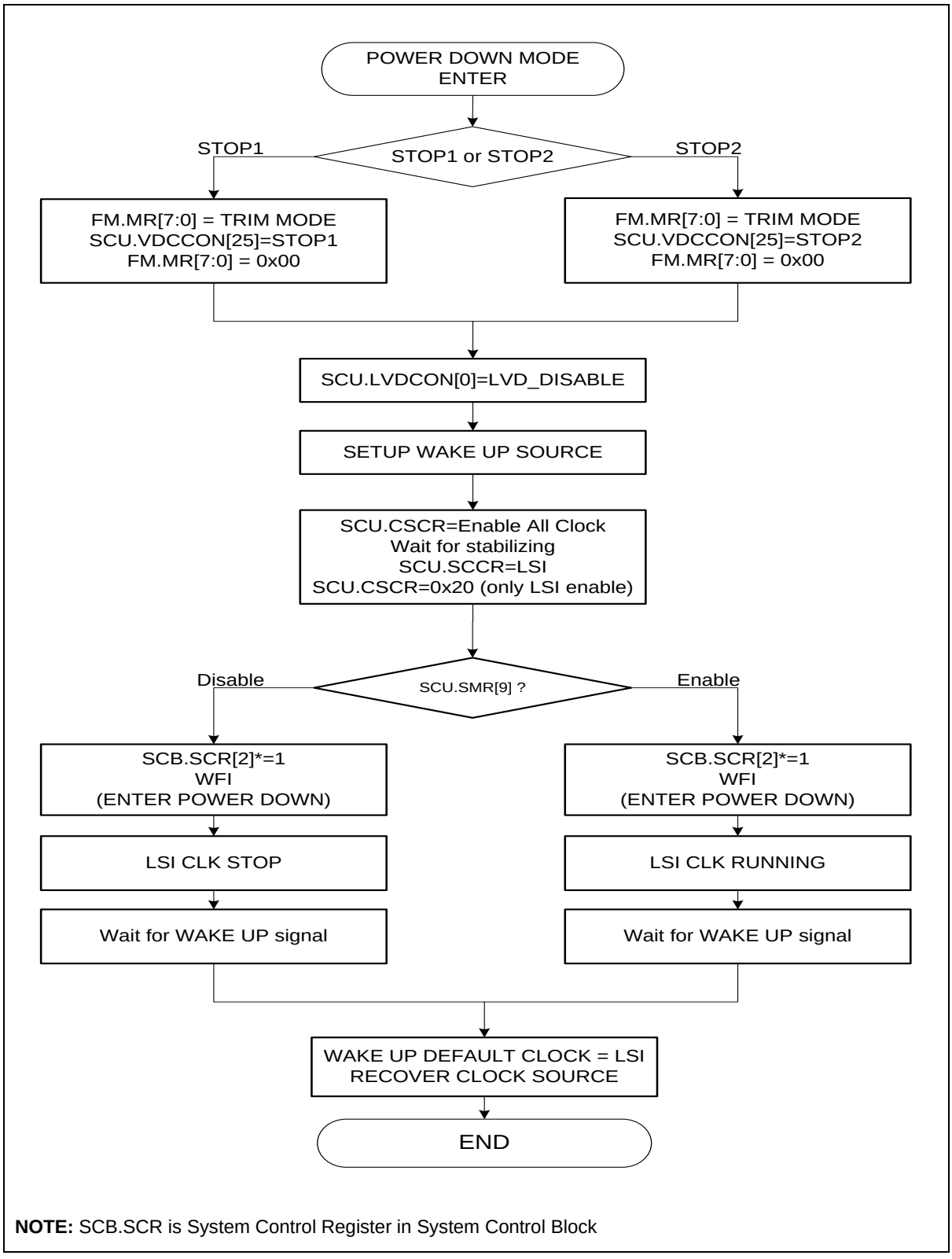


NOTE: SCB.SCR* is System Control Register in System Control Block

Figure 18. SLEEP Mode Operation Sequence

4.4.3 POWER DOWN mode

All the internal circuits are entered the stop state. Power down operation has special power off sequence as below picture.



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

Figure 19. Power Down Mode Operation Sequence

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
SMR	0x0004	RW	System Mode Register	0000_0000	4.5.1
SRCR	0x0008	RW	System Reset Control Register	0000_0000	4.5.2
WUER	0x0010	RW	Wake up source enable register	0000_0000	4.5.3
WUSR	0x0014	RO	Wake up source status register	0000_0000	4.5.4
RSER	0x0018	RW	Reset source enable register	0000_0049	4.5.5
RSSR	0x001C	RW	Reset source status register	0000_0080*	4.5.6
PRER1	0x0020	RW	Peripheral reset enable register 1	040F_0F2F*	4.5.7
PRER2	0x0024	RW	Peripheral reset enable register 2	0011_0311*	4.5.8
PER1	0x0028	RW	Peripheral enable register 1	0000_000F*	4.5.9
PER2	0x002C	RW	Peripheral enable register 2	0000_0101*	4.5.10
PCER1	0x0030	RW	Peripheral clock enable register 1	0000_000F*	4.5.11
PCER2	0x0034	RW	Peripheral clock enable register 2	0000_0101*	4.5.12
CSCR	0x0040	RW	Clock Source Control register	0000_0020	4.5.13
SCCR	0x0044	RW	System Clock Control register	0000_0000	4.5.14
CMR	0x0048	RW	Clock Monitoring register	0000_0090	4.5.15
NMIR	0x004C	RW	NMI control register	0000_0000	4.5.16
COR	0x0050	RW	Clock Output Control register	0000_000F	4.5.17
VDCCON	0x0064	WO	VDC Control register	040F_007F	4.5.18
LVDCON	0x0068	RW	LVD Control register	0001_0101	4.5.19

Table 10. SCU Register Map (continued)

Name	Offset	Type	Description	Reset value	Reference
HSIOSCTRIM	0x006C	RW	High Speed Internal OSC Trim Register	0XXX_XXXX	4.5.20
BISCCON	0x0070	RW	Built in self calibration control Register	0000_0000	4.5.21
MOSCR	0x0080	RW	External main Oscillator control register	0000_0301	4.5.22
EMODR	0x0084	RW	High Speed Internal OSC Trim Register	0000_0000	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 7	0001_0000	4.5.29

4.5.1 SMR: System Mode Register

The previous operating mode is shown in this register. The previous operating mode will be saved in this register after reset event. There are two controllable bits in power down mode. One is LSI On/Off control and the other is VDC On/Off control in power down mode. In addition, the System Mode Register is a 16-bit register.

SMR=0x4000_0004

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
						LSIAON	VDCAON			PREVMODE					
0	0	0	0	0	0	0	0	0	0	00		0	0	0	0
						RW	RW			RO					

9	LSIAON	LSI Always on select bit in power down mode	
		0	LSI is automatically off entering power down mode
		1	LSI isn't automatically off entering power down mode
8	VDCAON	VDC Always on select bit in power down mode	
		0	VDC is automatically off entering power down mode
		1	VDC isn't automatically off entering power down mode
5 4	PREVMODE	Previous operating mode before current reset event	
		00	Previous operating mode was RUN mode
		01	Previous operating mode was SLEEP mode
		10	Previous operating mode was Power Down mode
		11	Previous operating mode was INIT mode

4.5.2 SRCR: System Reset Control Register

It is possible to check if chip is in power down mode. To use STBO output function, it should be set as STBO that has output mode in Pin Mux. And it is possible to reset MCU as SWRST bit set. In addition, the System Reset Control Register is an 8-bit register.

SCR=0x4000_0008

7	6	5	4	3	2	1	0
			STBOP				SWRST
0	0	0	0	0	0	0	0
			RW				RW

5	STBOP	STBO pin output polarity select bit	
		0	Output Low when chip is in Power Down Output High when chip is in normal
		1	Output High when chip is in Power Down Output Low when chip is in normal
1	SWRST	Internal soft reset activation bit (check RSER[4] for reset)	
		0	Normal operation
		1	Internal soft reset generated and auto cleared

4.5.3 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						FRTWUE	WDTWUE	LVDWUE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
				RW	RW	RW	RW						RW	RW	RW

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
2	FRTWUE	Enable wakeup source of FRT event
	0	Not used for wakeup source
	1	Enable the wakeup event generation
1	WDTWUE	Enable wakeup source of WDT 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

4.5.4 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.**

WUSR=0x4000_0014

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

11	GPIODWU	Status of wakeup source of GPIOD port pin change event
	0	No wakeup event
	1	Wakeup event was generated
10	GPIOCWU	Status of wakeup source of GPIOC port pin change event
	0	No wakeup event
	1	Wakeup event was generated
9	GPIOBWU	Status of wakeup source of GPIOB port pin change event
	0	No wakeup event
	1	Wakeup event was generated
8	GPIOAWU	Status of wakeup source of GPIOA port pin change event
	0	No wakeup event
	1	Wakeup event was generated
2	FRTWU	Status of wakeup source of FRT event
	0	No wakeup event
	1	Wakeup event was generated
1	WDTWU	Status of wakeup source of WDT 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

4.5.5 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
LOCKUPRST	PINRST	CPURST	SWRST	WDTRST	MCKFRST	MOFRST	LVDRST
0	1	1	0	1	0	0	1
RW	RW	RW	RW	RW	RW	RW	RW

7	LOCKUPRST	CPU Lock up reset enable bit
		0 Reset from this event is masked
		1 Reset from this event is enabled
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	MOFRST	MOSC 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

4.5.6 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

8	7	6	5	4	3	2	1	0
LOCKUPRST	PORST	PINRST	CPURST	SWRST	WDTRST	MCKFRST	MOFRST	LVDRST
0	1	0	0	0	0	0	0	0
RC1	RC1	RC1	RC1	RC1	RC1	RC1	RC1	RC1

7	LOCKUPRST	CPU Lock up 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
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	MOFRST	MOSC 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

4.5.7 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
					FRT							TIMER3	TIMER2	TIMER1	TIMER0					GPIOD	GPIOC	GPIOB	GPIOA			DIV64		PCU	WDT	FMC	SCU
0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	1	0	1	1	1	1
					RW							RW	RW	RW	RW					RW	RW	RW	RW			RW		RW	RW	RW	RW

26	FRT	FRT reset enable
19	TIMER3	TIMER3 reset enable
18	TIMER2	TIMER2 reset enable
17	TIMER1	TIMER1 reset enable
16	TIMER0	TIMER0 reset enable
11	GPIOD	GPIOD reset enable
10	GPIOC	GPIOC reset enable
9	GPIOB	GPIOB reset enable
8	GPIOA	GPIOA reset enable
5	DIV64	DIV64 reset enable
3	PCU	Port Control Unit reset enable
2	WDT	Watchdog Timer reset enable
1	FMC	Flash memory controller reset enable
0	SCU	System Control Unit reset enable

4.5.8 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
											ADC				MPWM							UART1	UART0				I2C				SPI
0	0	0	0	0	0	0	0	0	0	0	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

20	ADC	ADC reset enable
16	MPWM0	MPWM reset enable
9	UART1	UART1 reset enable
8	UART0	UART0 reset enable
4	I2C	I^2C reset enable
0	SPI	SPI reset enable

4.5.9 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 PER1/PER2 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
					FRT							TIMER3	TIMER2	TIMER1	TIMER0					GPIOD	GPIOC	GPIOB	GPIOA			DIV64		Reserved	Reserved	Reserved	Reserved
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		RO	RO	RO	RO

26	FRT	FRT 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
5	DIV64	DIV64 function enable
3		Reserved
2		
1		
0		

4.5.10 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
											ADC				MPWM							UART1	UART0				I2C				SPI
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

20	ADC	ADC function enable
16	MPWM	MPWM function enable
9	UART1	UART1 function enable
8	UART0	UART0 function enable
4	I2C	I^2C function enable
0	SPI	SPI function enable

4.5.11 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
					FRT							TIMER3	TIMER2	TIMER1	TIMER0					GPIOD	GPIOC	GPIOB	GPIOA			DIV64		Reserved	Reserved	Reserved	Reserved
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	RO	RO	RO	RO

26	FRT	FRT 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
5	DIV64	DIV64 clock enable
3		Reserved
2		
1		
0		

4.5.12 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
											ADC				MWPM							UART1	UART0				I2C				SPI
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

20	ADC	ADC clock enable
16	MPWM	MPWM clock enable
9	UART1	UART1 clock enable
8	UART0	UART0 clock enable
4	I2C	I^2C clock enable
0	SPI	SPI clock enable

4.5.13 CSCR: Clock Source Control Register

The AC30M1x64/1x32 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
SOSCCON	LSICON	HSICON	MOSCCON
00	10	00	00
RW	RW	RW	RW

7	SOSCCON	External crystal sub oscillator control	
6		0X	Disable external sub crystal oscillator
		10	Enable external sub crystal oscillator
		11	Enable external sub crystal oscillator divide by 2
5	LSICON	Low speed internal oscillator control	
4		0X	Disable low speed internal oscillator
		10	Enable low speed internal oscillator
		11	Enable low speed internal oscillator divide by 2
3	HSICON	High speed internal oscillator control	
2		0X	Disable high speed internal oscillator
		10	Enable high speed internal oscillator
		11	Enable high speed internal oscillator divide by 2
1	MOSCCON	External crystal main oscillator control	
0		0X	Disable external main crystal oscillator
		10	Enable external main crystal oscillator
		11	Enable external main crystal oscillator divide by 2

4.5.14 SCCR: System Clock Control Register

Select the system clock source in SCCR, and selected clock source becomes MCLK. Before changing clock, clock sources have to be alive by CSCR register.

SCCR=0x4000_0044

7	6	5	4	3	2	1	0
					MCLKSEL		
					00		
					RW		

2 0	MCLKSEL	System clock select register	
		000	LSI (40kHz)
		001	SOSC (32.768kHz)
		100	HSI (40MHz)
		110	MOSC (4MHz ~ 16MHz)
NOTE: When change MCLKSEL, both clock sources should be alive.			
Example: Both of HSI and MOSC should be alive, otherwise the chip will do malfunction.			

4.5.15 CMR: Clock Monitoring Register

The clock can be monitored by LSI 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				SOSCMNT	SOSCIE	SOSCFAIL	SOSCSTS	MCLKMNT	MCLKIE	MCLKFAIL	MCLKSTS	MOSCMNT	MOSCIE	MOSCFAIL	MOSCSTS
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RW				RW	RW	RC1	RC1	RW	RW	RC1	RC1	RW	RW	RC1	RC1

15	MCLKREC	MCLK fail auto recovery
		0 MCLK is changed to LSI by default when MCLKFAIL issued
		1 MCLK auto recovery is disabled
11	SOSCMNT	External sub oscillator monitoring enable
		0 External sub oscillator monitoring disabled
		1 External sub oscillator monitoring enabled
10	SOSCIE	External sub oscillator fail interrupt enable
		0 External sub oscillator fail interrupt disabled
		1 External sub oscillator fail interrupt enabled
9	SOSCFAIL	External sub oscillator fail interrupt
		0 External sub oscillator fail interrupt not occurred
		1 Read: External sub oscillator fail interrupt is pending Write: Clear pending interrupt
8	SOSCSTS	External sub oscillator status
		0 Not oscillate
		1 External sub oscillator is working normally
7	MCLKMNT	MCLK monitoring enable
		0 MCLK monitoring disabled
		1 MCLK monitoring enabled
6	MCLKIE	MCLK fail interrupt enable
		0 MCLK fail interrupt disabled
		1 MCLK fail interrupt enabled
5	MCLKFAIL	MCLK fail interrupt
		0 MCLK fail interrupt not occurred
		1 Read: MCLK fail interrupt is pending Write: Clear pending interrupt
4	MCLKSTS	MCLK clock status
		0 No clock is present on MCLK
		1 Clock is present on MCLK
3	MOSCMNT	External main oscillator monitoring enable
		0 External main oscillator monitoring disabled
		1 External main oscillator monitoring enabled
2	MOSCIE	External main oscillator fail interrupt enable
		0 External main oscillator fail interrupt disabled
		1 External main oscillator fail interrupt enabled
1	MOSCFAIL	External main oscillator fail interrupt
		0 External main oscillator fail interrupt not occurred
		1 Read: External main oscillator fail interrupt is pending Write: Clear pending interrupt
0	MOSCSTS	External main oscillator status
		0 Not oscillate
		1 External main oscillator is working normally

4.5.16 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. It will jump to NMI handler if Selected NMI event occurred and it must check event status. For clearing occurred status, it should clear the interrupt flags of that peri occurred.

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

The AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. To use CLKO output function, it should be set as CLKO that has output mode in Pin Mux. Clock Output Register is 8-bit 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. To change VDCCON register value, it has to enter TRIM mode.

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	WDTINTSTS	MCLKFAILST	LVDSTS				PROTEN	OVPEN	WDTINTEN	MCLKFAILEN	LVDEN
-		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
WO		RO	RO	RO	RO	RO				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

4.5.17 COR: Clock Output Register

COR=0x4000_0050

7	6	5	4	3	2	1	0
-			CLKOEN	CLKODIV			
000			0	1111			
RO			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) $\text{CLKO} = \frac{MCLK}{2 * (\text{CLKODIV} + 1)} \quad (CLKODIV > 0)$

4.5.18 VDCCON: VDC Control Register

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
VDCME	Reserved					STOPSEL	Reserved	Reserved															VDCDE					VDCWDLY			
0						0																	0					0x04			
WO						WO																	WO					WO			

31	VDCME	VDCMODE value write enable. Write only with VDCMODE value.
		0 VDCMODE field is not updated by writing
		1 VDCMODE filed can be updated by writing
25	STOPSEL	STOP MODE Select bit.
		0 VDC STOP MODE 1
		1 VDC STOP MODE 2
8	VDCDE	VDCWDLY value write enable. Write only with VDCWDLY value
		0 VDCWDLY Write disable
		1 VDCWDLY Write Enable
3 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 4: 2msec

CAUTION!!: You must not set the reserved bit field.

NOTE: Refer to the following example code to see how to enter TRIM mode to change VDCCON value.

```
FM->MR=0xa5;
FM->MR=0x5a;                                    // TRIM mode enter
SCU->VDCCON = (1UL<<31) | (1UL<<25); // set VDC STOP MODE 2
FM->MR=0; // TRIM mode exit
```

4.5.19 LVDCON: LVD Control Register

On chip Low voltage 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
																SELEN						LVDSEL								LVDLVL	LVDEN
																0						00								0	1
																WO						RW								RO	RW

15	SELEN	LVD level SEL value write enable. Write only.
		0 LVDSEL field is not updated by writing
		1 LVDSEL filed can be updated by writing
9 8	LVDSEL	LVD detect level select
		00 LVD detect level is 1.73V
		01 LVD detect level is 2.65V
		10 LVD detect level is 3.70V
		11 Reserved
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

4.5.20 HSIOSCTRIM: High Speed Internal OSC Trim Register

Internal oscillator frequency trim register. This register is 32-bit register.

HSIOSCTRIM=0x4000_006C

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

31	BISCON	Build in self calibration function enable.
		0 BISC function disabled. IOSC supplies factory calibrated frequency.
		1 BISC function enabled. IOSC supplies self-calibrated frequency
30	REFSEL	Reference clock select for self-calibration
		0 Main oscillator clock source is reference clock
		1 Sub oscillator clock source is reference clock

CAUTION!!: You must not set the reserved bit field

NOTE: All trim bit can be writable when trim mode is enabled.

```
FM->MR=0xa5;
FM->MR=0x5a;                    // TRIM mode enter
…                               //  change HSIOSCTRIM value
FM->MR=0;                       // TRIM mode exit
```

4.5.21 BISCCON: Built In Self Calibration Control Register

Internal oscillator frequency trim register. This register is 32-bit register.

BISCCON=0x4000_0070

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
INTOSC_COMP		XTAL_COMP	
0		0	
RW		RW	

31 16	INTOSC_COMP[31:16]	INTOSC compare value
15 0	XTAL_COMP[15:0]	XTAL Compare value

Calibration supports below configurations on the table

Table 11. BISC Count Value

XTAL freq	TARGET freq	Update period	XTAL_COMP	INTOSC_COMP
MHz	MHz	Nano Sec	Count Value	Count Value
10	40	10,000	99	399
8	40	1,000,000	7999	39999
6	40	10,000	59	399

4.5.22 MOSCR: External Main 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.

EMOSCR=0x4000_0080

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FILSKIPWEN							FILSKIPEN	INVCLKWEN							INVCLKEN
0							0	0							0
WO							RW	WO							RW

15	FILSKIPWEN	Write enable of bit field FILSKIPEN.
		0 Write access of FILSKIPEN field is masked
		1 Write access of FILSKIPEN field is accepted
8	FILSKIPEN	Control External Main Oscillator Filter Skip bit
		0 External Main Oscillator Filter Skip Disable.
		1 External Main Oscillator Filter Skip Enable.
7	INVCLKWEN	Write enable of bit field FILSKIPEN.
		0 Write access of INVCLKEN field is masked
		1 Write access of INVCLKEN field is accepted
0	INVCLKEN	Control External Main Oscillator CLK Invert bit
		0 External Main Oscillator CLK Invert Disable.
		1 External Main Oscillator CLK Invert Enable.

4.5.23 EMODR: External 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			-	-	-
		RO			-	-	RO

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 AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. STCSEL bits and STCDIV bits of MCCR1 are used as SYSTICK external clock source. 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
Reserved	Reserved	Reserved	STCSEL	STCDIV
-	-	-	000	0x00
-	-	-	RW	RW

10 8	STCSEL	SYSTICK Clock source select bit	
		000	LSI
		100	MCLK
		101	HSI
		110	MOSC
		111	Reserved
7 0	STCDIV	SYSTICK Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing STCSEL.	

4.5.25 MCCR2: Miscellaneous Clock Control Register 2

The AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. PWMCSEL bits and PWMDIV bits of MCCR2 are used as MPWM clock source. If it is used MPWM, it must set this register. 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
Reserved	Reserved	PWMCSEL	PWMDIV
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	PWMCSEL	PWM Clock source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 Reserved
7 0	PWMDIV	PWM Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing PWMCSEL

4.5.26 MCCR3: Miscellaneous Clock Control Register 3

The AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. TIMERCSEL bits and TIMERDIV bits of MCCR3 are used as TIMER external clock source. WDTCSEL bits and WDTDIV bits of MCCR3 are used as WDT external clock source. 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 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
23 16	TIMERDIV	Timer Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing TIMERCSEL
10 8	WDTCSEL	WDT Clock source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
7 0	WDTDIV	WDT Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing WDTCSEL

4.5.27 MCCR4: Miscellaneous Clock Control Register 4

The AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. PxDCSEL bits and PxDDIV bits of MCCR4 are used as PORT debounce clock source. 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	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
					RW			RW														RW			RW							

26 24	PBDCSEL	Debounce Clock for Port B source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
23 16	PBDDIV	PORT B Debounce Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing PBDCSEL
10 8	PADCSEL	Debounce Clock for Port A source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
7 0	PADDIV	PORT A Debounce Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing PADCSEL

4.5.28 MCCR5: Miscellaneous Clock Control Register 5

The AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. PxDCSEL bits and PxDDIV bits of MCCR5 are used as PORT debounce clock source. 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	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
					RW			RW														RW			RW							

26 24	PDDCSEL	Debounce Clock for PORT D source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
23 16	PDDDIV	PORT D Debounce Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing PDDCSEL
10 8	PCDCSEL	Debounce Clock for PORT C source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
7 0	PCDDIV	PORT C Debounce Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing PCDCSEL

4.5.29 MCCR7: Miscellaneous Clock Control Register 7

The AC30M1x64/1x32 can drive the clock from internal MCLK clock with dedicated post divider. ADCCSEL bits and ADCDIV bits of MCCR7 are used as ADC external clock source. UARTCSEL bits and UARTDIV bits of MCCR7 are used as UART clock source. If it is used UART, it must set this register.

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														UARTCSEL			UARTCDIV							
0	0	0	0	0	000			0x01													000			0x01							
					RW			RW													RW				RW							

26 24	ADCCSEL	ADC clock source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 Reserved
23 16	ADCCDIV	ADC Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing ADCCSEL
10 8	UARTCSEL	UART clock source select bit 000 LSI 100 MCLK 101 HSI 110 MOSC 111 SOSC
7 0	UARTCDIV	UART Clock N divider 0x00: disabled 0xN: (selected clock) / N To change the value, set 0x0 first without changing UARTCSEL

4.6 Functional description

4.6.1 Built In Self Calibration

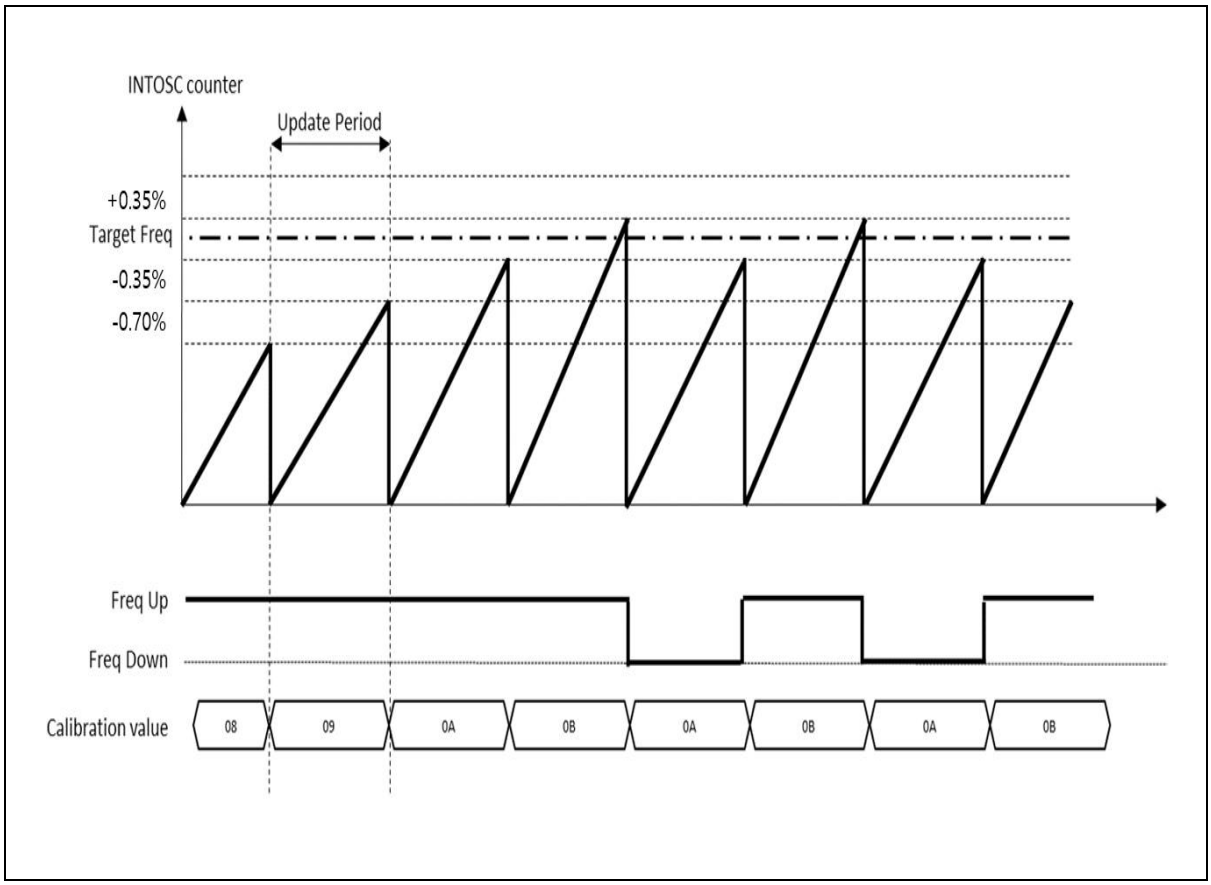


Figure 20. Built In Self Calibration

Self-calibration is a function to self-calibrate the INTOSC frequency at a specified interval based on an external clock source. Finely calibrate INTOSC using the external clock source and the counting error of INTOSC.

Self-calibration sets the INTOSC trim value area, and the trim value changes until the INTOSC frequency passes the target frequency level. At each stage, a frequency of 0.35% per 1-bit trim value is corrected, and the update period is determined by the reference clock counter value.

When the self-calibration function is activated, the factory calibrated INTOSC trim value is replaced by the self-calibration value.

5. Port Control Unit (PCU)

AC30M1x64/AC30M1x32 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 followings:

- Set pin function mux
- 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 12 are assigned for PCU blocks.

Table 12. 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 21 describes PCU in block diagram.

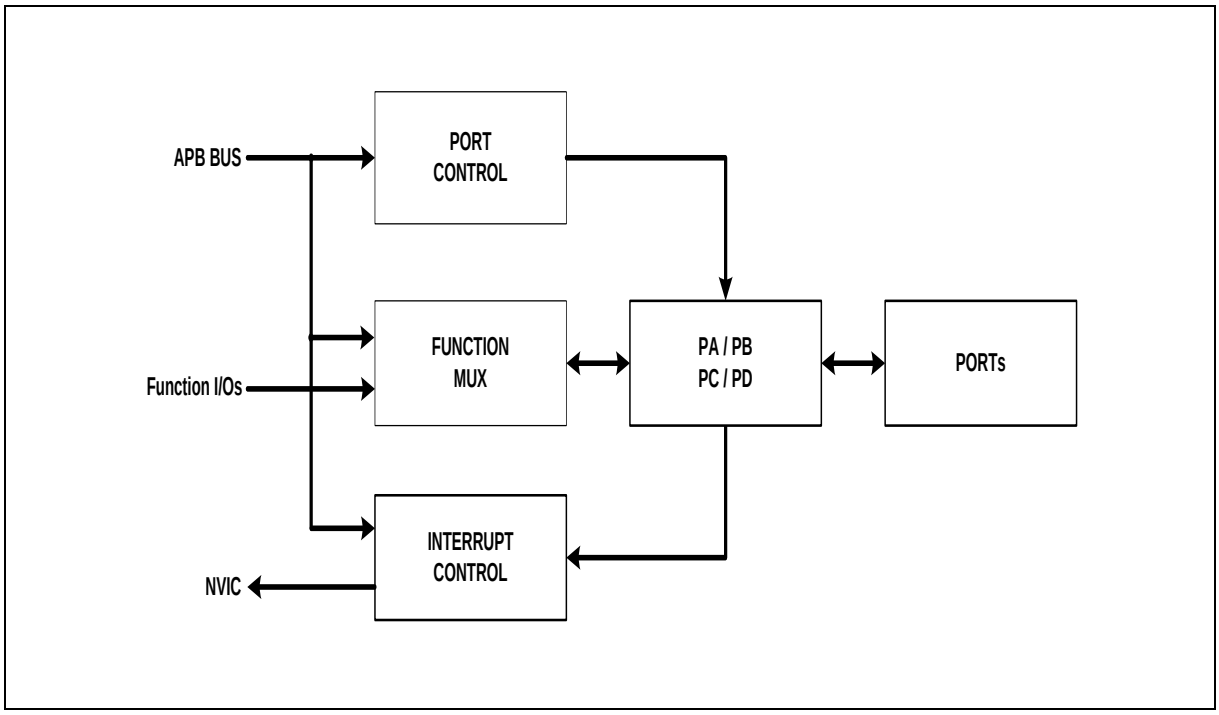


Figure 21. PCU Block Diagram

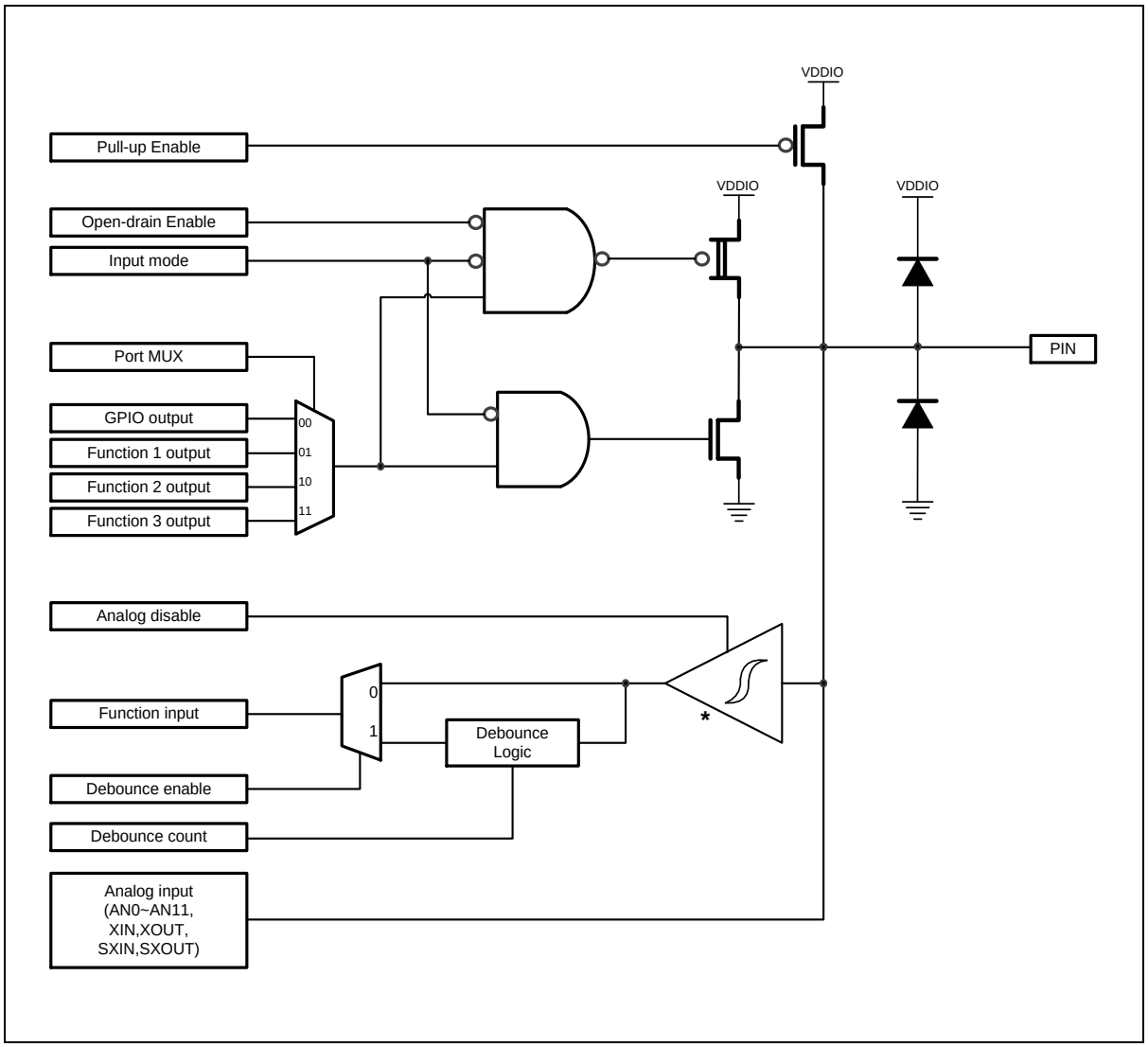


Figure 22. I/O Port Block Diagram (ADC and External Oscillator Pins)

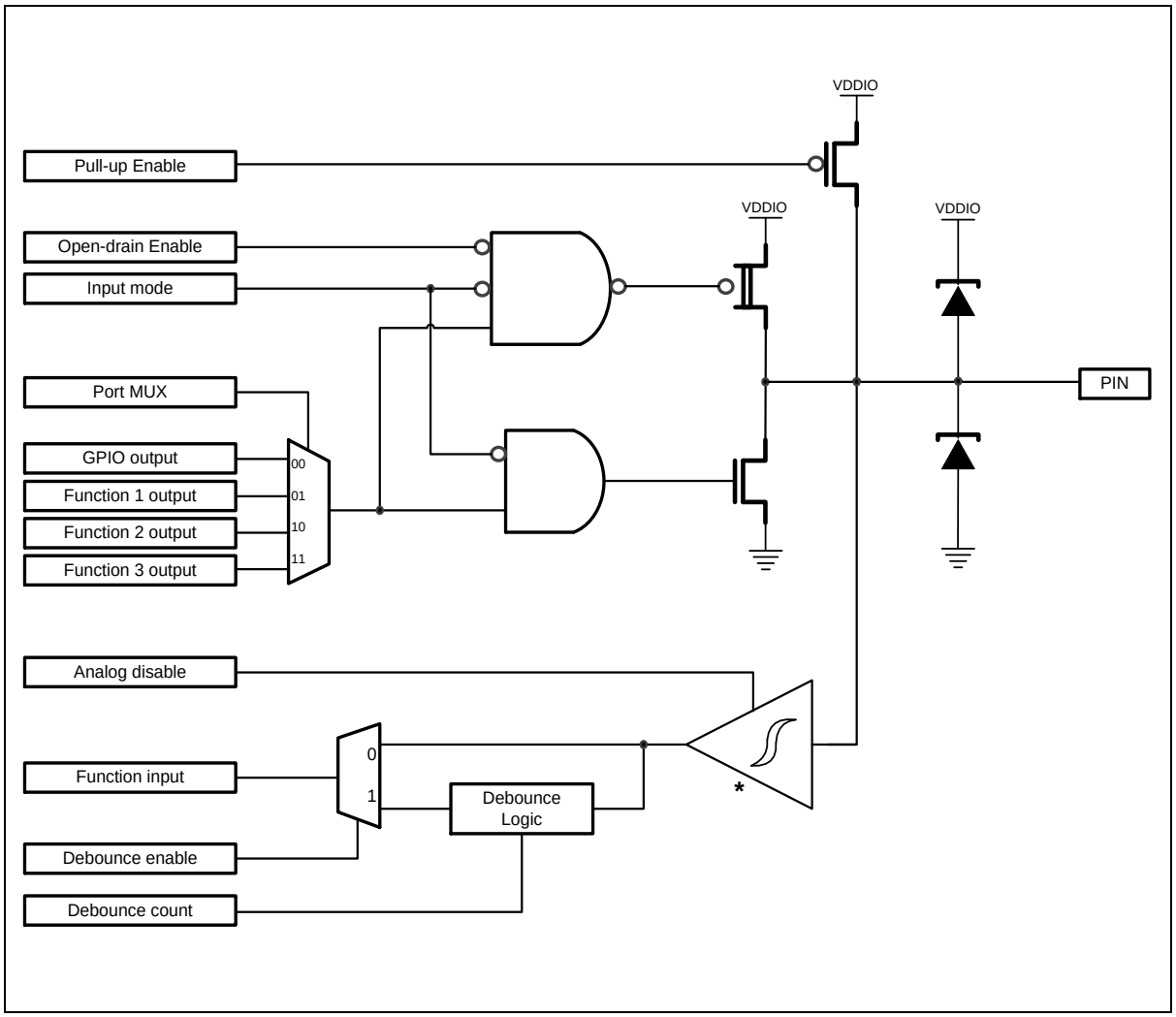


Figure 23. I/O Port Block Diagram (General I/O pins)

5.2 Pin multiplexing

GPIO pins have alternative function pins. Table 13 shows pin multiplexing information.

Table 13. GPIO Alternative Function

Pin name	Alternative function			
	00	01	10	11
PA0	PA0*	T2IO		AIN0
PA1	PA1*	T3IO		AIN1
PA2	PA2*	SS	WDTO	AIN2
PA3	PA3*	SCK	STBO	AIN3
PA4	PA4*			AIN4
PA5	PA5*			AIN5
PA6	PA6*	T0IO		AIN6
PA7	PA7*	T1IO		AIN7
PA8	PA8*	T2IO	T0IO	AIN8
PA9	PA9*	T3IO	T1IO	AIN9
PA10	PA10*			AIN10
PA11	PA11*			AIN11
PA12	PA12*	T0IO		
PA13	PA13*	T1IO		
PA14	PA14*	T2IO		
PA15	PA15*	T3IO		
PB0	PB0*	MPWMUH	SS	
PB1	PB1*	MPWMUL	SCK	
PB2	PB2*	MPWMVH	MOSI	
PB3	PB3*	MPWMVL	MISO	
PB4	PB4*	MPWMWH		
PB5	PB5*	MWMWL		
PB6	PB6*	PRTIN		
PB7	PB7*	OVIN		
PB8				
PB9				
PB10				
PB11				
PB12				
PB13				

Table 13. GPIO Alternative Function (continued)

Pin name	Alternative function			
	00	**01**	**10**	**11**
PB14				
PB15				
PC0	PC0	SWCLK*	RXD1	
PC1	PC1	SWDIO*	TXD1	
PC2	PC2*			
PC3	PC3*			
PC4	PC4*		T0IO	
PC5	PC5*	RXD1	T1IO	
PC6	PC6*	TXD1	T2IO	
PC7	PC7*	SCL	T3IO	
PC8	PC8*	SDA		VMRG
PC9	PC9*	CLKO		
PC10	PC10	nRESET*		
PC11	PC11	BOOT*	T0IO	
PC12	PC12*	T3IO		XIN
PC13	PC13*	T2IO		XOUT
PC14	PC14*	RXD0		
PC15	PC15*	TXD0		
PD0	PD0*	SS		
PD1	PD1*	SCK		
PD2	PD2*	MOSI	SCL	SXOUT
PD3	PD3*	MISO	SDA	SXIN

NOTES:

1. (*) mark indicates default pin setting.

5.3 Registers

Base address of PCU is introduced in the followings:

Table 14. Base Address of PCU

Name	Base address	Description
PA	0x4000_1000	General Port A
PB	0x4000_1100	General Port B
PC	0x4000_1200	General Port C
PD	0x4000_1300	General Port D

Table 15. 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	5.3.9
PC*n*.IER	0x--10	RW	Port *n* interrupt enable register	5.3.10
PC*n*.ISR	0x--14	RW	Port *n* interrupt status register	5.3.11
PC*n*.ICR	0x--18	RW	Port *n* interrupt control register	5.3.12
	0x--1C		Reserved	
PORTEN	0x1FF0	RW	Port Access enable	5.3.13

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 16. PCA.MR Selection Bit Information

Port	Selection bit			
	00	**01**	**10**	**11**
PA0	PA0*	T2IO		AIN0
PA1	PA1*	T3IO		AIN1
PA2	PA2*	SS	WDTO	AIN2
PA3	PA3*	SCK	STBO	AIN3
PA4	PA4*			AIN4
PA5	PA5*			AIN5
PA6	PA6*	T0IO		AIN6
PA7	PA7*	T1IO		AIN7
PA8	PA8*	T2IO	T0IO	AIN8
PA9	PA9*	T3IO	T1IO	AIN9
PA10	PA10*			AIN10
PA11	PA11*			AIN11
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 17. PCB.MR Selection Bit Information

Port	Selection bit			
	00	**01**	**10**	**11**
PB0	PB0*	MPWMUH	SS	
PB1	PB1*	MPWMUL	SCK	
PB2	PB2*	MPWMVH	MOSI	
PB3	PB3*	MPWMVL	MISO	
PB4	PB4*	MPWMWH		
PB5	PB5*	MPWMWL		
PB6	PB6*	PRTIN		
PB7	PB7*	OVIN		

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	00	00	00	01	01
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Table 18. PCC.MR Selection Bit Information

Port	Selection bit			
	00	**01**	**10**	**11**
PC0	PC0	SWCLK*	RXD1	
PC1	PC1	SWDIO*	TXD1	
PC2	PC2*			
PC3	PC3*			
PC4	PC4*		T0IO	
PC5	PC5*	RXD1	T1IO	
PC6	PC6*	TXD1	T2IO	
PC7	PC7*	SCL	T3IO	
PC8	PC8*	SDA		VMRG
PC9	PC9*	CLKO		
PC10	PC10	nRESET*		
PC11	PC11	BOOT*	T0IO	
PC12	PC12*	T3IO		XIN
PC13	PC13*	T2IO		XOUT
PC14	PC14*	RXD0		
PC15	PC15*	TXD0		

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 19. PCD.MR Selection Bit Information

PORT	SELECTION BIT			
	00	**01**	**10**	**11**
PD0	PD0*	SS		
PD1	PD1*	SCK		
PD2	PD2*	MOSI	SCL	SXOUT
PD3	PD3*	MISO	SDA	SXIN

5.3.5 PCn.CR: PORT 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

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	11	11	11	10	10
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

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 (Except for PCC.PCR)

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

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 PCC.PCR: PORT C Pull-up Resistor Control Register

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

PCC.PCR=0x4000_1208

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
0C03															
RW															

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

5.3.9 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															

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

5.3.10 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

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

5.3.11 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	

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

5.3.12 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
IC15		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	

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

5.3.13 PORTEN: Port Access Enable

PORTEN enables register writing permission 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	--							
								WO							

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

NOTE: Following example code shows how to use PORTEN.

```
PORTEN=0x15; PORTEN=0x51;        // enable PORTEN
…                                // set PCn.MR, PCn.CR PCn.PCR and etc.
PORTEN=0;                        // disable PORTEN
```

5.4 Functional description

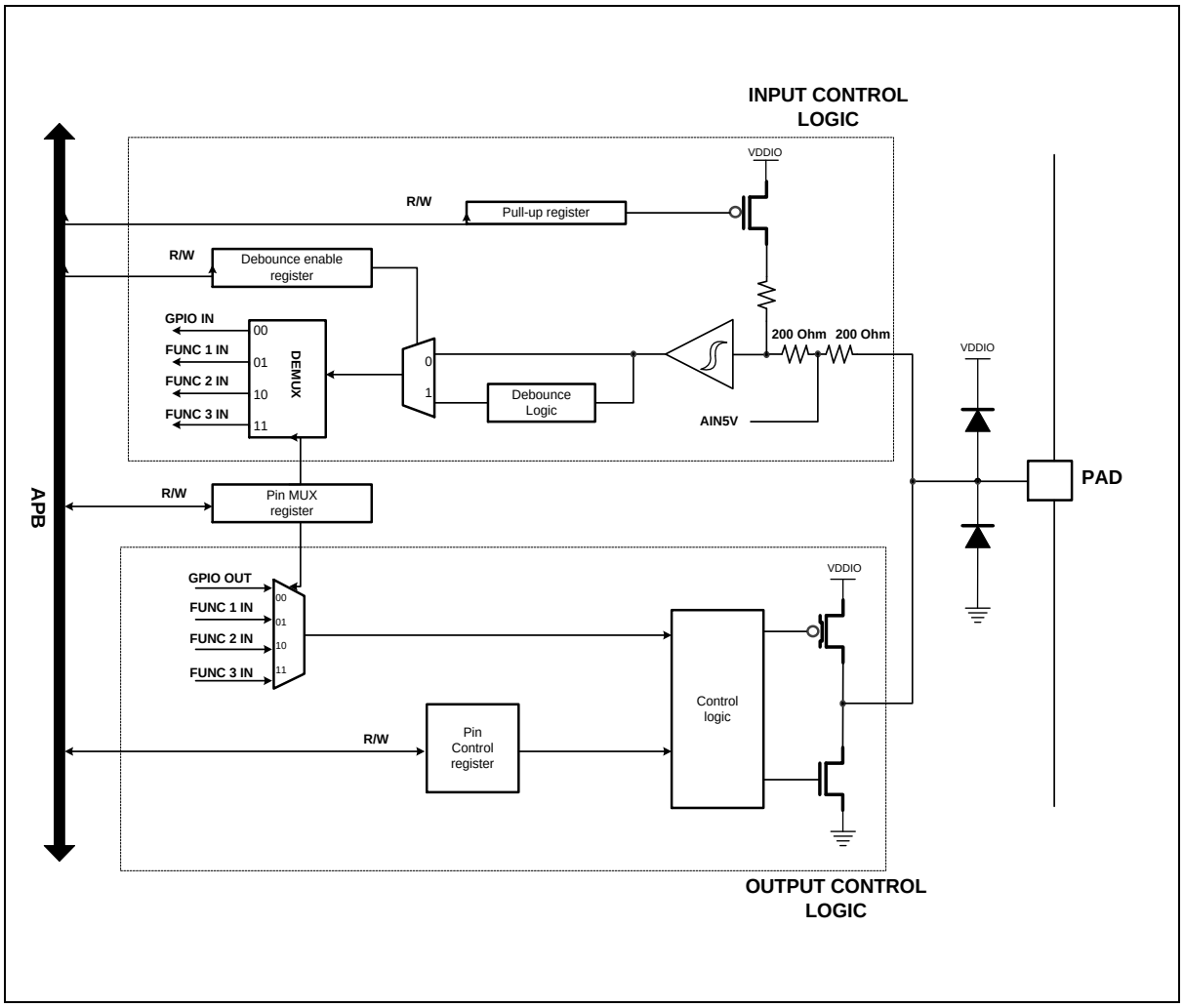


Figure 24. PCU Functional Block Diagram

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

The Port Function different according to the Pin Mux Register.

The Input Data Register capture the data present on the I/O pin or Debounced input data at every GPIO Clock cycle.

When the De-Bounce functions of Input Data is used by Debounce Enable Register. External input data 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.

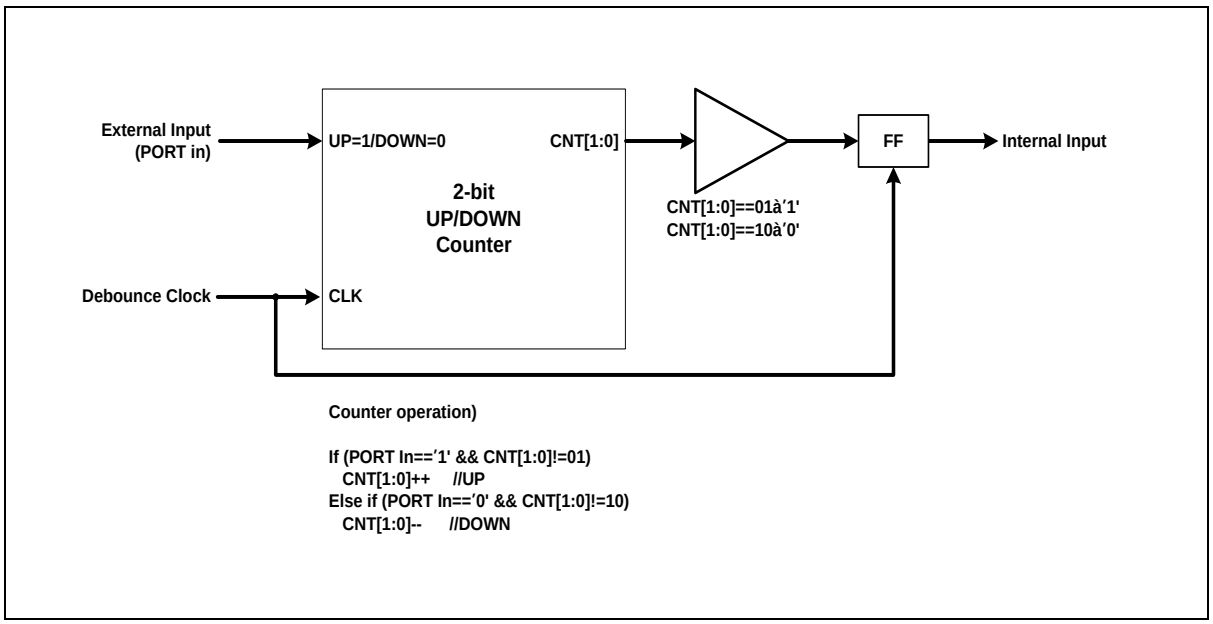


Figure 25. Debouncing Logic

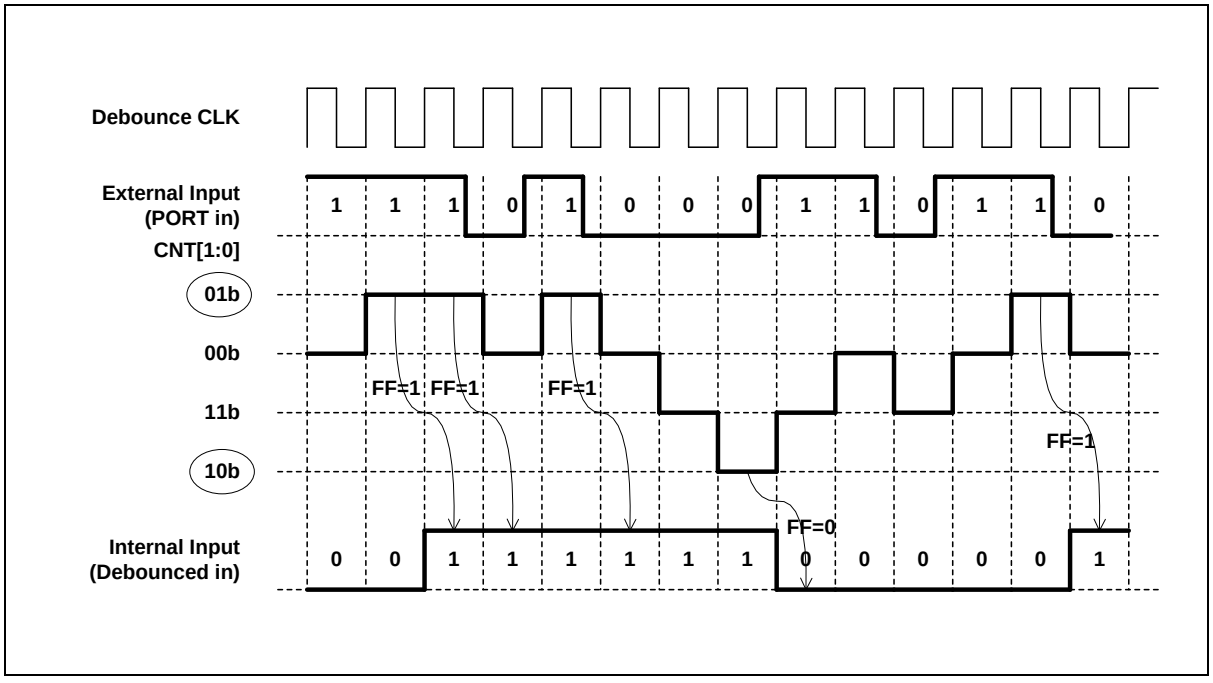


Figure 26. Port Debouncing Example

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) selection
- Read Input signal level

Table 20 are assigned for GPIO blocks.

Table 20. 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 27 describes GPIO in block diagram.

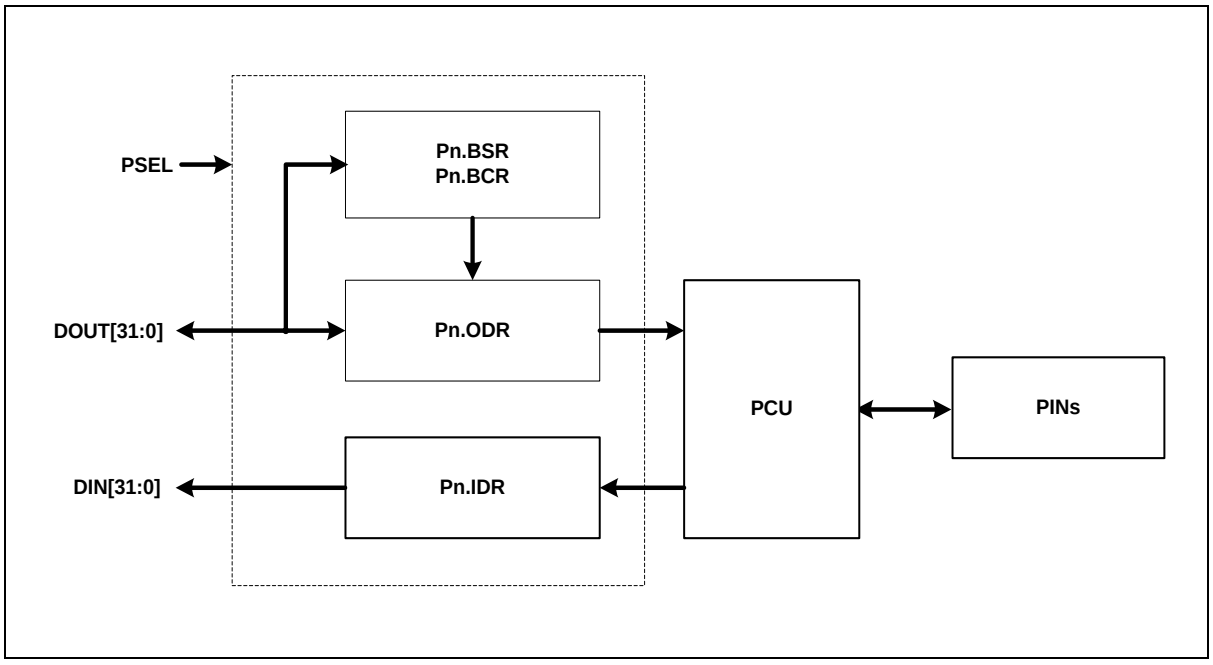


Figure 27. GPIO Block Diagram

6.2 Registers

Base address of PCU is introduced in the followings:

Table 21. Base Address of PCU

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

Table 22. 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 Pn.IDR 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
IDR															
0000 RO															

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

6.2.3 Pn.BSR: PORT 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 Pn.ODR register

6.2.4 Pn.BCR: PORT 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
BCR															
0000															
WO															

BCR	Pin current level
0	Not effect
1	Clear correspondent bit in Pn.ODR register

6.3 Functional description

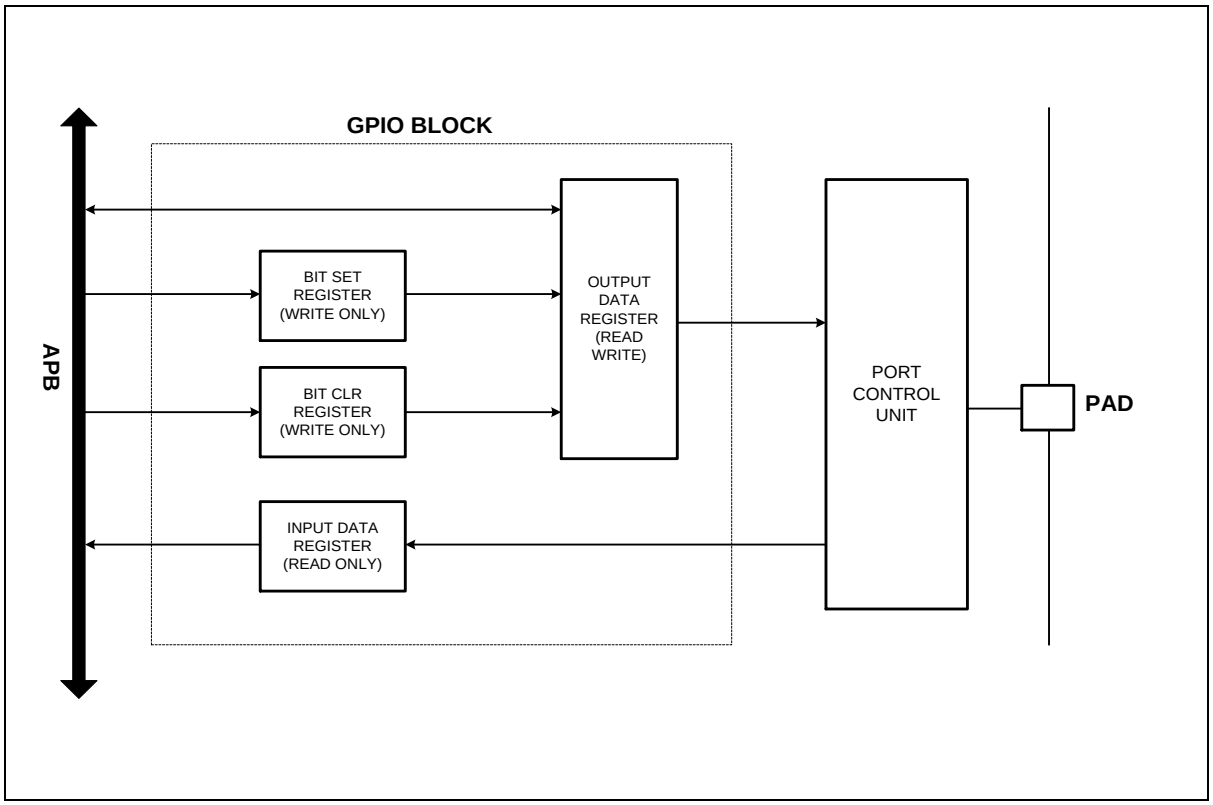


Figure 28. Functional Block Diagram

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

- When set the Bit Set Register, GPIO Output Data Register set the high.
- When set the Bit Clr Register, GPIO Output Data Register set the Low.

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

7. Flash Memory Controller (FMC)

The flash memory controller (FMC) is an interface controller of internal flash memories:

- 64/32KB flash memory with protection bits
- 32-word length program or erase at a time
- Bulk erase for 64/32KB memory at a time
- 32-word size OTP area
- 50ns flash access time of read
- wait(under 20MHz), 1-wait, 2-wait, and pre-fetch(read acceleration) access support
- Use internal 40MHz OSC clock to make timing control for Erase/Program

7.1 Flash memory map

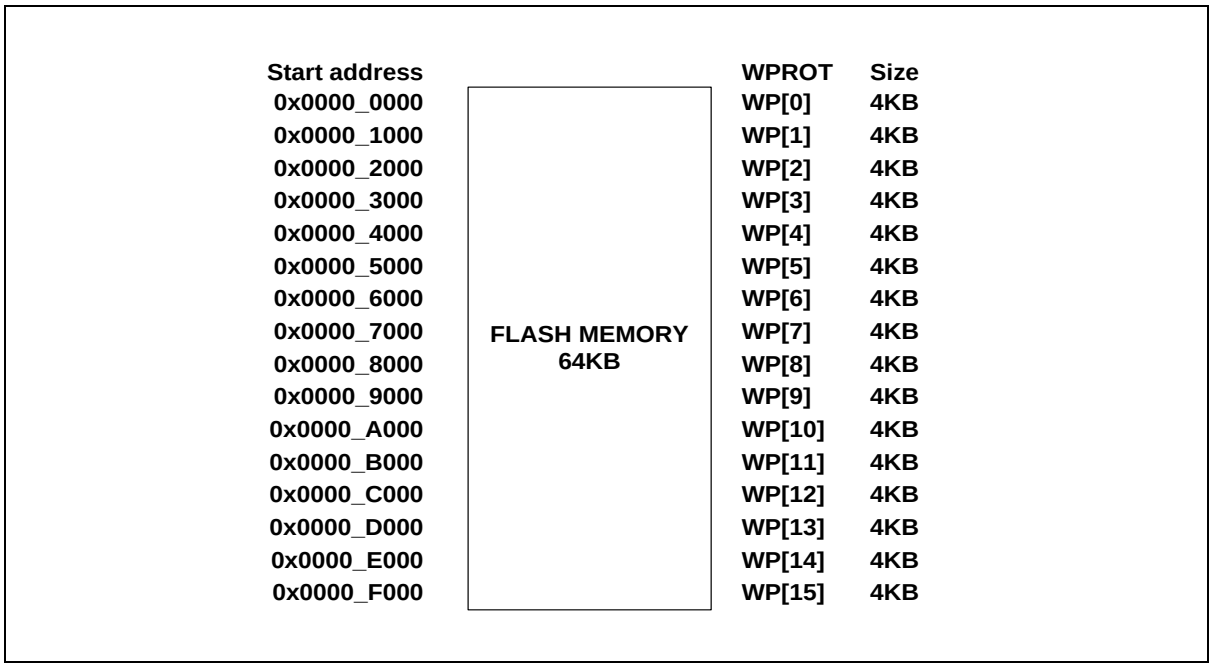


Figure 29. Code Flash Memory Map (64 KB Code Flash)

7.2 Registers

Tables shown below introduce the 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	0x05000000	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	0x00018FFF	7.2.5
FM.TICK	0x001C	R	Flash Memory Tick Timer	0x00000000	7.2.6
FM.CRC	0x0020	R	Flash CRC16 check value	0x00000000	7.2.7
FM.CFG	0x0030	RW	Flash Memory Configuration value	0x00008200	7.2.8
FM.HWID	0x0040	R	Second HW ID for AC30M1x64/1x32	0x30146400	7.2.9
BOOTCR	0x0074	RW	Boot ROM clear, SRAM Remap register	0x00000000	7.2.10
FM.WPROT	0x0078	RW	Write Protection register	0x00FFFF00	7.2.11
FM.RPROT	0x007C	RW	Read Protection register	0x000000FF	7.2.12

7.2.1 FM.MR: Flash Memory Mode Register

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
							IDLE	TESTEN	AMBAEN	PROTEN				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	RW							

24	IDLE	0	Flash Idle state bit ("0" means flash busy for PGM or ERS)
		1	Flash Idle state bit ("1" means flash idle, free to read)
23	TESTEN (test only)	0	Flash test register disable ("0" means cannot set TEST reg)
		1	Flash test register enable ("0" means can set TEST reg)
22	AMBAEN	0	AMBA mode disabled status
		1	AMBA mode enable (can change wait state and etc)
21	PROTEN	0	Flash protection register disable ("0" means cannot access protection register)
		1	Flash protection register enable ("1" means can access protection register)
17	TRMEN	0	TRIM mode disabled status
		1	Trim mode entry status(read only)
16	TRM	0	TRIM mode disabled
		1	Trim mode status(read only) must be set with TRMEN
9	FEMOD	0	Flash (program/erase) mode disabled
		1	Flash mode entry status(read only)
8	FMOD	0	Flash (program/erase) mode disabled
		1	Flash mode status(read only) must be set with FEMOD
7 0	ACODE	5A à A5	Flash mode entry sequence
		A5 à 5A	Trim mode entry sequence
		81 à 28	AMBA mode entry sequence
		66 à 99	PROT mode entry sequence
		39 à 7D	TESTEN mode entry sequence (test only)

7.2.2 FM.CR: Flash Memory Control Register

Internal flash memory control register. FM.CR[17:0] bits can be accessed while flash mode entry is activated. FM.CR[31], FM.CR[27:24] bit can be accessed in AMBA mode and FMCR[30:28] bits can be accessed in trim mode.

FM.CR=0x4000_0108

31	30	29	28	27 … 24	23…21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OTP3	OTP2	OTP1	OTP0			TMREN			TEST1	TEST0	VPPOUT	EVER	PVER	BLKE	DMYE	OTPE	AEE	AEF	SUBACT	PPGM	PMOD	WE	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	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31	OTP3	0	
		1	OTP area 3 access enable (user can access)
30	OTP2	0	
		1	OTP area 2 access enable (user can access)
29	OTP1	0	
		1	OTP area 1 access enable (user can access in a certain condition), OTP1 is used for read protection
28	OTP0	0	
		1	OTP area 0 access enable (user can not erase/program this area)
20	TMREN	0	Flash Tick timer enable
		1	Flash tick timer enable Tick timer runs by system clock while PGM or ERS undergoing
17 16	TEST[1:0]	00	Normal operation
		01	(read) Row voltage mode
		01	(write) ODD Row program
		10	Even Row program
		11	All Row program
15	VPPOUT	0	
		1	Charge pump Vpp output
14	EVER	0	
		1	Erase verify mode
13	PVER	0	
		1	Program verify mode
12	BLKE	0	
		1	128page write enable for full chip writing to save program time
11	DMYE	0	
		1	DUMMY area enable.
10	OTPE	0	
		1	OTP area A, B, C, D enable (user cannot access otp directly)
9	AEE	0	
		1	Pre PGM enable , Page buffer set automatically
8	AEF(BLKERS)	0	
		1	All erase 64/32KB code area enable
7	SUBACT	0	
		1	SUB Active mode (System clock under 1MHz)
6	PPGM	0	
		1	Pre-PGM for Erase operation (pre-program before erase)

5	PMODE	0	
		1	PMODE enable(Address path changing)
4	WE	0	
		1	Write enable
3	PBLD	0	
		1	Page buffer load(WE should be set)
2	PGM	0	
		1	Program mode enable
1	ERS	0	
		1	Erase mode enable
0	PBR	0	
		1	Page buffer reset

7.2.3 FM.AR: Flash Memory Address Register

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	0	0x0000													
		RW													

13 0	FADDR	14-bit address covers 16K words address (one word = 4 bytes)

7.2.4 FM.DR: Flash Memory Data Register

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

Internal flash memory Timer value register (18-bit), Erase/Program timer runs up to {TMR[17:0]}

FM.TMR=0x4000_0114

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
																TMR															
0	0	0	0	0	0	0	0	0	0	0	0	0	0			0x18FFF															
								RW																							

	TMR	Erase/PGM timer (default, 0x18FFF) Timer counts up to {TMR[17:0]} by 40MHz HSI OSC clock

7.2.6 FM.TICK: Flash Memory Tick Timer Register

Internal flash memory Tick Timer 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			0x00000															
								RW																							

17 0	FTICK	TICK goes to 0x3FFFF from written TICK value while TMR runs by PCLK clock while Flash PGM or ERS (counts up only when IDLE bit of FMMR register is low)

7.2.7 FM.CRC: Flash CRC Check Register

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

FM.CRC=0x4000_012C

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
																CRC16															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x00000															
																RW															

15 0	CRC16	CRC16 check value read register polynomial: (1 + x5 + x12 + x16) data width: 32 (the first serial bit is D[31])

7.2.8 FM.CFG: Flash Memory Config Register

Internal flash memory Config register. This register has the same address with FMTRIM0 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
																HRESPD			TESTCLK			WAIT		CRCINIT	CRCEN	Reserved					
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	-					
																RW			RW			RW	RW	RW	RW	-					

15	HRESPDIS	0	By default, when CPU try to write ROM area directly, flash interface will return ERROR response by AMBA protocol
		1	Disable HRESP(error response function) of Data or System bus (HRESP is AMBA AHB signal) This bit only be written in AMBA mode and MSB 16-bit (bit [31:16]) must be 0x7858
12	TESTCLK	0	TEST Clock selection (test purpose only) Set "1" to use system bus clock instead of internal 40MHz OSC This bits only be written in AMBA mode and MSB 16-bit (bit [31:16]) must be 0x7858
9 8	WAIT		This bits only be written in AMBA mode and MSB 16-bit (bit [31:16]) must be 0x7858
		00	WAIT is 00, flash access in 1 cycle (0-wait)
		01	WAIT is 01, flash access in 2 cycles (1-wait)
		10	WAIT is 10, flash access in 3 cycles (2-wait) – default
		11	WAIT is 11, flash access in pre-fetch mode Note) In pre-fetch mode, OTP (0x3F0000xx~0x3F0005xx) read and Program/Erase operation would not work correctly. User must exit from pre-fetch mode to read OTP or program/erase flash memory
7	CRCINIT	0	When this bit is set('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	CRC16 enable CRC value will be calculated at every flash read timing

7.2.9 FM.HWID: Flash Hardware ID Register

Flash Hardware ID for correct size information. This register is 32-bit read only register.

FM.HWID=0x4000_0140

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
FHWID			
0x3014_6400 R			

31 0	FHWID	Flash HWID register It returns size option values 0x30146400: 64KB flash product option 0x30143200: 32KB flash product option 0x30FF0000: wrong size option code, 64KB flash enable

7.2.10 BOOTCR: Boot ROM Remap Clear Register

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

BOOTCR=0x4000_0174

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

4	SREMAP	SRAM remap enable register When this bit is set, SRAM will be located at 0x0000_0000 address. This bit location can be accessed in AMBA mode Flash memory also can be read at 0x3000_0000 while SREAMP enable
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, user cannot re-activate this bit. Always 0 in user mode.

NOTE: SREMAP bit can be writable when AMBA mode is enabled.

```
FM->MR=0x81;
FM->MR=0x28;                                  // AMBA mode enter
…                                             //  change
BOOTCR[4](SREMAP) value
FM->MR=0;                                     // AMBA mode exit
```

7.2.11 FM.WPROT: Flash Memory Write Protection Register

Internal flash memory write protection register. This register can also be updated from OTP area of Flash during the boot sequence.

FM.WPROT=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
																WP															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0xFFFF															
																RW															

15 2	WP[15:2]	Sector(4KB block each) protect Each bit enable write protect corresponding 4K block when WP bit is set ('1'). If user want to set WP[15:2], MSB 8-bit(bit [31:24]) must be 0x87
1 0	WP[1:0]	Sector(4KB block each) protect Each bit enable write protect corresponding 4K block when WP bit is set ('1'). If user want to set WP[1:0], MSB 8-bit(bit [31:24]) must be 0x98

7.2.12 FM.RPROT: Flash Memory Read Protection Register

Internal flash memory read protection register. This register can also be updated from OTP area of Flash during the boot sequence.

FM.RPROT=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
LOCK2	LOCK1	JTAGDIS	-	-		READ PROTECTIN
0	0	0	-	-		0xFF
RO	RO	RO		'		RW

31	LOCK2	Read protection level 2 state flag
30	LOCK1	Read protection level 1 state flag
29	JTAGDIS	JTAG disable state flag
7	Unlock	By default, read protection disable (FM.RPROT = 0xFF)
0	LOCK1	Read protection level 1 Code protection mode enable, debug can be connected Write 0x39 to activate LOCK1 (only can be written in Unlock state) Code in SRAM or debugger cannot read flash area When flash was read from SRAM or debugger, 0xA5A5A5A5 will be return as read data
	LOCK2	Read protection level 2 Code protection mode enable, debug cannot be connected Write any value except 0x39(include 0xFF) to activate LOCK2 (only can be written in Unlock state or LOCK1) When flash was read from SRAM, 0xA5A5A5A5 will be return as read data

7.3 Functional description

7.3.1 Flash Erase and Program examples

Basic steps of programming Flash memory consist of three steps introduced below. Minimum Program or Erase unit is a Page, and 32-word (128-byte) becomes a page.

1. Pre-program of target Page before erase operation
2. Page Erase
3. Page Program

For every erase operations, pre-program operation is needed to prevent over erase of flash memory cells. A user must enable 40MHz internal oscillator first to erase or program flash.

1. Pre-program and Erase example
    A. Flash mode enable to write FM.CR register (write 0x5A and then write 0xA5 into FMMR)
    B. Set FM.TMR register to be 0.5ms operation (based on 40MHz Int OSC clock)
    C. Set target Page address in FM.AR
    D. Set PMODE bit first
    E. set PPGM, WE, PGM bits of FMCR
    F. Wait until IDLE bit of FM.MR register become "1" after pre-program
    G. Clear WE, PGM bits of FMCR
    H. Wait 5us
    I. Clear PPGM bit of FM.CR
    J. Wait 30us before returning to normal operation
    K. Clear PMODE bit of FM.CR
    L. Clear Flash mode (write 0x00 into FM.MR)
    M. Insert at least 2 NOPs, and return to normal operation

2. Erase example
   A. Flash mode enable to write FM.CR register (write 0x5A and then write 0xA5 into FM.MR)
   B. Set FM.TMR register to be 2.5ms operation (based on 40MHz Int OSC clock)
   C. Set target Page address in FM.AR
   D. Set PMODE bit first
   E. set WE, ERS bits of FM.CR
   F. Wait until IDLE bit of FM.MR register become "1" after erase
   G. Clear WE, ERS bits of FM.CR
   H. Wait 30us before returning to normal operation
   I. Clear PMODE bit of FM.CR
   J. Clear Flash mode (write 0x00 into FM.MR)
   K. Insert at least 2 NOPs, and return to normal operation
3. Program example include page buffer load
   A. Flash mode enable to write FM.CR register (write 0x5A and then write 0xA5 into FM.MR)
   B. Set FM.TMR register to be 2.5ms operation (based on 40MHz Int OSC clock)
   C. Set target Page address in FM.AR
   D. Set PMODE bit first
   E. Set PBR bit of FM.CR and clear PBR bit of FM.CR(page buffer reset)
   F. Set PBLD bit of FM.CR to load data into page buffer
   G. Write word(32-bit) data into FM.DR (max 32 words), address increased automatically based on word address
   H. Clear PBLD bits of FM.CR
   I. Set target Page address in FM.AR again
   J. Set WE, PGM bits of FM.CR
   K. Wait until IDLE bit of FM.MR register become "1" after program
   L. Clear WE, PGM bits of FM.CR
   M. Wait 30us before returning to normal operation
   N. Clear PMODE bits of FM.CR
   O. Clear Flash mode (write 0x00 into FM.MR)
   P. Insert at least 2 NOPs, and return to normal operation

8. Watchdog Timer (WDT)

Watchdog Timer (WDT) monitors the operation of an 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.

AC30M1x64/AC30M1x32 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 AC30M1x64/AC30M1x32 series features the followings:

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

8.1 WDT block diagram

In this section, WDT block diagram is introduced in Figure 30.

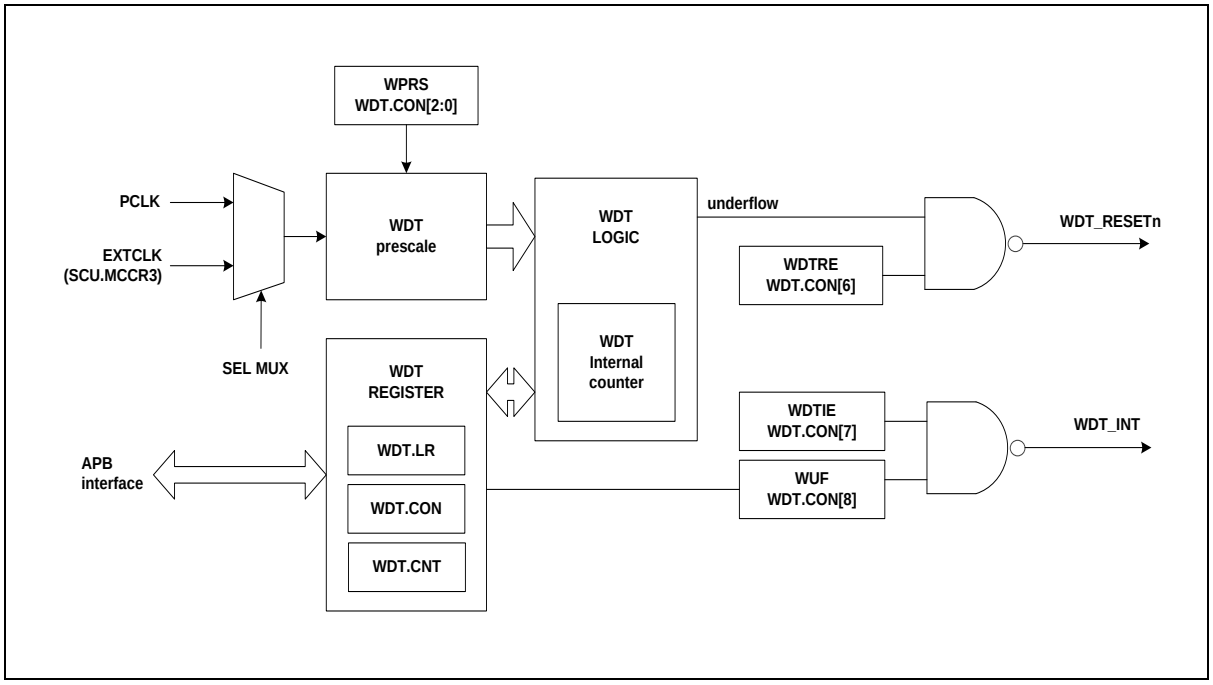


Figure 30. WDT Block Diagram

8.2 Registers

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

Table 25. Base Address of WDT

Name	Base address
WDT	0x4000_0200

Table 26. WDT Register Map

Name	Offset	Type	Description	Reset value	Reference
WDT.LR	0x0000	W	WDT load register	0x00000000	8.2.1
WDT.CNT	0x0004	R	WDT current count register	0x00000000	8.2.2
WDT.CON	0x0008	RW	WDT control register	0x0000805C	8.2.3

8.2.1 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 clocks to update WDTLR to WDTCNT. WDT ext clock source is controlled by WDTCSEL and WDTDIV in MCCR3.

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			
RW			

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

8.2.2 WDT.CNT: Watchdog Current Count Register

The WDTCNT register represent the current count value of 32-bit down counter .When the counter value reach to 0, the interrupt or reset will be aroused.

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_0A30																															
RO																															

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

8.2.3 WDT.CON: Watchdog Control Register

WDT module should be configured properly before running. When target purpose is defined, the WDT can be configured in the WDTCON register

WDT_CON=0x4000_0208

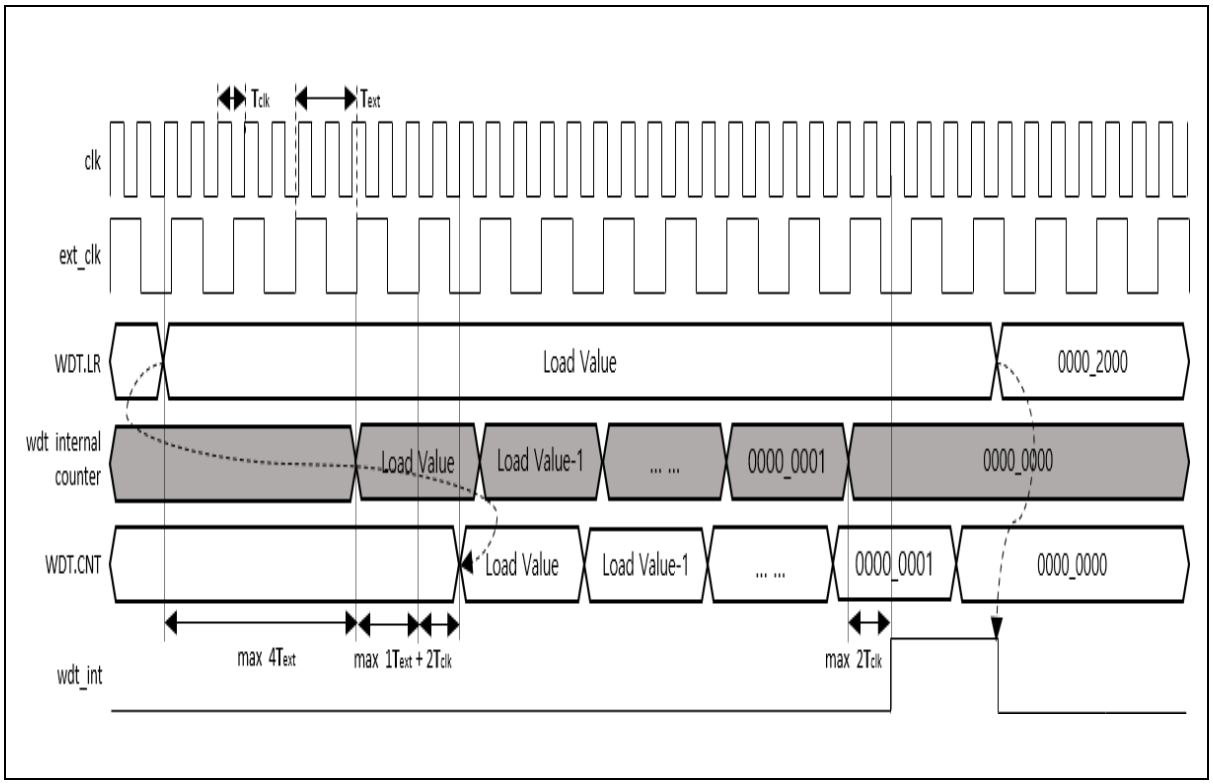
31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15	14 13 12 11 10 9	8	7	6	5	4	3	2 1 0
		WDBG		WUF	WDTIE	WDTRE		WDTEN	CKSEL	WPRS
		1		0	0	1	-	1	1	100
		RW		RW	RW	RW	RW	RW	RW	RW

15	WDBG	Watchdog operation control in debug mode
		0 0
		1 1
8	WUF	Watchdog timer underflow flag
		0 0
		1 1
7	WDTIE	Watchdog timer counter underflow interrupt enable
		0 0
		1 1
6	WDTRE	Watchdog timer counter underflow interrupt enable
		0 0
		1 1
4	WDTEN	Watchdog Counter enable
		0 0
		1 1
3	CKSEL	WDTCLKIN clock source select
		0 0
		1 1
2 0	WPRS[2:0]	Counter clock prescaler WDTCLK = WDTCLKIN/WPRS
		000 000
		001 001
		010 010
		011 011
		100 100
		101 101
		110 110
		111 111

8.3 Functional description

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 WDT.LR value. Before WDT counter down to 0, software can write a certain value to register WDT.LR to reload WDT counter.

8.3.1 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.

8.3.2 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 5external clock sources. External clock source can be enable 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 5Text} + max 4Tclk

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

Table 27. Prescaled WDT Counter Clock Frequencies

External clock source (WDTCLKIN)	WDTCLKIN	WDTCLKIN / 4	WDTCLKIN / 8	WDTCLKIN / 16	WDTCLKIN / 32	WDTCLKIN / 64	WDTCLKIN / 128	WDTCLKIN / 256
LSI	40kHz	10kHz	5kHz	2.5kHz	1.25kHz	0.625kHz	0.3125kHz	0.15625kHz
MCLK	Bus clock	MCLK/4	MCLK/8	MCLK/16	MCLK/32	MCLK/32	MCLK/128	MCLK/256
HSI	40MHz	10MHz	5MHz	2.5MHz	1.25MHz	0.625MHz	0.3125MHz	0.15625MHz
MOSC	XTAL frequency (4MHz~ 16MHz)	XTAL/4	XTAL/8	XTAL/16	XTAL/32	XTAL/64	XTAL/128	XTAL/256
SOSC	32.768kHz	8.192kHz	4.096kHz	2.048kHz	1.024kHz	0.512kHz	0.256kHz	0.128kHz

9. 16-bit timer

The timer block is consisted with 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 28 introduces pins assigned for 16-bit timer.

Table 28. Pin Assignment of 16-bit Timer: External Pins

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

9.1 16-bit timer block diagram

In this section, 16-bit timer is described in a block diagram in Figure 31.

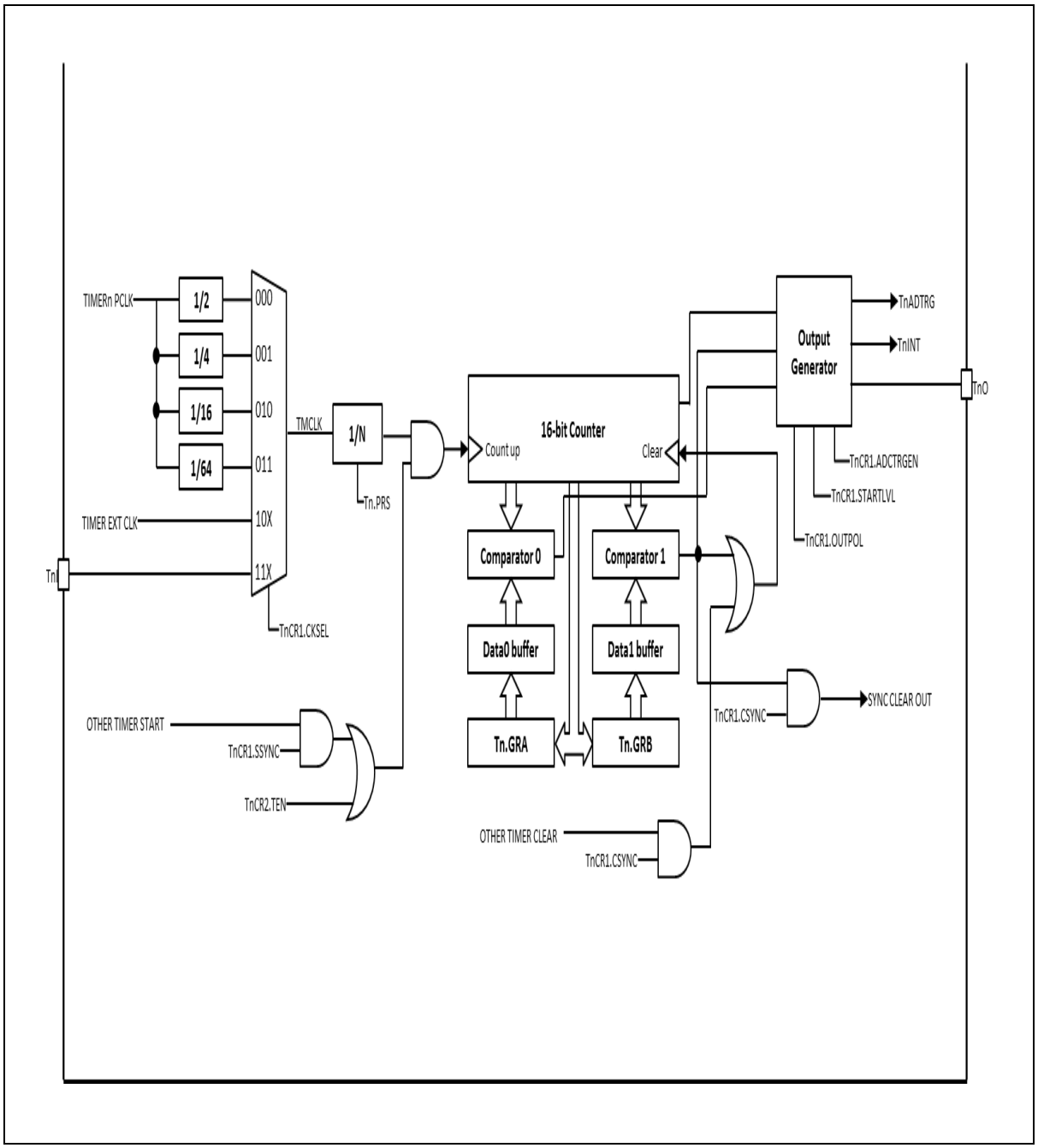


Figure 31. 16-bit Timer Block Diagram

9.2 Registers

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

Table 29. Base Address of 16-bit Timer

Name	Base address
T0	0x4000_3000
T1	0x4000_3020
T2	0x4000_3040
T3	0x4000_3060

Table 30. TIMER Register Map

Name	Offset	Type	Description	Reset value	Reference
Tn.CR1	0x--00	RW	Timer control register 1	0x00000000	9.2.1
Tn.CR2	0x--04	RW	Timer control register 2	0x00000000	9.2.2
Tn.PRS	0x--08	RW	Timer prescaler register	0x00000000	9.2.3
Tn.GRA	0x--0C	RW	Timer general data register A	0x00000000	9.2.4
Tn.GRB	0x--10	RW	Timer general data register B	0x00000000	9.2.5
Tn.CNT	0x--14	RW	Timer counter register	0x00000000	9.2.6
Tn.SR	0x--18	RW	Timer status register	0x00000000	9.2.7
Tn.IER	0x--1C	RW	Timer interrupt enable register	0x00000000	9.2.8

NOTE: n = 0, 1, 2, 3

9.2.1 Tn.CR1: Timer n Control Register 1

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.

Timer Control Register 1 is 16-bit register.

T0.CR1=0x4000_3000, T1.CR1=0x4000_3020

T2.CR1=0x4000_3040, T3.CR1=0x4000_3060

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSYNC	CSYNC	UAO	OUTPOL				ADCTRGEN	STARTLVL	CKSEL			CLRMD		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 at same time of GRA match
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

9.2.2 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

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

7	PWMO	PWM output for read
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 '1'.

9.2.3 Tn.PRS: Timer n Prescaler Register

Timer Prescaler Register is 16-bit register which is used to prescale the counter input clock.

T0.PRS=0x4000_3008, T1.PRS=0x4000_3028

T2.PRS =0x4000_3048, T3.PRS=0x4000_3068

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)

9.2.4 Tn.GRA: Timer 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

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

9.2.5 Tn.GRB: Timer 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

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

9.2.6 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

31	30	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRECLR															
0x0000															
W															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT															
0x0000															
RW															

31 16	PRECLR	Prescaler initialize when timer count value write operation	
		0x00	Prescaler will be initialized when write timer count value on Tn.CNT[15:0]. After writing the count value, prescaler restarted from initial state to make accurate period for first count.
		0xFF	Prescaler will not be initialized and maintain current conditions even writing timer count value on Tn.CNT[15:0]. First count period is not accurate depends on its status when writing operation.
15 0	CNT	Timer count value register	
		R	Read current timer count value
		W	Set count value

9.2.7 Tn.SR: Timer n Status Register

This register indicates the current status of timer module.

Timer Status Register is 8-bit register.

T0.SR=0x4000_3018, T1.SR=0x4000_3038

T2.SR=0x4000_3058, T3.SR=0x4000_3078

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

9.2.8 Tn.IER: Timer n Interrupt Enable Register

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

Timer Interrupt Enable Register is 8-bit register.

T0.IER=0x4000_301C, T1.IER=0x4000_303C

T2.IER=0x4000_305C, T3.IER=0x4000_307C

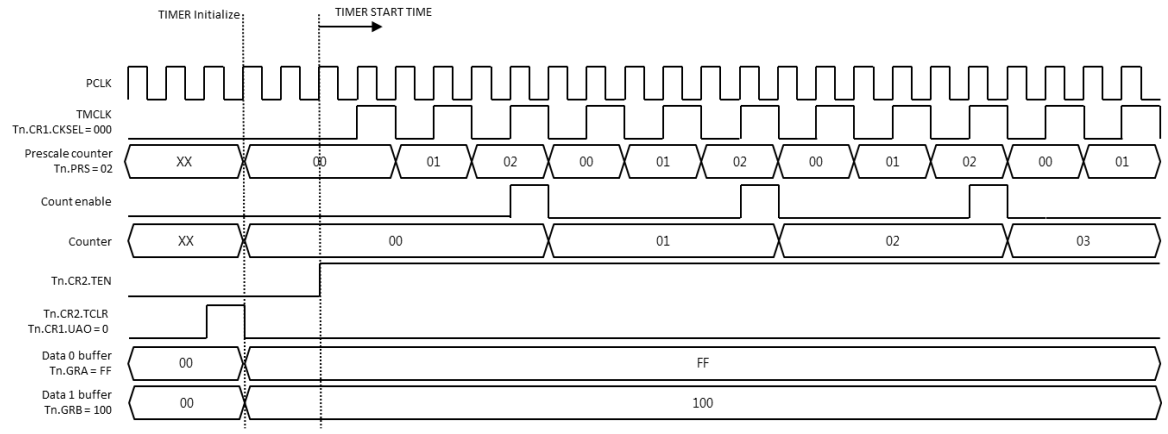
7	6	5	4	3	2	1	0
					MAIE	MBIE	OVIE
0	0	0	0	0	0 RW	0 RW	0 RW

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

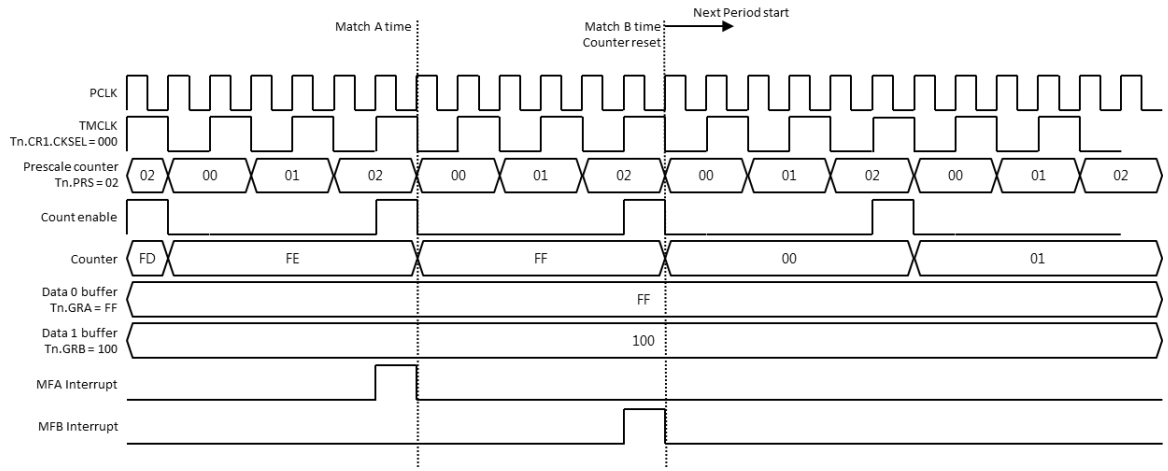
9.3 Functional description

9.3.1 Basic timer operations

TMCLK shown in Figure 32 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.



(b) Timer initialization is executed by the TCLR command, and the timer is started by the TEN command.



(a) Timer will be reset by matching GRB timing and count again from 00.

Figure 32. Basic Start and Match Operations

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.

9.3.2 Normal periodic mode

Figure 33 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.

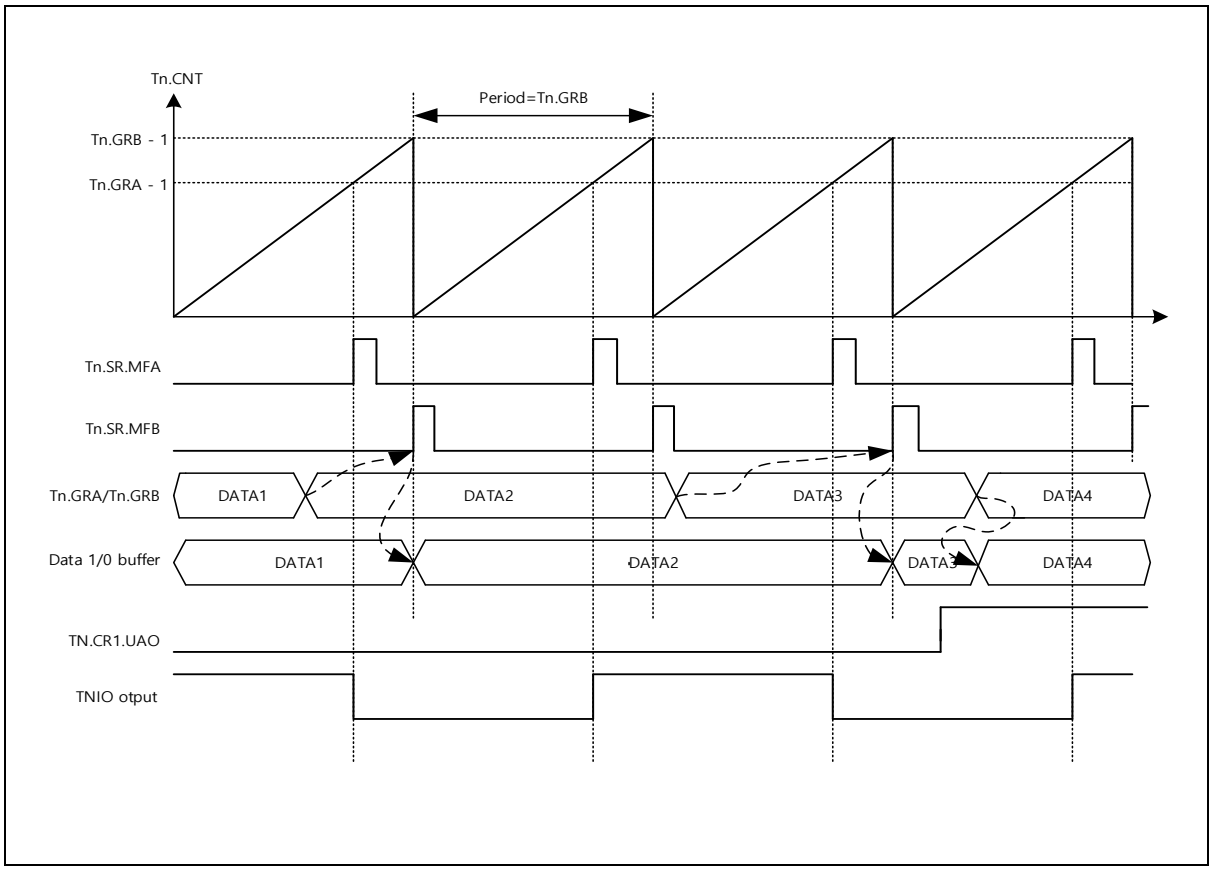


Figure 33. Normal Periodic Mode Operation

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.

9.3.3 One-shot mode

Figure 34 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.

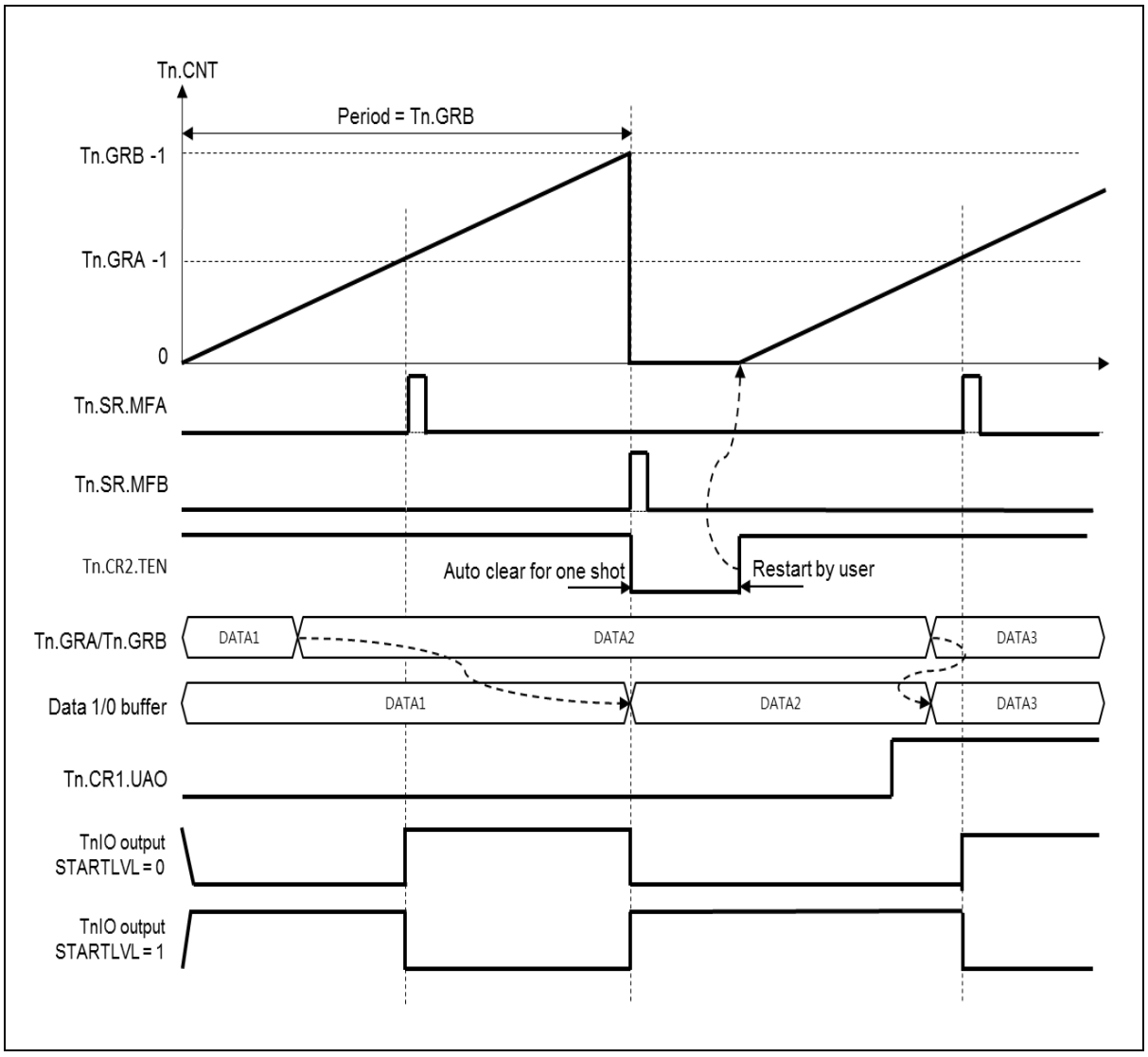


Figure 34. One-Shot Mode Operation

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.

9.3.4 PWM timer output

Figure 35 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.

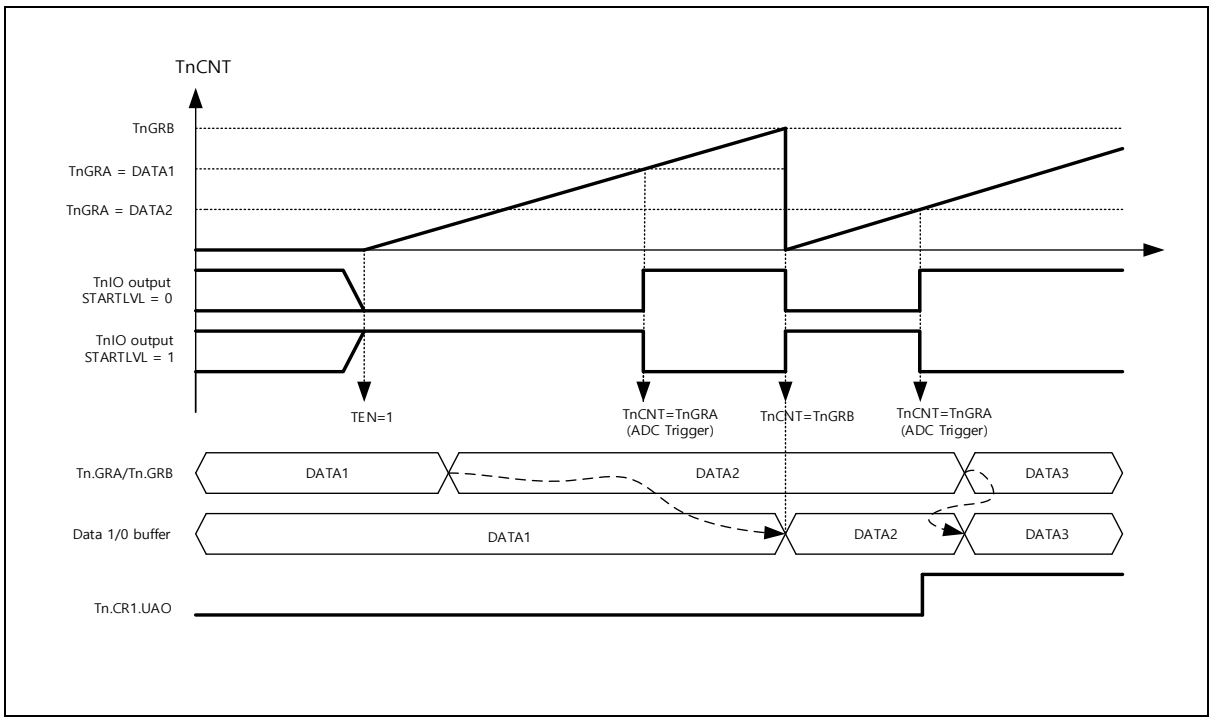


Figure 35. PWM Output Operation

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.

9.3.5 PWM synchronization

2-PWM outputs are usually used as synchronous PWM signal control. This function is provided with synchronous start. Figure 36 shows synchronous PWM generation.

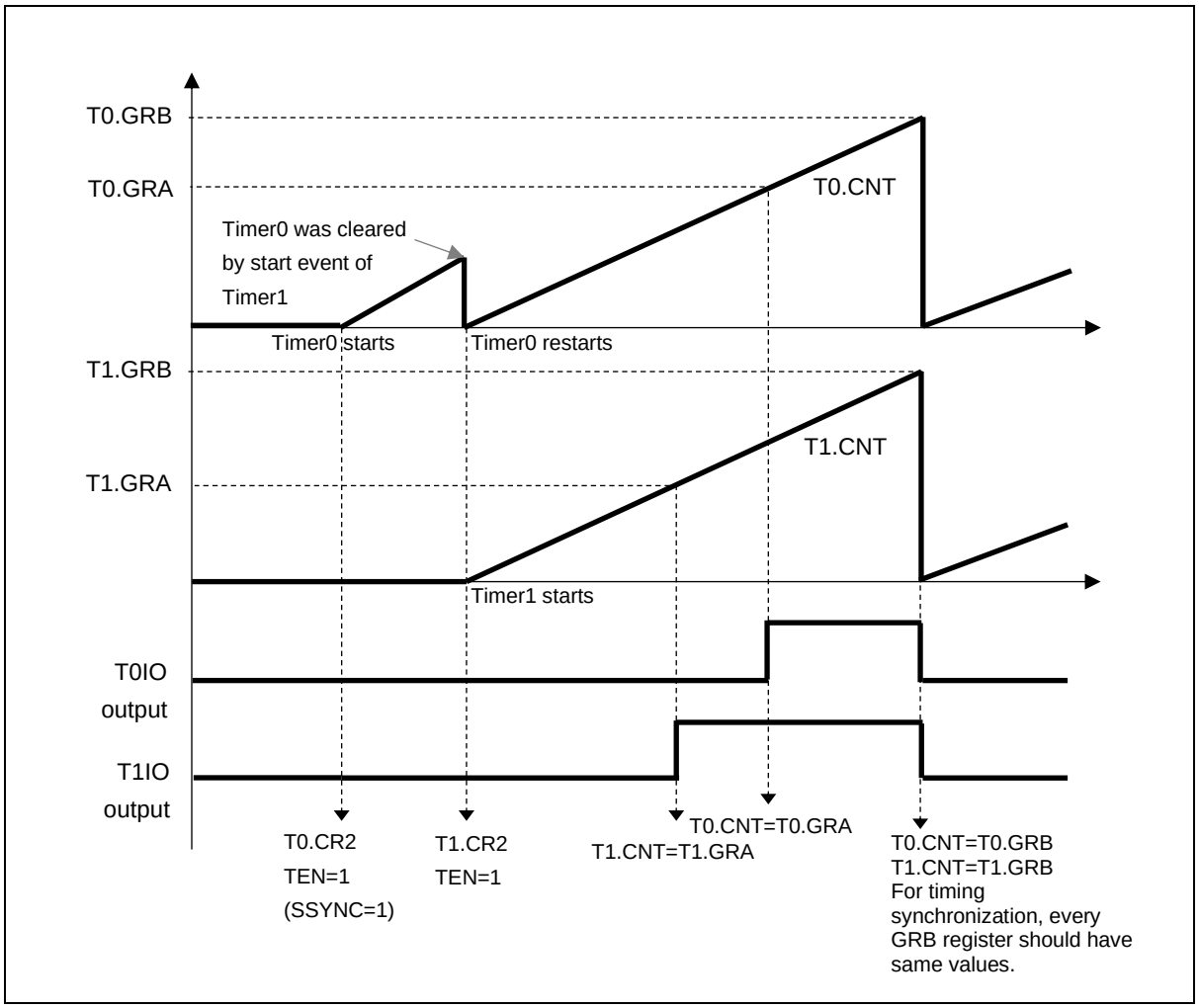


Figure 36. Timer Synchronization Example (When SSYNC = 1)

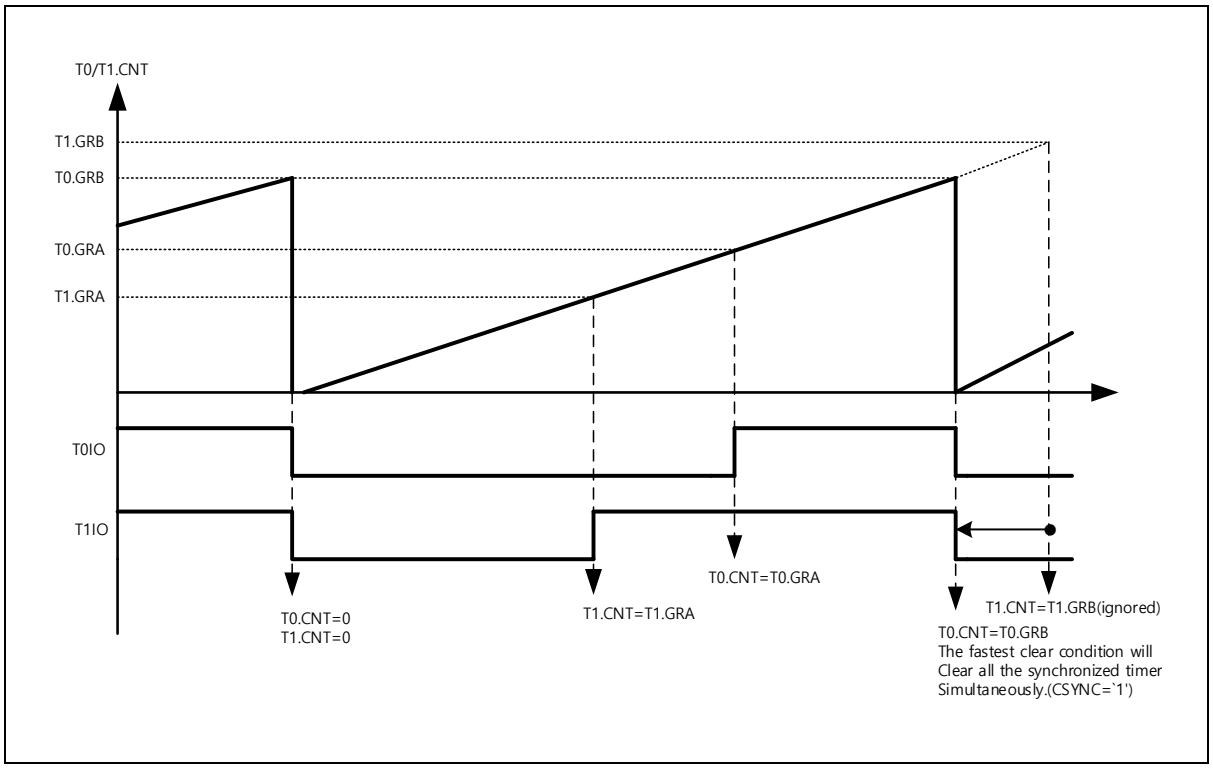


Figure 37. Timer Synchronization Example (When CSYNC = 1)

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.

9.3.6 Capture mode

Figure 38 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.

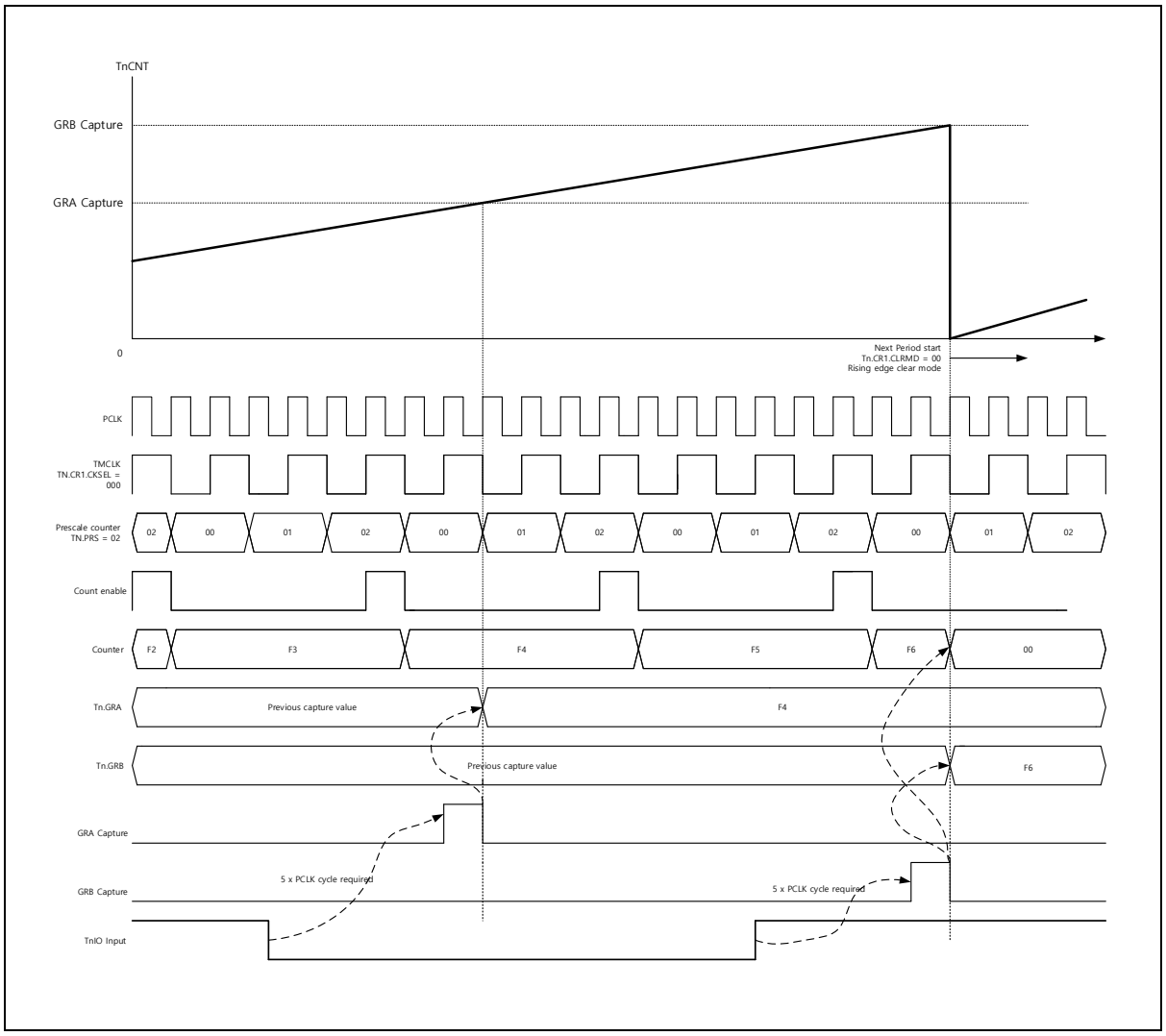


Figure 38. Capture Mode Operation

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.

The example case in Figure 38 describes the rising edge clear mode.

9.3.7 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. (Figure39 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.

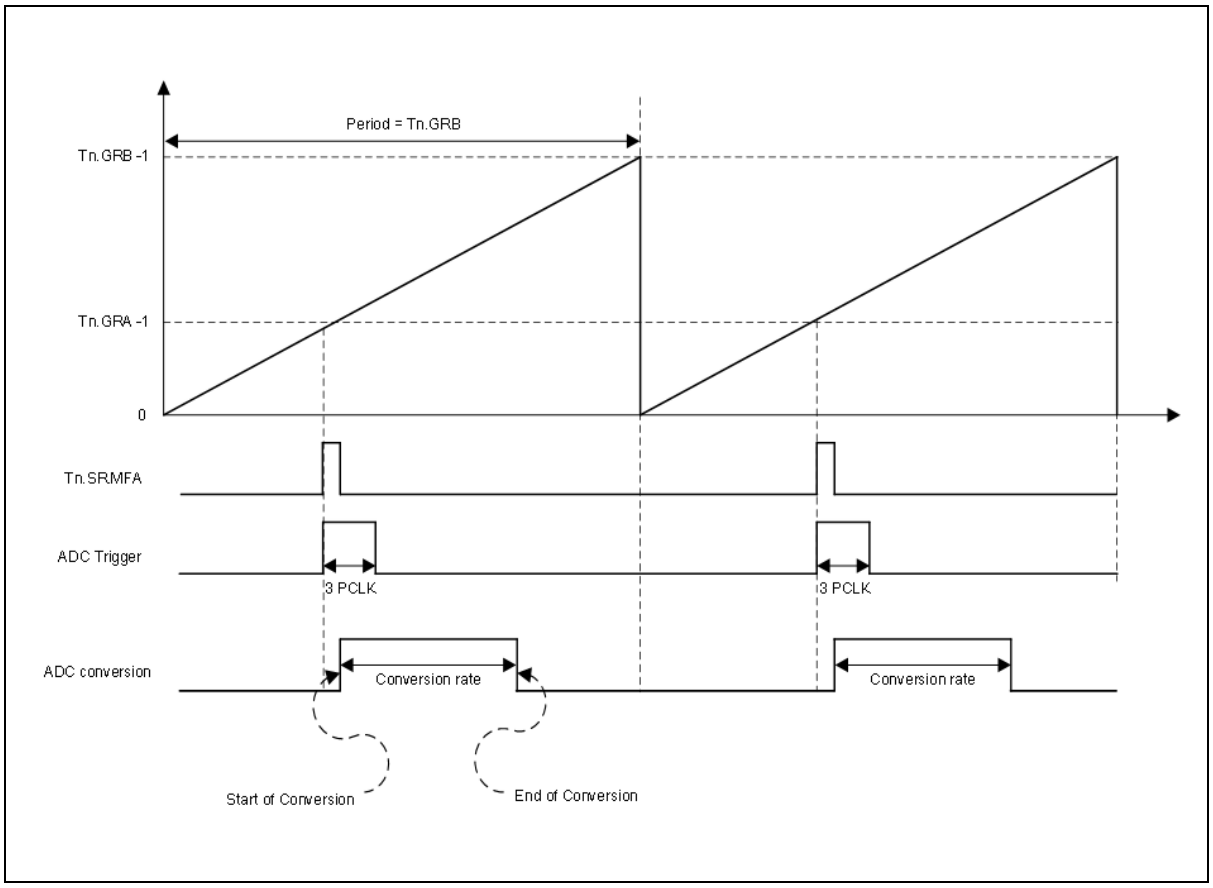


Figure 39. ADC Triggering Timing Diagram

10. Free Run Timers (FRT)

The AC30M1x64/AC30M1x32 series has one free-run timers (FRTs) built in, which are 32-bit up-count timers. These timers can run with the overflow or match interrupt according to their uses and can remain active in stop mode.

FRT of AC30M1x64/AC30M1x32 series features the followings:

- 32-bit up-count timers
- FRT match interrupts supported

10.1 FRT block diagram

In this section, FRT block diagram is introduced in Figure 40.

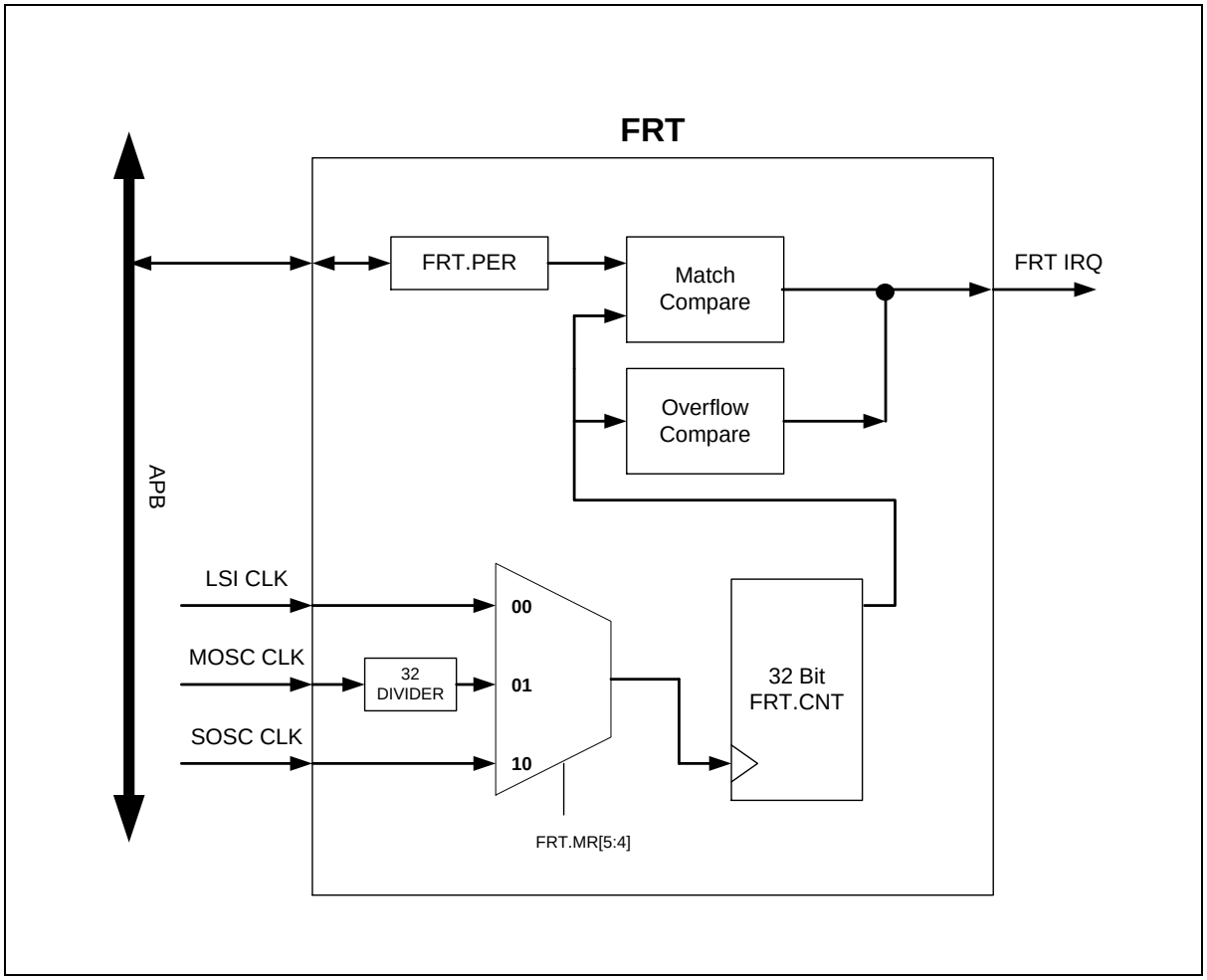


Figure 40. FRT Block Diagram

10.2 Registers

Base address of FRT block is introduced in the followings:

Table 31. Base Address of FRT Interface

Name	Base address
FRT	0x4000_0600

Table 32. FRT Register Map

Name	Offset	Type	Description	Reset value	Reference
FRT.MR	0x0000	RW	FRT mode register	0x00000000	10.2.1
FRT.CR	0x0004	RW	FRT control register	0x00000000	10.2.2
FRT.PER	0x0008	RW	FRT period match register	0x00000000	10.2.3
FRT.CNT	0x000C	RO	FRT counter register	0x00000000	10.2.4
FRT.SR	0x0010	RW	FRT status register	0x00000000	10.2.5

10.2.1 FRT.MR: FRT Mode Register

FRT is a 32-bit up counter. It can be used in power down mode when using SUB OSC. The SUB OSC clock is directly connected to FRT. Timer Control Register is 8-bit register.

FRT.MR=0x4000_0600

7	6	5	4	3	2	1	0
		CLKSEL			MCD	OVIE	MIE
0	0	0	0		0	0	0
		RW	RW		RW	RW	RW

5	CLKSEL	FRT counter clock source control	
4		0	Low Speed Internal Oscillator clock (40kHz)
		1	External Oscillator clock divided by 32
		2	Sub Oscillator clock
		3	Reserved
2	MCD	Counter Match Clear Disable bit	
		0	Counter Match Clear function is enabled. Whenever the counter matches FRT.PER, the counter will be set zero and waiting for MF to be cleared.
		1	Counter Match Clear function is disabled. The counter will keep countering without set zero
1	OVIE	Over Flow Interrupt Enable bit	
		0	Not effect
		1	Interrupt enabled
0	MIE	Match Interrupt Enable bit	
		0	Not effect
		1	Interrupt enabled

10.2.2 FRT.CR: FRT Control Register

FRT Control Register is 8-bit register.

FRT.CR=0x4000_0604

7	6	5	4	3	2	1	0
				RREQ	CLR	HOLD	EN
0	0	0	0	0	0	0	0
				RW	WO	RW	RW

3	RREQ	FRT Counter read request bit
	0	No action
	1	Request to read FRTn.CNT (cleared when CNTACK(FSR[1]) is high)
2	CLR	FRT Counter register clear bit
	0	No action
	1	Clear the counter
1	HOLD	FRT Counter register hold bit
	0	No action
	1	Hold the counter
0	EN	FRT enable bit
	0	FRT Disabled
	1	FRT Enabled

10.2.3 FRT.PER: FRT Period Match Register

FRT Period Match Register is 32-bit register

FRT.PER=0x4000_0608

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
PERIOD			
0x0000_0000			
RW			

32 0	PERIOD	FRT Period Match Data

10.2.4 FRT.CNT: FRT Counter Register

FRT Counter Register is 32-bit register

FRT.CNT=0x4000_060C

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
CNT			
0x0000_0000 RO			

32 0	CNT	FRT Counter

10.2.5 FRT.SR: FRT Status Register

FRT Status Register is 8-bit register.

FRT.SR=0x4000_0610

7	6	5	4	3	2	1	0
					RACK	OVF	MF
0	0	0	0	0	0 WC1	0 WC1	0 WC1

2	RACK	Read Counter Acknowledge bit	
		0	Not ready to read CNT value
		1	Ready to read CNT value
1	OVF	OverFlow Interrupt flag bit	
		0	Overflow interrupt did not occur
		1	Overflow interrupt occurred
0	MF	Interrupt flag bit	
		0	Match interrupt did not occur.
		1	Match Interrupt occurred In Counter Match Clear mode, this bit should be cleared for restarting the counter.

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 channelUART of AC30M1x64/AC30M1x32 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

Table 33 introduces pins assigned for the UART.

Table 33. 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.

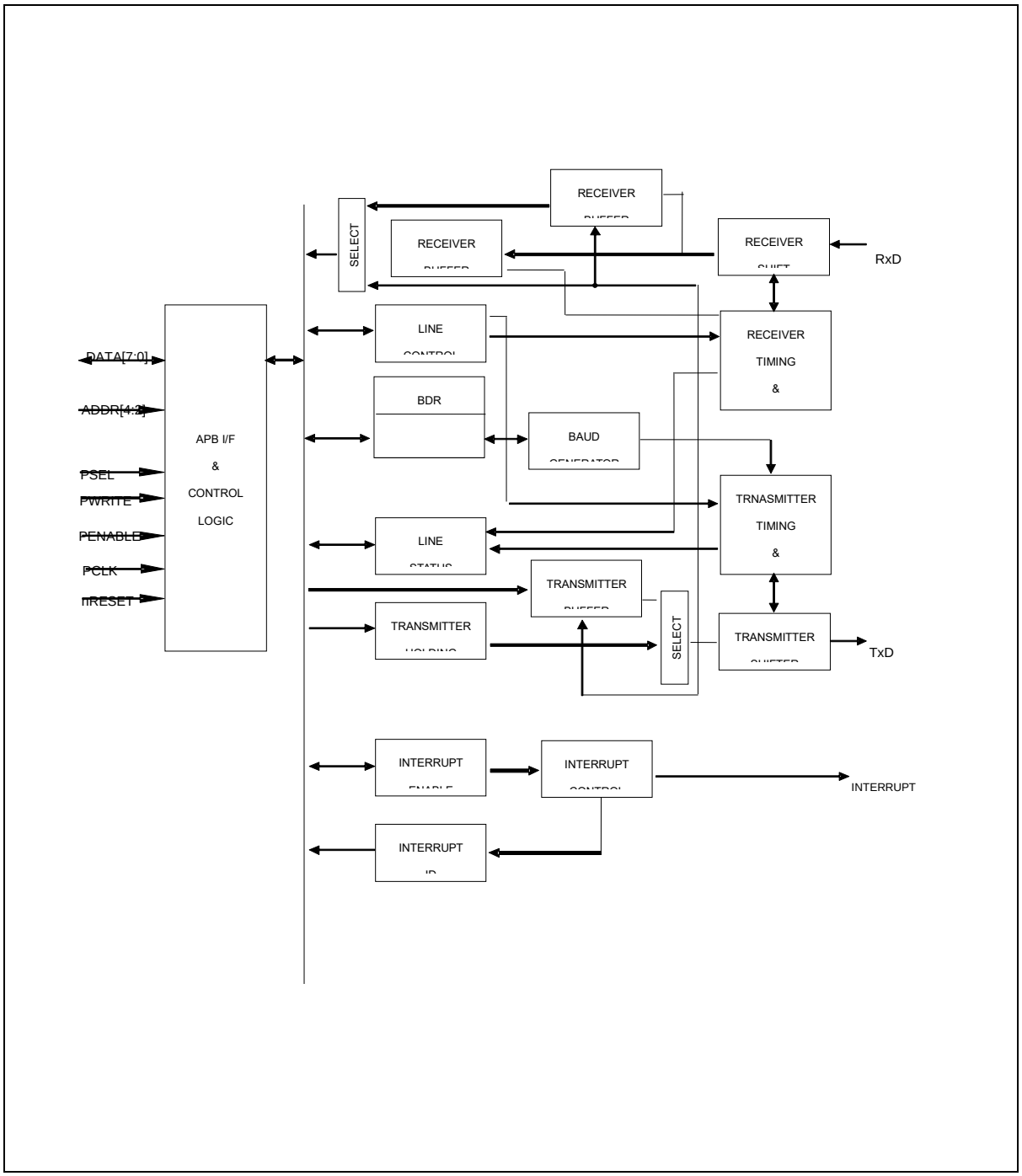


Figure 41. UART Block Diagram

11.2 Registers

Base address of UART is introduced in the followings:

Table 34. Base Address of UART

Name	Base address
U0	0x4000_8000
U1	0x4000_8100

Table 35. 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		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 Un.THR 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
-	-	-	-	TXEIE	RLSIE	THREIE	DRIE
0	0	0	0	0 RW	0 RW	0 RW	0 RW

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 Un.IIR: UART Interrupt ID Register

UART Interrupt ID Register is 8-bit register.

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

23	22	21	20	19	18	17	16
				-	RLS	THRE	DR
0	0	0	0	0	0	0	0
			R	R	R	R	R
15	14	13	12	11	10	9	8
				0			
7	6	5	4	3	2	1	0
			TXE	IID			IPEN
0	0	0	0	000			0
			R	R			R

18	RLS	Receiver line status flag
17	THRE	Transmit holding register empty flag
16	DR	Data receive interrupt flag
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 empty interrupt

11.2.5 Un.LCR: UART 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[1:0]	
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 Un.LCR register. The table shows the variation of parity bit generation.

Table 36. Interrupt ID and Control

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 set 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
			LBON	RXINV	TXINV		
0	0	0	0 RW	0 RW	0 RW	0	0

4	LBON	Local loopback test mode enable
	0	Normal mode
	1	Local loopback mode (TxD connected to RxD internally)
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

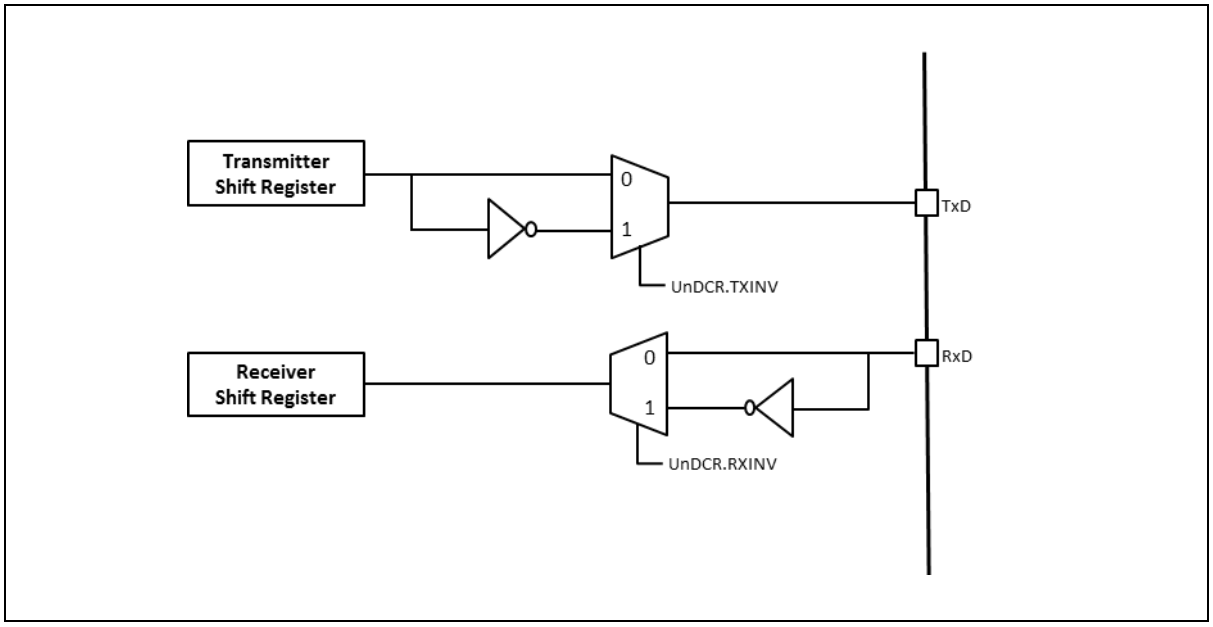


Figure 42. Data Inversion Control Diagram

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 Un.IEN register is set. Other bits can generate its interrupt when its interrupt enable bit in Un.IEN 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. $UART_{clock}$ gets from MCCR7.
Baud rate calculation formula is below.

$$\text{BDR} = \frac{UART_{clock}}{16 \times BaudRate}$$

In case of 40 MHz $UART_{clock}$ speed, the divider value and error rate is described in Table 37.

Table 37. Example of Baud Rate Calculation (without BFR)

$UART_{clock}$=40 MHz		
Baud rate	**Divider**	**Error (%)**
1200	2083	0.02%
2400	1041	0.06%
4800	520	0.16%
9600	260	0.16%
19200	130	0.16%
38400	65	0.16%
57600	43	0.94%
115200	21	3.34%

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 38. Example of Baud Rate Calculation

$UART_{clock}$ =40 MHz			
Baud rate	**Divider**	**FCNT**	**Error (%)**
1200	2083	85	0.00%
2400	1041	170	0.00%
4800	520	213	0.00%
9600	260	106	0.00%
19200	130	53	0.00%
38400	65	262	0.00%
57600	43	103	0.00%
115200	21	179	0.01%

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

FCNT value can calculated above equation. For example, the target baud rate is 4800 bps and $UART_{clock}$ is 40MHz case, the BDR value is 520.8333. The integer number 520 should be the BDR value and the floating number 0.8333 will make the FNCT value as below:

FCNT = 0.8333 * 256 = 213.3333, so the FCNT value is 213.

8-bit fractional counter will count up by FCNT value every (baud rate)/16 periods and whenever fractional counter overflow is happen, the divisor value will increment by 1. So this period will be compensated. Then next period, the divisor value will return to original set value.

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 2 continuous transmits.

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

7	6	5	4	3	2	1	0
SMS	DMS				WAITVAL		
1	0	0	0	0	000		
RW	RW				RW		

7	SMS	Start Bit Multi sampling enable
		0 Multi sampling is disable for start bit, Single sample will be done at 8/16 baud rate for the start bit
		1 Multi sampling is enabled for start bit. Sampling is done 3 times at 7/16, 8/16 and 9/16 baud rate. Dominant value of 3 samples will be selected for the start bit
6	DMS	Data Bit Multi sampling enable
		0 Multi sampling is disable for data bit, Single sample will be done at 8/16 baud rate for the data bit
		1 Multi sampling is enabled for data bit. Sampling is done 3 times at 7/16, 8/16 and 9/16 baud rate. Dominant value of 3 samples will be selected for the data bit
2 0	WAITVAL	Wait time is decided by this value $\text{Wait Time} = \frac{WAITVAL}{BAUDRATE}$

11.3 Functional description

The UART module is compatible with 16450 UART. A fractional baud rate compensation logic is provided additionally.

Since it doesn't have internal FIFO block, data transfer will establish interactive support.

11.3.1 Receive data sampling timing

The UART operates to the timing shown in Figure 43. In the figure, the UART judges the falling edge on the receive line as the start bit. From the start timing, UART oversamples 16 times of 1-bit and detects the bit value at the 7th sample of 16 samples.

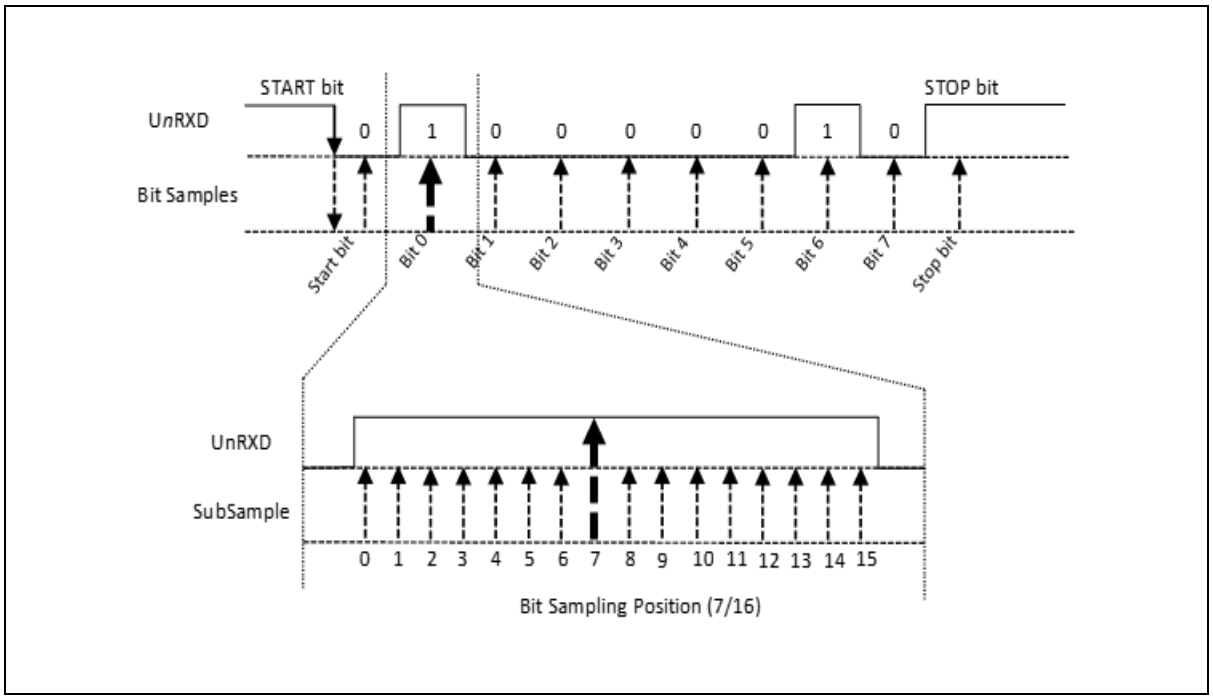


Figure 43. Sampling Timing of a UART Receiver

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.

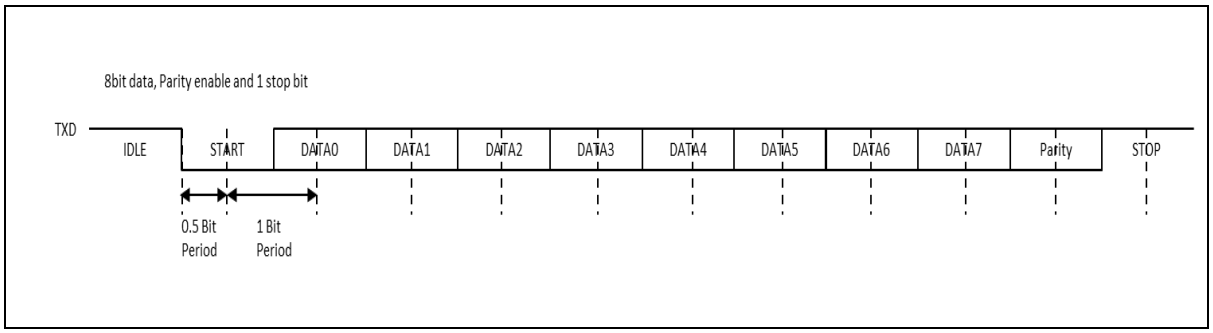


Figure 44. Transmit Data Format Example

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.

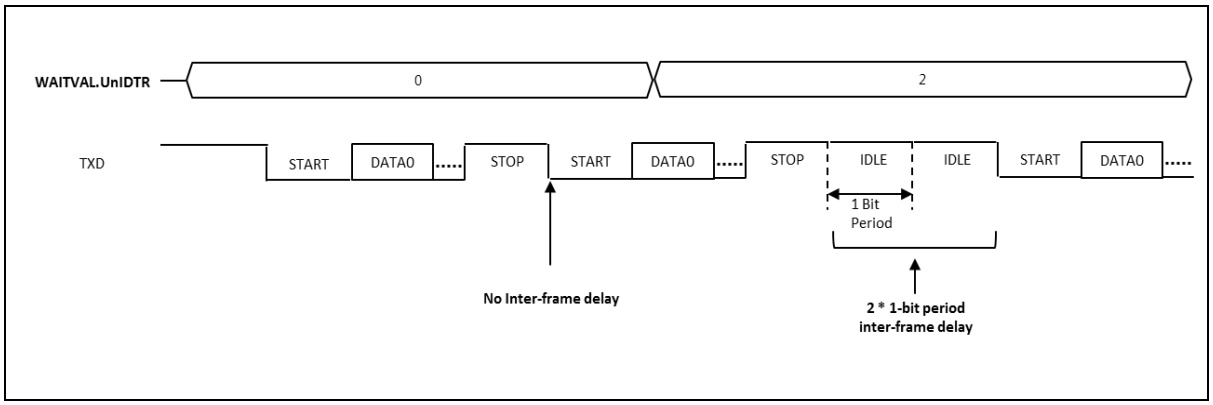


Figure 45. Inter-Frame Delay Timing Diagram

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.

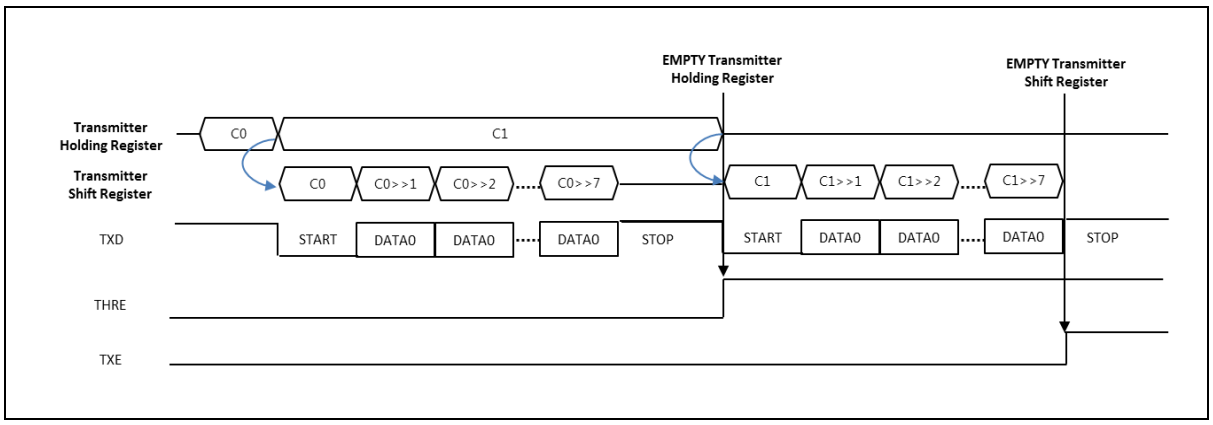


Figure 46. Transmit Interrupt Timing Diagram

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.

Table 39. 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 47.

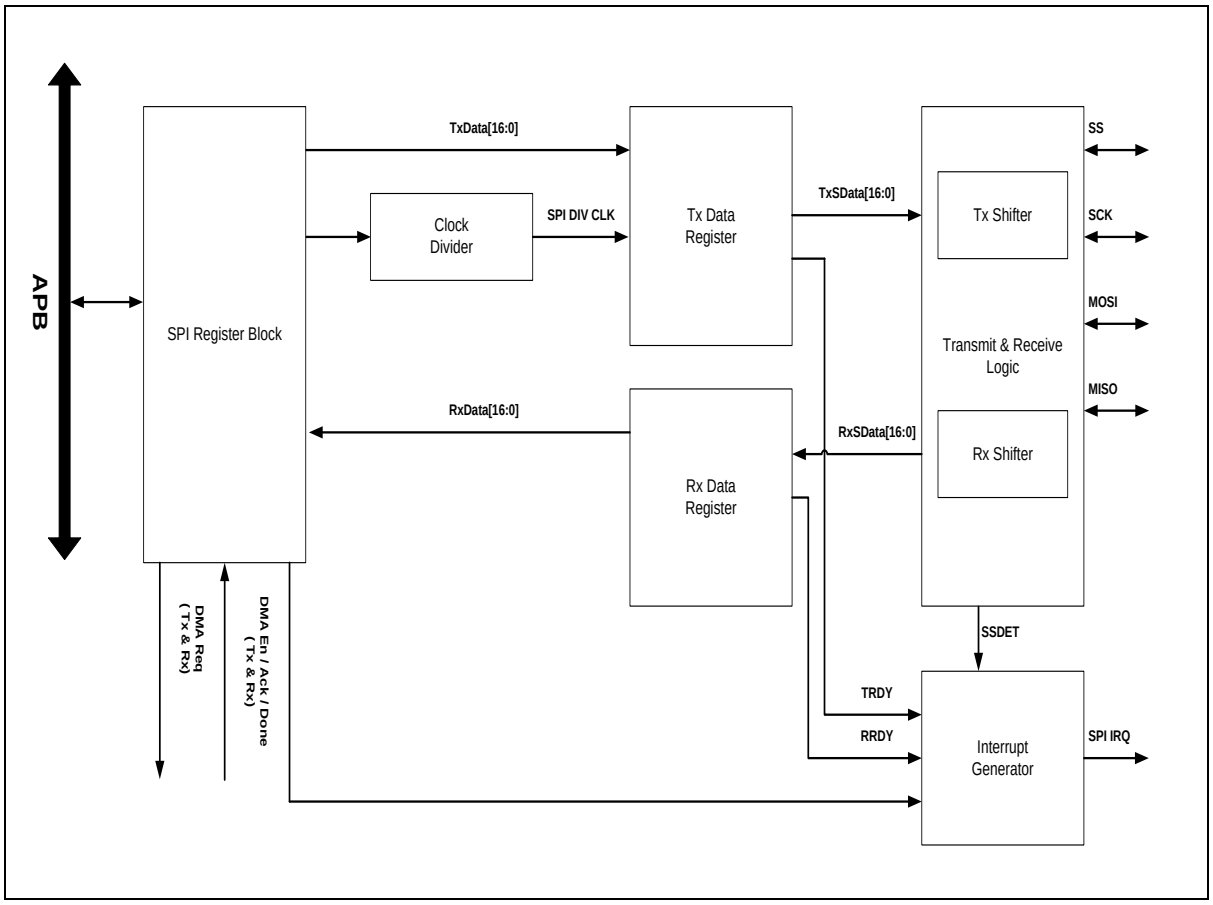


Figure 47. SPI Block Diagram

12.2 Registers

Base address of SPI is introduced in the followings:

Table 40. Base Address of SPI

Name	Base address
SPI	0x4000_9000

Table 41. SPI Register Map

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

12.2.1 SP.TDR: SPI Transmit Data Register

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

SP.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 SP.RDR: SPI Receive Data Register

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

SP.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 SP.CR: SPI Control Register

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

SP.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			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	

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
16	SSCIE	SS Edge Change Interrupt Enable bit.
		0 nSS interrupt is disabled.
		1 nSS interrupt is enabled for both edges (LàH, HàL)
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 (SP.CR[12]). SS signal is in normal operation mode.
		1 SS output signal is set by SSOUT.
12	SSOUT	SS output signal select bit.
		0 SS output is 'L.'
		1 SS output is 'H'.
11	LBE	Loop-back mode select bit in master mode.
		0 Loop-back mode is disabled.
		1 Loop-back mode is enabled.
10	SSMASK	SS signal masking bit in slave mode.
		0 SS signal masking is disabled. Receive data when SS signal is active.
		1 SS signal masking is enabled. Receive data at SCLK edges. SS signal is ignored.
9	SSMO	SS output signal select bit.
		0 SS output signal is disabled.
		1 SS output signal is enabled.
8	SSPOL	SS signal Polarity select bit.
		0 SS signal is Active-Low.
		1 SS signal is Active-High.
5	MS	Master/Slave select bit.
		0 SPI is in Slave mode.
		1 SPI is in Master mode.
4	MSBF	MSB/LSB Transmit select bit.
		0 LSB is transferred first.
		1 MSB is transferred first.
3	CPHA	SPI Clock Phase bit.
		0 Sampling of data occurs at odd edges (1,3,5,…,15).

		1	Sampling of data occurs at even edges (2,4,6,…,16).
2	CPOL	SPI Clock Polarity bit.	
		0	Active-high clocks selected.
		1	Active-low clocks selected.
1	BITSZ	Transmit/Receive Data Bits select bit.	
0		0 0	8 bits
		0 1	9 bits
		1 0	16 bits
		1 1	17 bits

- CPOL=0, CPHA=0: data sampling at rising edge, data changing at falling edge
- CPOL=0, CPHA=1: data sampling at falling edge, data changing at rising edge
- CPOL=1, CPHA=0: data sampling at falling edge, data changing at rising edge
- CPOL=1, CPHA=1: data sampling at rising edge, data changing at falling edge

12.2.4 SP.SR: SPI Status Register

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

SP.SR=0x4000_9008

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

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 SP.RDR.
3	UDRF	Transmit Underrun Error flag.
		0 Transmit Underrun is not occurred.
		1 Transmit Underrun is occurred. This bit is cleared by writing or reading SP.TDR.
2	TXIDLE	Transmit/Receive Operation flag.
		0 SPI is transmitting data
		1 SPI is in IDLE state.
1	TRDY	Transmit buffer Empty flag.
		0 Transmit buffer is busy.
		1 Transmit buffer is ready. This bit is cleared by writing data to SP.TDR.
0	RRDY	Receive buffer Ready flag.
		0 Receive buffer has no data.
		1 Receive buffer has data. This bit is cleared by writing data to SP.RDR.

12.2.5 SP.BR: SPI Baud Rate Register

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

SP.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 SP.EN: SPI Enable Register

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

SP.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. SP.SR 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 SP.TDR before this bit is active.

12.2.7 SP.LR: SPI Delay Length Register

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

SP.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	StoP Length value
16		0x01 ~ 0xFF: 1 ~ 255 SCLKs. (SPL ≥ 1)
15	BTL	BursT Length value
8		0x01 ~ 0xFF: 1 ~ 255 SCLKs. (BTL ≥ 1)
7	STL	STart Length value
0		0x01 ~ 0xFF: 1 ~ 255 SCLKs. (STL ≥ 1)

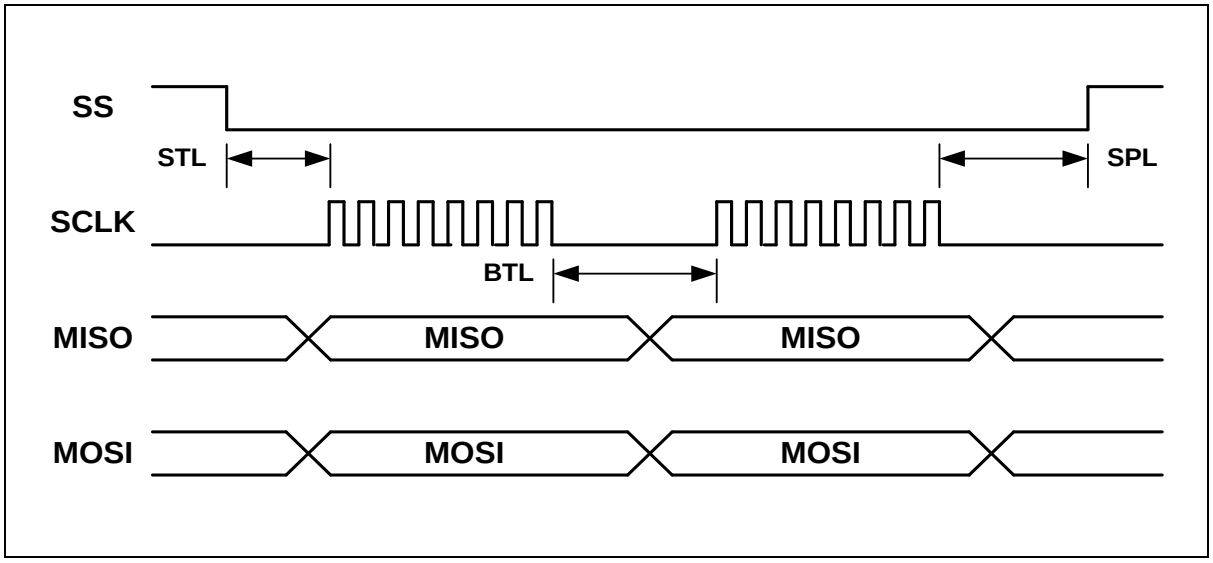


Figure 48. SPI Wave form (STL, BTL and SPL)

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 49 and Figure 50. 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.

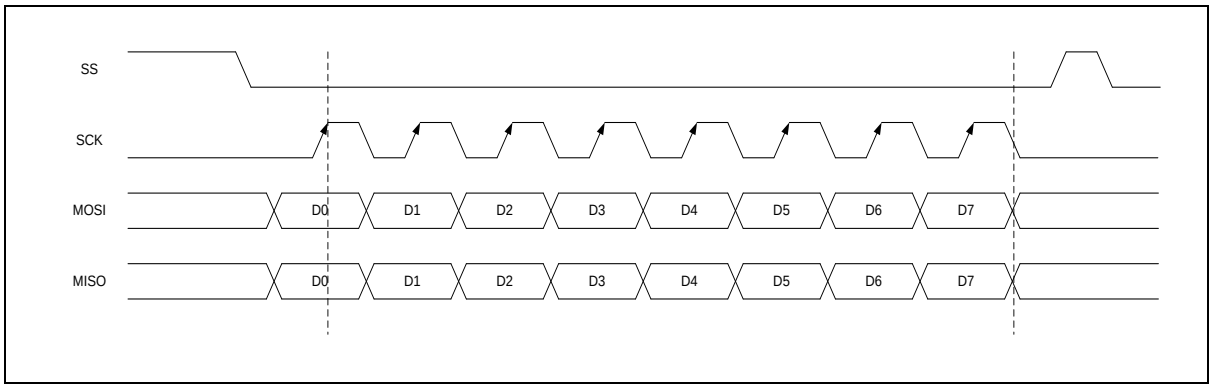


Figure 49. SPI Transfer Timing 1/4 (CPHA = 0, CPOL = 0, MSBF = 0)

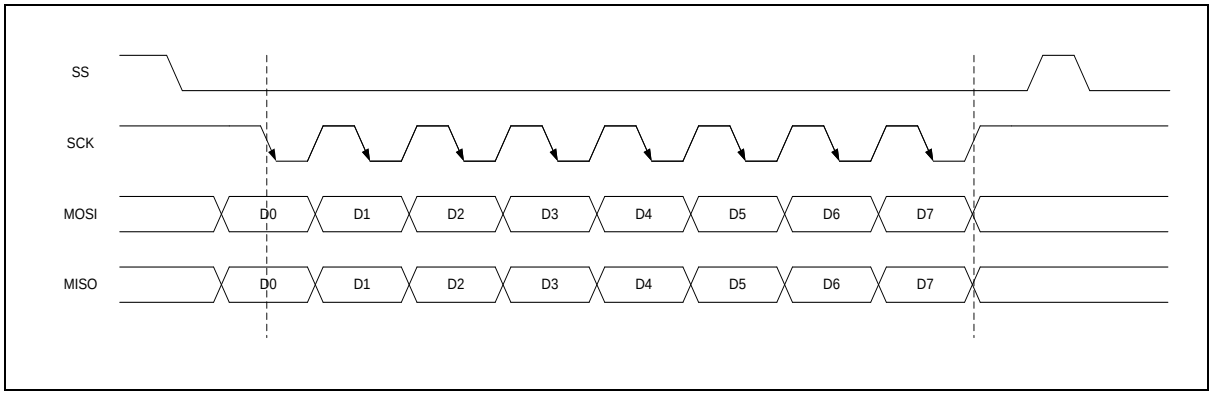


Figure 50. SPI Transfer Timing 2/4 (CPHA = 0, CPOL = 1, MSBF = 1)

The timing of a SPI transfer where CPHA is one is shown in Figure 51 and Figure 52.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 49 and Figure 50, 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.

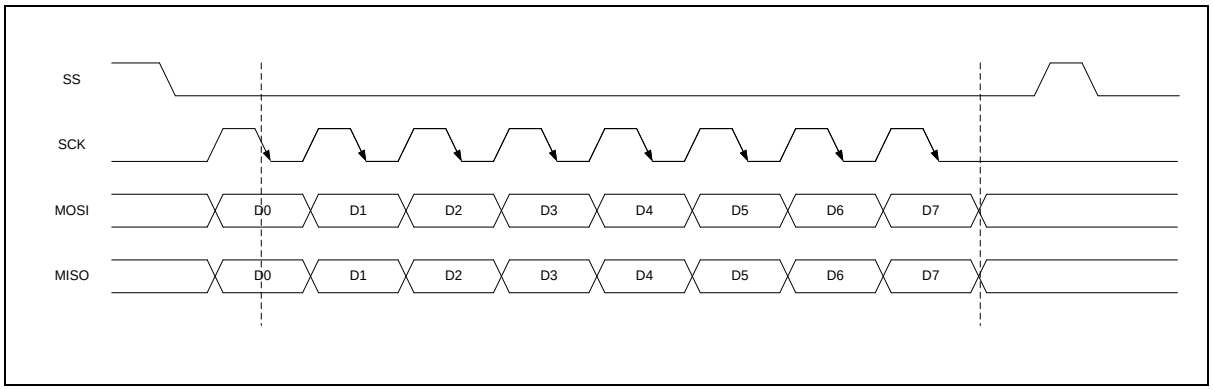


Figure 51. SPI Transfer Timing 3/4 (CPHA = 1, CPOL = 0, MSBF = 0)

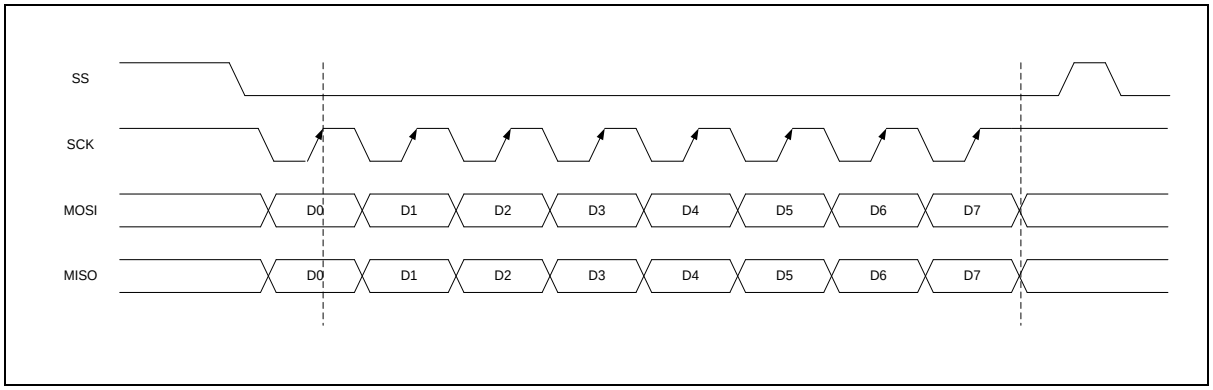


Figure 52. SPI Transfer Timing 4/4 (CPHA = 1, CPOL = 1, MSBF = 1)

13. Inter Integrated Circuit (I2C)

I2C (Inter-Integrated Circuit) 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 features the followings:

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

Table 42. 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.

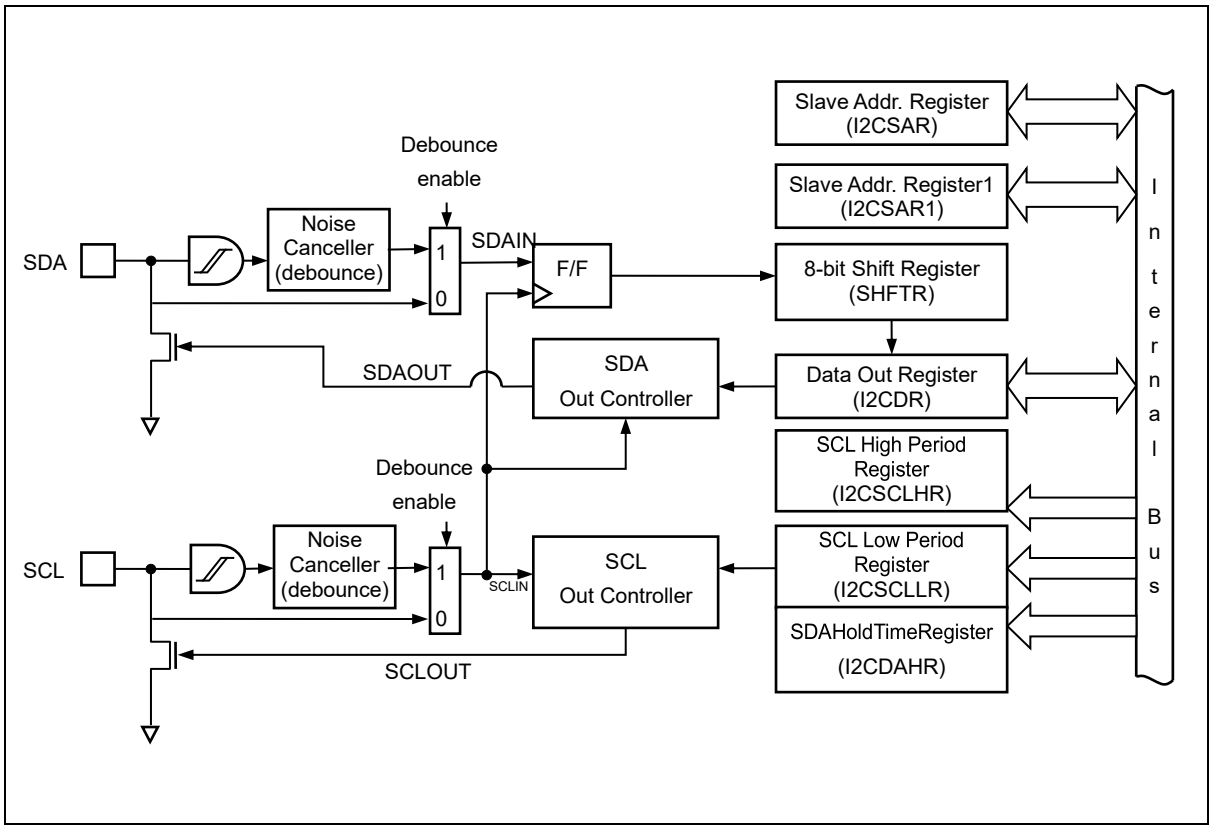


Figure 53. I2C Block Diagram

13.2 Registers

Base address of I2C is introduced in the followings:

Table 43. Base Address of I2C Interface

Name	Base address
I2C	0x4000_A000

Table 44. I2C Register Map

Name	Offset	Type	Description	Reset value	Reference
IC.DR	0x00	RW	I²C Data Register	0xFF	13.2.1
IC.SR	0x08	RW	I²C Status Register	0x00	13.2.2
IC.SAR	0x0C	RW	I²C Slave Address Register	0x00	13.2.3
IC.CR	0x14	RW	I²C Control Register	0x00	13.2.4
IC.SCLL	0x18	RW	I²C SCL LOW duration Register	0xFFFF	13.2.5
IC.SCLH	0x1C	RW	I²C SCL HIGH duration Register	0xFFFF	13.2.6
IC.SDH	0x20	RW	I²C SDA Hold Register	0x7F	13.2.7

13.2.1 IC.DR: I²C Data Register

IC.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.

IC.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 IC.SR: I²C Status Register

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

IC.SR=0x4000_A008

7	6	5	4	3	2	1	0
GCALL	TEND	STOP	SSEL	MLOST	BUSY	TMODE	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 IC.SAR:I²C Slave Address Register

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

IC.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 IC.CR: I²C Control Register

IC.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.

IC.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.

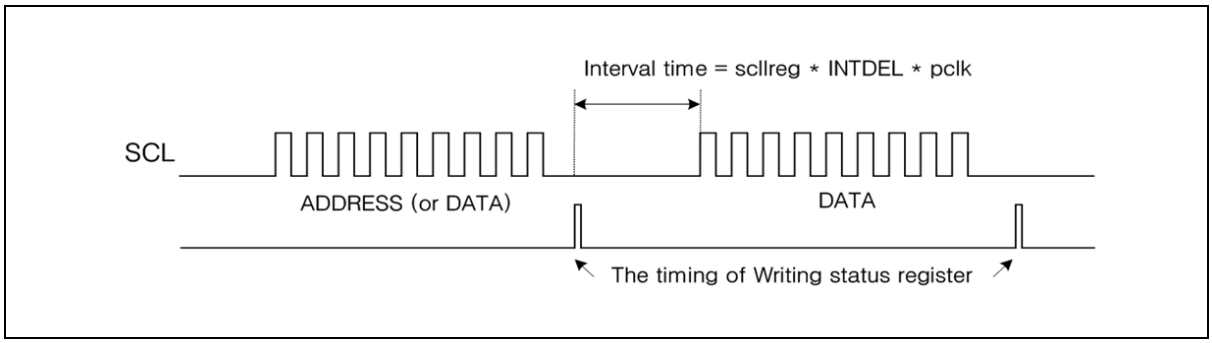


Figure 54. INTDEL in Master Mode

13.2.5 IC.SCLL: I²C SCL LOW Duration Register

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

IC.SCLL=0x4000_A018

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

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

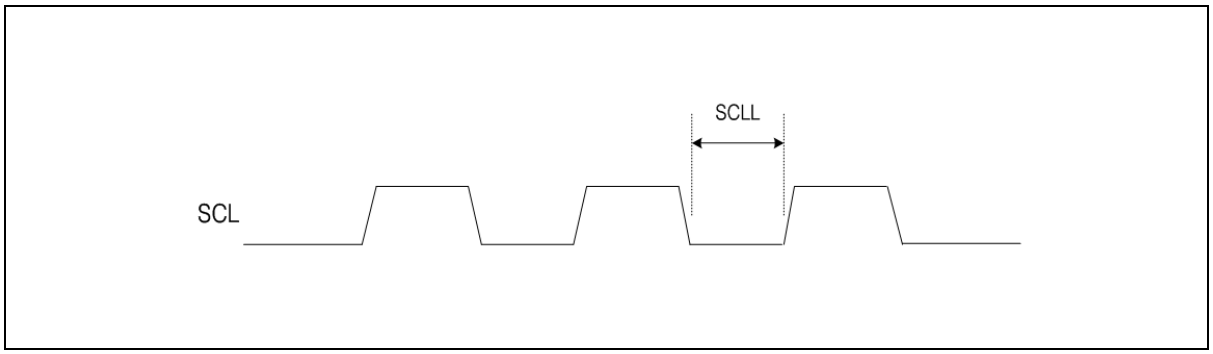


Figure 55. SCL LOW Timing

13.2.6 IC.SCLH: I²C SCL HIGH Duration Register

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

IC.SCLH=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.

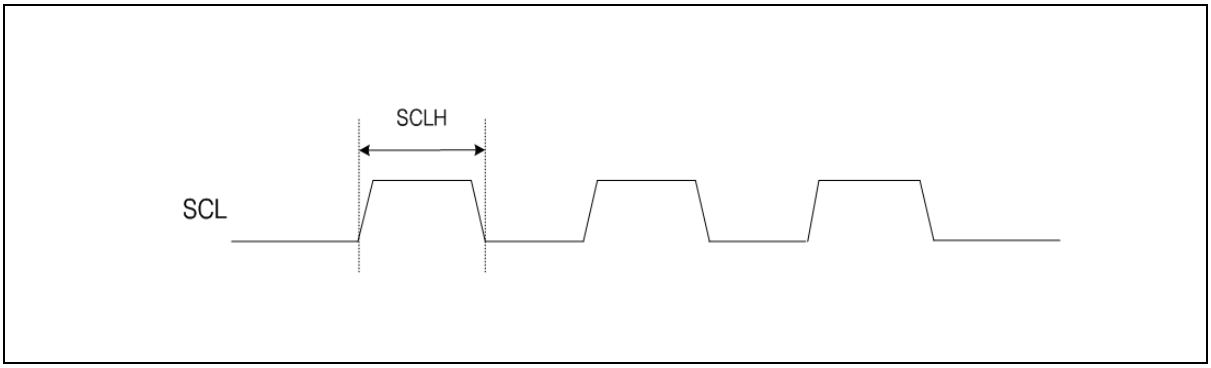


Figure 56. SCL HIGH Timing

13.2.7 IC.SDH:SDA Hold Register

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

IC.SDH=0x4000_A020

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDH														
	0x7FFF														
	RW														

14	SDH	SDA HOLD time setting value. SDH = (PCLK * SDH[14:0]) + 4 PCLKs
0		Default value is 0x7FFF.

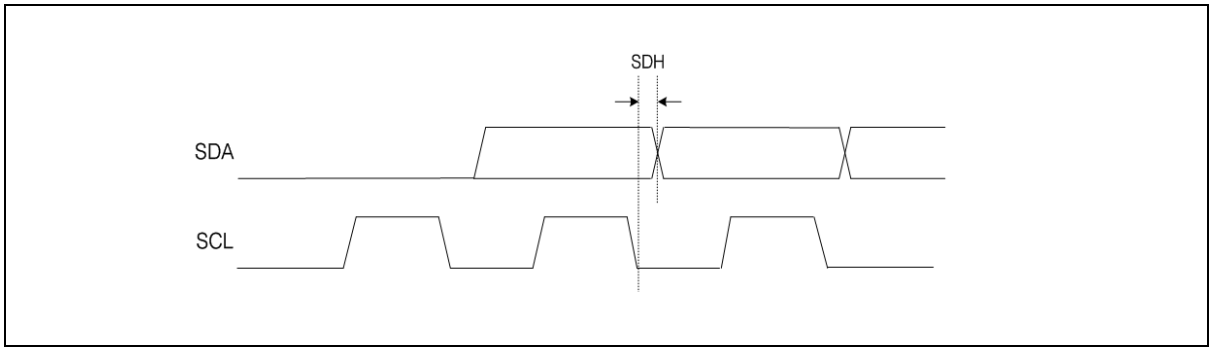


Figure 57. SDA HOLD Timing

13.3 Functional description

13.3.1 I2C bit transfer

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" (Refer to Figure 58).

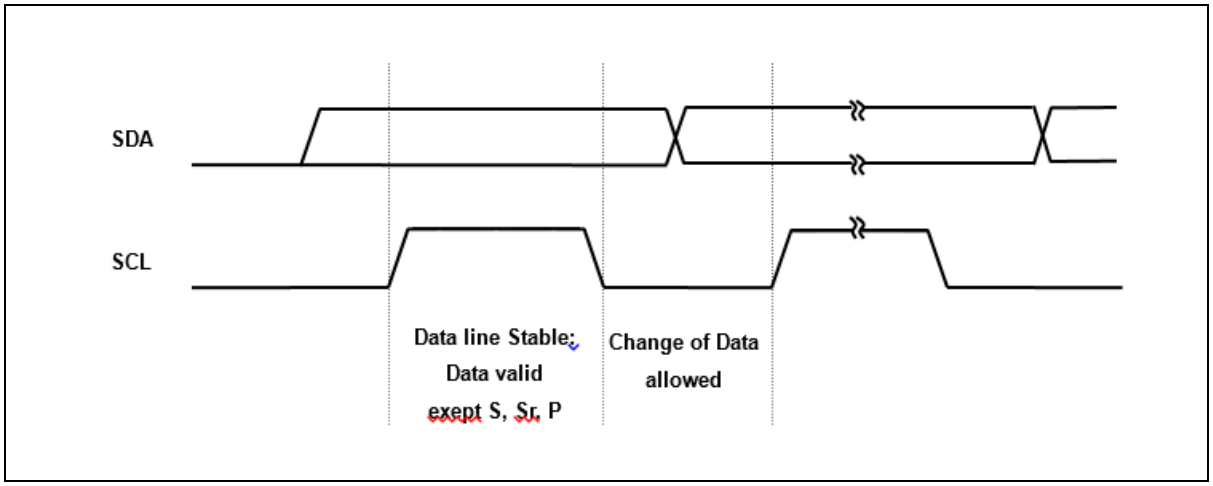


Figure 58. I²C Bus Bit Transfer

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 (Refer to Figure 59).

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.

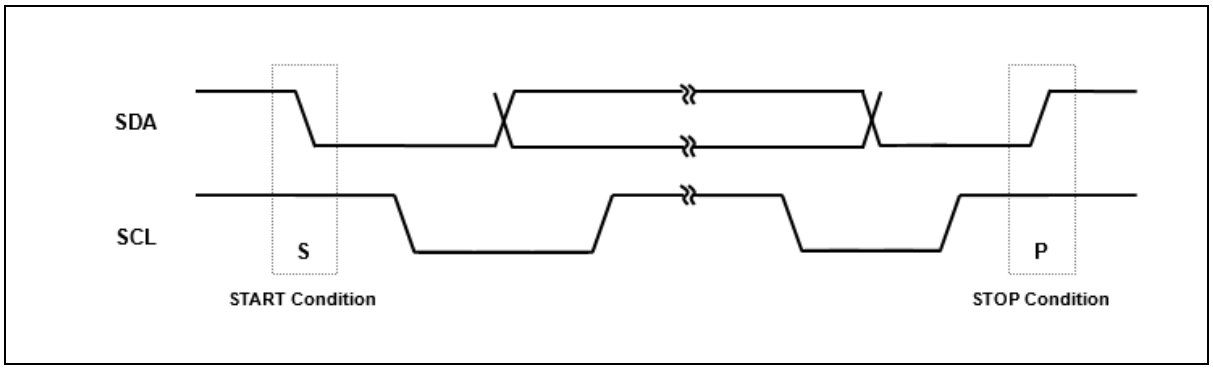


Figure 59. START and STOP Conditions

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 (Refer to Figure 60).

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.

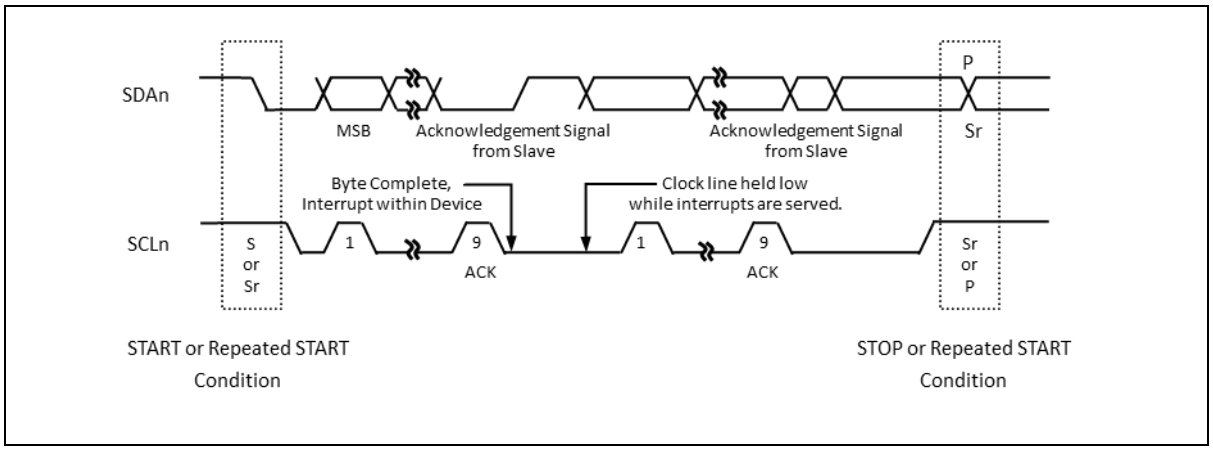


Figure 60. I2C Bus Data Transfer

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 (Refer to Figure 61). 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.

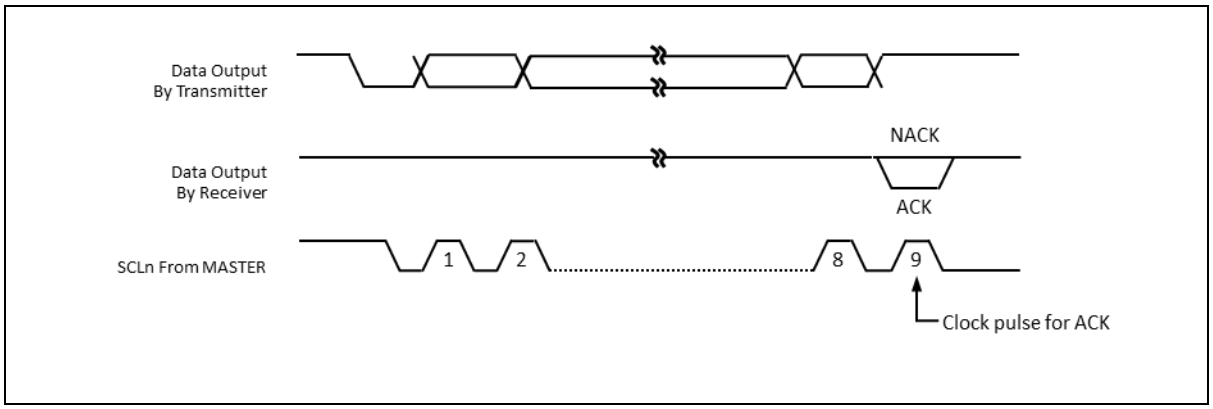


Figure 61. I2C Bus Response

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 (Refer to Figure 62).

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".

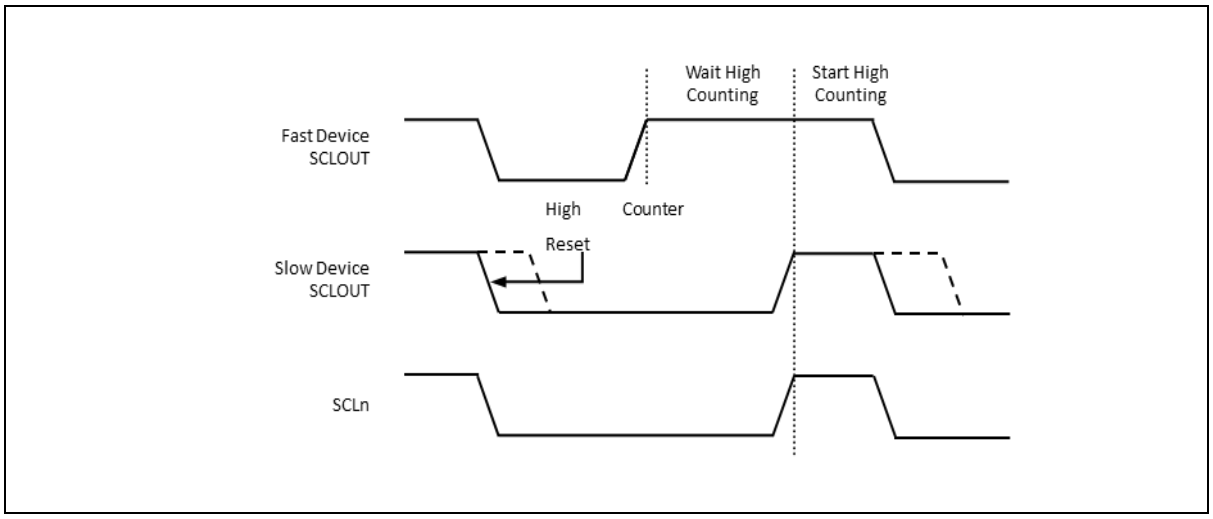


Figure 62. Clock Synchronization during Arbitration

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 63 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.

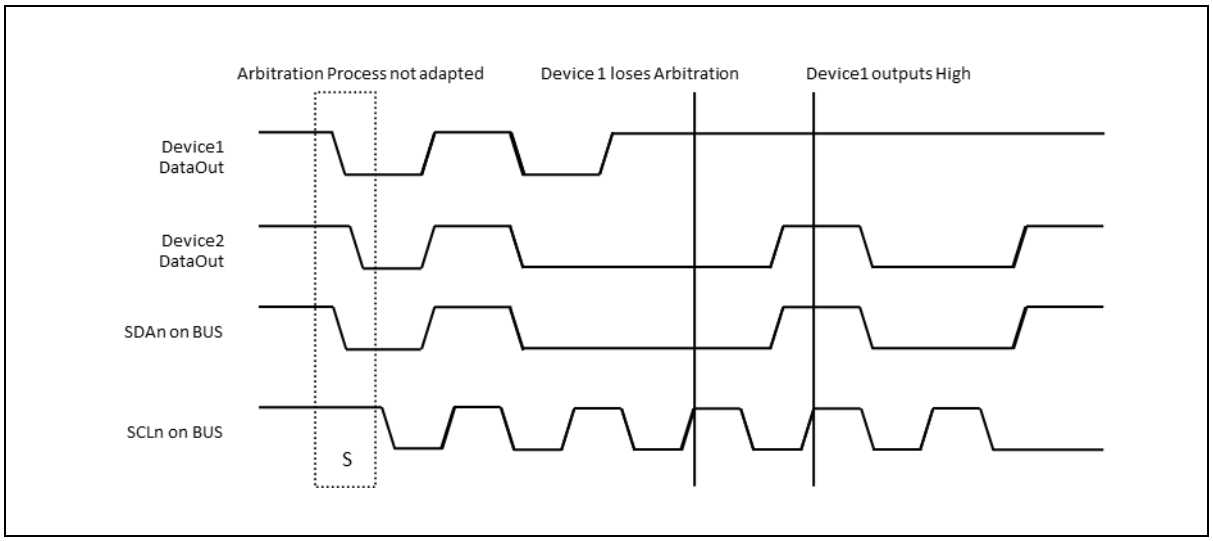


Figure 63. Arbitration Process between Two Masters

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 64 shows a flowchart of the transmitter in master mode.

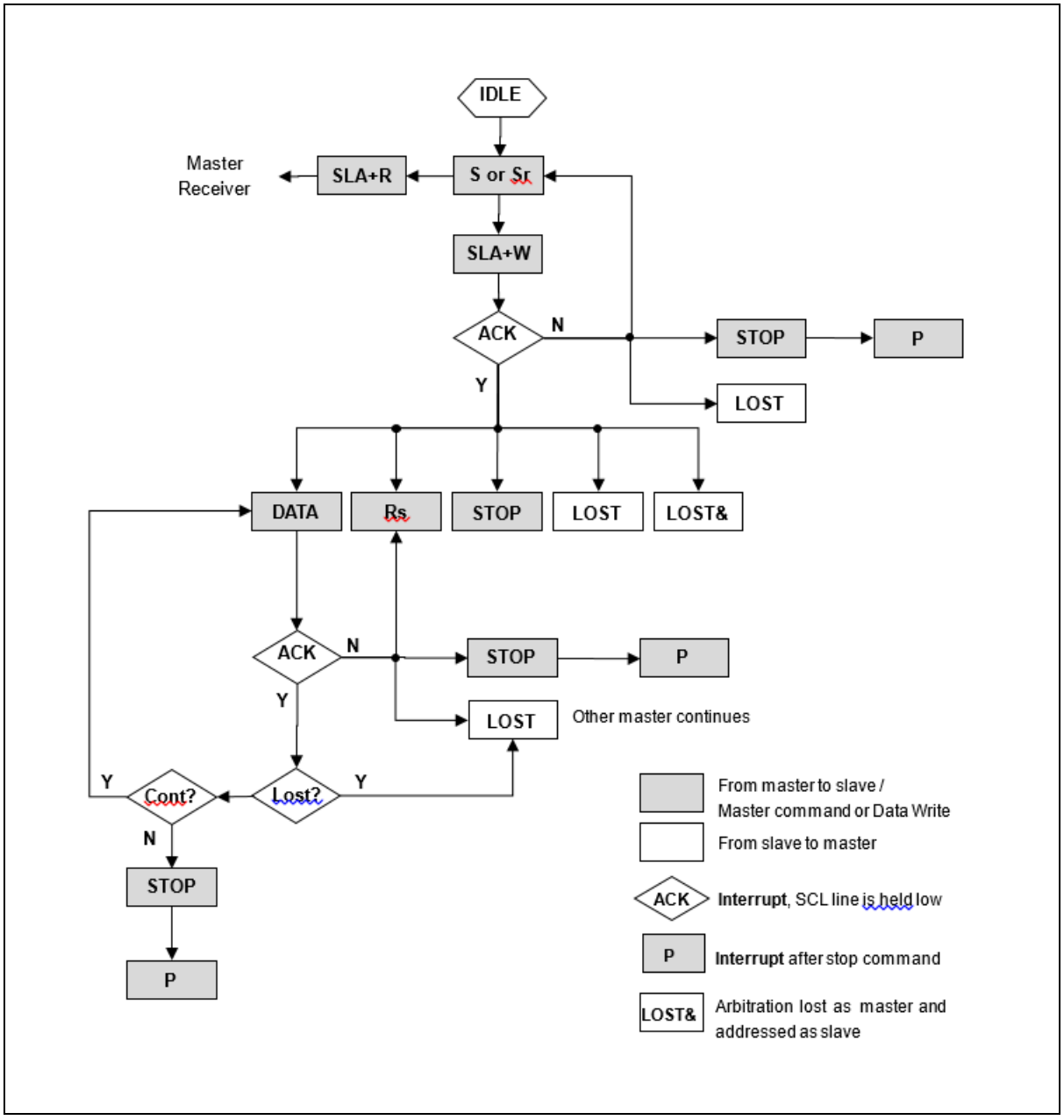


Figure 64. Master Transmitter Flowchart

13.3.8 Master receiver

Figure 65 shows a flowchart of the receiver in master mode.

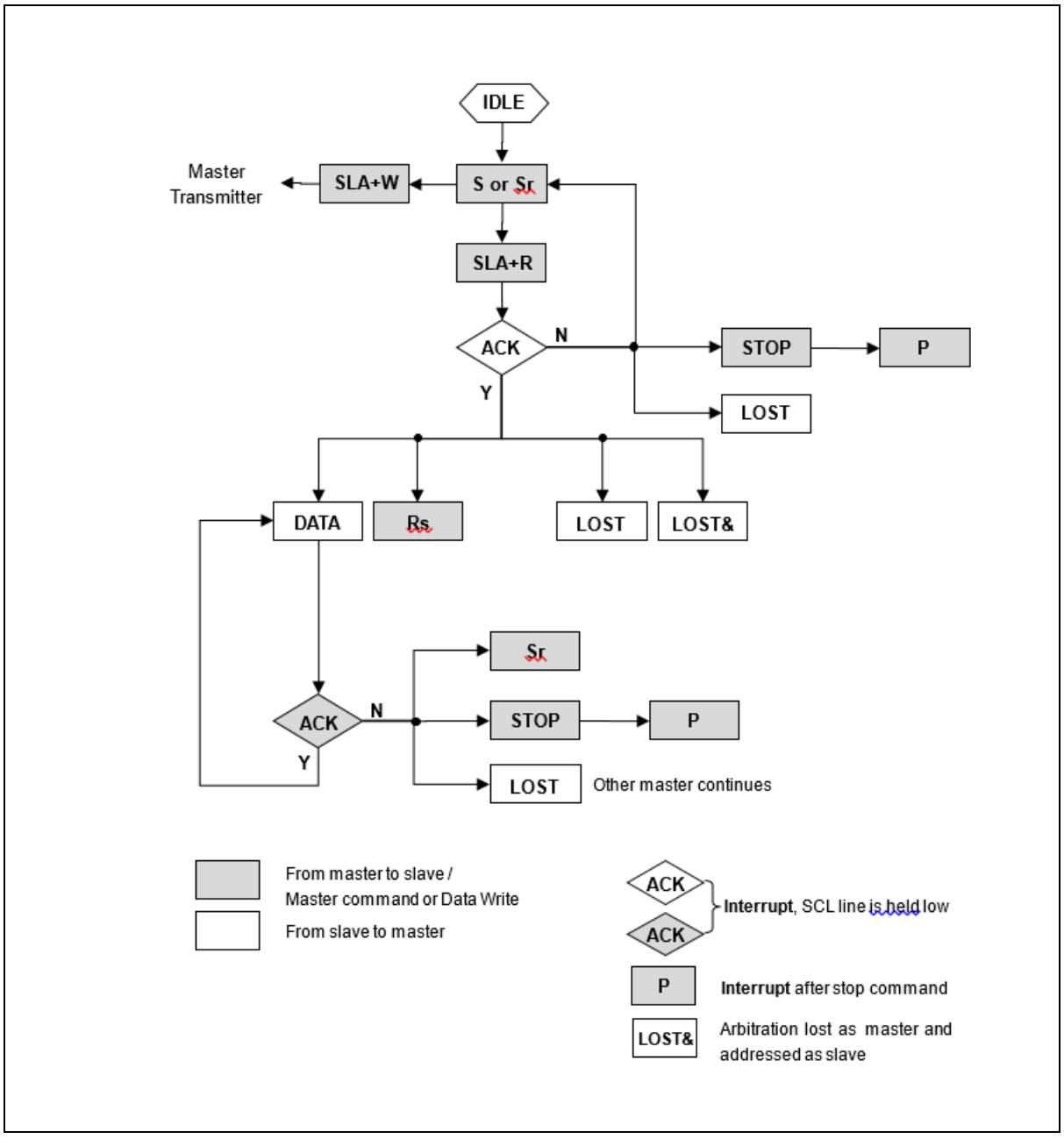


Figure 65. Master Receiver Flowchart

13.3.9 Slave transmitter

Figure 66 shows a flowchart of the transmitter in slave mode.

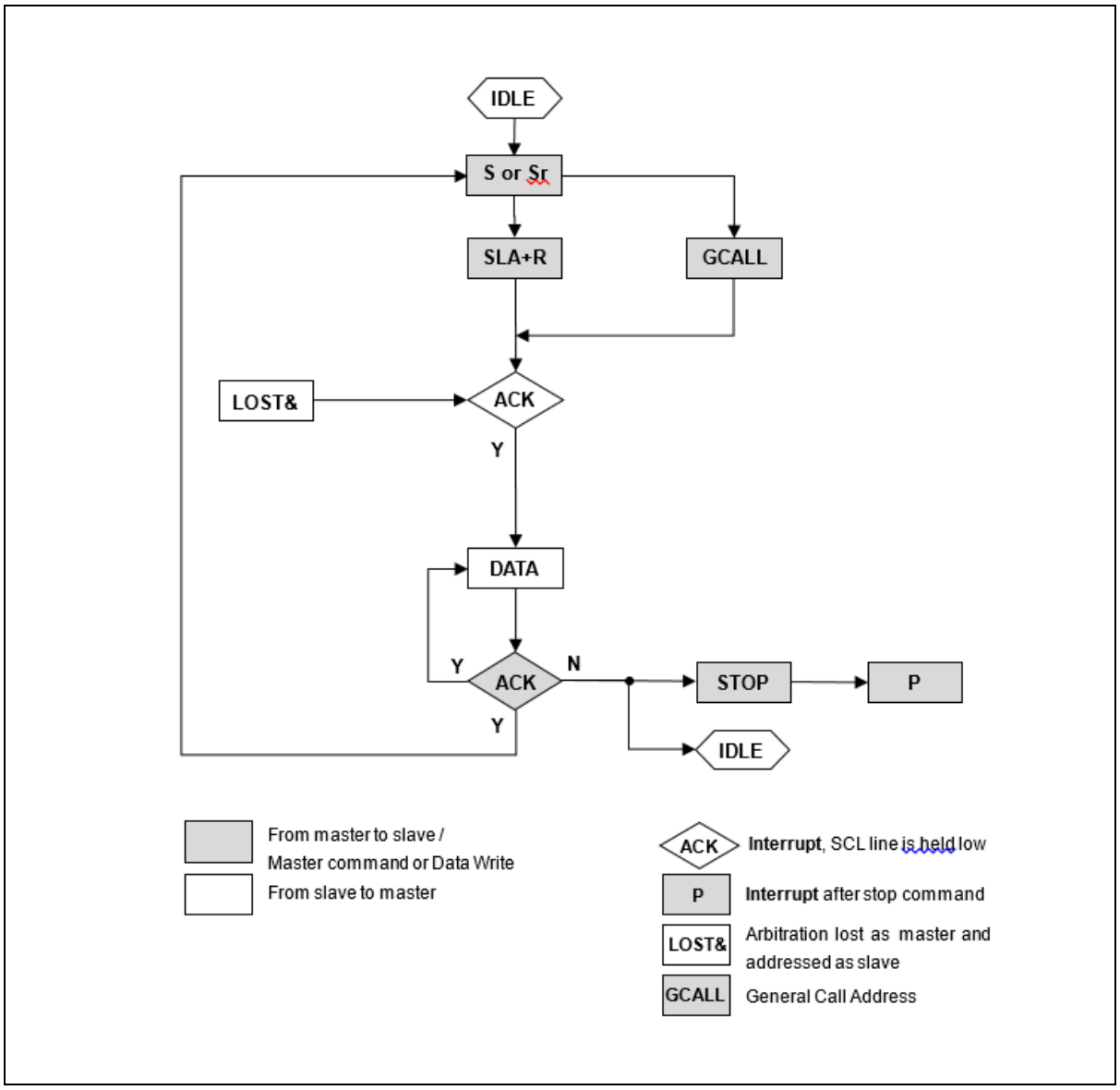


Figure 66. Slave Transmitter Flowchart

13.3.10 Slave receiver

Figure 67 shows a flowchart of the Receiver in slave mode.

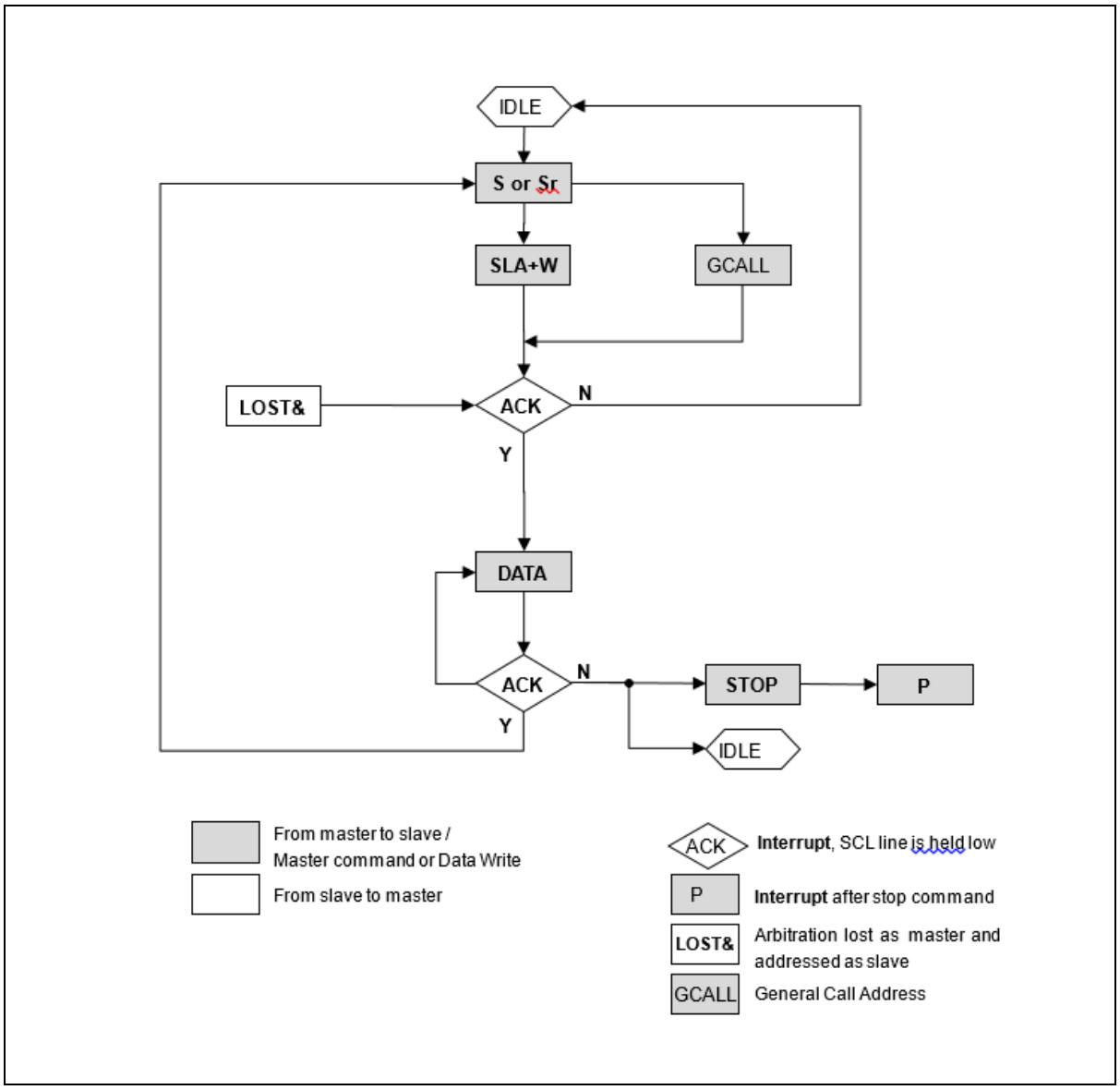


Figure 67. Slave Receiver Flowchart

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 AC30M1x64/AC30M1x32 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 RINGOSC clock. Before enable MPWM module, the proper MPWM clock selection should be required.

Table 45 introduces pins assigned for MPWM.

Table 45. Pin Assignment of MPWM: External Pins

Pin name	Type	Description
MPWMUH	O	MPWM Phase-U H-side output
MPWMUL	O	MPWM Phase-U L-side output
MPWMVH	O	MPWM Phase-V H-side output
MPWMVL	O	MPWM Phase-V L-side output
MPWMWH	O	MPWM Phase-W H-side output
MPWMWL	O	MPWM Phase-W L-side output
PRTIN	I	MPWM Protection Input
OVIN	I	MPWM Over-voltage Input

14.1 MPWM block diagram

Figure 68 describes normal mode of MPWM in block diagram.

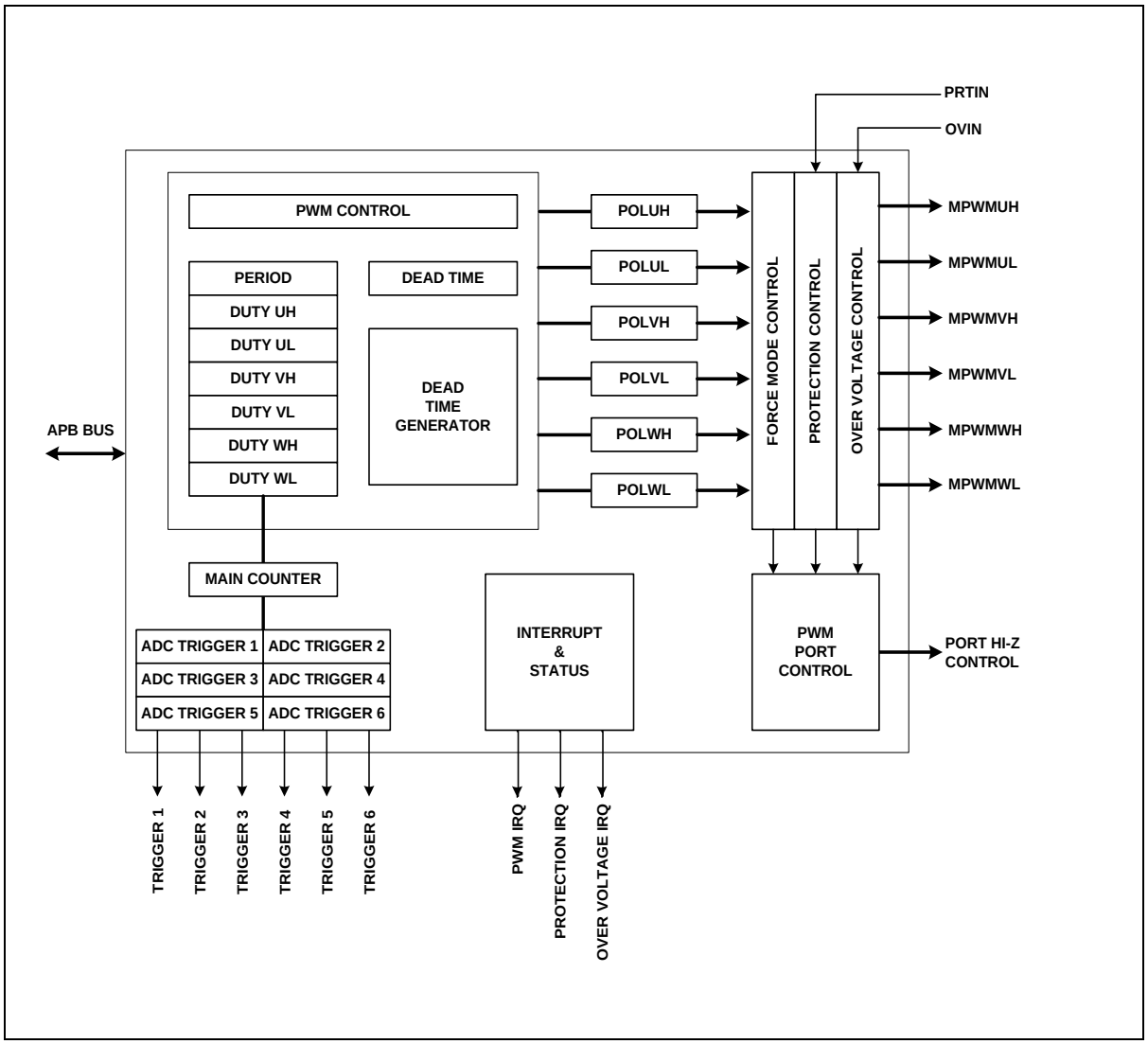


Figure 68. MPWM Block Diagram

14.2 Registers

Base address of MPWM is introduced in the followings:

Table 46. Base Address of MPWM Interface

Name	Base address
MPWM	0x4000_4000

Table 47. MPWM Register Map

Name	Offset	Type	Description	Reset value	Reference
MP.MR	0x0000	RW	MPWM Mode register	0x0000_0000	14.2.1
MP.OLR	0x0004	RW	MPWM Output Level register	0x0000_0000	14.2.2
MP.FOLR	0x0008	RW	MPWM Force Output register	0x0000_0000	14.2.3
MP.PRD	0x000C	RW	MPWM Period register	0x0000_0002	14.2.6
MP.DUH	0x0010	RW	MPWM Duty UH register	0x0000_0001	14.2.7
MP.DVH	0x0014	RW	MPWM Duty VH register	0x0000_0001	14.2.8
MP.DWH	0x0018	RW	MPWM Duty WH register	0x0000_0001	14.2.9
MP.DUL	0x001C	RW	MPWM Duty UL register	0x0000_0001	14.2.10
MP.DVL	0x0020	RW	MPWM Duty VL register	0x0000_0001	14.2.11
MP.DWL	0x0024	RW	MPWM Duty WL register	0x0000_0001	14.2.12
MP.CR1	0x0028	RW	MPWM Control register 1	0x0000_0000	14.2.4
MP.CR2	0x002C	RW	MPWM Control register 2	0x0000_0000	14.2.5
MP.SR	0x0030	R	MPWM Status register	0x0000_0000	14.2.14
MP.IER	0x0034	RW	MPWM Interrupt Enable	0x0000_0000	14.2.13
MP.CNT	0x0038	R	MPWM counter register	0x0000_0001	14.2.15
MP.DTR	0x003C	RW	MPWM dead time control	0x0000_0000	14.2.16
MP.PCR0	0x0040	RW	MPWM protection 0 control register	0x0000_0000	14.2.17
MP.PSR0	0x0044	RW	MPWM protection 0 status register	0x0000_0080	14.2.18
MP.PCR1	0x0048	RW	MPWM protection 1 control register	0x0000_0000	14.2.17
MP.PSR1	0x004C	RW	MPWM protection 1 status register	0x0000_0000	14.2.18
-	0x0054	-	Reserved		

Table 47. MPWM Register Map (continued)

Name	Offset	Type	Description	Reset value	Reference
MP.ATR1	0x0058	RW	MPWM ADC Trigger reg1	0x0000_0000	14.2.19
MP.ATR2	0x005C	RW	MPWM ADC Trigger reg2	0x0000_0000	14.2.19
MP.ATR3	0x0060	RW	MPWM ADC Trigger reg3	0x0000_0000	14.2.19
MP.ATR4	0x0064	RW	MPWM ADC Trigger reg4	0x0000_0000	14.2.19
MP.ATR5	0x0068	RW	MPWM ADC Trigger reg5	0x0000_0000	14.2.19
MP.ATR6	0x006C	RW	MPWM ADC Trigger reg6	0x0000_0000	14.2.19

14.2.1 MP.MR: MPWM Mode Register

Motor PWM operation Mode register is 16-bit register.

MP.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 PWM mode
		1	Normal PWM 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 MP.MR field is only effective when MOTORB in MP.MR is clear "0". Otherwise the MCHMOD field value will be ignored internally and will keep "00" value.

The UPDOWN in MP.MR field is only effective when MOTORB in MP.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 MP.OLR: MPWM Output Level Register

PWM output level register is 8-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.

MP.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

WHL	0	Default Output Level
	1	Inversion Output Level
VHL	0	Default Output Level
	1	Inversion Output Level
UHL	0	Default Output Level
	1	Inversion Output Level
WLL	0	Default Output Level
	1	Inversion Output Level
VLL	0	Default Output Level
	1	Inversion Output Level
ULL	0	Default Output Level
	1	Inversion Output Level

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

Table 48. MPWM Default Output Level (MP.OLR = 0x00)

PWM Output	Level	NORMAL PWM mode (MOTORB = 1)		MOTOR PWM mode (MOTORB = 0)
		UP mode (UPDOWN = 0)	UPDOWN mode (UPDOWN = 1)	
WH	Default level	LOW	HIGH	LOW
	Inversion level	HIGH	LOW	HIGH
WL	Default level	LOW	LOW	HIGH
	Inversion level	HIGH	HIGH	LOW
VH	Default level	LOW	HIGH	LOW
	Inversion level	HIGH	LOW	HIGH
VL	Default level	LOW	LOW	HIGH
	Inversion level	HIGH	HIGH	LOW
UH	Default level	LOW	HIGH	LOW
	Inversion level	HIGH	LOW	HIGH
UL	Default level	LOW	LOW	HIGH
	Inversion level	HIGH	HIGH	LOW

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

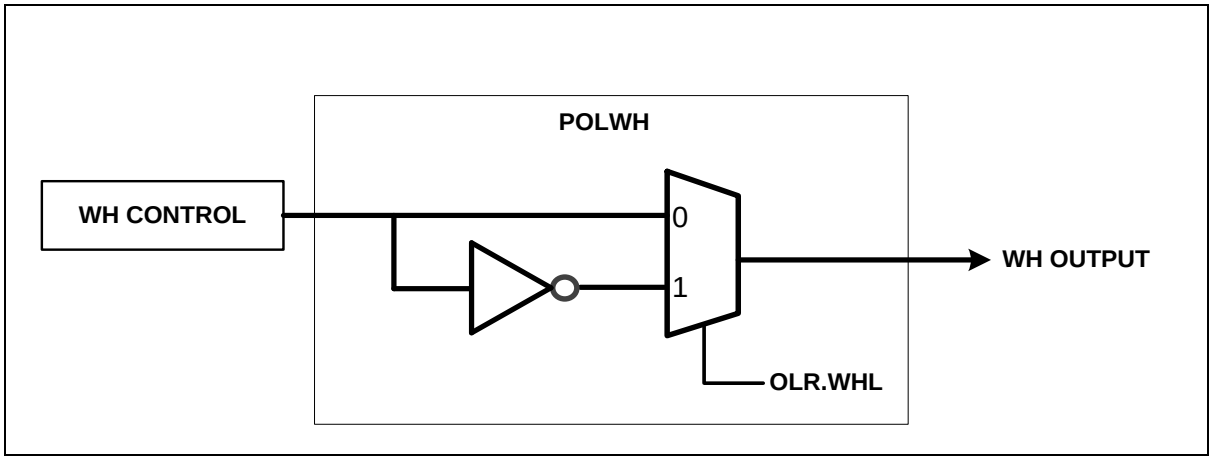


Figure 69. Polarity Control Block

14.2.3 MP.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.

MP.FOLR=0x4000_4008

7	6	5	4	3	2	1	0
		WHFL	VHFL	UHFL	WLFL	VLFL	ULFL
0	0	0 RW	0 RW	0 RW	0 RW	0 RW	0 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 MP.CR1: MPWM Control Register 1

PWM Control Register 1 is 16-bit register.

MP.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 MP.CR2: MPWM Control Register 2

PWM Control Register 2 is 8-bit register.

MP.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 MP.PRD: MPWM Period Register

PWM Period Register is 16-bit register.

MP.PRD=0x4000400C

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 MP.DUH: MPWM Duty UH Register

PWM UH channel duty register is 16-bit register.

MP.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 MP.DVH: MPWM Duty VH Register

PWM VH channel duty register is 16-bit register.

MP.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 MP.DWH: MPWM Duty WH Register

PWM WH channel duty register is 16-bit register.

MP.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 MP.DUL: MPWM Duty UL Register

PWM UL channel duty register is 16-bit register.

MP.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 MP.DVL: MPWM Duty VL Register

PWM VL channel duty register is 16-bit register.

MP.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 MP.DWL: MPWM Duty WL Register

PWM WL channel duty register is 16-bit register.

MP.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 MP.IER: PWM Interrupt Enable Register

PWM Interrupt Enable Register is 8-bit register.

MP.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

MP.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 MP.SR: MPWM Status Register

PWM Status Register is 16-bit register.

MP.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
R	R							RW	RW	RW	RW	RW	RW	RW	RW

15	DOWN	0	PWM Count Up
		1	PWM Count 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

MP.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 MP.CNT: MPWM Counter Register

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

MP.CNT=0x4000_4038

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

	CNT	PWM Counter Value

14.2.16 MP.DTR: MPWM Dead Time Register

PWM Dead Time Register is 16-bit register.

MP.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 0	DT	Dead Time value (Dead time setting makes output delay of 'low to high transition' in normal polarity) 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 MP.PCRn: MPWM Protection 0,1 Control Register

PWM Protection Control Register is 16-bit register.

MP.PCR0=0x4000_4040, MP.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 to OVIN		

14.2.18 MP.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.

MP.PSR0=0x4000_4044, MP.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

If PROTEN bit in MP.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 MP.FOLR register.

In addition, a user can prohibit the output manually by writing the designated value into the MP.PSRn register.
Note) MP.PSR0 is related to PRTIN pin and MP.PSR1 is related to OVIN .

14.2.19 MP.ATRn: MPWM ADC Trigger Counter n Register (n = 1 to 6)

PWM ADC Trigger Counter Register is 32-bit register.

MP.ATR1=0x4000_4058

MP.ATR2=0x4000_405C

MP.ATR3=0x4000_4060

MP.ATR4=0x4000_4064

MP.ATR5=0x4000_4068

MP.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.

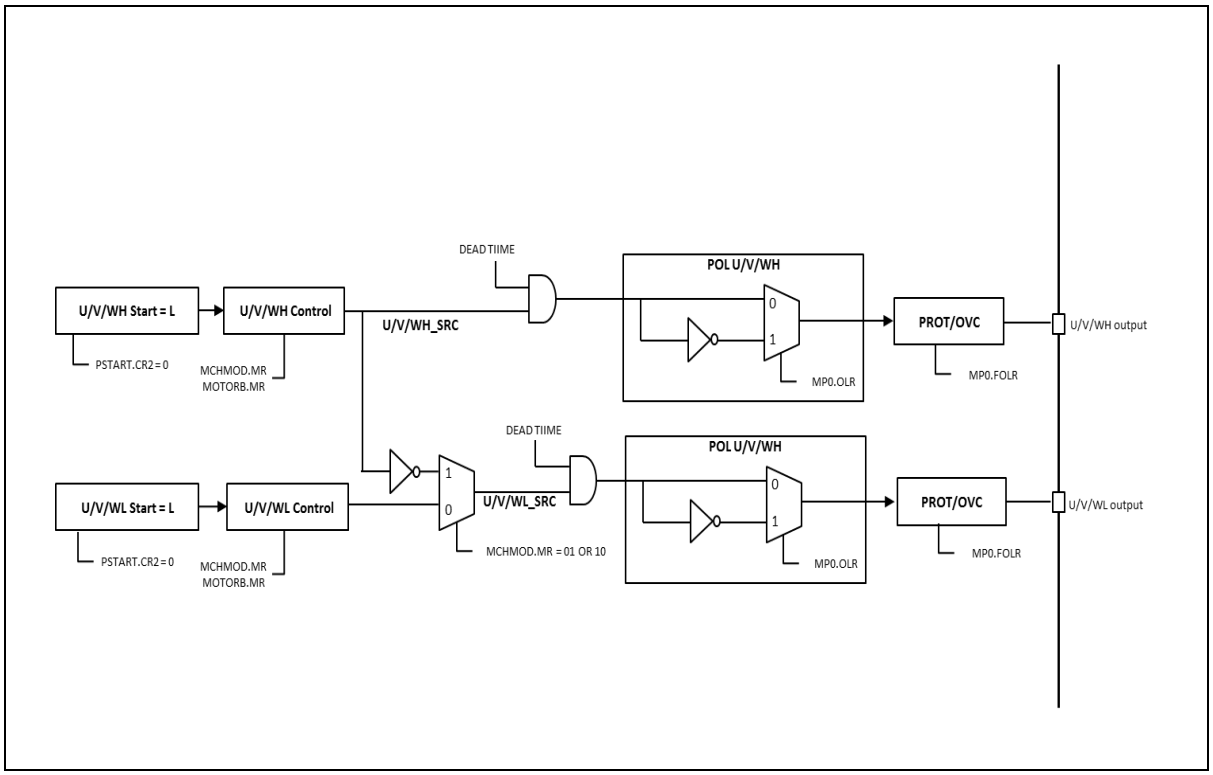


Figure 70. PWM Output Generation Chain

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.

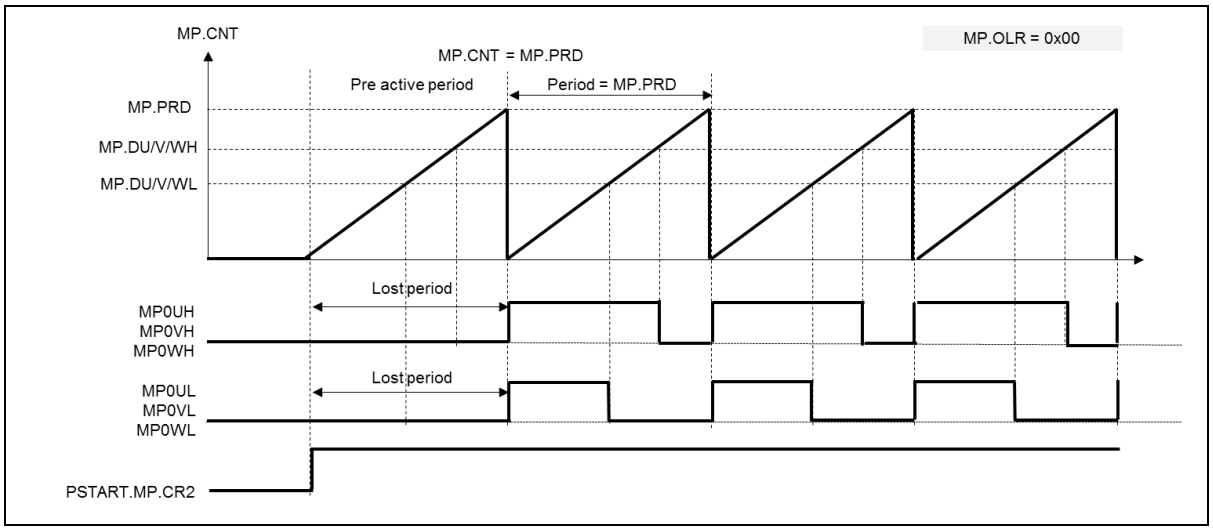


Figure 71. UP Count Mode Wave Form (MOTORB=1, UPDOWN=0)

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.

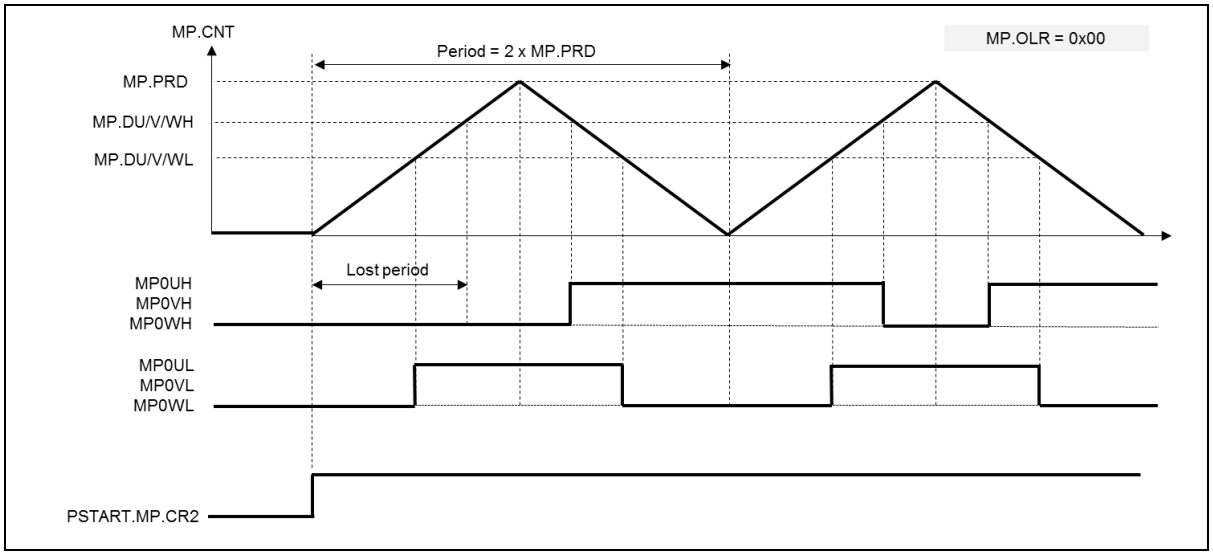


Figure 72. UP/DOWN Count Mode Wave Form (MOTORB=0, MCHMOD=0, UPDOWN=1)

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 73 shows 2 channel symmetric mode waveform.

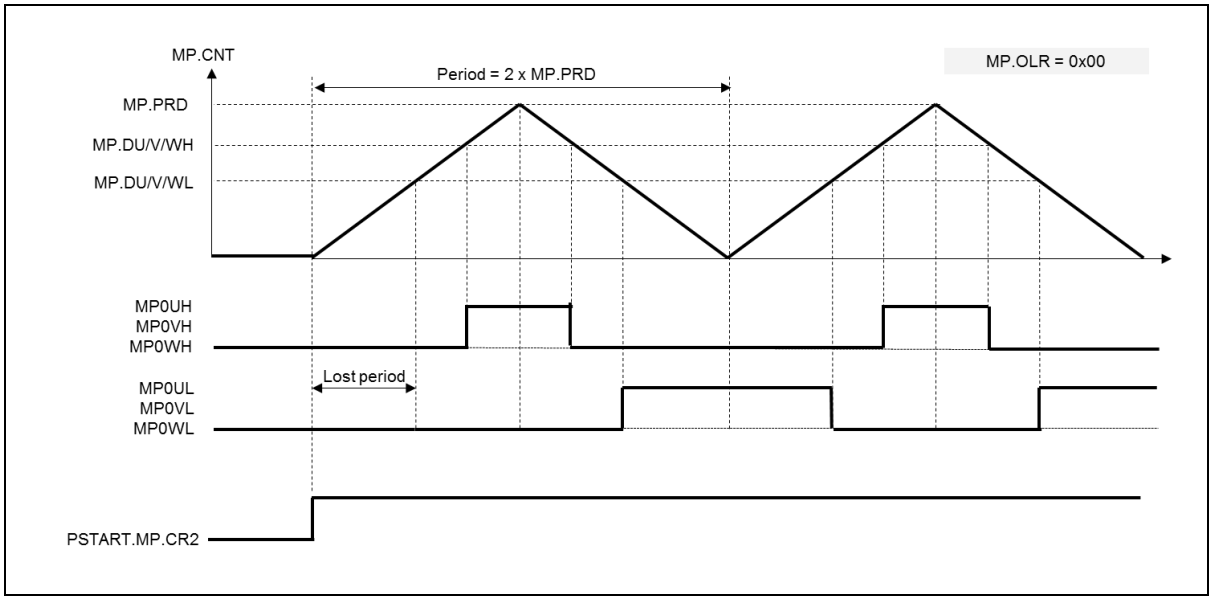


Figure 73. 2-Channel Symmetric Mode Wave Form (MOTORB=0, MCHMOD=00)

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.

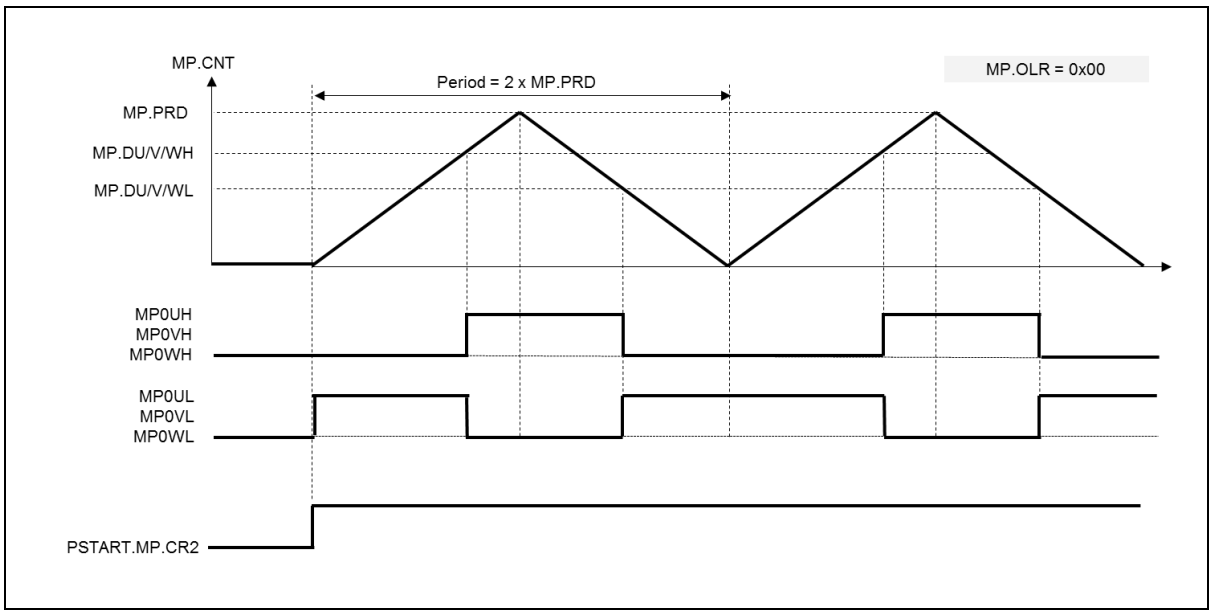


Figure 74. 1-Channel Asymmetric Mode Wave Form (MOTORB=0, MCHMOD=01)

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.

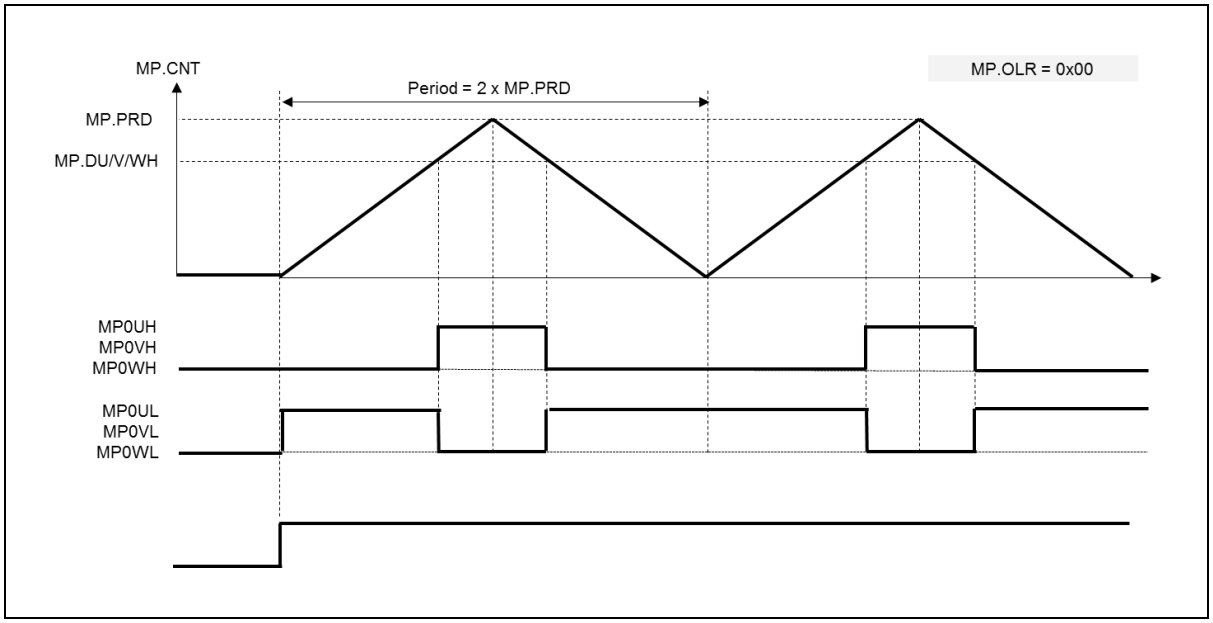


Figure 75. 1-Channel Symmetric Mode Wave Form (MOTORB=0, MCHMOD=10)

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 by the value in DT.MP.DTR[7:0] field:

- 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 76 is an example of dead time operation in 1-channel symmetric mode.

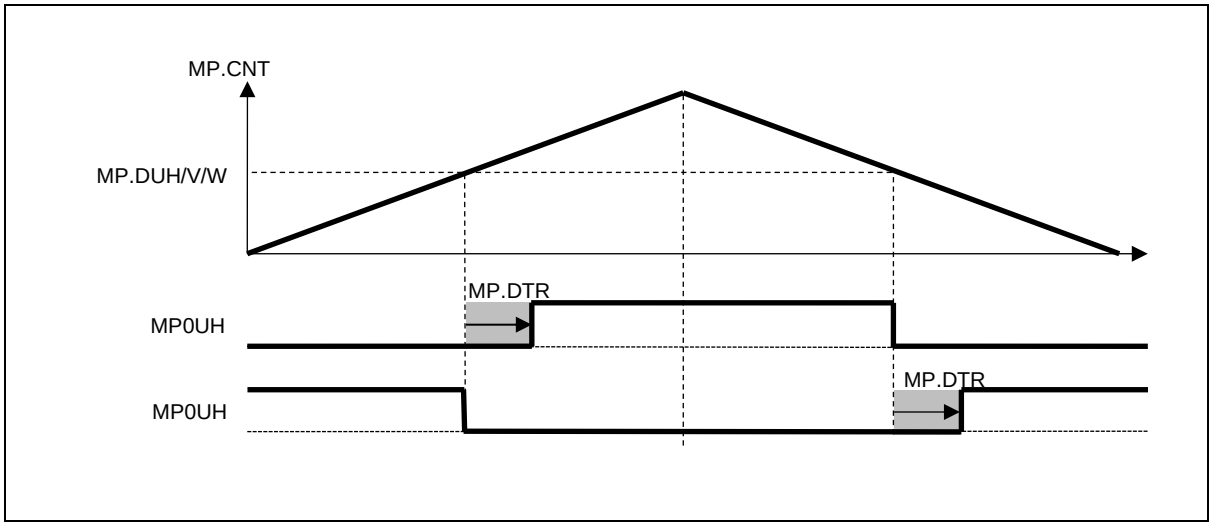


Figure 76. PWM Dead-Time Operation Timing Diagram (Symmetric mode)

Figure 77 shows in case of 1-channel asymmetric mode operation

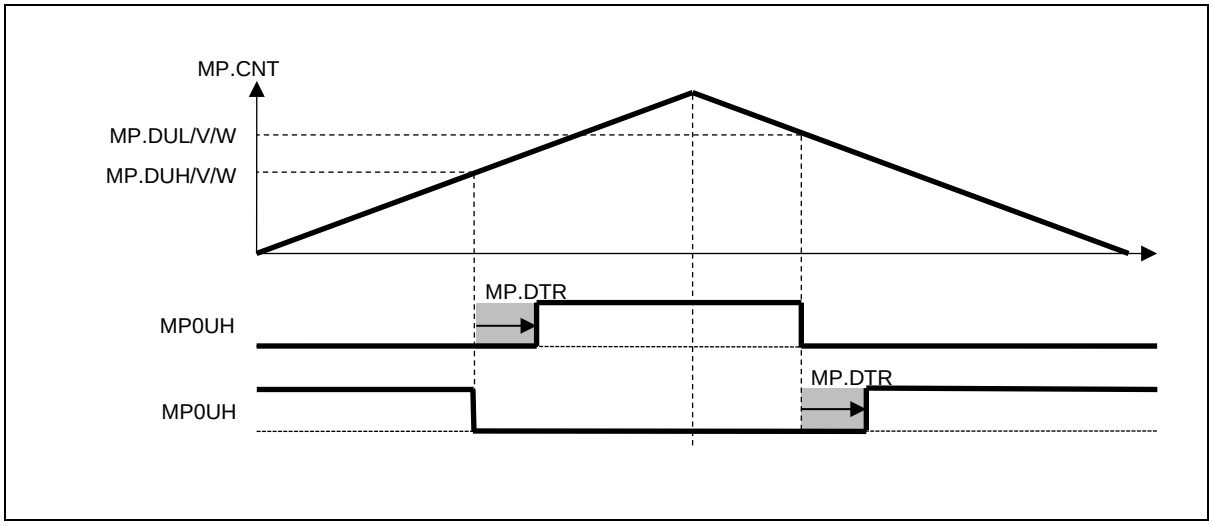


Figure 77. PWM Dead-Time Operation Timing Diagram (Asymmetric mode)

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

Figure 78 show how the dead-time operates with an example case of normal dead time. 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.

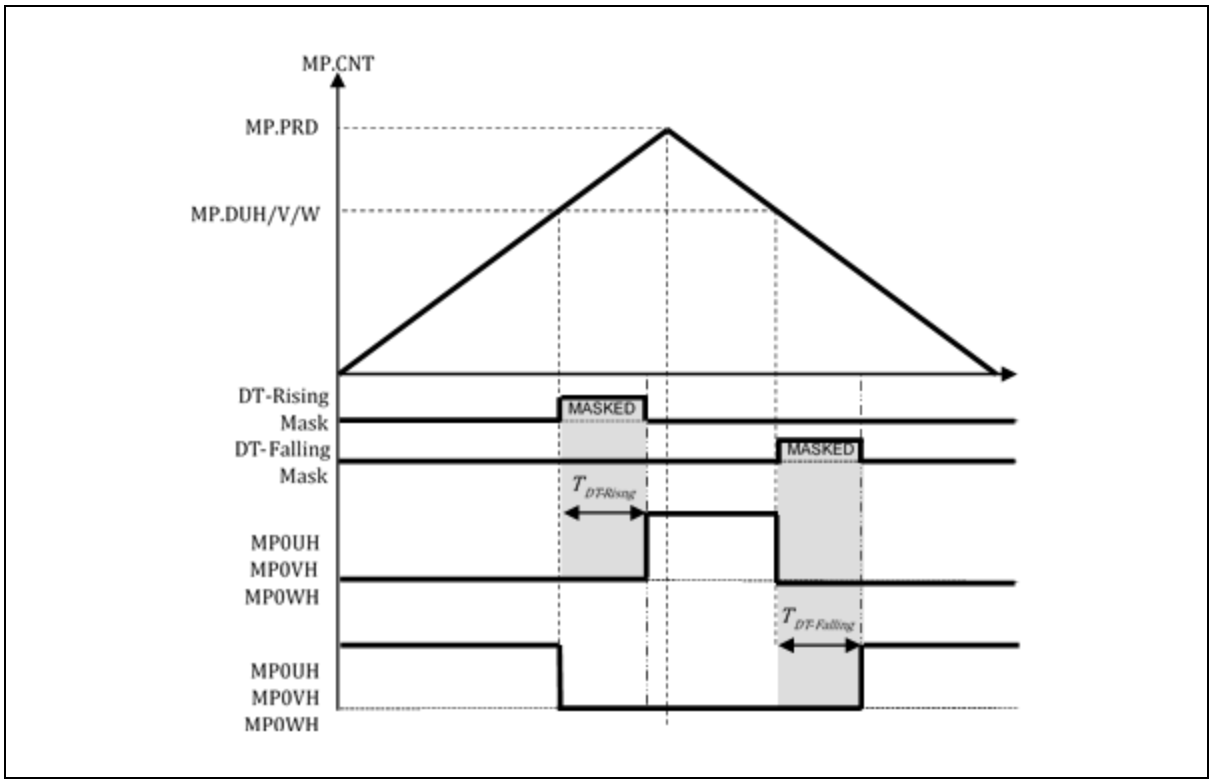


Figure 78. Normal Dead-Time Operation (T_{DUTY}>T_{DT})

A couple of figures below show special case of dead time configurations:

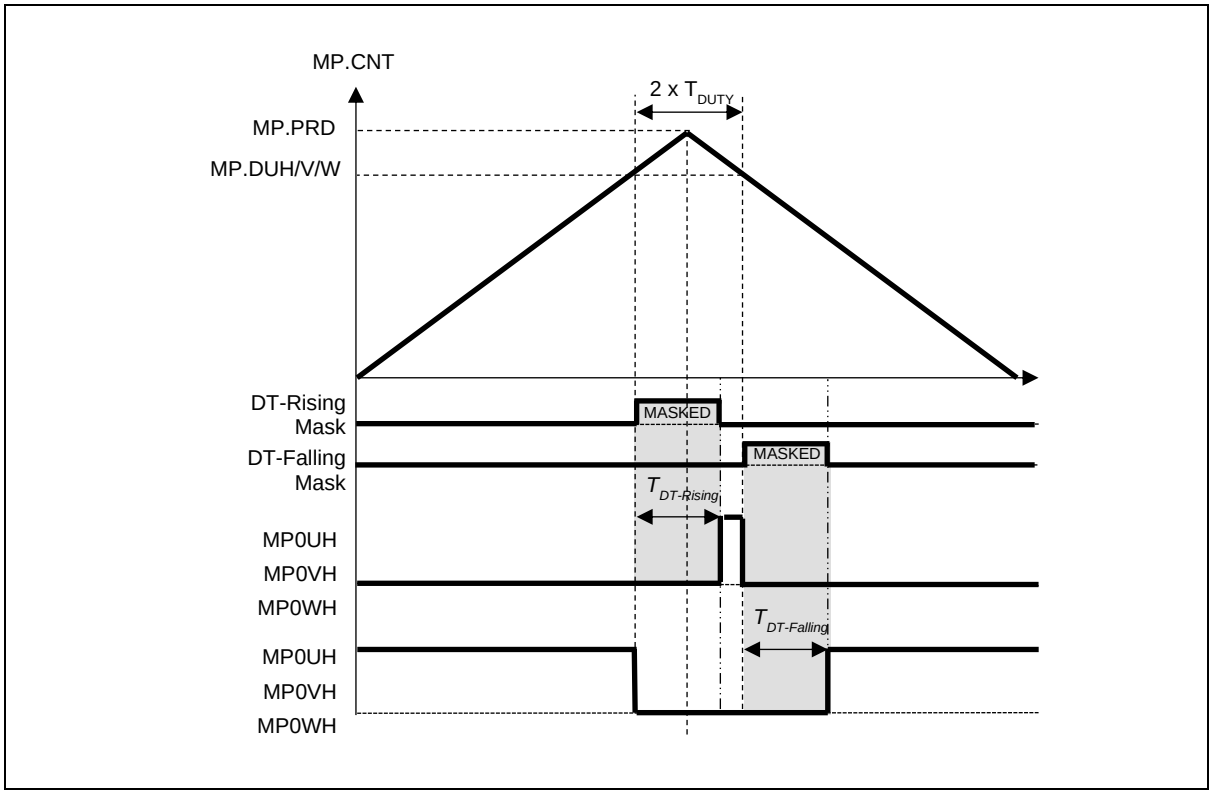


Figure 79. Minimum H-Side Pulse Timing (T_{DUTY}<T_{DT}<2xT_{DUTY})

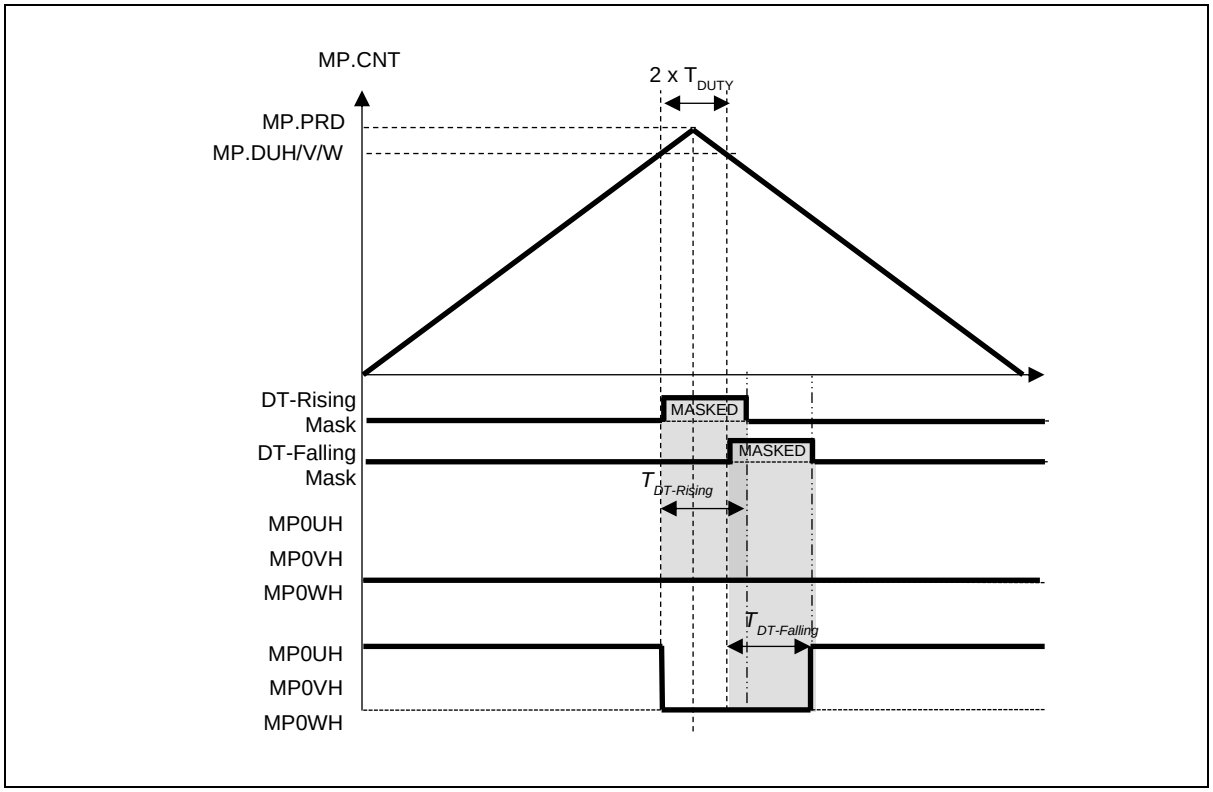


Figure 80. Zero H-Side Pulse Timing (T_{DT}>2xT_{DUTY})

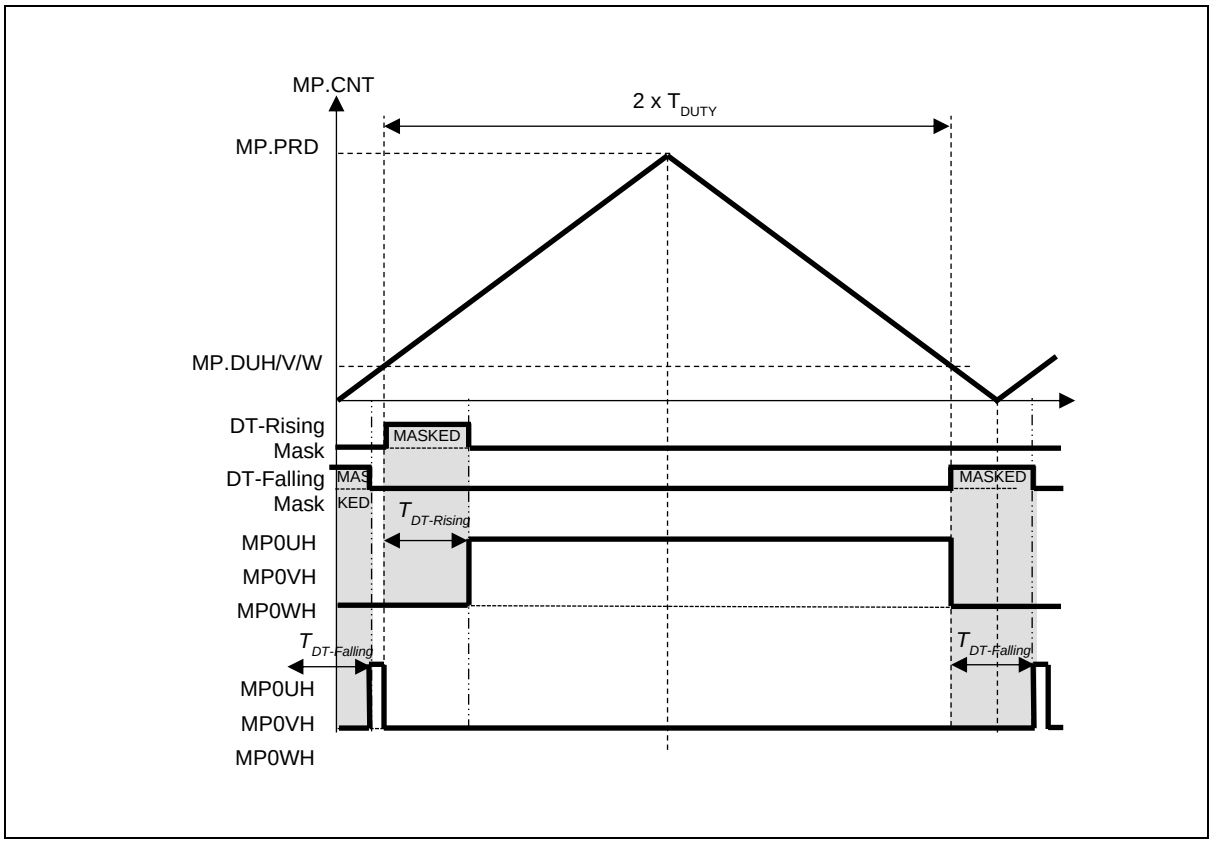


Figure 81. Minimum L-Side Pulse Timing (T_{DT}<Period-T_{DUTY})

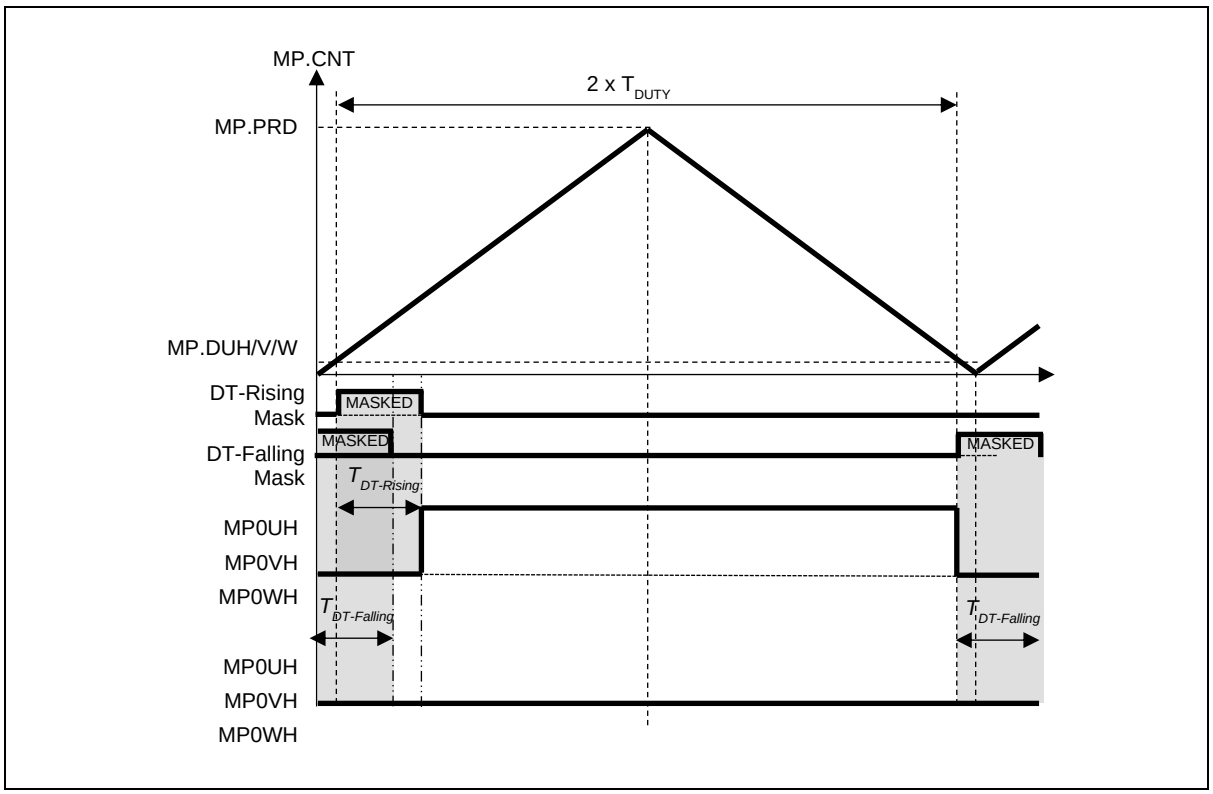


Figure 82. Zero L-Side Pulse Timing (T_{DT}>Period-T_{DUTY})

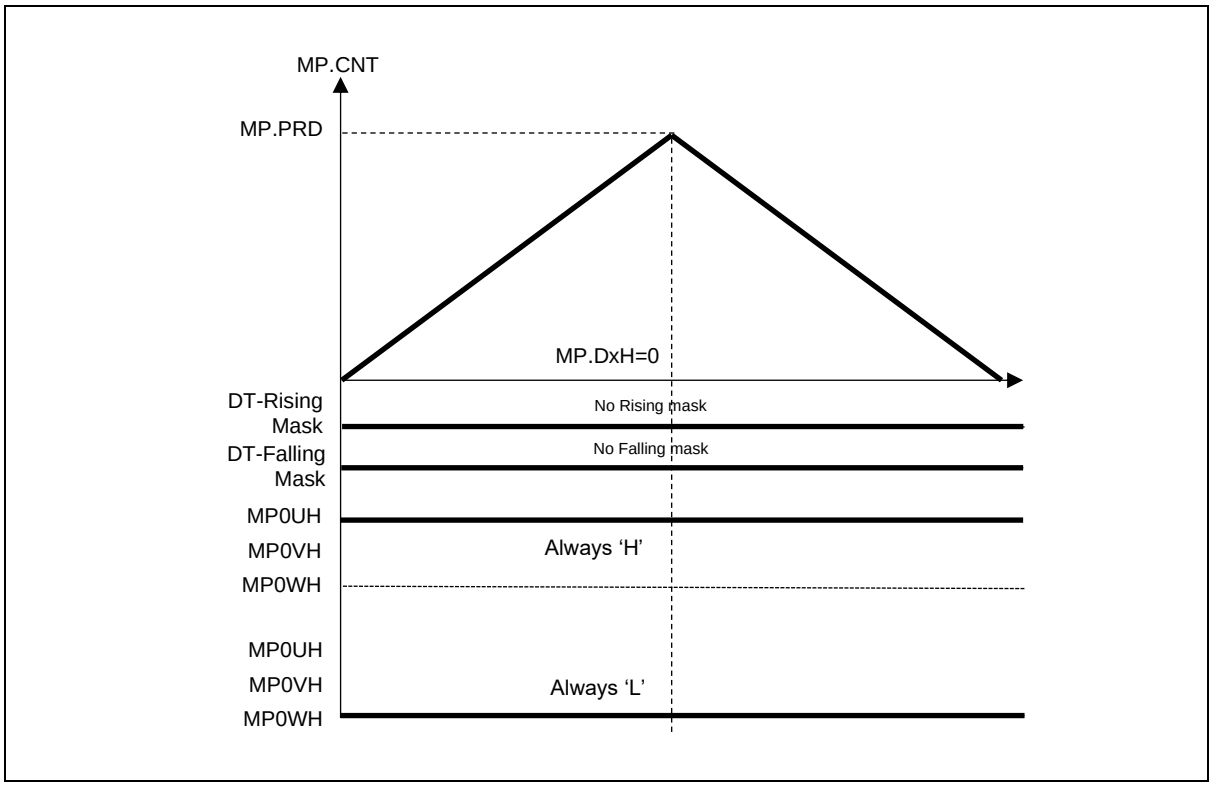


Figure 83. H-Side Always On (T_{DUTY}=Period: dead-time disabled)

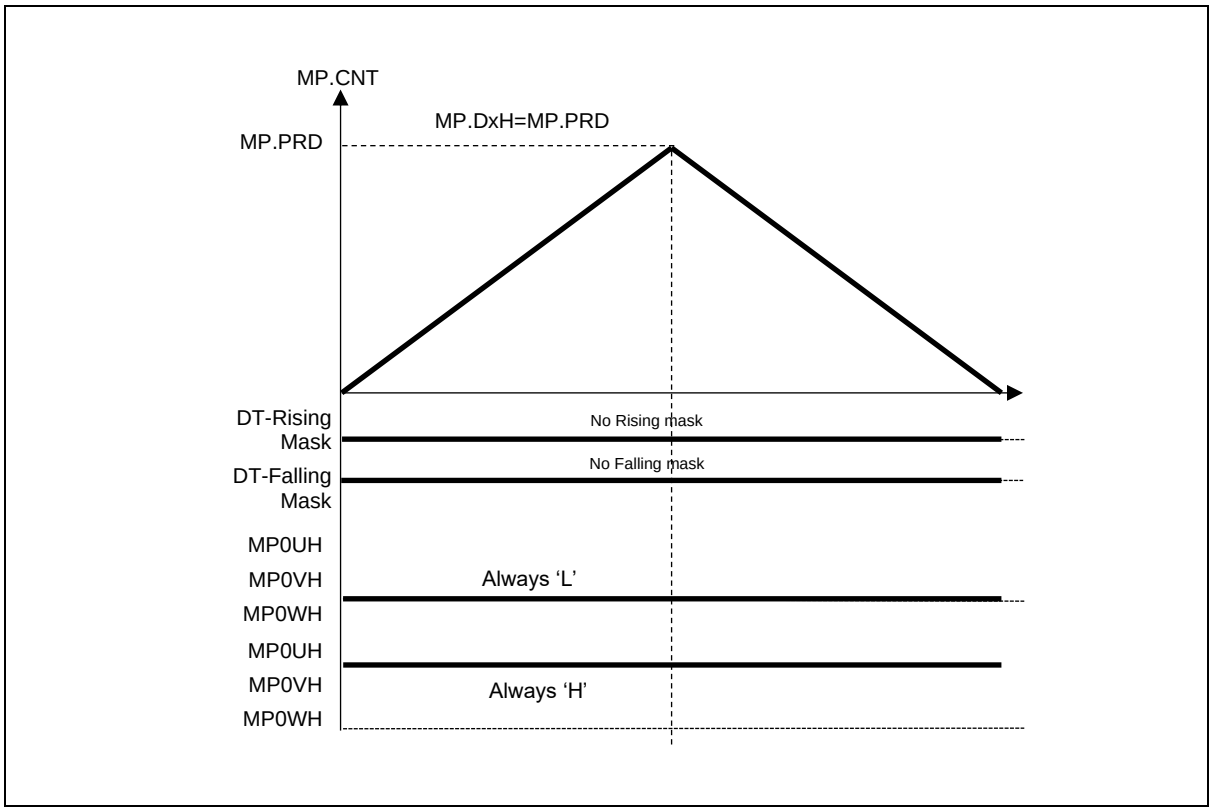


Figure 84. L-Side Always On (T_{DUTY}='0': dead-time disabled)

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.

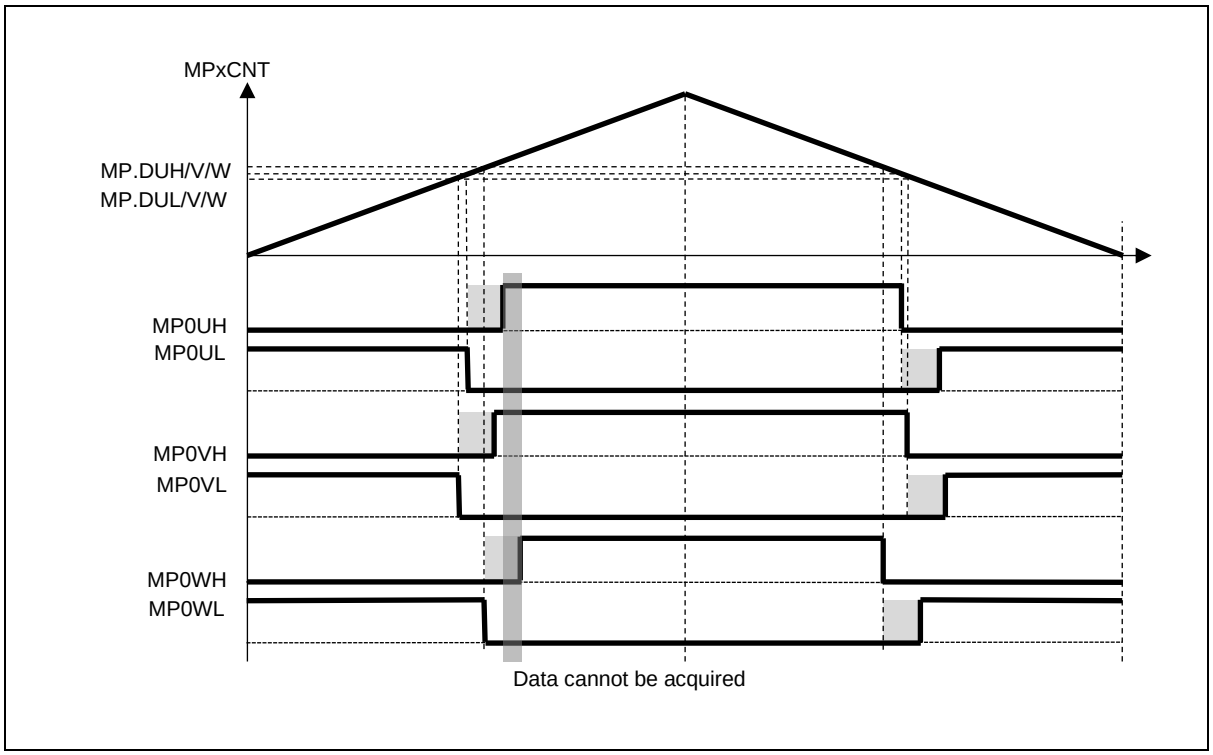


Figure 85. Symmetrical PWM Timing

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.

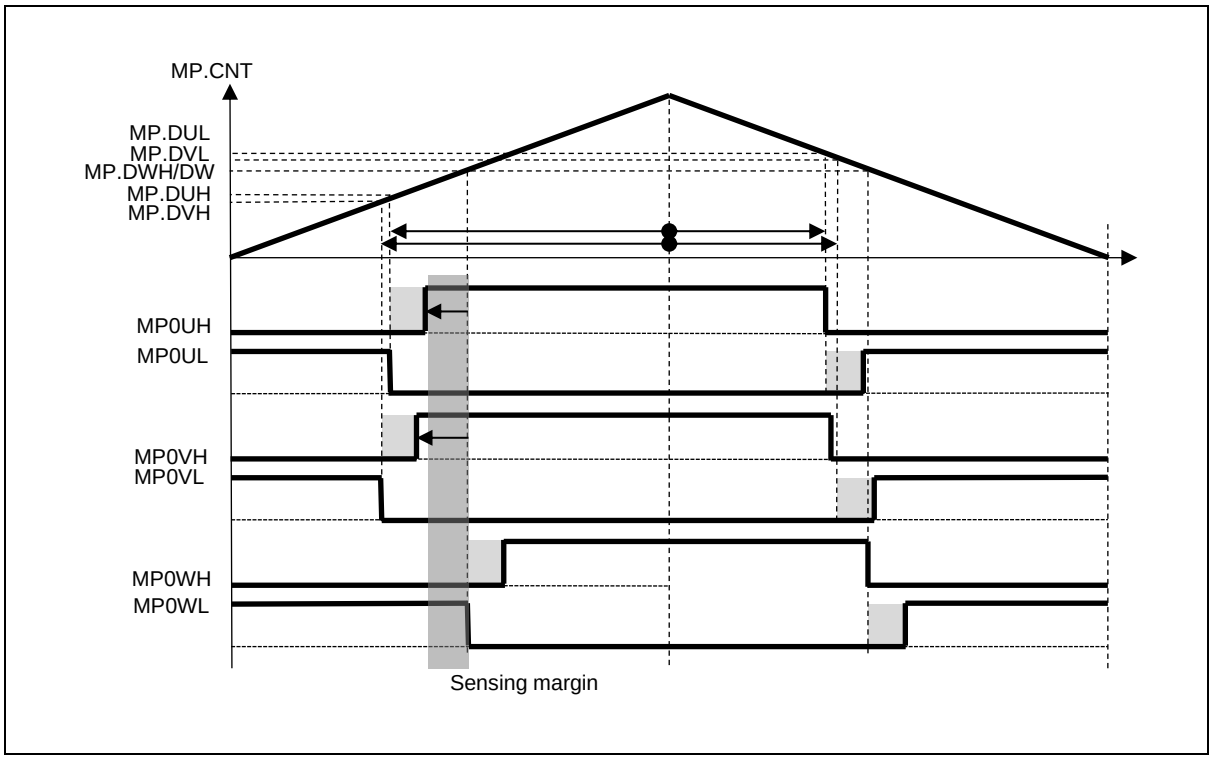


Figure 86. Asymmetrical PWM Timing and Sensing Margin

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.

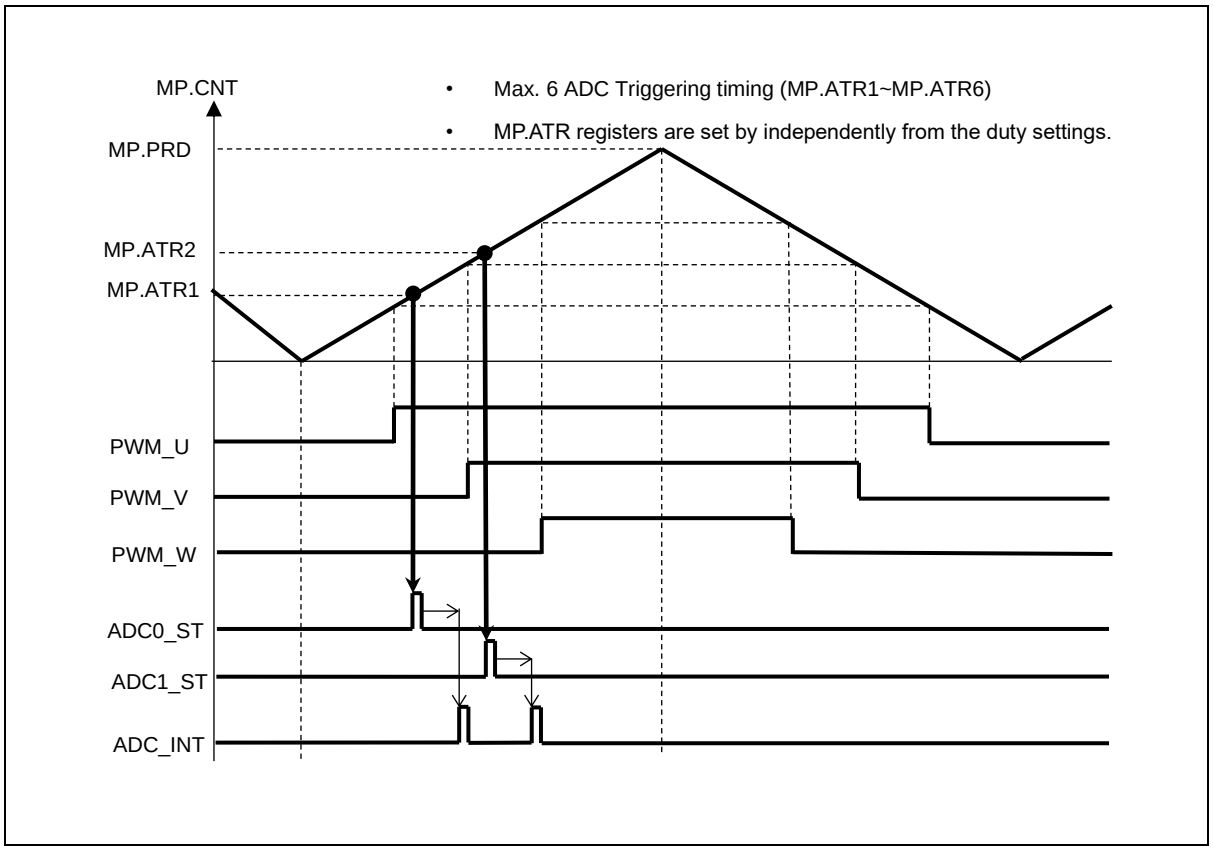


Figure 87. ADC Triggering Function Timing Diagram

Figure 88 shows an example of ADC Data acquisition:

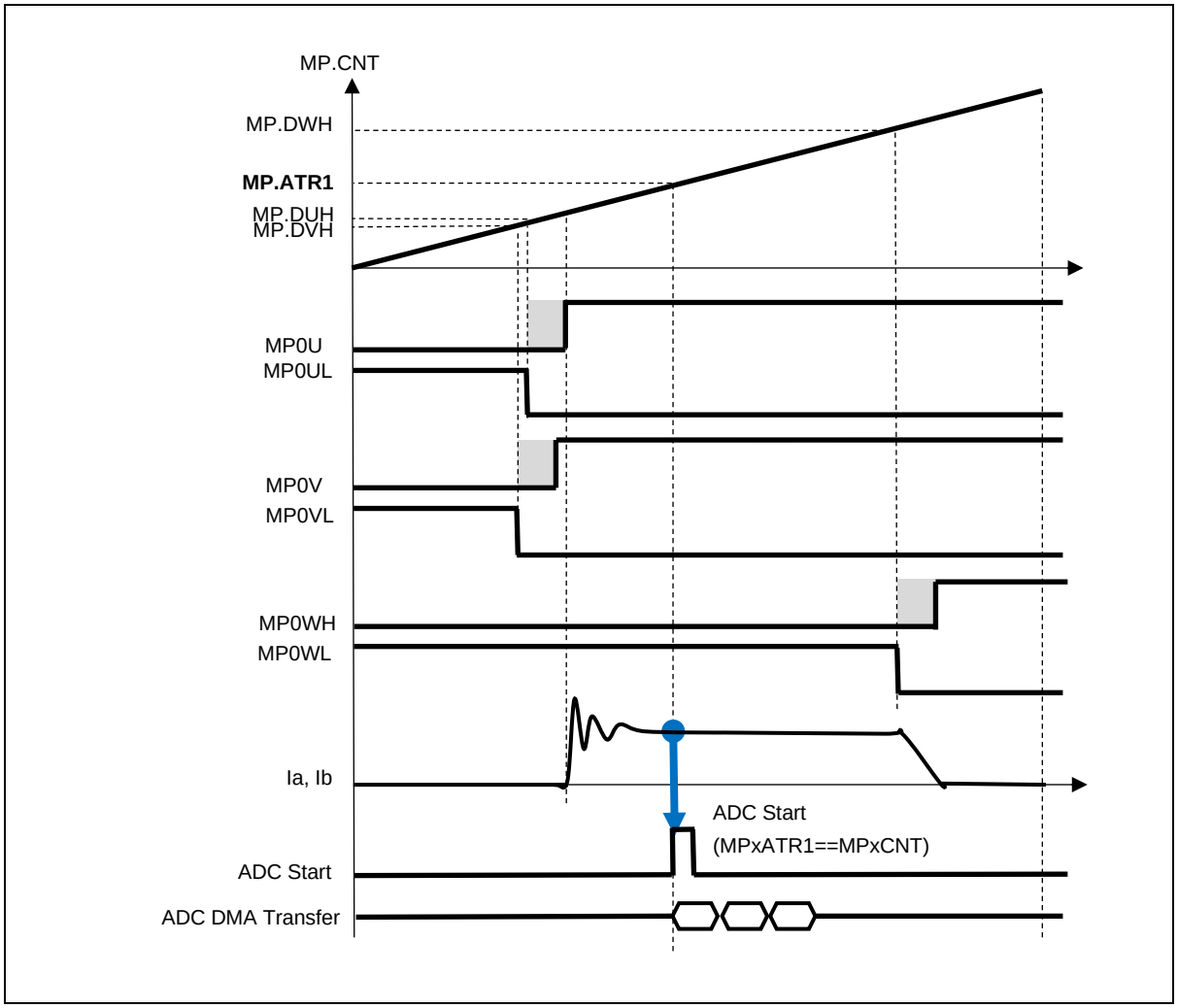


Figure 88. An Example of ADC Acquisition Timing by Event from MPWM

14.3.10 Interrupt generation timing

Each timing event can make interrupt request to the CPU.

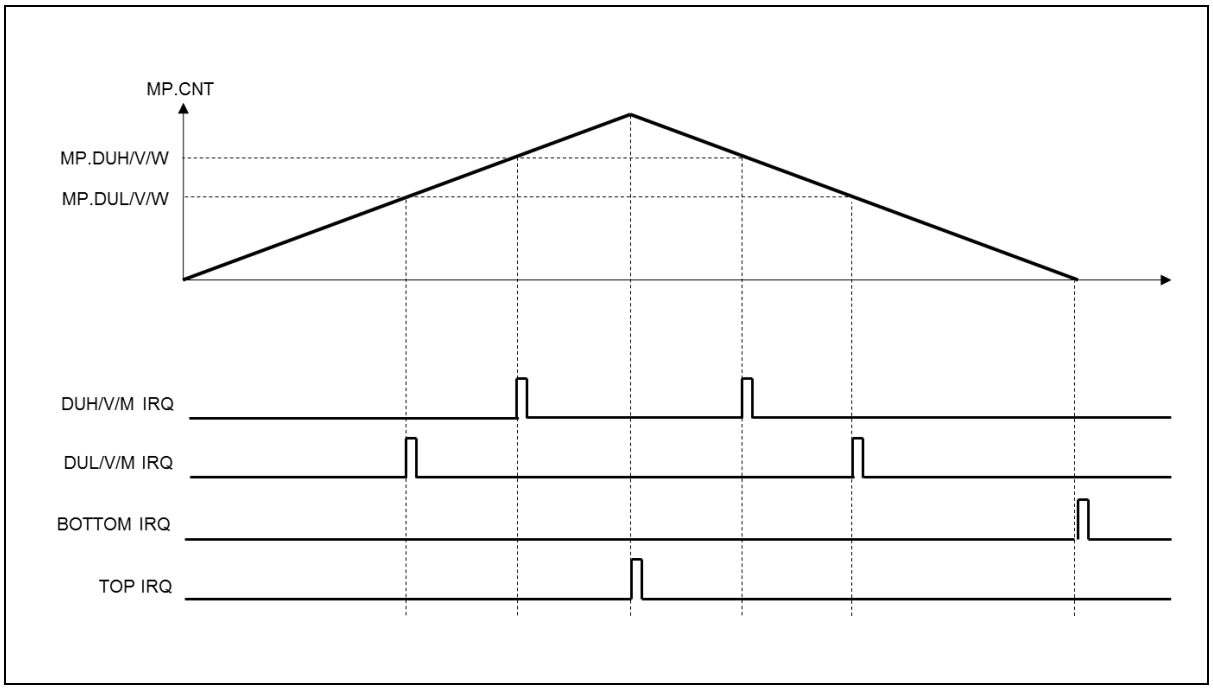


Figure 89. Interrupt Generation Timing

15. Divider (DIV64)

Divider module (DIV64) provides hardware divider ability to accelerate complicated calculation. This divider is sequential 64bit/32bit divider, and requires 32 clock cycles for one operation.

The equation of the operation is shown below:

(AREGH,AREGL) / BREG = (QREGH,QREGL)

The DIV64 supports division of the following dividends:

- Unsigned 64bit dividend
- Unsigned 32bit divisor
- Unsigned 64bit quotient
- Unsigned 32bit remainder
- Unsigned 32 cycle operating time.

15.1 Drvider block diagram

Figure 90 describes normal mode of Drvider in block diagram.

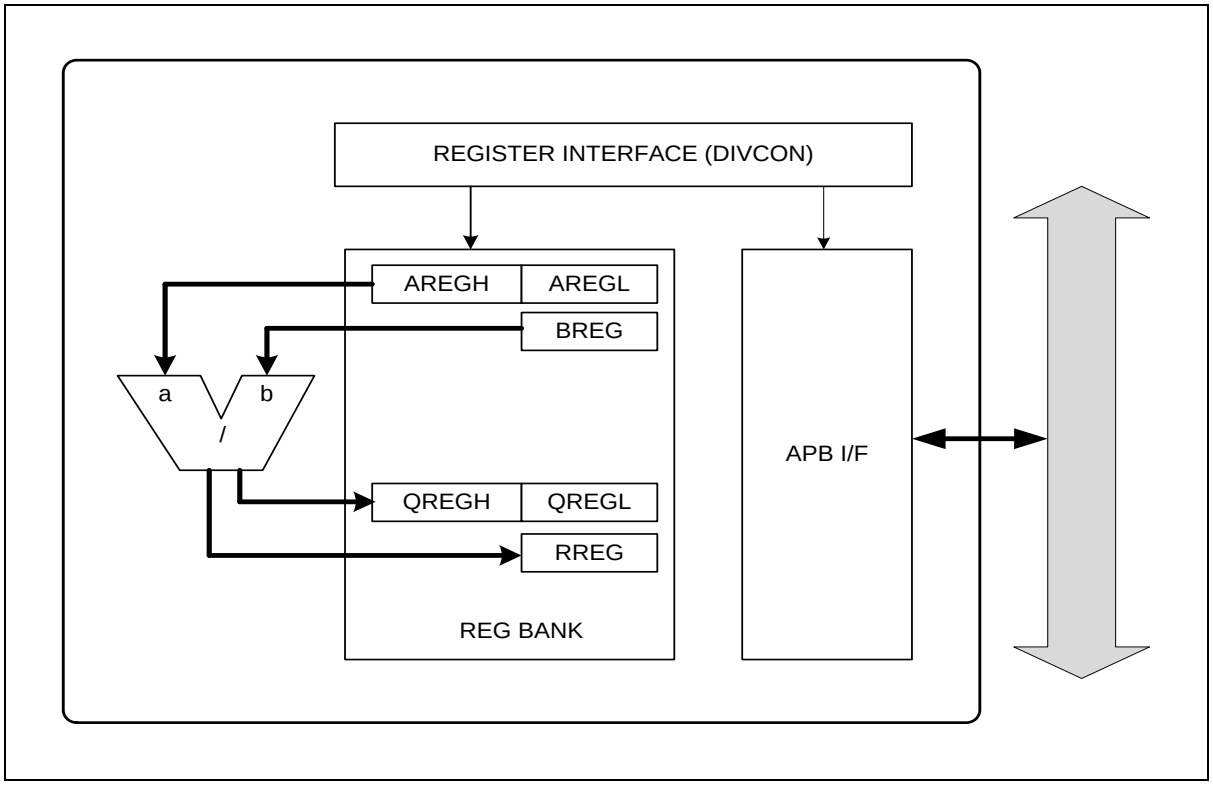


Figure 90. Drvider Block Diagram

15.2 Registers

Base address of DIV64 is introduced in the followings:

Table 49. Base Address of Divder

Name	Base address
DIV64	0x4000_0500

Table 50. DIV64 Register Map

Name	Offset	Type	Description	Reset value	Reference
CR	0x0000	RW	DIV control register	0x00000000	15.2.1
AREGL	0x0004	RW	Most 32bit data register for dividend	0x00000000	15.2.2
AREGH	0x0008	RW	Least 32bit data register for dividend	0x00000000	15.2.3
BREG	0x000C	RW	32bit data register for divisor	0x00000000	15.2.4
QREGL	0x0010	R	Most 32bit data register for quotient	0x00000000	15.2.5
QREGH	0x0014	R	Least 32bit data register for quotient	0x00000000	15.2.6

15.2.1 CR: Divider Control Register

DIVCON register control the hardware divider module.

CR=0x4000_0500

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					I_ERROR	BUSY	DONE				MODE				START
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
					RO	RO	RO				RW				RW

10	I_ERROR	Divide by zero flag	
		0	Not divide by zero
		1	Divide by zero
9	BUSY	Divider is now under operating	
		0	Divider is not busy
		1	Divider is busy
8	DONE	Divider operation done flag	
		0	Divider is now operating
		1	Divider operation is done
4	MODE	Start operation mode	
		0	START bit write operation will trigger the divide operation
		1	BREG register write operation will trigger the divide operation
0	START	Divide operation start command. This bit is effective when MODE bit is 0	
		0	No effect
		1	Start divider

15.2.2 AREGL: AREG (Dividend) Lower 32bit Register.

Lower 32bit value of dividend should be written this register

AREGL=0x4000_0504

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
AREGL[31:0]			
0x0000_0000 RW			

31 0	AREGL	Lower 32 bit value for dividend A.

15.2.3 AREGH: AREG (Dividend) High 32bit Register

High 32bit value of dividend should be written this register

AREGH=0x4000_0508

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
AREGH[63:32]			
0x0000_0000 RW			

31 0	AREGH	High 32 bit value for dividend A.

15.2.4 BREG: BREG (Divisor) Register.

32bit value of divisor should be written this register.

When MODE bit set 1, the divide operation will be started automatically as soon as writing this register.

BREG=0x4000_050C

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
BREG[31:0]			
0x0000_0000 RW			

31 0	BREG	32 bit value for divisor B.

15.2.5 QREGL: QREG (Quotient) Lower 32bit Register

The divider will store lower 32bit value of quotient in this register

QREGL=0x4000_0510

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
QREGL[31:0]			
0x0000_0000 R			

31 0	QREGL	Lower 32 bit value for quotient.

15.2.6 QREGH: QREG (Quotient) High 32bit Register

The divider will store high 32bit value of quotient in this register

QREGH=0x4000_0514

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
QREGH[63:32]																															
0x0000_0000 R																															

31 0	QREGH	High 32 bit value for quotient.

15.2.7 RREG: RREG (Remainter) Register.

The divider will store 32bit value of remainter in this register

RREG=0x4000_0518

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
RREG[31:0]																															
0x0000_0000 R																															

31 0	RREG	32 bit value for remainder.

16. 12-bit Analog-to-Digital Converter (ADC)

ADC block of AC30M1x64/AC30M1x32 series consists of an independent ADC unit featuring the followings:

- 12 channels of analog inputs
- Single and Continuous conversion mode
- Up to 8 times sequential conversion supports
- Software trigger supports
- 3 internal trigger sources supports (Soft-trig, MPWM, 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(2.4V~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
AN11	A	ADC Input 11

16.1 12-bit ADC block diagram

In this section, 12-bit ADC is described in a block diagram in Figure 91.

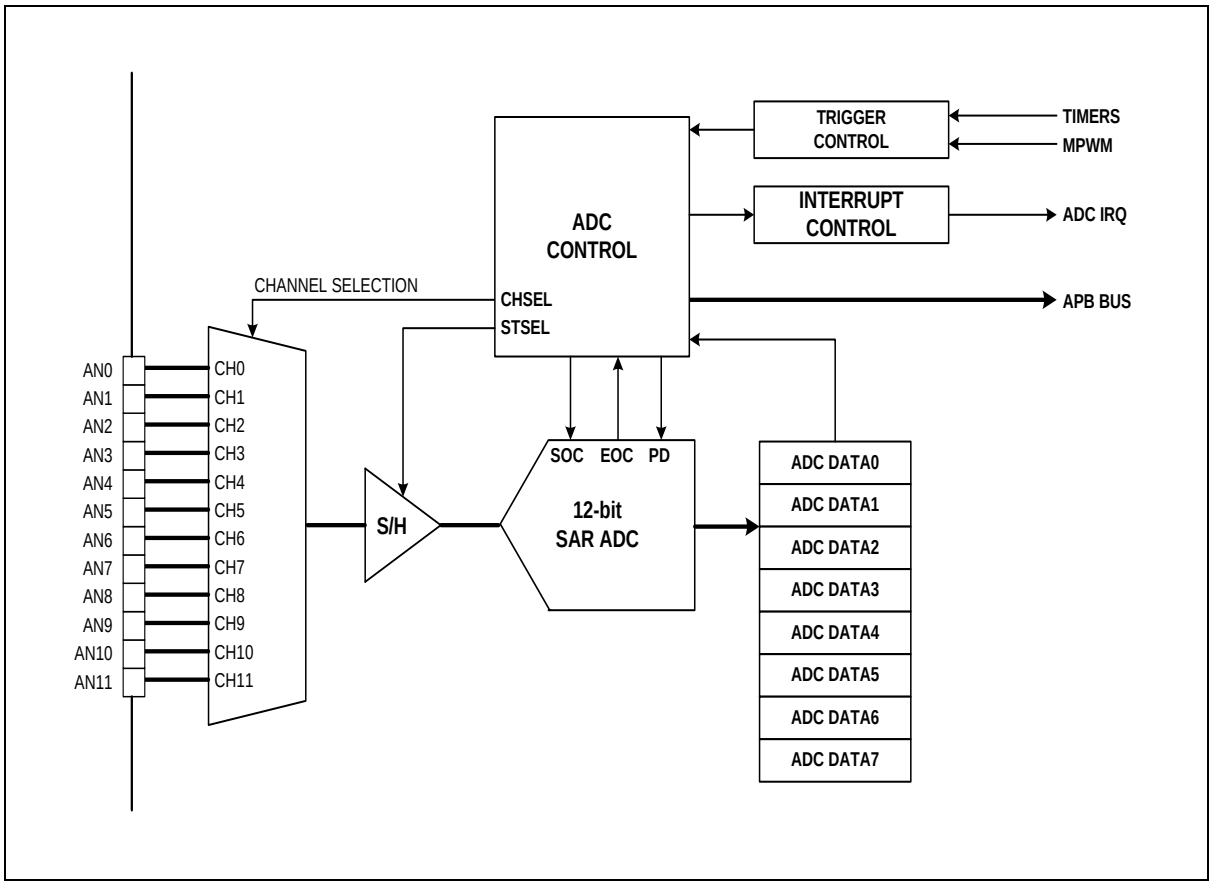


Figure 91. 12-bit ADC Block Diagram

16.2 Registers

Base address of ADC is introduced in the followings:

Table 52. Base Address of ADC

Name	Base address
ADC	0x4000_B000

Table 53. ADC Register Map

Name	Offset	Type	Description	Reset value	Reference
AD.MR	0x0000	RW	ADC Mode register	0x00	16.2.1
AD.CSCR	0x0004	RW	ADC Current Sequence/Channel register	0x00	16.2.2
AD.CCR	0x0008	RW	ADC Clock Control register	0x80	16.2.3
AD.TRG	0x000C	RW	ADC Trigger Selection register	0x00	16.2.4
-	0x0010	-	Reserved	-	
-	0x0014	-	Reserved	-	
AD.SCSR	0x0018	RW	ADC Burst mode channel select	0x00	16.2.5
AD.CR	0x0020	RW	ADC Control register	0x00	16.2.6
AD.SR	0x0024	RW	ADC Status register	0x00	16.2.7
AD.IER	0x0028	RW	ADC Interrupt Enable register	0x00	16.2.8
-	0x002C	-	Reserved	-	
AD.DR0	0x0030	R	ADCn Sequence 0 Data register	0x00	16.2.9
AD.DR1	0x0034	R	ADCn Sequence 1 Data register	0x00	16.2.9
AD.DR2	0x0038	R	ADCn Sequence 2 Data register	0x00	16.2.9
AD.DR3	0x003C	R	ADCn Sequence 3 Data register	0x00	16.2.9
AD.DR4	0x0040	R	ADCn Sequence 4 Data register	0x00	16.2.9
AD.DR5	0x0044	R	ADCn Sequence 5 Data register	0x00	16.2.9
AD.DR6	0x0048	R	ADCn Sequence 6 Data register	0x00	16.2.9
AD.DR7	0x004C	R	ADCn Sequence 7 Data register	0x00	16.2.9

16.2.1 AD.MR: ADC Mode Register

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

AD.MR=0x4000_B000

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

16 12	STSEL	Sampling Time Selection ADC Sample & Hold circuit sampling time become (2 + STSEL[4:0]) MCLK cycles Minimum sampling time is 2 MCLK cycles			
10 8	SEQCNT	Number of coversion in a sequence If ADMOD is 2'h0 and SEQCNT is not 3'h0, CSEQN will be increased up to SEQCNT by trigger event. SEQCNT is burst count In burst mode. See 14.4.1~3 to know between Triggers and SEQCNT in burst conversion mode and in single sequential convertion mode			
		000	1st single sequential conversion or 1 burst count	100	5st single sequential conversion or 5 burst count
		001	2nd single sequential conversion or 2 burst counts	101	6st single sequential conversion or 6 burst counts
		010	3rd single sequential conversion or 3 burst counts	110	7st single sequential conversion or 7 burst counts
		011	4st single sequential conversion or 4 burst counts	111	8st single sequential conversion or 8 burst counts
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 (single sequential conversion mode when SEQCNT is not 0x0)		
		01	Burst conversion mode		
		10	Reserved		
		11	Reserved		
1 0	TRGSEL	00	Event Trigger Disabled/Soft-Trigger Only		
		01	Timer Event Trigger		
		10	MPWM Event Trigger		
		11	Reserved		

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

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

16.2.2 AD.CSCR: ADC Current Sequence/Channel Register

ADC Current Seqneuce/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 change next sequence number. Writing CSEQN as 0x7 when CSEQN is 0x3 and AD.MR.SEQCNT is 0x7, the next sequence number is 0x7 and AD converts the channel of AD.SCSR.SEQ7CH and the 4,5,6 sequence are skipped. This register should be written first before AD.SCSR.

AD.CSCR=0x4000_B004

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

6 4	CSEQN	Current Sequence Number, can write when not abusy AD starts conversion the AD.SCSR.SEQ*CH's channel by AD.TRG.SEQTRG* in Single sequential mode. AD starts conversion the AD.SCSR.SEQ*CH's channel by AD.TRG.BSTTRG in Burst mode	
		0000	Current Sequence is 0 the AD.SCSR.SEQ0CH's channel is converted by AD.TRG.SEQTRG0 in Single sequential mode or by AD.TRG.BSTTRG in Burst mode
		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	ADC channel 8 is active
		1001	ADC channel 9 is active
		1010	ADC channel 10 is active
		1011	ADC channel 11 is active
		1100	reserved
		1101	reserved
		1110	reserved
		1111	reserved

16.2.3 AD.CCR: ADC Clock Control Register

ADC Control Registers are 16-bit registers. ADC period register

AD.CCR=0x4000_B008

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCPDA	CLKDIV							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: 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-MCCR7)
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%

16.2.4 AD.TRG: ADC Trigger Selection Register

ADC Trigger registers are 32-bit registers including the 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.

AD.TRG=0x4000_B00C

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	RW RW RW RW	RW RW RW RW	RW RW RW RW	RW RW RW RW	RW RW RW RW	RW RW RW RW

31 28	SEQTRG7	8th Sequence Trigger Source
27 24	SEQTRG6	7th Sequence Trigger Source
23 20	SEQTRG5	6th Sequence Trigger Source
19 16	SEQTRG4	5th Sequence Trigger Source
15 12	SEQTRG3	4th Sequence Trigger Source
11 8	SEQTRG2	3rd Sequence Trigger Source
7 4	SEQTRG1	2nd Sequence Trigger Source
3 0	SEQTRG0 BSTTRG	1st Sequence Trigger Source Burst conversion Trigger Source

Table 54. AD.TRG Trigger Selection Register

Value	Timer (TRGSEL '2'h1)	MPWM (TRGSEL '2'h2)
0	Timer 0	MP.ATR1
1	Timer 1	MP.ATR2
2	Timer 2	MP.ATR3
3	Timer 3	MP.ATR4
4		MP.ATR5
5		MP.ATR6
6	-	BOTTOM
7	-	PERIOD

16.2.5 AD.SCSR: ADC Sequence Channel Selection Register

ADC Burst Mode Channel Select Register is 32-bit register. For ADC single mode, it uses SEQ0CH to select channel.

AD.SCSR=0x4000_B018

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

16.2.6 AD.CR: ADC Control Register

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

AD.CR=0x4000_B020

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

7	ASTOP	0	No operation
		1	ADC conversion stop (will be clear next @ADC clock) If ASTOP is set after a conversion starts, the conversion is completed and AD stops.
0	ASTART	0	No ADC conversion
		1	ADC conversion start when single mode (AD.MR.ADMOD and AD.MR.SEQCNT are 0x0. this bit will be cleared by coming @AD clock. If ASTART is set as 0 when ARST is 0 in Timer/MPWM trigger event mode, AD converts to AD.MR.SEQCNT once and AD stops. ASTART should be written to start the conversion sequence again

16.2.7 AD.SR: ADC Status Register

ADC Status Register is 8-bit register.

AD.SR=0x4000_B024

7	6	5	4	3	2	1	0
EOC	ABUSY	-	-	TRGIRQ	EOSIRQ	-	EOCIRQ
0	0	-	-	0	0	-	0
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
-	-		Reserved.
-	-		Reserved.
3	TRGIRQ		ADC Trigger interrupt flag (Write "1" to clear flag) (0: no int / 1: int occurred)
2	EOSIRQ		This flag will be set at the end of a burst conversion or a sequence convrersion set (Write "1" to clear flag). *Sequence conversion set is the operation that AD converts to AD.MR.SEQCNT.
		0	None.
		1	End-of-Sequence Interrupt occurred in burst or single sequential mode
0	EOCIRQ		This flag will be set upon each conversion in a single is occurred (Write "1" to clear flag)
		0	None.
		1	End-of-Conversion Interrupt occurred

16.2.8 AD.IER:Interrupt Enable Register

AD.IER=0x4000_B028

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

3	TRGIRQE	ADC trigger conversion interrupt enable
2	EOSIRQE	ADC sequence conversion interrupt enable
1	-	Reserved.
0	EOCIRQE	ADC single conversion interrupt enable

16.2.9 AD.DRm: ADC Sequence Data Register 0 to 7

ADC Data Registers are 16-bit registers. ADC conversion result register.

AD.DR0=0x4000_B030, AD.DR1=0x4000_B034, AD.DR2=0x4000_B038, AD.DR3=0x4000_B03C

AD.DR4=0x4000_B040, AD.DR5=0x4000_B044, AD.DR6=0x4000_B048, AD.DR7=0x4000_B04C

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

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

16.3 Functional description

16.3.1 AD Conversion timing diagram

When AD.MR.ADMOD is 0x0 and AD.MR.SEQCNT is 0x0, ADC conversion will be started by writing '1' in the AD.CR.ASTART.

Once AD.CR.ASTART is set, SOC (start of Conversion) will be activated in 3 ADC clocks and AD.SR.EOCIRQ will be set in 2 ADC clocks and 2 PCLKs after the End of Conversion.

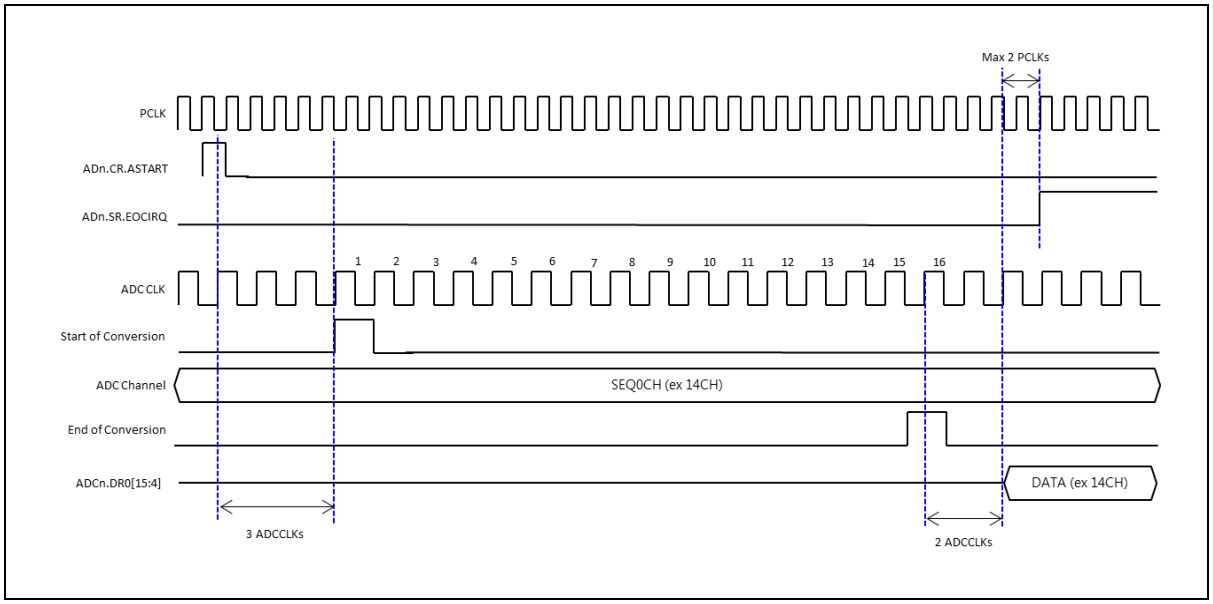


Figure 92. ADC Single Mode Timing (when ADCn.MR.AMOD = '0')

16.3.2 ADC burst conversion mode timing diagram

Burst conversion mode (burst mode) is when AD.MR.ADMOD is 0x1. When there are two sources to make SOC in burst mode. First is TRG event (TIMER and MPWM) and the other is AD.CR.ASTART.

When AD.MR.TRGSEL is set as timer event trigger or mpwm event trigger, SOC will be made by the trigger of AD.TRG.BSTTRG (AD.TRG[3:0]). For example, ADC conversion will be started by the trigger of TIMER3 if AD.TRG.BSTTRG is set as TIMER3. Once the BSTTRG's trigger events, ADC will covert ADC Channels as much as AD.MR.SEQCNT set. See Figure 93.

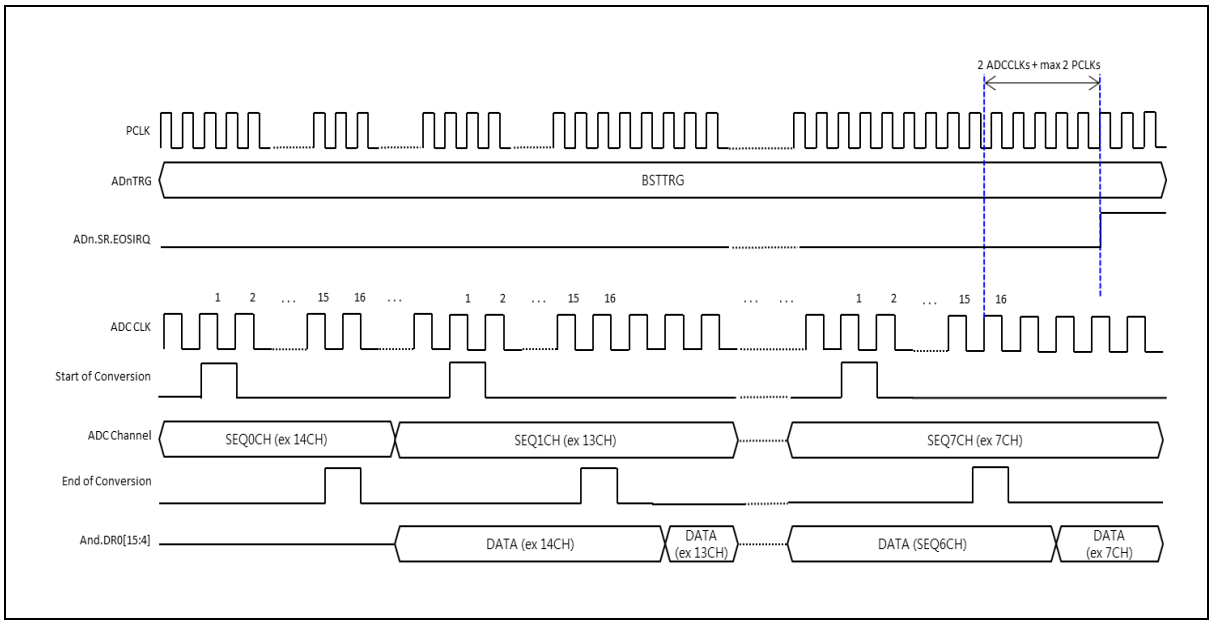


Figure 93. ADC Burst Mode Timing (when AD.MR.AMOD = '1')

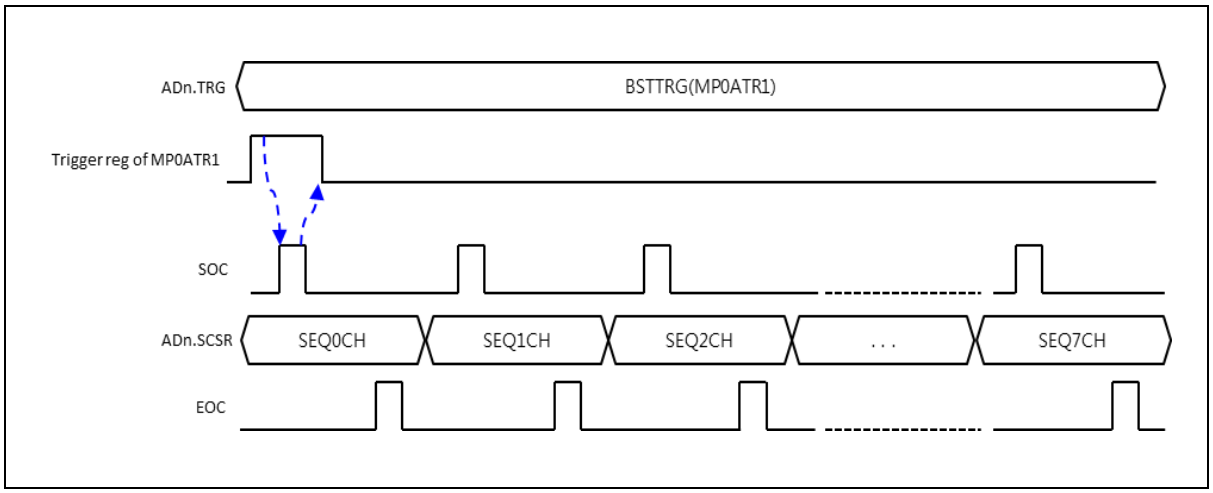


Figure 94. ADC Trigger Timing in Burst Mode (SEQCNT = 3'b111, 8 sequence coversion)

16.3.3 ADC sequential conversion mode timing diagram

Single sequential conversion mode (Single sequential mode) is when AD.MR.ADMOD is 0x0 and AD.MR.SEQCNT is not 0x0. To set sequential conversion mode, AD.MR.AMOD is 2'b00 and AD.MR.SEQCNT is not 2'b00.

The operation of sequential mode is the almost same as the burst mode. The difference is a source of SOC. Each SOC is made by the trigger of the SEQTRGx as each SEQCNT. Refer to Figure 95.

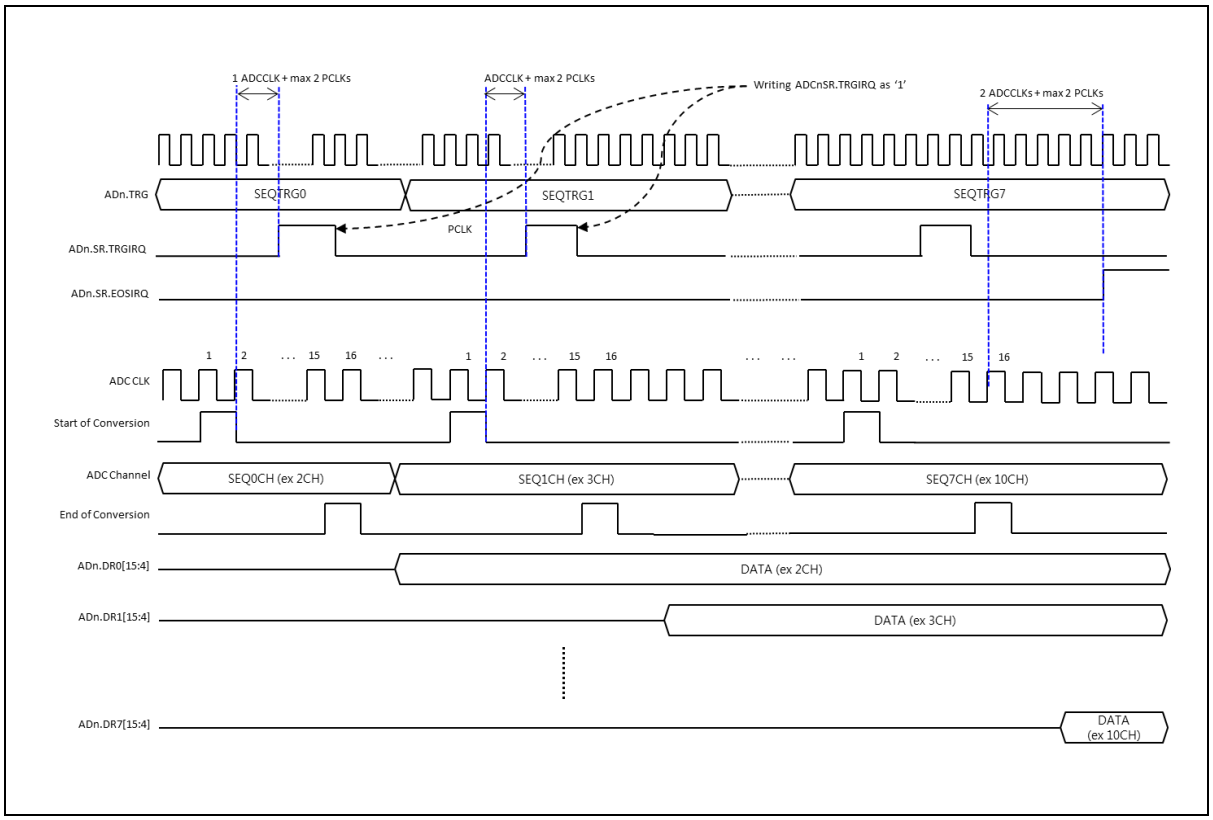


Figure 95. ADC Sequential Mode Timing (when AD.MR.AMOD = '0 and AD.MR.SEQCNT ≠ '0')

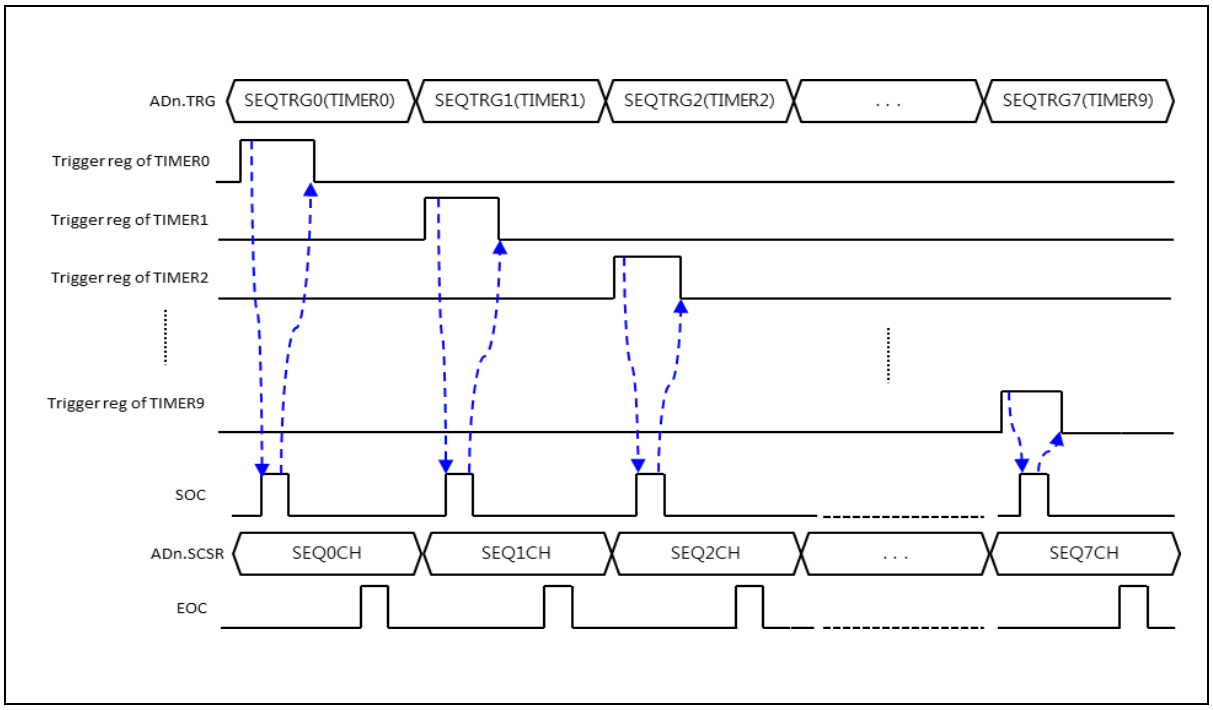


Figure 96. ADC Trigger Timing in Sequential Mode (SEQCNT = 3'b111, 8 sequence conversion)

17. Electrical characteristics

17.1 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 (VDD)	VDD	-0.5	+6	V
Analog power supply (AVDD)	AVDD	-0.5	+6	V
VDC output voltage	VDD18			V
Input high voltage		-	VDD+0.5	V
Input low voltage		VSS – 0.5	-	V
Output low current per pin	I_{OL}		5	mA
Output low current total	$\sum I_{OL}$		40	mA
Output high current per pin	I_{OH}		5	mA
Output high current total	$\sum I_{OH}$		40	mA
Power consumption				mW
Input main clock range		4	16	MHz
Operating frequency		-	40	MHz
Storage temperature	Tst	-55	+125	℃
Operating temperature	Top	-40	+105	℃

17.2 DC characteristics

Table 56. Recommended Operating Condition

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Supply voltage	VDD		2.2	-	5.5	V
Supply voltage	AVDD		2.2	-	5.5	V
Operating frequency	FREQ	MOSC	4	-	16	MHz
		SOSC	-	32.768	-	kHz
		HSI	38.8	40	41.2	MHz
		LSI	32	40	48	kHz
Operating temperature	Top	Top	-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	-	-	0.2VDD	V
Input high voltage	V_{IH}	Schmitt input	0.8VDD	-	-	V
Output low voltage	V_{OL}	I_{OL} = 3mA	-	-	VSS+1.0	V
Output high voltage	V_{OH}	I_{OH} = -3mA	VDD–1.0	-	-	V
Input high leakage	I_{IH}				4	uA
Input low leakage	I_{IL}		-4			
Pull-up resister	R_{PU}	VDD=5V	30	-	90	kΩ

17.3 Current consumption

Table 58. Current Consumption in Each Mode

(Temperature: +25℃ Only)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Normal operation	IDD_{NORMAL}	LSIOSC=RUN, HSIOSC=RUN MXOSC=RUN, SXOSC=RUN HCLK=LSIOSC	-	2.6	-	mA
		LSIOSC=RUN, HSIOSC=OFF MXOSC=OFF, SXOSC=OFF HCLK=LSIOSC	-	0.7	-	mA
		LSIOSC=RUN, HSIOSC=RUN MXOSC=RUN, SXOSC=RUN HCLK=HSIOSC	-	10.3	-	mA
		LSIOSC=OFF, HSIOSC=RUN MXOSC=OFF, SXOSC=OFF HCLK=HSIOSC	-	9.4	-	mA
		LSIOSC=RUN, HSIOSC=RUN MXOSC=RUN, SXOSC=RUN HCLK=MXOSC	-	4.2	-	mA
		LSIOSC=OFF, HSIOSC=OFF MXOSC=RUN, SXOSC=OFF HCLK=MXOSC	-	3.2	-	mA
		LSIOSC=RUN, HSIOSC=RUN MXOSC=RUN, SXOSC=RUN HCLK=SXOSC	-	2.6	-	mA
		LSIOSC=OFF, HSIOSC=OFF MXOSC=OFF, SXOSC=RUN HCLK=SXOSC	-	0.7	-	mA
SLEEP mode	IDD_{SLEEP}	LSIOSC=RUN, HSIOSC=RUN SXOSC=RUN, MXOSC=RUN HCLK =LSIOSC	-	2.5	-	mA
		LSIOSC=RUN, HSIOSC=OFF SXOSC=OFF, MXOSC=OFF HCLK =LSIOSC		0.6		mA
		LSIOSC=RUN, HSIOSC=RUN SXOSC=RUN, MXOSC=RUN HCLK =HSIOSC		7.6		mA

Table 58. Current consumption in each mode (continued)

(Temperature: +25℃ Only)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
SLEEP mode	IDD_{SLEEP}	LSIOSC=OFF, HSIOSC=RUN SXOSC=OFF, MXOSC=OFF HCLK =HSIOSC		6.8		mA
		LSIOSC=RUN, HSIOSC=RUN SXOSC=RUN, MXOSC=RUN HCLK =MXOSC		3.5		mA
		LSIOSC=OFF, HSIOSC=OFF SXOSC=OFF, MXOSC=RUN HCLK =MXOSC		2.5		mA
		LSIOSC=RUN, HSIOSC=RUN SXOSC=RUN, MXOSC=RUN HCLK =SXOSC		2.5		mA
		LSIOSC=OFF, HSIOSC=OFF SXOSC=RUN, MXOSC=OFF HCLK =SXOSC		0.6		mA
POWER DOWN mode	IDD_{STOP}	LSIOSC=STOP, SIOSC=STOP SXOSC=STOP, MXOSC=STOP HCLK=STOP	-	5	10	uA

NOTES:

1. uart en, 1 port toggle @5VLSIOSC (40KHz)
2. HSIOSC (40MHz), MXOSC (8MHz), SXOSC (32.768KHz)

17.4 POR electrical characteristics

Table 59. POR Electrical Characteristics

(Temperature: -40 ~ +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	VDD18		1.6	1.8	2.0	V
Operating current	IDD_{PoR}	Typ. <6uA, If always on	-	60	-	nA
POR set level	VR_{PoR}	VDD rising (slow)	1.3	1.4	1.55	V
POR reset level	VF_{PoR}	VDD falling (slow)	1.1	1.2	1.4	V

17.5 LVD electrical characteristics

Table 60. LVD Electrical Characteristics

(Temperature: -40 ~ +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	VDD		1.7		5	V
Operating current	IDD_{LVD}	Typ. <6uA when always on	-	1	-	mA
LVD set level 0	VLVD0	VDD falling (slow)	1.58	1.73	2.2	V
LVD set level 1	VLVD1	VDD falling (slow)	2.4	2.65	3.1	V
LVD set level 2	VLVD2	VDD falling (slow)	3.55	3.7	4.15	V

17.6 VDC electrical characteristics

Table 61. VDC Electrical Characteristics

(Temperature: -40 ~ +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	VDD_{VDC}		2.2	-	5.5	V
Current consumption	IDD_{NORM}	@RUN	-	100	150	uA
	IDD_{STOP}	@STOP	-	1	2	uA

17.7 External OSC characteristics

Table 62. External OSC Characteristics

(Temperature: -40 ~ +105℃)

Parameter	Symbol	Condition	Min	Typ	Max	unit
Operating voltage	VDD		2.2	-	5.5	V
IDD		@4MHz/5V	-	240		uA
Frequency	$OSCF_{req}$		4	-	16	MHz
Output voltage	OSC_{VOUT}		1.2	2.4	-	V
Load capacitance	$LOAD_{CAP}$		5	22	35	pF

17.8 ADC electrical characteristics

Table 63. ADC Electrical Characteristics

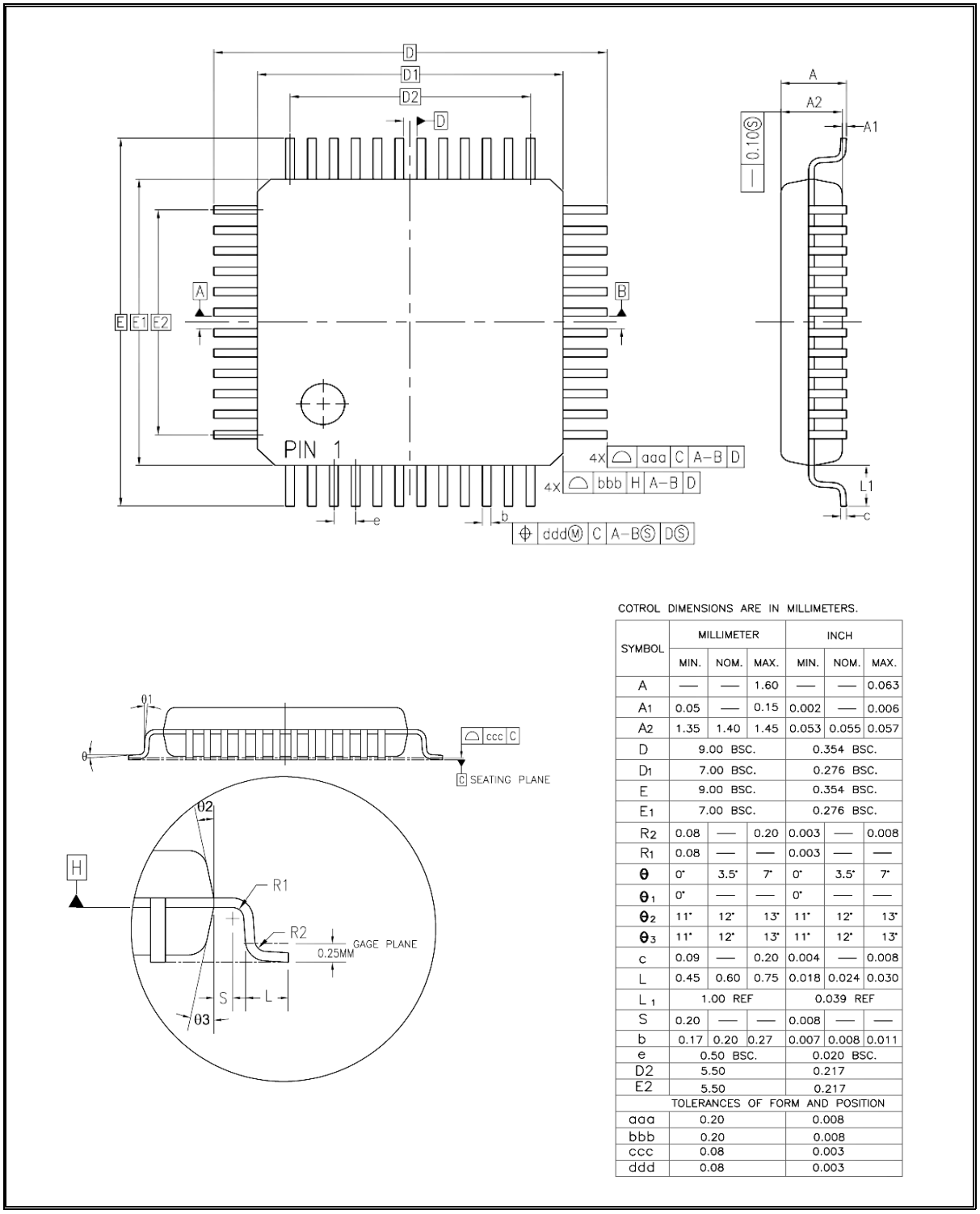
(Temperature: -40 ~ +105℃)

Parameter	Symbol	Condition	Min	Typ.	Max	unit
Operating voltage	AVDD		2.4	5	5.5	V
Resolution				12		Bit
Operating current	IDDA				2.8	mA
Analog input range			0		AVDD	V
Conversion rate				-	1.0	MSPS
Operating frequency	ACLK				16	MHz
DC accuracy	INL			±3.5		LSB
	DNL			±2.5		LSB
Offset error				±1.5		LSB
Full scale error				±1.5		LSB
SNDR	SNDR			68		dB
THD				-70		dB

18. Package information

This chapter provides AC30M1x64/AC30M1x32 series package information.

18.1 48 LQFP package information

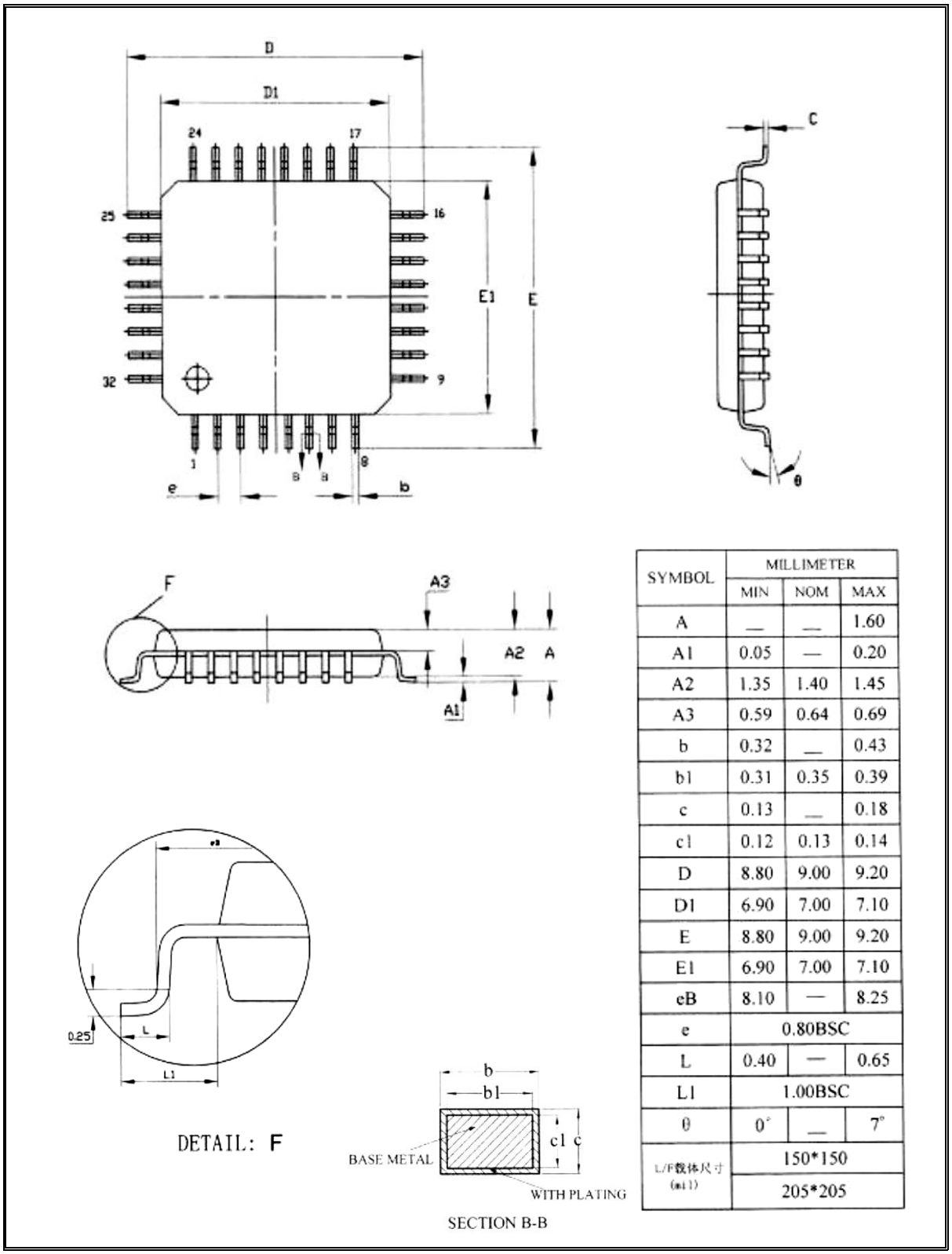


COTROL DIMENSIONS ARE IN MILLIMETERS.

SYMBOL	MILLIMETER			INCH		
	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	—	—	1.60	—	—	0.063
A1	0.05	—	0.15	0.002	—	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
D	9.00 BSC.			0.354 BSC.		
D1	7.00 BSC.			0.276 BSC.		
E	9.00 BSC.			0.354 BSC.		
E1	7.00 BSC.			0.276 BSC.		
R2	0.08	—	0.20	0.003	—	0.008
R1	0.08	—	—	0.003	—	—
θ	0°	3.5°	7°	0°	3.5°	7°
θ1	0°	—	—	0°	—	—
θ2	11°	12°	13°	11°	12°	13°
θ3	11°	12°	13°	11°	12°	13°
c	0.09	—	0.20	0.004	—	0.008
L	0.45	0.60	0.75	0.018	0.024	0.030
L1	1.00 REF			0.039 REF		
S	0.20	—	—	0.008	—	—
b	0.17	0.20	0.27	0.007	0.008	0.011
e	0.50 BSC.			0.020 BSC.		
D2	5.50			0.217		
E2	5.50			0.217		
TOLERANCES OF FORM AND POSITION						
aaa	0.20			0.008		
bbb	0.20			0.008		
ccc	0.08			0.003		
ddd	0.08			0.003		

Figure 97. 48 LQFP Package Outline

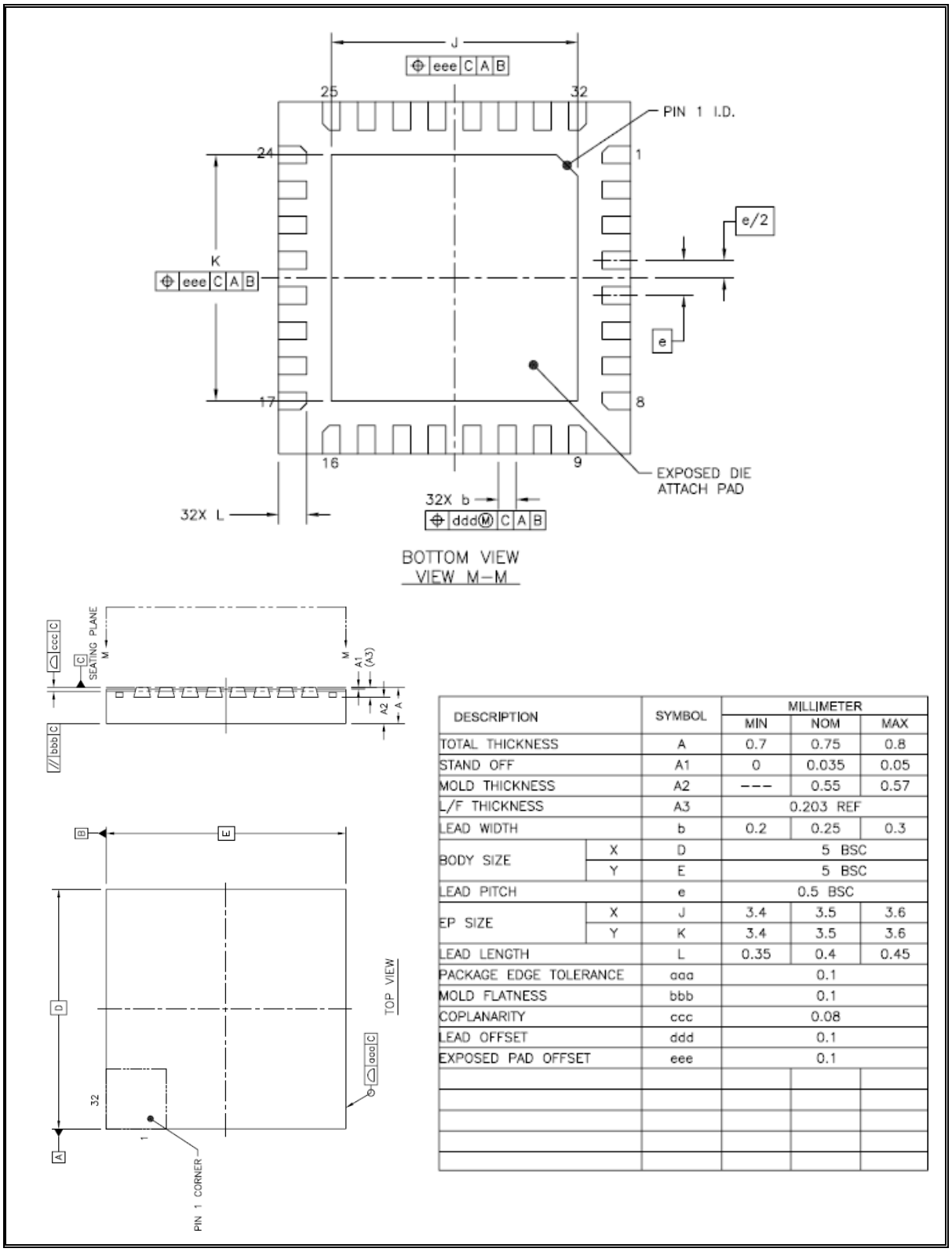
18.2 32 LQFP package information



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.20
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.32	—	0.43
b1	0.31	0.35	0.39
c	0.13	—	0.18
c1	0.12	0.13	0.14
D	8.80	9.00	9.20
D1	6.90	7.00	7.10
E	8.80	9.00	9.20
E1	6.90	7.00	7.10
eB	8.10	—	8.25
e	0.80BSC		
L	0.40	—	0.65
L1	1.00BSC		
θ	0°	—	7°
L/F载体尺寸 (unit)	150*150		
	205*205		

Figure 98. 32 LQFP Package Outline

18.3 32 QFN package information



DESCRIPTION		SYMBOL	MILLIMETER		
			MIN	NOM	MAX
TOTAL THICKNESS		A	0.7	0.75	0.8
STAND OFF		A1	0	0.035	0.05
MOLD THICKNESS		A2	---	0.55	0.57
L/F THICKNESS		A3		0.203 REF	
LEAD WIDTH		b	0.2	0.25	0.3
BODY SIZE	X	D		5 BSC	
	Y	E		5 BSC	
LEAD PITCH		e		0.5 BSC	
EP SIZE	X	J	3.4	3.5	3.6
	Y	K	3.4	3.5	3.6
LEAD LENGTH		L	0.35	0.4	0.45
PACKAGE EDGE TOLERANCE		aaa		0.1	
MOLD FLATNESS		bbb		0.1	
COPLANARITY		ccc		0.08	
LEAD OFFSET		ddd		0.1	
EXPOSED PAD OFFSET		eee		0.1	

Figure 99. 32 QFN Package Outline

19. Ordering information

Table 64. AC30M1x64/AC30M1x32 Series Device Ordering Information

Device name	Flash	SRAM	UART	SPI	I2C	MPWM	ADC	I/O ports	Package
AC30M1464LBN*	64KB	4KB	2	1	1	1	12 ch	44	LQFP-48
AC30M1364LBN	64KB	4KB	2	1	1	1	10 ch	30	LQFP-32
AC30M1364UB*	64KB	4KB	2	1	1	1	10 ch	30	QFN-32
AC30M1332LBN*	32KB	4KB	2	1	1	1	10 ch	30	LQFP-32
AC30M1332UB*	32KB	4KB	2	1	1	1	10 ch	30	QFN-32

: For available options or further information on the devices with an "" mark, please contact the ABOV sales office.

20. Development tools

This chapter introduces wide range of development tools for AC30M1x64/AC30M1x32. 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.

20.1 Compiler

ABOV semiconductor does not provide any compiler for AC30M1x64/AC30M1x32. However, since AC30M1x64/AC30M1x32 have ARM's high-speed 32-bit Cortex-M0 Cores for their CPU, you can use all kinds of third party's standard compiler such as Keil C Compiler. These compilers' output debug information can be integrated with our A-Link and A-Link Pro. Please visit our website www.abovsemi.com for more information regarding the A-Link and A-Link Pro.

20.2 Debugger

The A-Link and A-Link Pro support ABOV Semiconductor's AC30M1x64/AC30M1x32 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 100. 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.

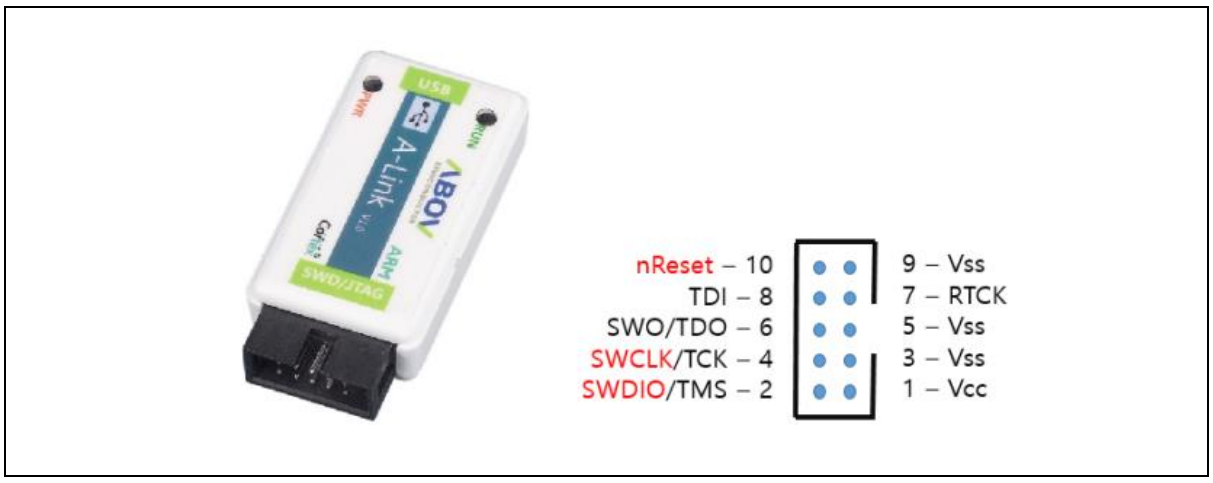


Figure 100. A-Link and Pin Descriptions

20.3 Programmer

20.3.1 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

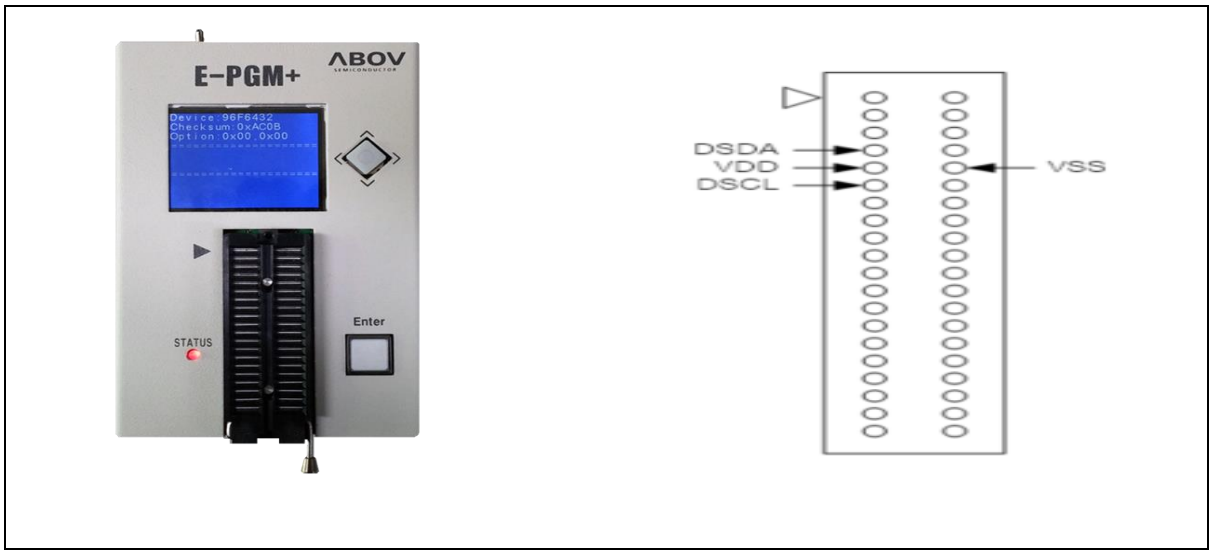


Figure 101. E-PGM+ (Single Writer) and Pin Descriptions

20.3.2 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 102. E-Gang4 and E-Gang6 (for Mass Production)

Revision history

Date	Version	Description
2016/7/22	1.0.0	File created
2016/8/17	1.1.0	32pin map diagram changed
2016/10/10	1.1.1	description errors were corrected Explanation of Un.IER was modified.
2016/10/17	1.1.2	Modified MP.Duty, MP.SR, MP.OLR Explanation . Modified figure of MPWM functional description.
2016/10/26	1.1.3	Modified LVD Voltage Level . Delete 4.35V
2016/11/18	1.1.4	Add Low Voltage Reset (chapter1.3.3)
2016/11/23	1.1.5	Electrical Characteristic Output High/Low Current Total Value Modifed, Current Consumption Condition Addtion.
2017/01/16	1.1.6	Modified SWD mode description.
2020/6/09	1.00	1st creation(PMO)
2022/10/17	1.01	ADC description updated Update the template of this document

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