Confidentiality level: external use

UM800Y User Manual

V1.7.1

UNICMICRO
广芯微电子

Unicmicro (Guangzhou) Co., Ltd.

http://www.unicmicro.com/

Copyright & Disclaimer

Contents

List of Figures

List of Tables

1 Overview

UM800Y is a universal MCU with 1-T 8051 core, which runs faster and with better performance than the traditional 8051 at the same system clock, and the instruction code is fully compatible with the traditional.

1.1 Main Features

- 8-bit MCU based on 1-T 8051 instruction pipeline structure
- eFlash: up to 64 KB (supporting IAP)
- RAM: 256 B Idata, and up to 2048 B Xdata
- Operating voltage: 2.5–5.5 V
- Clock source
  - Internal RCH: 24 MHz
  - Internal RCL: 38 kHz
  - External clock input: < 24 MHz
  - Crystal resonator: < 24 MHz
- 17 bidirectional CMOS I/O pins (built-in pull-up/pull-down resistors)
- Up to 3 GTimers, supporting input capture and complementary PWM with dead-time insertion
- Up to 11 × 16-bit PWM outputs
- 1 x LPTimer with two input captures and two PWM outputs
- UART: UART0 / UART1 / UART2 / UART3
- One SPI, master / slave
- 8-channel 12-bit ADC @ 3 V, 1 Msps
- CAN: 1x, CAN2.0 A/B (supported for some types)
- WDT
- Buzzer
- Interrupt source
  - EFC interrupt
  - External interrupt supported on all IOs

- UART0/UART1/UART2/UART3
- ADC
- PWM cycle interrupt
- SPI
- I2C
- LPTIMER
- GTimer

- Reset source
  - POR
  - LVR
  - LVD
  - WDT reset
  - Pin reset
- Built-in low-voltage detection (LVD) module
- Power-saving mode
  - Typical current in Stop mode: 0.75 μA
  - Typical current in DeepSleep mode: 1.1 μA
  - Typical current in Sleep mode: 245 μA
  - Typical current in Active mode: 80 μA/MHz

1.2 Functional Block Diagram

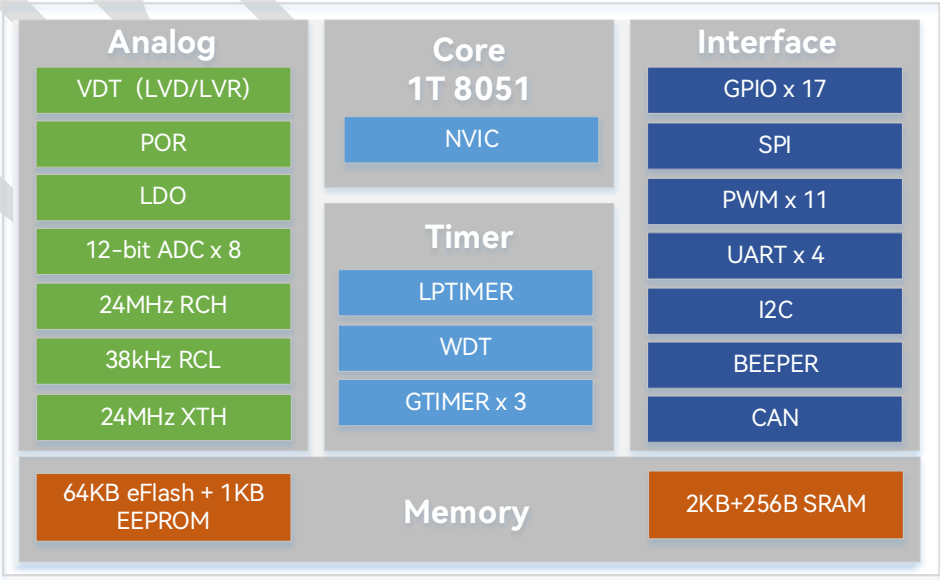


Figure 1-1: Functional Block Diagram

2 Core

2.1 Main Features

CPU core registers: ACC, B, PSW, SP, DPS, DPL, DPH, DPL1, DPH1.

2.2 Program Status Word (PSW) Register

The PSW register contains the program status information.

2.3 Accumulator (ACC)

Accumulator is a common special register used to store the operands and the operation results.

2.4 B Register

B register is used for the multiply-divide instruction, or used as a general temporary register in other instructions.

2.5 Data Pointer (DPTR)

The data pointer (DPTR) is a 16-bit special register with the higher-byte register denoted by DPH and the lower-byte register by DPL, which can be handled either as a 16-bit register DPRT or as two separate 8-bit registers DPH and DPL. It contains double data pointers DPTR & DPTR1 that can be selected via the DPS (bit0) register.

3 Memory

3.1 Main Features

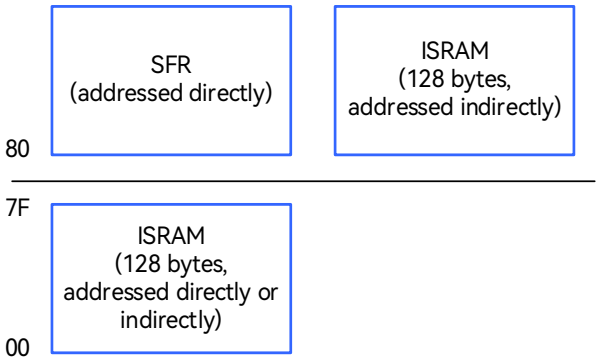
The memory includes internal RAM (ISRAM) and external storage memory, the latter of which contains a program memory EFLASH for storing user programs.

3.2 Internal RAM (ISRAM)

The chip provides a 256-byte internal RAM (ISRAM) for data storage, which is accessed by MOV instructions. ISRAM is divided into lower 128 bytes and higher 128 bytes.

- ISRAM of lower 128 bytes (00H–7FH) can be addressed directly or indirectly.
- ISRAM of higher 128 bytes (80H–FFH) can only be addressed indirectly.
- Special function register (SFR) (80H–FFH) can only be addressed directly.
- External RAM can be accessed directly via MOVX instruction.

The higher 128-byte ISRAM occupies the same address space as the SFR, but is physically separated from the SFR space. When an instruction accesses an internal address higher than 7FH, the CPU can distinguish whether to access the ISRAM of higher 128 bytes or the SFR according to the type of instruction accessed.



3.3 External Storage Memory

The chip also offers a 2-KB external storage memory XRAM for data storage.

3.4 External Storage Memory Mapping

There are two boot modes for the chip: Boot and Main. The specific boot process is as follows:

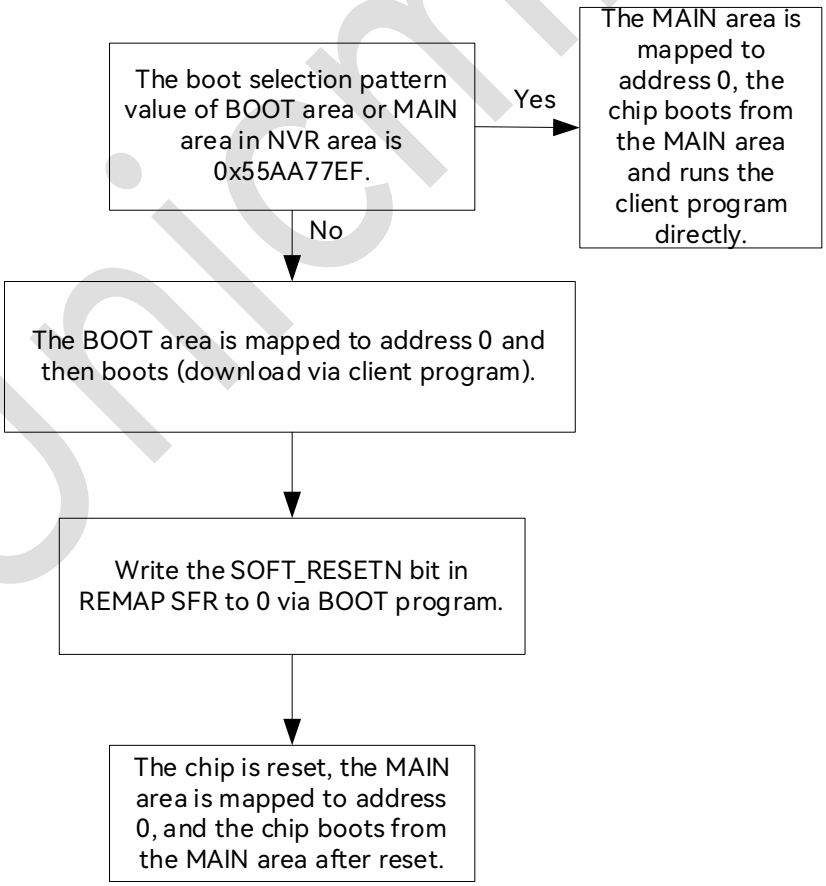
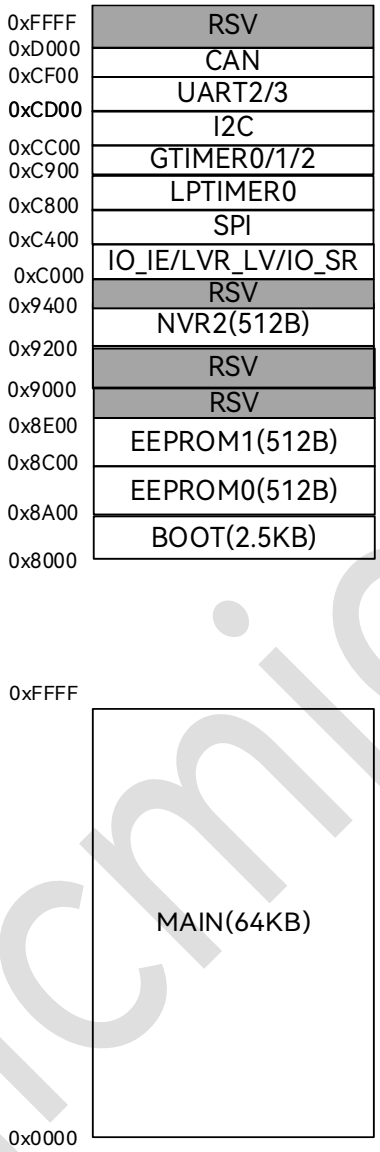


Figure 3-1: Chip Boot Flow Chart

The address mapping after remapping is shown below:



Note:

- The BOOT area, NVR area, EEPROM0 area and EEPROM1 area are read with xdata keyword.
- The data in SRAM are read with xdata keyword.
- The base address of the Main area is 0x0000, which is read with code keyword.

3.5 Program Memory EFLASH

Main features of EFLASH:

- 64 KB program memory
- IAP supported
- Data retention: at least 10 years

EFLASH programming (EFLASH can be read and written in the following two ways):

1. In IAP mode, program the unused EFLASH area via user program code, as described in the chapter of "EFC".
2. Program EFLASH via the system Boot program and serial port.

4 System Configuration Utility (SFR)

4.1 Register Description

Table 4-1: List of Registers

Address	Name	Description
80H	P0	P0 register
81H	SP	Stack pointer register
82H–85H	DPTR	Data pointer register
87H	PCON	PCON register
8EH	PDSEL	Power-down mode selection bit register
8FH	POREN	Power-down reset enable register
90H	P1	P1 register
91H	LDOTRIML	Bandgap trimming register
92H	DPS	Data pointer selection register
97H	P0DR	Port P0 driving capability configuration register
9AH	IEN2	Interrupt enable register
BCH	RCLTRIML	On-chip RCL trimming value low-order register
9FH	RCLTRIM	On-chip RCL trimming value high-order register
A0H	P2	P2 register
A1H	OUS	Flash erase time scale register
A4H	P0AL	P0_0–3 port rising-edge/falling-edge triggered interrupt enable register
A8H	IEN0	Interrupt enable register
A9H	IP0	Interrupt priority register
B9H	IP1	Interrupt priority register
ABH	P0AH	P0_4 port rising-edge/falling-edge triggered interrupt enable register
AEH	P1AL	P1_0–3 port rising-edge/falling-edge triggered interrupt enable register
AFH	REMAP	REMAP register
B0H	P1AH	P1_4–5 port rising-edge/falling-edge triggered interrupt enable register
B1H	CLKST	System clock setting register

Address	Name	Description
B2H	ESTCR	External reset enable register
B3H	XTHCTR	External XTH register
B6H	ADCDR0	A/D channel RX data low-order register
B7H	ADCDR1	A/D channel RX data high-order register
B8H	IEN1	Interrupt enable register
BDH	LDOTRIMH	Bandgap trimming register
BEH	RCHTRIMH	On-chip RCH trimming value high-order register
BFH	RCHTRIML	On-chip RCH trimming value low-order register
C0H	P2AL	P2_0~3 port rising-edge/falling-edge triggered interrupt enable register
D0H	PSW	Program status word register
D5H	P0PD	Port P0 pull-down configuration register
D6H	P0OD	Port P0 open-drain output configuration register
D7H	P0CS	Port P0 input type configuration register
D9H	SYSDIV	Frequency division control register of high frequency clock (RC24M or XCLK)
DAH	P1PD	Port P1 pull-down configuration register
DBH	P1OD	Port P1 open-drain output configuration register
DCH	P1CS	Port P1 input type configuration register
DEH	PCLK0	Clock enable/disable register
DFH	PCLK1	Clock enable/disable register
E0H	ACC	Accumulator register
E1H~E3H	PxIRQ	Port interrupt flag register
E4H	P2PD	Port P2 pull-down configuration register
E5H	P1DR	Port P1 driving capability configuration register
E6H	PRESET0	Reset release register
E7H	PRESET1	Reset release register
P2AH	P2AH	P2_5~7 port rising-edge/falling-edge triggered interrupt enable register
E9H~EBH	PxIEN	Port interrupt enable register
ECH	P2OD	Port P2 open-drain output configuration register
EEH	RCHDIV	RCH frequency division setting register
F0H	B	B register
F1H~F3H	PxPUN	Port pull-up enable register
F4H	P2CS	Port P2 input type configuration register
F8H	CLKCON	System clock register

Address	Name	Description
F9H–FBH	PxOEN	Port direction control register
FCH	P2DR	Port P2 driving capability configuration register

4.1.1 P0

80H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0	-	-	-	P0.4	P0.3	P0.2	P0.1	P0.0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	1	1	1	1	1

Bit No.	Bit Designator	Description
7-5	-	-
4	P0.4	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read.
3	P0.3	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read.
2	P0.2	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read.
1	P0.1	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read.
0	P0.0	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read.

4.1.2 SP

81H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SP	SP							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	1	1	1

Bit No.	Bit Designator	Description
7-0	SP	The stack pointer SP is an 8-bit special register whose value can be updated via write operation. ● In the case of executing PUSH, various subroutine calls, interrupt response and other instructions, the SP value is added by 1 first and then the data is pushed. ● In the case of executing instructions such as POP, RET and RETI, the SP value is decreased by 1 after the stack pop. The top of the stack can be any address (00H–FFH) of the on-chip internal RAM. After system reset, SP is initialized to 07H, so that the stack actually starts from address 08H.

4.1.3 DPTR

82H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DPL	DPL							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	DPL	Lower 8 bits of the data pointer DPTR0

83H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DPH	DPH							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	DPH	Higher 8 bits of the data pointer DPTR0

84H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DPL1	DPL1							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	DPL1	Lower 8 bits of the data pointer DPTR1

85H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DPH1	DPH1							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	DPH1	Higher 8 bits of the data pointer DPTR1

4.1.4 PCON

87H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCON	-	-	-	-	-	-	STOP	IDLE
R/W	R	R	R	R	R	R	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-2	-	-
1	STOP	Writing 1 enters the Stop mode; the read operation always returns 0.
0	IDLE	Writing 1 enters the Idle mode; the read operation always returns 0.

4.1.5 PDSEL

8EH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PDSEL	-	-	-	-	-	-	-	PDSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-1	-	-
0	PDSEL	Power-down mode selection bit: ● 1: power-down mode enabled. When this bit is 1, writing 1 to the STOP bit of PCON will enter Power-down mode and all clock sources of the system will be turned off. ● 0: power-down mode disabled. When this bit is 0, writing 1 to the STOP bit of PCON will enter Stop mode with the RCL running and the XTH and RCH off.

4.1.6 POREN

8FH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
POREN	-							LVR_ENB
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	1	0

Bit No.	Bit Designator	Description
7-1	-	-
0	LVR_ENB	Power-down reset enable bit: 0: power-down reset enabled 1: power-down reset disabled

4.1.7 P1

90H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1	-	-	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	-	-	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-6	-	-
5	P1.5	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read

4	P1.4	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read
3	P1.3	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read
2	P1.2	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read
1	P1.1	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read
0	P1.0	● When the port is configured as output state: 0: output low level 1: output high level ● When the port is configured as input state: the port state is read

4.1.8 LDOTRIML

91H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LDOTRIML	-			VTRM_BGRVT				
R/W	R			R/W				
Reset value	0			5'h0F				

Bit No.	Bit Designator	Description
7-5	-	-
4-0	VTRM_BGRVT	Bandgap trimming bit

4.1.9 DPS

92H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DPS	-	-	-	-	-	-	-	DPS
R/W	R	R	R	R	R	R	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-1	-	-
0	DPS	1: select DPTR1 as data pointer 0: select DPTR0 as data pointer

4.1.10 P0DR

97H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0DR	-	-	-	P0_4DR	P0_3DR	P0_2DR	P0_1DR	P0_0DR
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-5	-	-
4	P0_4DR	Port P0_4 driving capability configuration register: 0: high driving capability 1: low driving capability
3	P0_3DR	Port P0_3 driving capability configuration register: 0: high driving capability 1: low driving capability
2	P0_2DR	Port P0_2 driving capability configuration register: 0: high driving capability 1: low driving capability
1	P0_1DR	Port P0_1 driving capability configuration register: 0: high driving capability 1: low driving capability
0	P0_0DR	Port P0_0 driving capability configuration register: 0: high driving capability 1: low driving capability

4.1.11 IEN2

9AH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IEN2	-	-	-	-	-	-	CANINTEN	UART3INTEN
R/W	R	-	R	R	R	R	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-2	-	-
1	CANINTEN	CAN interrupt enable: 1: enabled 0: disabled
0	UART3INTEN	UART3 interrupt enable: 1: enabled 0: disabled

4.1.12 RCLTRIML

BCH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RCLTRIML	RCLTCTRIM				-			RCLTRIML
R/W	R/W				R			R/W
Reset value	0x7				0			0x1

Bit No.	Bit Designator	Description
7-4	RCLTCTRIM	On-chip RCL temperature drift trimming value
3-1	-	-
0	RCLTRIML	On-chip RCL trimming value low-order bit

4.1.13 RCLTRIM

9FH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RCLTRIM	RCLTRIMH							
R/W	R/W							
Reset value	0x7F							

Bit No.	Bit Designator	Description
7-0	RCLTRIMH	On-chip RCL trimming value high-order bit

4.1.14 P2

A0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2	P2.7	P2.6	P2.5	-	P2.3	P2.2	-	P2.0
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W
Reset value	1	1	1	0	1	1	0	1

Bit No.	Bit Designator	Description
7	P2.7	**When the port is configured as output state:** 0: output low level 1: output high level **When the port is configured as input state:** the port state is read.
6	P2.6	**When the port is configured as output state:** 0: output low level 1: output high level **When the port is configured as input state:** the port state is read.
5	P2.5	**When the port is configured as output state:** 0: output low level 1: output high level **When the port is configured as input state:** the port state is read.
4	-	-
3	P2.3	**When the port is configured as output state:** 0: output low level 1: output high level **When the port is configured as input state:** the port state is read
2	P2.2	**When the port is configured as output state:** 0: output low level 1: output high level **When the port is configured as input state:** the port state is read.
1	-	-
0	P2.0	**When the port is configured as output state:**

		0: output low level 1: output high level **When the port is configured as input state:** the port state is read.

4.1.15 OUS

A1H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OUS	-	-	-	OUS				
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
Reset value	-	-	-	0	0	1	1	1

Bit No.	Bit Designator	Description
7-5	-	-
4-0	OUS	Flash erase time scale The default value of this register is 0xF, which is set according to NVR table in actual application.

4.1.16 P0AL

A4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0AL	P0AL.7	P0AL.6	P0AL.5	P0AL.4	P0AL.3	P0AL.2	P0AL.1	P0AL.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7	P0AL.7	Port P0_3 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
6	P0AL.6	Port P0_3 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
5	P0AL.5	Port P0_2 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
4	P0AL.4	Port P0_2 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt

		1: rising edge of the port enabled to trigger interrupt
3	P0AL.3	Port P0_1 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
2	P0AL.2	Port P0_1 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
1	P0AL.1	Port P0_0 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
0	P0AL.0	Port P0_0 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt

4.1.17 IEN0

A8H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IEN0	EA	EADC	EPWM	ES0	-	ES1	-	EX0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	EA	Master interrupt switch: 0: disabled 1: enabled
6	EADC	ADC interrupt enable: 0: disabled 1: enabled
5	EPWM	PWM interrupt enable: 0: disabled 1: enabled
4	ES0	UART0 interrupt enable: 0: disabled 1: enabled
3	-	-
2	ES1	UART1 interrupt enable: 0: disabled

		1: enabled
1	-	-
0	EX0	External interrupt primary enable: 0: disabled 1: enabled

4.1.18 IP

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IP0 (A9H)	-	-	IP0.5	IP0.4	IP0.3	IP0.2	IP0.1	IP0.0
IP1 (B9H)	-	-	IP1.5	IP1.4	IP1.3	IP1.2	IP1.1	IP1.0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Each interrupt source can be individually set to one of the four interrupt priorities, which are set by the corresponding bits of IP0 and IP1 respectively, as described in the following table:

IP1.x	IP0.x	Priority Level
0	0	Level 0 (lowest)
0	1	Level 1
1	0	Level 2
1	1	Level 3 (highest)

Bit	Corresponding Interrupt
IP1.0, IP0.0	External interrupt 0EX0, ADC interrupt, UART2 interrupt and CAN interrupt
IP1.1, IP0.1	SPI interrupt, UART3 interrupt
IP1.2, IP0.2	UART1 interrupt ES1
IP1.3, IP0.3	GTimer2 interrupt, EFC interrupt
IP1.4, IP0.4	UART0 interrupt ES0, GTimer1 interrupt and LPTimer interrupt
IP1.5, IP0.5	PWM interrupt EPWM, GTimer0 interrupt and I2C interrupt

4.1.19 P0AH

ABH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0AH	-	-	-	-	-	-	P0AH.1	P0AH.0
R/W	R	R	R	R	R	R	R/W	R/W

Reset value	0	0	0	0	0	0	1	1

Bit No.	Bit Designator	Description
7-2	-	-
1	P0AH.1	Port P0_4 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
0	P0AH.0	Port P0_4 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt

4.1.20 P1AL

AEH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1AL	P1AL.7	P1AL.6	P1AL.5	P1AL.4	P1AL.3	P1AL.2	P1AL.1	P1AL.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7	P1AL.07	Port P1_3 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
6	P1AL.06	Port P1_3 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
5	P1AL.05	Port P1_2 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
4	P1AL.04	Port P1_2 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
3	P1AL.3	Port P1_1 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
2	P1AL.2	Port P1_1 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt

1	P1AL.1	Port P1_0 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
0	P1AL.0	Port P1_0 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt

4.1.21 REMAP

AFH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
REMAP	-	-	-	-	-	REMAP_FLAG	REMAP_IM	REMAP
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	1	1

Bit No.	Bit Designator	Description
7-3	-	-
2	REMAP_FLAG	Flag of whether REMAP occurs in the system: 1: REMAP occurred in the system. 0: REMAP did not occur in the system.
1	REMAP_IM	Write 0, the address REMAP occurs directly.
0	REMAP	Write 0, the address REMAP occurs and a system reset is generated, after which the system boots from the main area of eFlash.

4.1.22 P1AH

B0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1AH	-	-	-	-	P1AH.3	P1AH.2	P1AH.1	P1AH.0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	1	1	1	1

Bit No.	Bit Designator	Description
7-4	-	-
3	P1AH.3	Port P1_5 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt

2	P1AH.2	Port P1_5 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
1	P1AH.1	Port P1_4 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
0	P1AH.0	Port P1_4 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt

4.1.23 CLKST

B1H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CLKST	WACKDELAY		XTHSTAB		RCHSTAB		RCLSTAB	
RW	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	1	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	WACKDELAY	Wake-up delay setting: 00: 64 system clock cycles 01: 128 system clock cycles 10: 161 system clock cycles 11: 184 system clock cycles
5-4	XTHSTAB	XTH stabilization time setting: 00: 1024 XTH clock cycles 01: 4096 XTH clock cycles 10: 16384 XTH clock cycles 11: 32768 XTH clock cycles
3-2	RCHSTAB	RCH stabilization time setting: 00: 1 RCH48M clock cycles 01: 4 RCH48M clock cycles 10: 32 RCH48M clock cycles 11: 256 RCH48M clock cycles

1-0	RCLSTAB	RCL stabilization time setting: 00: 1 RCL38K clock cycles 01: 4 RCL38K clock cycles 10: 32 RCL38K clock cycles 11: 256 RCL38K clock cycles

4.1.24 ESTCR

B2H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ESTCR	-	-	-	-	ERSTEN	ERSTLVT		ERSTLVEN
R/W	R	R	R	R	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	1	0	0	0

Bit No.	Bit Designator	Description
7-4	-	-
3	ERSTEN	Enable bit of external reset pin: 1: enabled 0: disabled
2-1	ERSTLVT	External reset filter time setting: 11: 2 RCL38K clock cycles 10: 15 RCL38K clock cycles 01: 23 RCL38K clock cycles 00: 30 RCL38K clock cycles
0	ERSTLVEN	1: external reset filter enabled 0: external reset filter disabled

4.1.25 XTHCTR

B3H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
XTHCTR	-	-	-	-	EXTH_GSEL			EXTH_EN
R/W	R	R	R	R	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	1	0	0

Bit No.	Bit Designator	Description
7-4	-	-
3-1	EXTH_GSEL	XTH trimming position
0	EXTH_EN	External XTH input control:

		1: XTH is input from P0_0. 0: XTH is generated by crystal oscillator. Note: XCLKEN shall be set to 1 in the case of P0_0 being adopted for clock input.

Crystal Frequency	GSEL[2:0>	RSEL[1:0]
F ≤ 1 MHz	000	00
1 MHz < F ≤ 6 MHz	001	01
6 MHz < F ≤ 12 MHz	010	10
12 MHz < F ≤ 16 MHz	011	10
16 MHz < F ≤ 20 MHz	101	11
20 MHz < F ≤ 24 MHz	110	11

4.1.26 ADCDR0

B6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCDR0	CHDATAL							
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	CHDATAL	A/D channel RX data low-order register

4.1.27 ADCDR1

B7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCDR1	-	-	-	-	CHDATAH			
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	-	-
6-4	-	-
3-0	CHDATAH	A/D channel RX data high-order register

4.1.28 IEN1

B8H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IEN1	UART2	GTIMER2INTEN	I2CINTEN	LPTIMERINTEN	EFCINTEN	GTIMER1INTEN	SPIINTEN	GTIMER0INTEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	UART2INTEN	UART2 interrupt enable: 1: enabled 0: disabled
6	GTIMER2INTEN	GTimer2 interrupt enable: 1: enabled 0: disabled
5	I2CINTEN	I2C interrupt enable: 1: enabled 0: disabled
4	LPTIMERINTEN	LPTimer interrupt enable: 1: enabled 0: disabled
3	EFCINTEN	EFC interrupt enable: 1: enabled 0: disabled
2	GTIMER1INTEN	GTimer1 interrupt enable: 1: enabled 0: disabled
1	SPIINTEN	SPI interrupt enable: 1: enabled 0: disabled
0	GTIMER0INTEN	GTimer0 interrupt enable: 1: enabled 0: disabled

4.1.29 LDOTRIMH

BDH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LDOTRIMH	-			VTRM_BGRTT				
R/W	R			R/W				
Reset value	0			5'h0				

Bit No.	Bit Designator	Description
7-5	-	-
4-0	VTRM_BGRTT	Bandgap trimming bit

4.1.30 RCHTRIMH

BEH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RCHTRIMH	RCHTRIMH							
R/W	R/W							
Reset value	0x37							

Bit No.	Bit Designator	Description
7-0	RCHTRIMH	On-chip RCH trimming value high-order bit

4.1.31 RCHTRIML

BFH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RCHTRIML	RCHTCTRIM				-	-	RCHTRIML	-
R/W	R/W				R	R	R/W	R
Reset value	0x9				0	0	0	0

Bit No.	Bit Designator	Description
7-4	RCHTCTRIM	On-chip RCH temperature drift trimming value
3-2	-	-
1	RCHTRIML	On-chip RCH trimming value low-order bit
0	-	-

4.1.32 P2AL

C0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2AL	P2AL.7	P2AL.6	P2AL.5	P2AL.4	-	-	P2AL.1	P2AL.0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W
Reset value	1	1	1	1	0	0	1	1

Bit No.	Bit Designator	Description
7	P2AL.7	Port P2_3 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
6	P2AL.6	Port P2_3 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
5	P2AL.5	Port P2_2 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
4	P2AL.4	Port P2_2 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
3-2	-	-
1	P2AL.1	Port P2_0 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
0	P2AL.0	Port P2_0 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt

4.1.33 PSW

D0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PSW	CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	CY	Carry flag: When the last arithmetic operation produces a carry (addition) or a borrow (subtraction), this bit is set to 1, and it will be cleared by other arithmetic operations.
6	AC	Auxiliary carry flag: When the last arithmetic operation produces a carry (addition) or a borrow (subtraction) to the higher half byte, this bit is set to 1, and it will be cleared by other arithmetic operations.
5	F0	User flag 0: This is a bit-addressable general flag bit for software control.
4-3	RS[1:0]	RS1-RS0: register area selection: 00: page 0 (mapped to 00H–07H) 01: page 1 (mapped to 08H-0FH) 10: page 2 (mapped to 10H-17H) 11: page 3 (mapped to 18H–1FH)
1	F1	User flag 1: This is a bit-addressable general flag bit for software control.
0	P	Parity bit: 0: The sum of the 8 bits in the accumulator is even. 1: The sum of the 8 bits in the accumulator is odd.

4.1.34 P0PD

D5H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0PD	-	-	-	P0_4PD	P0_3PD	P0_2PD	P0_1PD	P0_0PD
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-5	-	-
4	P0_4PD	Port P0_4 pull-down configuration register: 0: disabled 1: enabled
3	P0_3PD	Port P0_3 pull-down configuration register: 0: disabled 1: enabled

2	P0_2PD	Port P0_2 pull-down configuration register: 0: disabled 1: enabled
1	P0_1PD	Port P0_1 pull-down configuration register: 0: disabled 1: enabled
0	P0_0PD	Port P0_0 pull-down configuration register: 0: disabled 1: enabled

4.1.35 P0OD

D6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0OD	-	-	-	P0_4OD	P0_3OD	P0_2OD	P0_1OD	P0_0OD
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-5	-	-
4	P0_4OD	Port P0_4 open-drain output configuration register: 0: disabled 1: enabled
3	P0_3OD	Port P0_3 open-drain output configuration register: 0: disabled 1: enabled
2	P0_2OD	Port P0_2 open-drain output configuration register: 0: disabled 1: enabled
1	P0_1OD	Port P0_1 open-drain output configuration register: 0: disabled 1: enabled
0	P0_0OD	Port P0_0 open-drain output configuration register: 0: disabled 1: enabled

4.1.36 P0CS

D7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0CS	-	-	-	P0_4CS	P0_3CS	P0_2CS	P0_1CS	P0_0CS
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-5	-	-
4	P0_4CS	Port P0_4 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
3	P0_3CS	Port P0_3 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
2	P0_2CS	Port P0_2 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
1	P0_1CS	Port P0_1 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
0	P0_0CS	Port P0_0 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer

4.1.37 SYSDIV

D9H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SYSDIV	-	-	-	-	-	SYSDIV[2]	SYSDIV[1]	SYSDIV[0]
R/W	R	R	R	R	R	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-3	-	-

2-0	SYSDIV[1:0]	Frequency division control of high frequency clock (RC24M or XCLK), which is then output as system clock: 000: HSCLK frequency not divided 001: HSCLK frequency divided by 2 010: HSCLK frequency divided by 4 011: HSCLK frequency divided by 8 100: HSCLK frequency divided by 16 101: HSCLK frequency divided by 32 110: HSCLK frequency divided by 64 111: HSCLK frequency divided by 128

4.1.38 P1PD

DAH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1PD	-	-	P1_5PD	P1_4PD	P1_3PD	P1_2PD	P1_1PD	P1_0PD
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	-	-
5	P1_5PD	Port P1_5 pull-down configuration register: 0: disabled 1: enabled
4	P1_4PD	Port P1_4 pull-down configuration register: 0: disabled 1: enabled
3	P1_3PD	Port P1_3 pull-down configuration register: 0: disabled 1: enabled
2	P1_2PD	Port P1_2 pull-down configuration register: 0: disabled 1: enabled
1	P1_1PD	Port P1_1 pull-down configuration register: 0: disabled 1: enabled
0	P1_0PD	Port P1_0 pull-down configuration register: 0: disabled 1: enabled

4.1.39 P1OD

DBH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1OD	-	-	P1_5OD	P1_4OD	P1_3OD	P1_2OD	P1_1OD	P1_0OD
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	-	-
5	P1_5OD	Port P1_5 open-drain output configuration register: 0: disabled 1: enabled
4	P1_4OD	Port P1_4 open-drain output configuration register: 0: disabled 1: enabled
3	P1_3OD	Port P1_3 open-drain output configuration register: 0: disabled 1: enabled
2	P1_2OD	Port P1_2 open-drain output configuration register: 0: disabled 1: enabled
1	P1_1OD	Port P1_1 open-drain output configuration register: 0: disabled 1: enabled
0	P1_0OD	Port P1_O open-drain output configuration register: 0: disabled 1: enabled

4.1.40 P1CS

DCH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1CS	-	-	P1_5CS	P1_4CS	P1_3CS	P1_2CS	P1_1CS	P1_0CS
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	-	-
5	P1_5CS	Port P1_5 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
4	P1_4CS	Port P1_4 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
3	P1_3CS	Port P1_3 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
2	P1_2CS	Port P1_2 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
1	P1_1CS	Port P1_1 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
0	P1_0CS	Port P1_0 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer

4.1.41 PCLK0

DEH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCLK0	I2CCEN	LPTIMCEN	PWMCEN	ADCCEN	SPICEN	WDTCEN	UART1CEN	UART0CEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7	I2CCEN	1: I2C clock enabled 0: I2C clock disabled
6	LPTIMCEN	1: LPTimer clock enabled 0: LPTimer clock disabled
5	PWMCEN	1: PWM clock enabled 0: PWM clock disabled

4	ADCCEN	1: ADC clock enabled 0: ADC clock disabled
3	SPICEN	1: SPI clock enabled 0: SPI clock disabled
2	WDTCEN	1: WDT clock enabled 0: WDT clock disabled
1	UART1CEN	1: UART1 clock enabled 0: UART1 clock disabled
0	UART0CEN	1: UART0 clock enabled 0: UART0 clock disabled

4.1.42 PCLK1

DFH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCLK1	UART2CEN	GTIMER2CEN	GTIMER1CEN	-	GTIMER0CEN	GIO2CEN	GIO1CEN	GIO0CEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7	UART2CEN	1: UART2 clock enabled 0: UART2 clock disabled
6	GTIMER2CEN	1: GTIMER2 clock enabled 0: GTIMER2 clock disabled
5	GTIMER1CEN	1: GTIMER1 clock enabled 0: GTIMER1 clock disabled
4	-	-
3	GTIMER0CEN	1: GTIMER0 clock enabled 0: GTIMER0 clock disabled
2	GIO2CEN	1: GPIO2 clock enabled 0: GPIO2 clock disabled
1	GIO1CEN	1: GPIO1 clock enabled 0: GPIO1 clock disabled
0	GIO0CEN	1: GPIO0 clock enabled 0: GPIO0 clock disabled

4.1.43 ACC

E0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ACC	ACC							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	ACC	Accumulator is a common special register used to store the operands and the operation results.

4.1.44 PxIRQ

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0IRQ (E1H)	-	-	-	P0IRQ.4	P0IRQ.3	P0IRQ.2	P0IRQ.1	P0IRQ.0
P1IRQ (E2H)	-	-	P1IRQ.5	P1IRQ.4	P1IRQ.3	P1IRQ.2	P1IRQ.1	P1IRQ.0
P2IRQ (E3H)	P2IRQ.7	P2IRQ.6	P2IRQ.5	-	P2IRQ.3	P2IRQ.2	-	P2IRQ.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	PxIRQ.y x = 0-4, y = 0-7	Port interrupt flag bit: 0: no interrupt occurred on the port 1: interrupt occurred on the port Write 0 to clear this bit.

4.1.45 P2PD

E4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2PD	P2_7PD	P2_6PD	P2_5PD	-	P2_3PD	P2_2PD	-	P2_0PD
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	P2_7PD	Port P2_7 pull-down configuration register: 0: disabled

		1: enabled
6	P2_6PD	Port P2_6 pull-down configuration register: 0: disabled 1: enabled
5	P2_5PD	Port P2_5 pull-down configuration register: 0: disabled 1: enabled
4	-	-
3	P2_3PD	Port P2_3 pull-down configuration register: 0: disabled 1: enabled
2	P2_2PD	Port P2_2 pull-down configuration register: 0: disabled 1: enabled
1	-	-
0	P2_0PD	Port P1_0 pull-down configuration register: 0: disabled 1: enabled

4.1.46 P1DR

E5H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1DR	-	-	P1_5DR	P1_4DR	P1_3DR	P1_2DR	P1_1DR	P1_0DR
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-6	-	-
5	P1_5DR	Port P1_5 driving capability configuration register: 0: high driving capability 1: low driving capability
4	P1_4DR	Port P1_4 driving capability configuration register: 0: high driving capability 1: low driving capability
3	P1_3DR	Port P1_3 driving capability configuration register: 0: high driving capability 1: low driving capability

2	P1_2DR	Port P1_2 driving capability configuration register: 0: high driving capability 1: low driving capability
1	P1_1DR	Port P1_1 driving capability configuration register: 0: high driving capability 1: low driving capability
0	P1_0DR	Port P1_0 driving capability configuration register: 0: high driving capability 1: low driving capability

4.1.47 PRESET0

E6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PRESET0	I2CREN	LPTIMREN	PWMREN	ADCREN	SPIREN	WDTREN	UART1REN	UART0REN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7	I2CREN	1: I2C reset released 0: I2C reset
6	LPTIMREN	1: LPTimer reset released 0: LPTimer reset
5	PWMREN	1: PWM reset released 0: PWM reset
4	ADCREN	1: ADC reset released 0: ADC reset
3	SPIREN	1: SPI reset released 0: SPI reset
2	WDTREN	1: WDT reset released 0: WDT reset
1	UART1REN	1: UART1 reset released 0: UART1 reset
0	UART0REN	1: UART0 reset released 0: UART0 reset

4.1.48 PRESET1

E7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PRESET1	UART2REN	GTIMER2REN	GTIMER1REN	-	GTIMER0REN	GIO2REN	GIO1REN	GIO0REN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7	UART2REN	1: UART2 reset released 0: UART2 reset
6	GTIMER2REN	1: GTIMER2 reset released 0: GTIMER2 reset
5	GTIMER1REN	1: GTIMER1 reset released 0: GTIMER1 reset
4	-	-
3	GTIMER0REN	1: GTIMER0 reset released 0: GTIMER0 reset
2	GIO2REN	1: GPIO2 reset released 0: GPIO2 reset
1	GIO1REN	1: GPIO1 reset released 0: GPIO1 reset
0	GIO0REN	1: GPIO0 reset released 0: GPIO0 reset

4.1.49 P2AH

E8H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2AH	P2AH.7	P2AH.6	P2AH.5	P2AH.4	P2AH.1	P2AH.0	-	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
Reset value	1	1	1	1	1	1	0	0

Bit No.	Bit Designator	Description
7	P2AH.7	Port P2_7 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt

		1: falling edge of the port enabled to trigger interrupt
6	P2AH.6	Port P2_7 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
5	P2AH.5	Port P2_6 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
4	P2AH.4	Port P2_6 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
3	P2AH.1	Port P2_5 falling-edge triggered interrupt enable bit: 0: falling edge of the port disabled to trigger interrupt 1: falling edge of the port enabled to trigger interrupt
2	P2AH.0	Port P2_5 rising-edge triggered interrupt enable bit: 0: rising edge of the port disabled to trigger interrupt 1: rising edge of the port enabled to trigger interrupt
1-0	-	-

4.1.50 PxIEN

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0IEN (E9H)	-	-	-	P0IEN.4	P0IEN.3	P0IEN.2	P0IEN.1	P0IEN.0
P1IEN (EAH)	-	-	P1IEN.5	P1IEN.4	P1IEN.3	P1IEN.2	P1IEN.1	P1IEN.0
P2IEN (EBH)	P2IEN.7	P2IEN.6	P2IEN.5	-	P2IEN.3	P2IEN.2	-	P2IEN.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	PxIEN.y x = 0-4, y = 0-7	Port interrupt enable bit: 0: disabled 1: enabled

4.1.51 P2OD

ECH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2OD	P2_7OD	P2_6OD	P2_5OD	-	P2_3OD	P2_2OD	-	P2_0OD
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W

Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	P2_7OD	Port P2_7 open-drain output configuration register: 0: disabled 1: enabled
6	P2_6OD	Port P2_6 open-drain output configuration register: 0: disabled 1: enabled
5	P2_5OD	Port P2_5 open-drain output configuration register: 0: disabled 1: enabled
4	-	-
3	P2_3OD	Port P2_3 open-drain output configuration register: 0: disabled 1: enabled
2	P2_2OD	Port P2_2 open-drain output configuration register: 0: disabled 1: enabled
1	-	-
0	P2_0OD	Port P2_0 open-drain output configuration register: 0: disabled 1: enabled

4.1.52 RCHDIV

EEH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RCHDIV	-	-	-	-	-	-	RCHDIV	
R/W	R	R	R	R	R	R	R/W	R/W
Reset value	0	0	0	0	0	0	1	0

Bit No.	Bit Designator	Description
7-2	-	-
1-0	RCHDIV	RCH frequency division setting: 00: 16 MHz 01: 24 MHz 10: 16 MHz (default) 11: 12 MHz

4.1.53 B

F0H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
B	B							
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	B	B register is used for the multiply-divide instruction, or used as a general temporary register in other instructions.

4.1.54 PxPUN

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0PUN (F1H)	-	-	-	P0PUN.4	P0PUN.3	P0PUN.2	P0PUN.1	P0PUN.0
P1PUN (F2H)	-	-	P1PUN.5	P1PUN.4	P1PUN.3	P1PUN.2	P1PUN.1	P1PUN.0
P2PUN (F3H)	P2PUN.7	P2PUN.6	P2PUN.5	-	P2PUN.3	P2PUN.2	-	P2PUN.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-0	PxPUN.y x = 0-4, y = 0-7	Port pull-up enable (reference value of pull-up resistor is 60 kΩ) : 0: internal pull-up resistor enabled 1: internal pull-up resistor disabled

4.1.55 P2CS

F4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2CS	P2_7CS	P2_6CS	P2_5CS	-	P2_3CS	P2_2CS	-	P2_0CS
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	P2_7CS	Port P2_7 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer

6	P2_6CS	Port P2_6 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
5	P2_5CS	Port P2_5 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
4	-	-
3	P2_3CS	Port P2_3 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
2	P2_2CS	Port P2_2 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer
1	-	-
0	P2_0CS	Port P2_0 input type configuration register: 0: Schmitt input buffer 1: CMOS input buffer

4.1.56 CLKCON

F8H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CLKCON	SYSCLKSEL	RC38KF	RC24MF	XCLKF	RC38KEN	RC24MEN	XCLKEN	HSCLKSEL
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
Reset value	0	1	1	0	1	1	0	0

Bit No.	Bit Designator	Description
7	SYSCLKSEL	System clock source selection bit, i.e. STSCLK from HSCLK or RCL38K: 0: HSCLK selected as system clock source 1: RCL38K selected as system clock source
6	RC38KF	RC38K clock source flag bit: 0: RC38K did not start normally 1: RC38K has started normally
5	RC24MF	RC24M clock source flag bit: 0: RC24M did not start normally 1: RC24M has started normally

4	XCLKF	XCLK clock source status flag bit: 0: no valid external input clock detected 1: valid external input clock detected
3	RC38KEN	RC38K clock source enable bit: 0: disabled 1: enabled
2	RC24MEN	RC24M clock source enable bit: 0: internal RC24M clock source disabled; writing 0 is invalid if the current system clock is RC24M. 1: internal RC24M clock source enabled.
1	XCLKEN	XCLK clock source enable bit: 0: XCLK disabled, P0.0 and P0.1 are GPIO functions; if XCLK is the current system clock, writing 0 to XCLKEN cannot disable XCLK. 1: XCLK enabled, P0.0 and P0.1 serving as crystal or external clock input.
0	HSCLKSEL	High-frequency clock source selection bit, i.e. HSCLK from RCH24M or XCLK: 0: RC24M selected as system clock source 1: XCLK selected as system clock source

4.1.57 PxOEN

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0OEN (F9H)	-	-	-	P0OEN.4	P0OEN.3	P0OEN.2	P0OEN.1	P0OEN.0
P1OEN (FAH)	-	-	P1OEN.5	P1OEN.4	P1OEN.3	P1OEN.2	P1OEN.1	P1OEN.0
P2OEN (FBH)	P2OEN.7	P2OEN.6	P2OEN.5	-	P2OEN.3	P2OEN.2	-	P2OEN.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-0	PxOEN.y x = 0-4, y = 0-7	Port direction control bit: 0: output mode 1: input mode

4.1.58 P2DR

FCH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2DR	P2_7DR	P2_6DR	P2_5DR	-	P2_3DR	P2_2DR	-	P2_0DR
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W
Reset value	1	1	1	0	1	1	0	1

Bit No.	Bit Designator	Description
7	P2_7DR	Port P2_7 driving capability configuration register: 0: high driving capability 1: low driving capability
6	P2_6DR	Port P2_6 driving capability configuration register: 0: high driving capability 1: low driving capability
5	P2_5DR	Port P2_5 driving capability configuration register: 0: high driving capability 1: low driving capability
4	-	-
3	P2_3DR	Port P2_3 driving capability configuration register: 0: high driving capability 1: low driving capability
2	P2_2DR	Port P2_2 driving capability configuration register: 0: high driving capability 1: low driving capability
1	-	-
0	P2_0DR	Port P2_0 driving capability configuration register: 0: high driving capability 1: low driving capability

4.2 System Clock

4.2.1 Main Features

- Internal 24-MHz RCH
- Internal 38-kHz RCL
- 2~24 MHz XCLK used as system clock source
- Internal system clock divider

4.2.2 Clock Definition

- RCH48M: the internal 48-MHz high-frequency oscillator
- RCL38K: the internal 38-kHz low-frequency oscillator
- XCLK: 2-24 MHz crystal resonator or external clock input

4.2.3 Clock Architecture Diagram

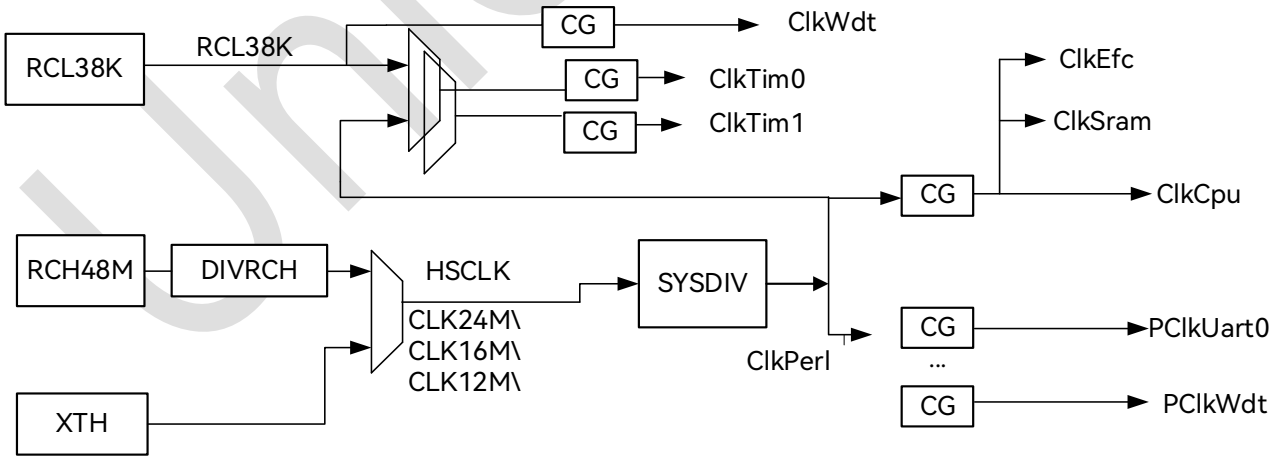


Figure 4-1: Clock Architecture Diagram

4.3 Reset Source

4.3.1 Main Features

Reset sources include:

- Pin reset
- LVD reset
- LVR
- WDT reset
- Power-on reset (POR)

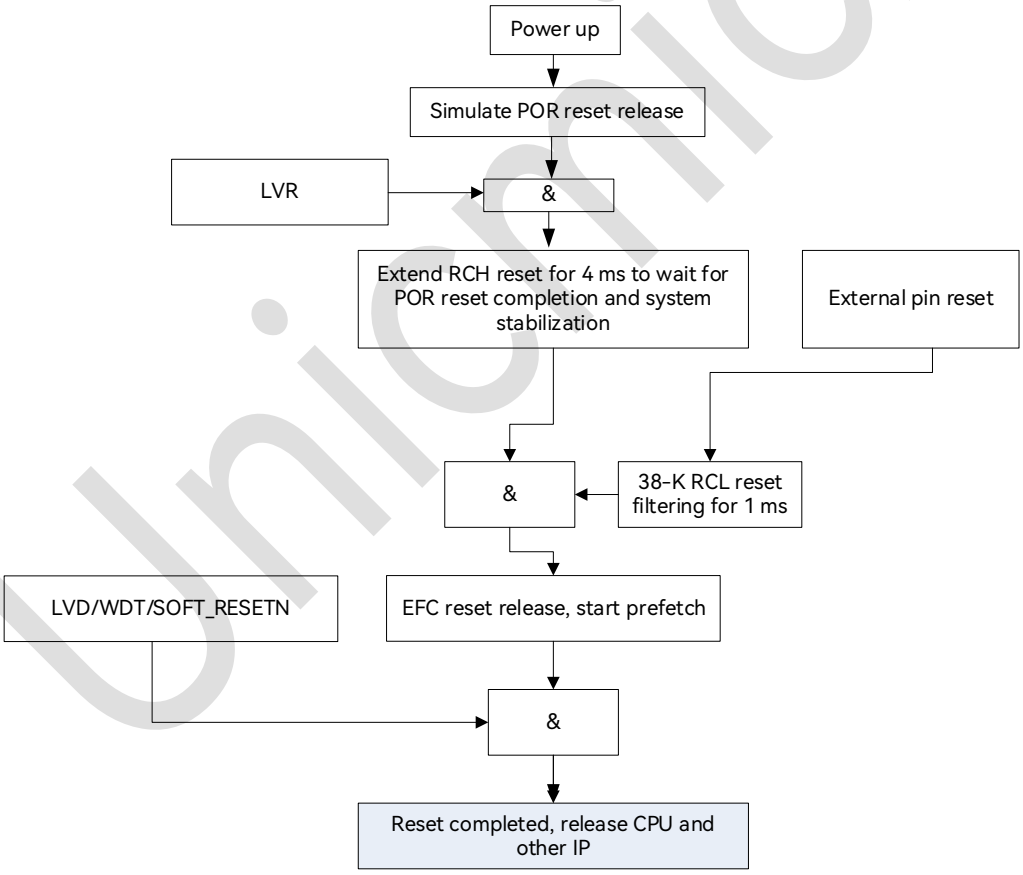


Figure 4-2: Reset Flow Chart

Note:

- In LVD reset, the LVDCON register will not be reset.
- Both LVR and POR are global resets that reset all digital logics.

4.3.2 WDT reset

WDT is an up-counter with RC38K as the clock source. To enable WDT, RC38K must be ensured to be activated first. In power-down mode, WDT will reset the chip when the timer overflows.

Reading or writing the RSTSTAT register automatically clears the WDT count.

4.3.3 LVD and LVR

LVD is a low-voltage detection unit that can be configured by software to generate a reset when the external supply voltage falls below the detection threshold. LVD can reset CPU and other peripherals except EFC.

LVR is a power-down reset unit with the same scope as POR, which is a global reset. LVR will reset the whole chip when the external supply voltage is less than the fixed voltage of LVR (2.2 V).

4.3.4 External Reset

External reset is a global reset with the same scope as POR. When RESETN IO is 0 and RESETN is used for external reset, the external reset is effective and the whole chip is reset.

4.3.5 Register

Refer to Chapters POREN, LVD_RSTSTAT and LVD_CON for reset related registers.

4.4 Low-power Mode

4.4.1 Main Features

- Three low-power modes: Sleep mode, DeepSleep mode and Stop mode
- Interrupt and reset allow to exit from these three low-power modes.

4.4.2 Low-power Mode

In addition to the normal operating mode, there are three low-power modes available for reducing current consumption of the chip: Sleep mode, Deepsleep mode and Stop mode.

In Sleep mode, CPU stops working and retains the interrupt handling function. The module clock and reset of other peripherals can be set by software. The Sleep mode is accessed by software writing 1 to the specific SFR (PCON -> IDLE), and the wake-up is triggered by interrupt.

Deepsleep mode is an upgrade of Sleep mode, in which CPU and the high-speed clock stop running while the low-power modules (LPTimer and WDT) remain running. The DeepSleep mode is accessed by software writing 1 to the specific SFR (PCON -> STOP), and the wake-up is triggered by interrupt.

In Stop mode, both the high-speed clock and the low-speed clock stop running, the system has no running clock, and all peripheral modules stop running. The POR signal is valid; IO state is retained; IO interrupt is valid; all registers, RAM and CPU data are saved with low power consumption. The Stop mode is accessed by setting the PDSEL register in SFR to 1 first and then writing 1 to PCON -> STOP. The wake-up can be triggered only by the external GPIO level.

4.4.3 Low-power Modes Summary

Mode	Mode Description	Entry Condition	Exit Condition
Sleep	Most of CPU sleep; the software can turn off each module clock.	1. Turn off the clock of each peripheral module as required, leaving only the module needed for monitoring the interrupt event. 2. Write 1 to PCON -> IDLE.	1. CPU detects an interrupt or event. 2. Enter the interrupt service program to clear the interrupt and return. 3. Continue to execute subsequent instructions.
Deepsleep	Most of CPU sleep; the high-speed clock source is off and the low-speed clock source is on.	1. Turn off the clock of each peripheral module as required, leaving only the module needed for monitoring the interrupt event. 2. Write 1 to PCON -> STOP.	1. CPU detects an interrupt or event. 2. Enter the interrupt service program to clear the interrupt and return. 3. Continue to execute subsequent instructions.
Stop	All system clocks are off.	1. Set the conditions for IO wake-up as required. 2. Set the PDSEL register in SFR to 1. 3. Write 1 to PCON -> STOP.	1. External IO wake-up occurs. 2. CPU detects the interrupt triggered by the external IO wake-up event. 3. Enter the interrupt service program to clear the interrupt and return. 4. Continue to execute subsequent instructions.

5 EFC

5.1 Main Features

- Support EFLASH read/write (8 bits), sector erase, etc.
- Read waiting time is configurable.
- 128 sectors in the main area, each with 512 bytes
- Erase/write protection
- Automatic bus locking
- Sector erase time: up to 5 ms; chip erase time: up to 40 ms; word write time: up to 20 μs; read time: up to 25 ns.

5.2 Eflash Read Efficiency

When the RD_WAIT value is set to 0, there is no efficiency loss in CPU fetching, and the timing of reading Eflash is the same as that of reading ROM on the controller side. When RD_WAIT value is set to 1, the EFC bus will be pulled down one cycle per read operation.

5.3 Argument Address

SN number acquisition address: 0x9248; size: 16 bytes

VCAP acquisition address: 0x919C; size: 4 bytes

5.4 Register Description

Table 5-1: List of Registers

Address	Name	Description
A7H	EFC_OPSET	Setting register
A3H	OINTUS	Interrupt status register
D1H–D2H	EFC_OADRL/H	EFLASH programming address register
D3H	EFC_ODATA	EFLASH programming data register
D4H	EFC_OCTRL	Voltage output register
A2H	OINTEN	Interrupt enable register

5.4.1 EFC_OPSET Setting Register

A7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EFC_OPSET	EEPROM_SET	NVR_SET	RDWAIT			CHIPSERSET	PAGESERSET	PAGEWRSET
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	1

Bit No.	Bit Designator	Description
7	EEPROM_SET	Erase enable bit for the latter 4-K memory of Flash main area in Boot mode
6	NVR_SET	Erase enable bit of EEPROM area
5-3	RDWAIT	Setting bit for read waiting time
2	CHIPSERSET	1: CHIP erase mode enabled 0: CHIP erase mode disabled
1	PAGESERSET	1: PAGE erase mode enabled 0: PAGE erase mode disabled
0	PAGEWRSET	1: PAGE write mode enabled 0: PAGE write mode disabled

5.4.2 OINTUS Interrupt Status Register

A3H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OINTUS	-	NVRCERR	VDD_LOW	WPOGERR	BOOTERR	NVR1ERR	NVR0ERR	PRODONE
R/W	R	R	R	R	R	R	R	R

Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	-	-
6	NVRCERR	1: NVRC error interrupt status bit 0: NVRC error interrupt status bit Write 1 to clear this bit.
5	VDD_LOW	1: LVD interrupt status bit 0: LVD interrupt status bit Write 1 to clear this bit.
4	WPOGERR	1: operation error interrupt status bit 0: operation error interrupt status bit Write 1 to clear this bit.
3	BOOTERR	1: BOOT error interrupt status bit 0: BOOT error interrupt status bit Write 1 to clear this bit.
2	NVR1ERR	1: NVR1 error interrupt status bit 0: NVR1 error interrupt status bit Write 1 to clear this bit.
1	NVR0ERR	1: NVR0 error interrupt status bit 0: NVR0 error interrupt status bit Write 1 to clear this bit.
0	PRODONE	1: erase done interrupt status bit 0: erase done interrupt status bit Write 1 to clear this bit.

5.4.3 EFC_OADRL/H EFLASH Programming Address Register

D1H-D2H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EFC_OADRL	Low-order bit [7:0] of EFLASH programming address							
EFC_OADRH	High-order bit [15:8] of EFLASH programming address							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

5.4.4 EFC_ODATA EFLASH Programming Data Register

D3H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EFC_ODATA	EFLASH programming data [7:0]							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

5.4.5 EFC_OCTRL Voltage Output Register

D4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
EFC_OCTRL	VPPO_EN	PUMP_EN	PUMP_SEL<2:0>			-	PUMP_OK	PUMP_6O5V
R/W	R/W	R/W	R/W			-	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	VPPO_EN	EFLASH IAP start control signal: 0: off (cleared by hardware automatically) 1: start EFLASH programming (software writes 1 to start EFLASH IAP and hardware clears this bit upon completion of IAP.) After this bit is set, CPU will be in Idle state and will resume upon completion of IAP.
6	PUMP_EN	PUMP module enable bit: 0: internal PUMP function disabled 1: internal PUMP function enabled
5-3	PUMP_SEL<2:0>	Pump output voltage selection control bit: PUMP_SEL / VPP (V) / PUMP_SEL / VPP (V) 000 / 6.5 / 100 / 7.5 001 / 6.75 / 101 / 7.75 010 / 7.0 / 110 / 8 011 / 7.25 / 111 / 8.25 It is suggested that the actual IAP set as PUMP_SEL<2:0> = 001, 6.75 V.
2	-	-
1	PUMP_OK	VPP output voltage flag 1: 0: VPP output voltage lower than the set voltage of PUMP_SEL 1: VPP output voltage higher than the set voltage of PUMP_SEL

0	PUMP_6O5V	VPP output voltage flag 2: 1: VPP output voltage higher than 6.5 V 0: VPP output voltage lower than 6.5 V

5.4.6 OINTEN Interrupt Enable Register

A2H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OINTEN	-	NVRCERREN	VDDLOWEN	WPOGERREN	BOOTERREN	NVR1ERREN	NVR0ERREN	PRODONEEN
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	-	-
6	NVRCERREN	1: NVRC error interrupt enabled 0: NVRC error interrupt disabled
5	VDDLOWEN	1: LVD interrupt enabled 0: LVD interrupt disabled
4	WPOGERREN	1: operation error interrupt enabled 0: operation error interrupt disabled
3	BOOTERREN	1: BOOT error interrupt enabled 0: BOOT error interrupt disabled
2	NVR1ERREN	1: NVR1 error interrupt enabled 0: NVR1 error interrupt disabled
1	NVR0ERREN	1: NVR0 error interrupt enabled 0: NVR0 error interrupt disabled
0	PRODONEEN	1: write/erase done interrupt enabled 0: write/erase done interrupt disabled

5.5 Software Process

5.5.1 Read Operation

EFLASH can perform read operation as it is stablized after power-on. Pay attention to configure the read waiting time RD_WAIT for read operation.

5.5.2 Write Operation

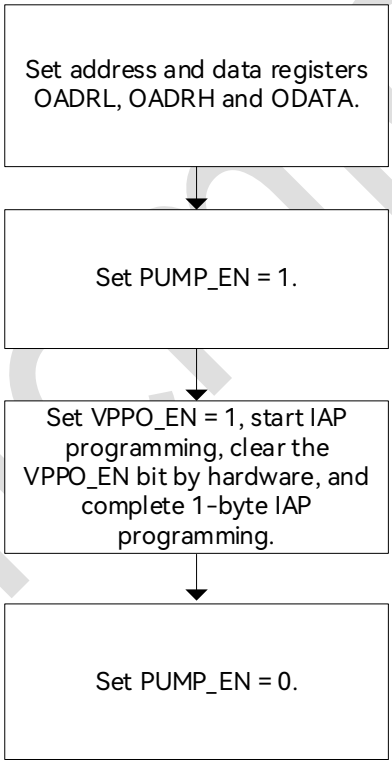


Figure 5-1: Write Operation Process

5.5.3 Erase Operation

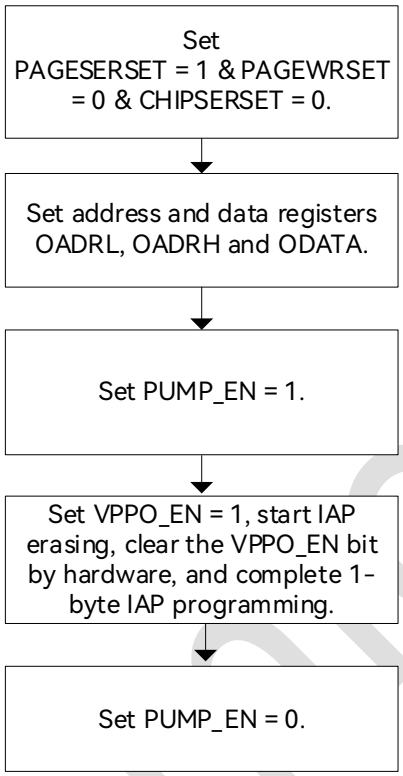


Figure 5-2: Erase Operation Process

5.5.4 Chip Erase Operation

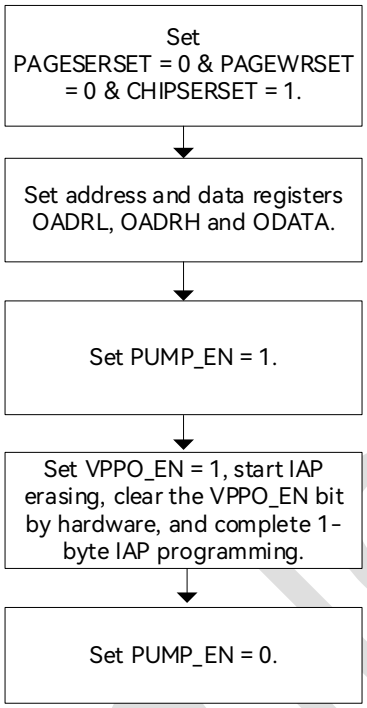


Figure 5-3: Erase Operation Process

6 Pulse Width Modulation (PWM)

6.1 Main Features

- 11 x 16-bit PWM modules
- Provide overflow interrupt for each PWM cycle
- Output polarity selection

6.2 Functional Description

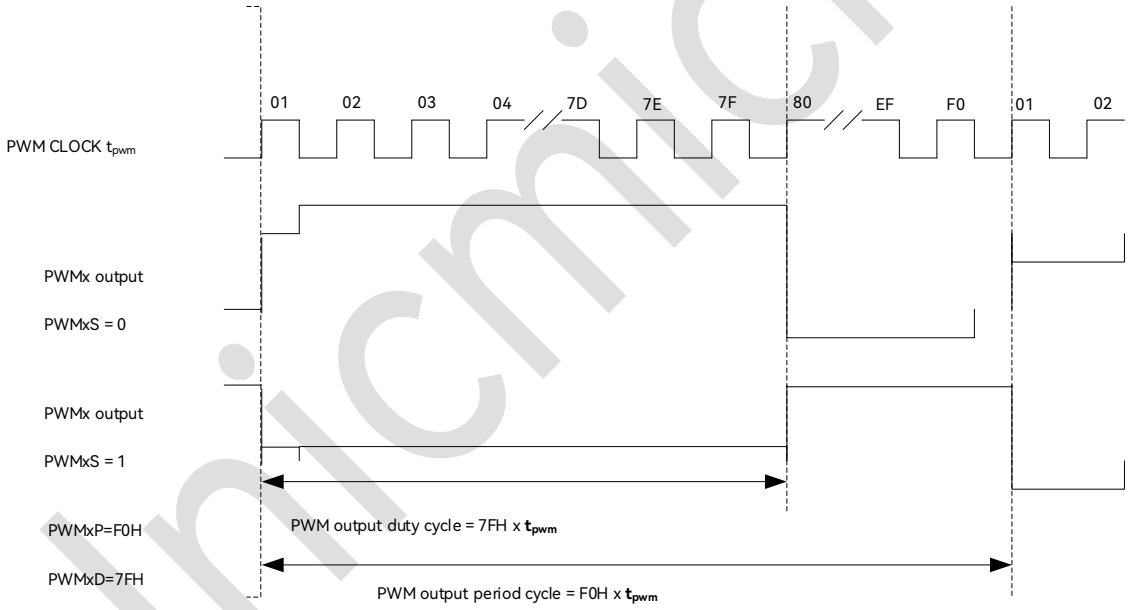


Figure 6-1: PWM Output Example

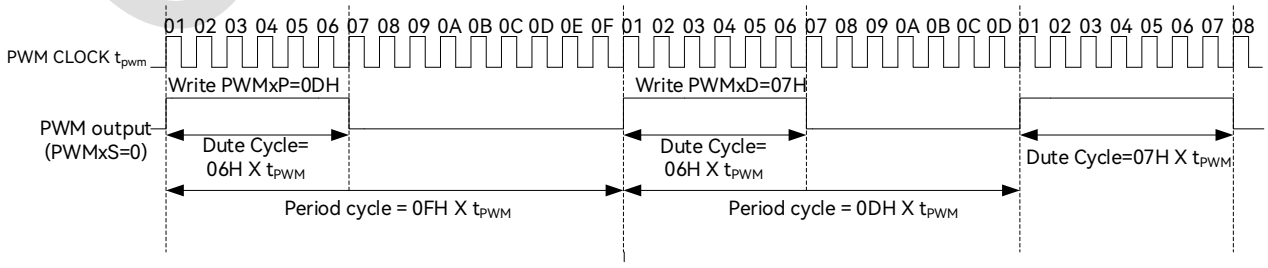


Figure 6-2: Example of PWM Output Cycle or Duty Cycle Change

6.3 PWM Output Timing

Table 6-1: Output Timing (PWMxSS = 1)

PWMxS	Condition	PWMx Port Output Status
0	PWMxP > PWMxD	To output square wave, first output high level of PWMxD clock cycles, and then output low level of PWMxP−PWMxD clock cycles.
	PWMxP ≤ PWMxD	High level
	PWMxD = 0, PWMxP = 0	High level
	PWMxD = 0, PWMxP ≠ 0	Low level
1	PWMxP > PWMxD	To output square wave, first output low level of PWMxD clock cycles, and then output high level of PWMxP−PWMxD clock cycles.
	PWMxP ≤ PWMxD	Low level
	PWMxD = 0, PWMxP = 0	Low level
	PWMxD = 0, PWMxP ≠ 0	High level

Notes:

- The PWMxEN bit controls on and off of the PWMx module.
- The PWMxSS (x = 0−2) bit can select whether the port is used as an I/O port or a PWM output port. When PWMxSS = 1, but PWMxEN = 0, the corresponding port is in the input state.
- The EPWM bit in the IENO register and the PWMxIE bit in the PWMxCON register jointly control the PWMx interrupt.
- The three PWM modules share the interrupt vector.
- When PWMENx = 1 and PWMxSS = 0, the PWMx module output is off and the PWM module can be used as a 16-bit timer. If the PWM interrupt is enabled and PWMxIE=1, the interrupt will be triggered every PWM cycle as well.

6.4 Register Description

Table 6-2: List of Registers

Address	Name	Description
CBH	PWM0_PL	PWM0 data register
CCH	PWM0_PH	PWM0 data register
CDH	PWM1_PL	PWM1 data register
CEH	PWM1_PH	PWM1 data register
C1H	PWM2_PL	PWM2 data register
C2H	PWM2_PH	PWM2 data register
C3H	PWM0_DL	PWM0 duty cycle control register
C4H	PWM0_DH	PWM0 duty cycle control register
C5H	PWM1_DL	PWM1 duty cycle control register
C6H	PWM1_DH	PWM1 duty cycle control register
CFH	PWM2_DL	PWM2 duty cycle control register
C7H	PWM2_DH	PWM2 duty cycle control register
C8H	PWM0_CON	PWM0 setting register
C9H	PWM1_CON	PWM1 setting register
CAH	PWM2_CON	PWM2 setting register

6.4.1 PWMx_PL/H PWMx Data Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0_PL (CBH)	PWM0P[7:0]							
PWM0_PH (CCH)	PWM0P[15:8]							
PWM1_PL (CDH)	PWM1P[7:0]							
PWM1_PH (CEH)	PWM1P[15:8]							
PWM2_PL (C1H)	PWM2P[7:0]							
PWM2_PH (C2H)	PWM2P[15:8]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
15-0	PWMxP [15:0] (x = 0, 1, 2)	PWMx data register

Note:

- Modifying the register PWMxPH will make the output of PWMx take effect in the next cycle.
- If the PWM duty cycle is required to be modified, modify PWMxPL first and then PWMxPH.

6.4.2 PWMx_DL/H PWMx Duty Cycle Control Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0_DL (C3H)	PWM0D[7:0]							
PWM0_DH (C4H)	PWM0D[15:8]							
PWM1_DL (C5H)	PWM1D[7:0]							
PWM1_DH (C6H)	PWM1D[15:8]							
PWM2_DL (CFH)	PWM2D[7:0]							
PWM2_DH (C7H)	PWM2D[15:8]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
15-0	PWMxD[15:0] (x = 0, 1, 2)	PWMx duty cycle control, controlling the output time of PWM0 waveform duty cycle. Refer to the Chapter PWM Output Timing for details of PWM output timing.

Note:

- Modifying the register PWMxDH will make the output of PWMx take effect in the next cycle.
- If the PWM duty cycle is required to be modified, modify PWMxDL first and then PWMxDH.

6.4.3 PWMx_CON PWMx Setting Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0_CON (C8H)	PWM0EN	PWM0S	-	-	-	PWM0IE	PWM0IF	PWM0SS
PWM1_CON (C9H)	PWM1EN	PWM1S	-	-	-	PWM1IE	PWM1IF	PWM1SS
PWM2_CON (CAH)	PWM2EN	PWM2S	-	-	-	PWM2IE	PWM2IF	PWM2SS
R/W	R/W	R/W	-	-	-	R/W	R/W	R/W
Reset value	0	0	-	-	-	0	0	0

Bit No.	Bit Designator	Description
7	PWMxEN	PWMx enable: 0: PWMx module disabled 1: PWMx module enabled
6	PWMxS	PWMx output mode: 0: output high level during PWMx duty cycle and low level after duty cycle overflow 1: output low level during PWMx duty cycle and high level after duty cycle overflow
5-3	-	-
2	PWMxIE	PWMx interrupt enable bit: 0: PWMx interrupt disabled 1: PWMx interrupt enabled The three PWM modules share the interrupt entry address.
1	PWMxIF	PWMx interrupt flag bit: 0: PWM cycle counter does not overflow. 1: PWM cycle counter overflows; this bit is set by hardware and is invalid by software writing 1; writing 0 clears this bit.
0	PWMxSS	PWMx pin output control bit: 0: disable P1_0 for PWM0 output, P1_1 for PWM1 output and P1_2 for PWM2 output, used as I/O. 1: enable P1_0 for PWM0 output, P1_1 for PWM1 output and P1_2 for PWM2 output. **Note:** ● This bit shall be set when PWM0_OUT is P1_0, PWM1_OUT is P1_1 and PWM2_OUT is P1_2. ● This bit is not required to be set when PWM0_OUT is P1_1 / P2_0, PWM1_OUT is P1_2 / P1_5 and PWM2_OUT is P1_3.

6.5 Software Operation Process

1. Configure the PWM output pin either by configuring the PWMxSS bit or by enabling the corresponding pin alternate function.
2. Write the PWMx_PL/PH bit to set the period of PWM.
3. Write the PWMx_DL/DH bit to set the duty cycle of PWM.
4. Write the PWMxS bit to configure the PWM output polarity.
5. If interrupt is to be used, write the EAL and EPWM bits first, then write the PWMxIE register to enable PWM interrupt.
6. Finally write the PWMxEN bit to enable the PWM module.

7 GPIO (I/O Port)

7.1 Main Features

- Up to 17 programmable bidirectional I/O ports
- Interrupt triggered by either edge supported on all IOs
- Built-in pull-up resistor
- I/O port with alternate functions

7.2 Port Module Diagram

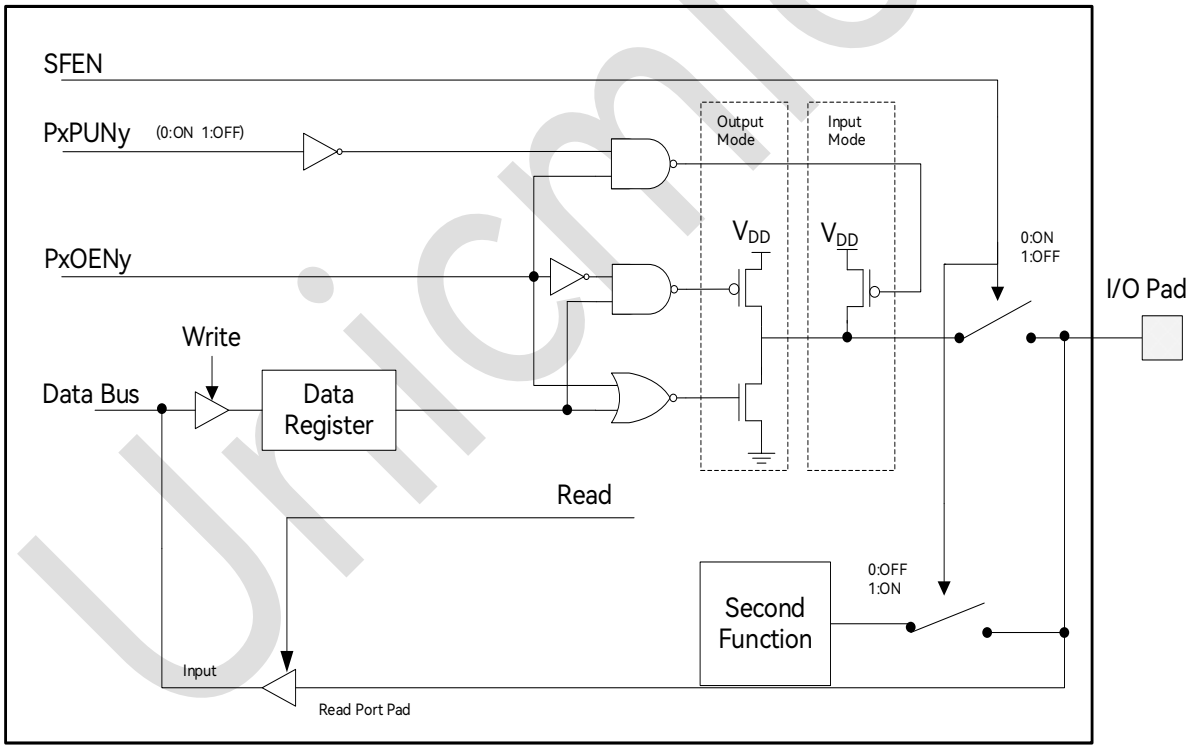


Figure 7-1: Port Module Diagram

Note:

- A read operation on a port configured as an input port is a direct read of the pin level.
- A read operation on a port configured as an output port is also a read of the pin level.
- Even if the I/O port is configured for other alternate functions, the write operation to the port is directed to the port data register.

7.3 Port Interrupt

All ports support interrupt function. If the interrupt is enabled, either rising or falling edge will trigger the interrupt, and all port interrupts share the INT0 interrupt entry. The edge triggering IO interrupt is configured by writing to the P0AL/P0AH/P1AL/P1AH/P2AL/P2AH registers.

The PxIEN register controls the enable of all IO interrupts. If the interrupt of a certain port is enabled, an external interrupt will be generated when the MCU detects a rising or falling edge on the pin, and the corresponding PxIRQ flag bit is set to 1. The user program can query which pin generated the interrupt flag bit in the service program of external interrupt 0.

If the port interrupt is enabled, the change in port level can generate interrupt to wake up the MCU when the MCU enters power-down mode.

7.4 Register Description

Table 7-1: Register Configuration

Address	Name	Description
C010H	P00_CFG	Port P0_0 function configuration register
C011H	P01_CFG	Port P0_1 function configuration register
C013H	P03_CFG	Port P0_3 function configuration register
C014H	P04_CFG	Port P0_4 function configuration register
C018H	P10_CFG	Port P1_0 function configuration register
C019H	P11_CFG	Port P1_1 function configuration register
C01AH	P12_CFG	Port P1_2 function configuration register
C01BH	P13_CFG	Port P1_3 function configuration register
C01CH	P14_CFG	Port P1_4 function configuration register
C01DH	P15_CFG	Port P1_5 function configuration register
C020H	P20_CFG	Port P2_0 function configuration register
C022H	P22_CFG	Port P2_2 function configuration register
C023H	P23_CFG	Port P2_3 function configuration register
C025H	P25_CFG	Port P2_5 function configuration register
C026H	P26_CFG	Port P2_6 function configuration register

C027H	P27_CFG	Port P2_7 function configuration register
C000H	P0_IE	P0 IO input control register
C001H	P1_IE	P1 IO input control register
C002H	P2_IE	P2 IO input control register
C005H	P0_SR	P0 IO speed control register
C006H	P1_SR	P1 IO speed control register
C007H	P2_SR	P2 IO speed control register

7.4.1 P00_CFG

C010H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P00_CFG	-	-	-	-	-	P00_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P00_SEL	000: P0_0 001: UART2_RX 010: SPI_CSN 011: LPOUT0 100: GTIMER1_CHN 101: GTIMER2_BKE 110: CAN_TX

7.4.2 P01_CFG

C011H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P01_CFG	-	-	-	-	-	P01_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-

2-0	P01_SEL	000: P0_1 001: UART2_TX 010: SPI_SCK 011: I2C_SDA 100: LPOUT1 101: GTIMER0_BKE 110: GTIMER2_CHN 111: CAN_RX

7.4.3 P03_CFG

C013H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P03_CFG	-	-	-	-	-	P03_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x01		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P03_SEL	000: P0_3 001: CLKOUT 010: UART2_TX 011: UART3_RX 100: SPI_CSN 101: LPOUT0 110: GTIMER1_CH 111: CAN_TX

7.4.4 P04_CFG

C014H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P04_CFG	-	-	-	-	-	P04_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x03		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P04_SEL	000: P0_4

		001: UART2_RX 010: SPI_SCK 011: I2C_SDA 100: LPOUT1 101: GTIMER1_BKE 110: GTIMER2_CHN 111: CAN_RX

7.4.5 P10_CFG

C018H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P10_CFG	-	-	-	-	-	P10_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x04		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P10_SEL	000: P1_0 001: UART1_RX 010: UART2_TX 011: PWM0 100: I2C_SCL 101: LP0_IN 110: GTIMER2_CH

7.4.6 P11_CFG

C019H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P11_CFG	-	-	-	-	-	P11_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-

2-0	P11_SEL	000: P1_1 001: UART1_TX 010: UART3_RX 011: PWM1 100: SPI_MISO 101: LP0_TRG 110: GTIMER1_CHN 111: PWM0

7.4.7 P12_CFG

C01AH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P12_CFG	-	-	-	-	-	P12_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P12_SEL	000: P1_2 001: UART0_RX 010: UART3_TX 011: PWM2 100: LP0_CAP 101: GTIMER1_CH 110: PWM1 111: CAN_TX

7.4.8 P13_CFG

C01BH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P13_CFG	-	-	-	-	-	P13_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P13_SEL	000: P1_3

		001: UART0_TX 010: UART2_RX 011: SPI_SCK 100: I2C_SDA 101: LP0_IN 110: GTIMER0_CH 111: PWM2

7.4.9 P14_CFG

C01CH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P14_CFG	-	-	-	-	-	P14_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x01		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P14_SEL	000: P1_4 001: UART1_RX 010: PWM2 011: SPI_MOSI 100: LP0_TRG 101: GTIMER0_CHN 110: GTIMER1_BKE 111: CAN_RX

7.4.10 P15_CFG

C01DH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P15_CFG	-	-	-	-	-	P15_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x01		

Bit No.	Bit Designator	Description
7-3	-	-

2-0	P15_SEL	000: P1_5 001: UART1_TX 010: PWM1 011: SPI_MISO 100: GTIMER0_CH 101: GTIMER1_BKE 110: GTIMER2_CH 111: LP0_CAP1

7.4.11 P20_CFG

C020H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P20_CFG	-	-	-	-	-	P20_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P20_SEL	000: P2_0 001: UART3_RX 010: PWM0 011: SPI_MOSI 100: I2C_SCL 101: LPOUT0 110: GTIMER0_CHN

7.4.12 P22_CFG

C022H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P22_CFG	-	-	-	-	-	P22_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P22_SEL	000: P2_2 001: UART3_TX

		010: SPI_CSN 011: SPI_MISO 100: I2C_SDA 101: GTIMER0_BKE 110: GTIMER2_CHN 111: LP0_CAP1

7.4.13 P23_CFG

C023H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P23_CFG	-	-	-	-	-	P23_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P23_SEL	000: P2_3 001: UART3_RX 010: SPI_SCK 011: SPI_MOSI 100: LP0_CAP 101: GTIMER0_CHN 110: GTIMER2_BKE 111: CAN_RX

7.4.14 P25_CFG

C025H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P25_CFG	-	-	-	-	-	P25_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x06		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P25_SEL	000: P2_5 001: UART3_TX 010: SPI_CSN

		011: I2C_SCL 100: GTIMER0_CH 101: GTIMER0_BKE 110: BUZZER_OUT 111: UART0_RX

7.4.15 P26_CFG

C026H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P26_CFG	-	-	-	-	-	P26_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x01		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P26_SEL	000: P2_6 001: UART0_TX 010: UART2_TX 011: SPI_MISO 100: LPOUT1 101: GTIMER1_CH 110: GTIMER2_CH

7.4.16 P27_CFG

C027H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P27_CFG	-	-	-	-	-	P27_SEL		
R/W	R	R	R	R	R	R/W		
Reset value	0	0	0	0	0	0x01		

Bit No.	Bit Designator	Description
7-3	-	-
2-0	P27_SEL	000: P2_7 001: UART0_RX 010: UART2_RX 011: SPI_MOSI

		100: I2C_SCL 101: GTIMER1_CHN 110: GTIMER2_BKE 111: BUZZER_OUT

7.4.17 P0_IE

C000H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0_IE	-	-	-	P0_4_IE	P0_3_IE	P0_2_IE	P0_1_IE	P0_0_IE
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	1	1	1	1	1

Bit No.	Bit Designator	Description
7-5	-	-
4	P0_4_IE	P0_4 IO input control bit: 0: P0_4 input disabled 1: P0_4 input enabled
3	P0_3_IE	P0_3 IO input control bit: 0: P0_3 input disabled 1: P0_3 input enabled
2	P0_2_IE	P0_2 IO input control bit: 0: P0_2 input disabled 1: P0_2 input enabled
1	P0_1_IE	P0_1 IO input control bit: 0: P0_1 input disabled 1: P0_1 input enabled
0	P0_0_IE	P0_0 IO input control bit: 0: P0_0 input disabled 1: P0_0 input enabled

7.4.18 P1_IE

C001H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1_IE	-	-	P1_5_IE	P1_4_IE	P1_3_IE	P1_2_IE	P1_1_IE	P1_0_IE
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-6	-	-
5	P1_5_IE	P1_5 IO input control bit: 0: P1_5 input disabled 1: P1_5 input enabled
4	P1_4_IE	P1_4 IO input control bit: 0: P1_4 input disabled 1: P1_4 input enabled
3	P1_3_IE	P1_3 IO input control bit: 0: P1_3 input disabled 1: P1_3 input enabled
2	P1_2_IE	P1_2 IO input control bit: 0: P1_2 input disabled 1: P1_2 input enabled
1	P1_1_IE	P1_1 IO input control bit: 0: P1_1 input disabled 1: P1_1 input enabled
0	P1_0_IE	P1_0 IO input control bit: 0: P1_0 input disabled 1: P1_0 input enabled

7.4.19 P2_IE

C002H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2_IE	P2_7_IE	P2_6_IE	P2_5_IE	-	P2_3_IE	P2_2_IE	-	P2_0_IE
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W
Reset value	1	1	1	0	1	1	0	1

Bit No.	Bit Designator	Description
7	P2_7_IE	P2_7 IO input control bit: 0: P2_7 input disabled 1: P2_7 input enabled
6	P2_6_IE	P2_6 IO input control bit: 0: P2_6 input disabled 1: P2_6 input enabled

5	P2_5_IE	P2_5 IO input control bit: 0: P2_5 input disabled 1: P2_5 input enabled
4	-	-
3	P2_3_IE	P2_3 IO input control bit: 0: P2_3 input disabled 1: P2_3 input enabled
2	P2_2_IE	P2_2 IO input control bit: 0: P2_2 input disabled 1: P2_2 input enabled
1	-	-
0	P2_0_IE	P2_0 IO input control bit: 0: P2_0 input disabled 1: P2_0 input enabled

7.4.20 P0_SR

C005H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P0_SR	-	-	-	P0_4_SR	P0_3_SR	P0_2_SR	P0_1_SR	P0_0_SR
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	1	1	1	1	1

Bit No.	Bit Designator	Description
7-5	-	-
4	P0_4_SR	P0_4 IO speed control bit: 0: P0_4 fast 1: P0_4 slow
3	P0_3_SR	P0_3 IO speed control bit: 0: P0_3 fast 1: P0_3 slow
2	P0_2_SR	P0_2 IO speed control bit: 0: P0_2 fast 1: P0_2 slow
1	P0_1_SR	P0_1 IO speed control bit: 0: P0_1 fast 1: P0_1 slow

0	P0_0_SR	P0_0 IO speed control bit: 0: P0_0 fast 1: P0_0 slow

7.4.21 P1_SR

C006H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P1_SR	-	-	P1_5_SR	P1_4_SR	P1_3_SR	P1_2_SR	P1_1_SR	P1_0_SR
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-6	-	-
5	P1_5_SR	P1_5 IO speed control bit: 0: P1_5 fast 1: P1_5 slow
4	P1_4_SR	P1_4 IO speed control bit: 0: P1_4 fast 1: P1_4 slow
3	P1_3_SR	P1_3 IO speed control bit: 0: P1_3 fast 1: P1_3 slow
2	P1_2_SR	P1_2 IO speed control bit: 0: P1_2 fast 1: P1_2 slow
1	P1_1_SR	P1_1 IO speed control bit: 0: P1_1 fast 1: P1_1 slow
0	P1_0_SR	P1_0 IO speed control bit: 0: P1_0 fast 1: P1_0 slow

7.4.22 P2_SR

C007H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
P2_SR	P2_7_SR	P2_6_SR	P2_5_SR	-	P2_3_SR	P2_2_SR	-	P2_0_SR
R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W
Reset value	1	1	1	0	1	1	0	1

Bit No.	Bit Designator	Description
7	P2_7_SR	P2_7 IO speed control bit: 0: P2_7 fast 1: P2_7 slow
6	P2_6_SR	P2_6 IO speed control bit: 0: P2_6 fast 1: P2_6 slow
5	P2_5_SR	P2_5 IO speed control bit: 0: P2_5 fast 1: P2_5 slow
4	-	-
3	P2_3_SR	P2_3 IO speed control bit: 0: P2_3 fast 1: P2_3 slow
2	P2_2_SR	P2_2 IO speed control bit: 0: P2_2 fast 1: P2_2 slow
1	-	-
0	P2_0_SR	P2_0 IO speed control bit: 0: P2_0 fast 1: P2_0 slow

8 Beeper

The chip integrates a beeper signal generator that can automatically output 1-, 2- and 4-kHz square waves by hardware.

8.1 Register Description

8.1.1 BEEPCTR

86H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
BEEPCTR	UART3CEN	UART3RSTEN	CANCEN	BEEPEN	CANRSTN	BEEPCOLSET	BEEPSEL [1:0]	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset value	1	1	1	0	1	0	0	0

Bit No.	Bit Designator	Description
7	UART3CEN	1: UART3 clock enabled 0: UART3 clock disabled
6	UART3RSTEN	1: UART3 reset released 0: UART3 reset
5	CANCEN	1: CAN clock enabled 0: CAN clock disabled
4	BEEPEN	BEEP module enable bit: 0: BEEP module disabled 1: BEEP module enabled Note: BEEP defaults to P2.5 output.
3	CANRSTN	1: CAN reset released 0: CAN reset
2	BEEPCOLSET	BEEP polarity control: 0: BEEP outputs low level by default 1: BEEP outputs high level by default
1-0	BEEPSEL	BEEP output frequency control: 00/11: 1 kHz 01: 2 kHz 10: 4 kHz

9 UART0/1 (Enhanced Serial Port)

9.1 Main Features

- UART0/1 come with baud rate generator.
- UART0 with four operation modes
- UART1 with two operation modes

Both serial ports are composed of a shift register, a serial control register, a baud rate generator and two independent data buffers (for transmitting and receiving data respectively). The two data buffers are collectively referred to as S0BUF (S1BUF) sharing the address of 99H (9CH). Write data to S0BUF or S1BUF to start serial data transmission, and read S0BUF or S1BUF to return the data already received by the buffer.

When the serial port is receiving data, the data enters the shift register first and is moved into S0BUF (S1BUF) after receiving a frame of data, then the next frame of data is received immediately. The master shall ensure that the data in S0BUF (S1BUF) buffer is taken away before receiving this frame of data, otherwise the previous frame of data may be overwritten by this frame of data, resulting in data loss.

9.2 UART0 Operation Mode

UART0 has four operation modes. Before communication, the user must initialize the relevant registers and select the appropriate operation mode and baud rate. Different operation modes can be selected by setting SM0/SM1.

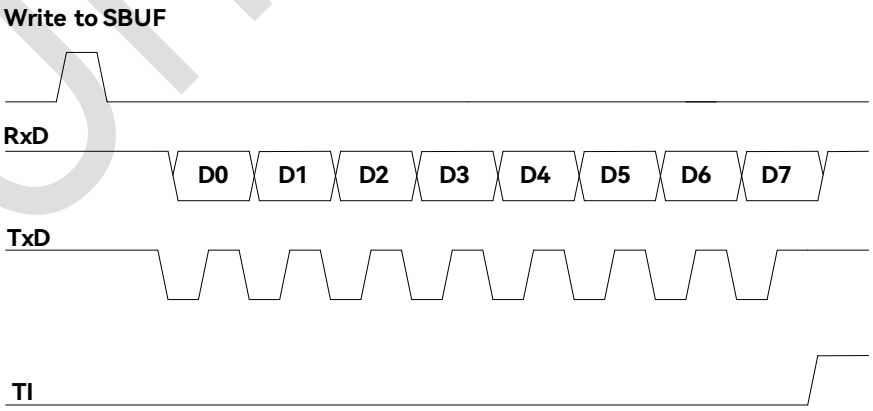
Table 9-1: List of UART0 Operation Mode

SM0	SM1	Mode	Description	Baud Rate
0	0	0	Shift register	SYSCLK / 12
0	1	1	8-bit UART	Configurable
1	0	2	9-bit UART	SYSCLK / 16
1	1	3	9-bit UART	Configurable

- Mode 0: synchronous and half-duplex communication

  Mode 0 supports synchronous communication with external devices. Serial data is transmitted and received on RX pin, and shift clock is transmitted on TX pin. In this mode, 8 bits are transmitted and received per frame, with the lower bits being received or transmitted first.

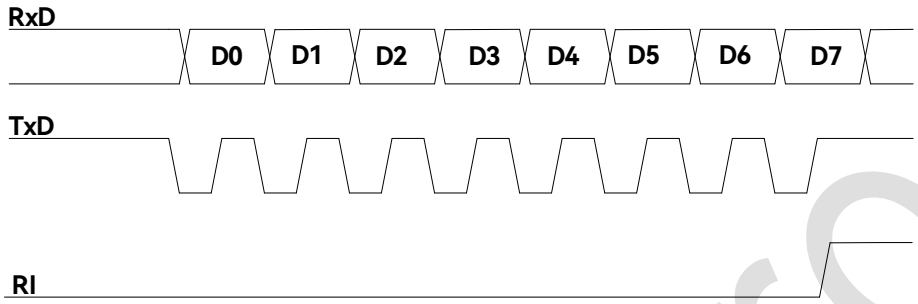
  Any write operation with SBUF as the destination register will initiate transmission, and the next system clock TX control block will start transmitting. Data conversion occurs on the falling edge of the shift clock, where the shift register contents are shifted left to right one by one, with the null bit set as 0. Upon completion of transmission, the TX control block stops transmitting and then sets TI on the rising edge of the next system clock.



Transmit Timing of Mode 0

Set REN to 1 and clear RI to initiate reception. The data is latched on the rising edge of the shift clock and the contents of the RX conversion register are successively shifted to the

left. As all the 8-bit data are moved to the shift register, the RX control block stops receiving, RI is set at the rising edge of the next system clock, and the next reception is not allowed until the software cleared this bit.

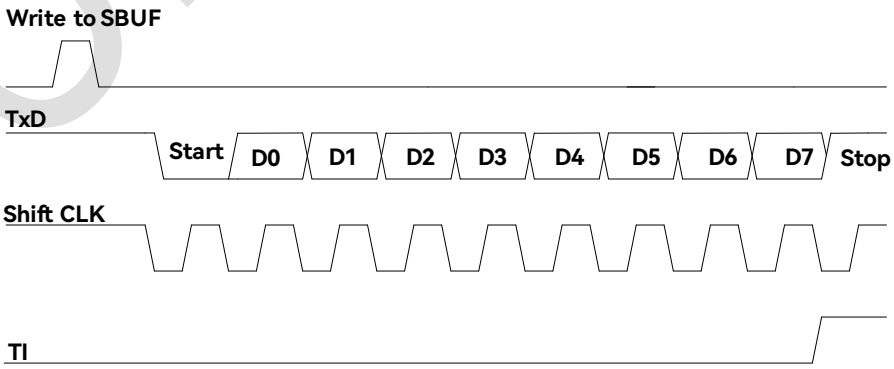


Receive Timing of Mode 0

- Mode 1: 8-bit UART, variable baud rate, asynchronous full-duplex

  Mode 1 supports 10-bit full-duplex asynchronous communication, with the 10 bits consisting of a start bit (logic 0), 8 data bits (the lower bit is the first bit) and a stop bit (logic 1). On receiving, 8 data bits are stored in SBUF and the stop bit is stored in RB8.

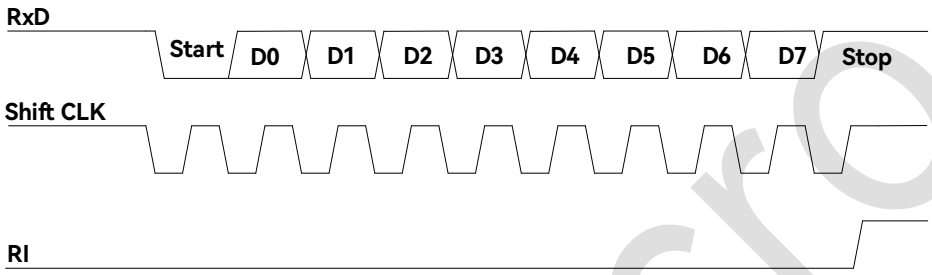
  Any write operation with SBUF as the destination register will initiate transmission, with the start bit shifted out on the TX pin first, followed by the 8 data bits. After all the 8 data bits in the TX shift register are transmitted, the stop bit is shifted out from the TX pin, and the TI flag is set to issue an interrupt request at the same time as the stop bit is sent out.



Transmit Timing of Mode 1

Reception is allowed only when REN is set. The serial port starts receiving serial data when

a falling edge is detected on the RX pin. If the start bit is valid, it is shifted into the shift register, followed by other bits into the shift register. After the 8 data bits and 1 stop bit are shifted in, the contents of the shift register are loaded into SBUF and RB8 respectively, followed by RI being set. At this point, the receiver continues to detect the next falling edge on RX pin. The user needs to clear the RI by software before it can receive again.

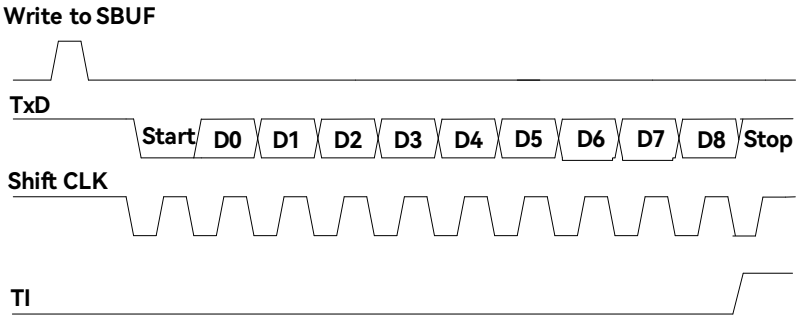


Receive Timing of Mode 1

- Mode 2: 9-bit UART, fixed baud rate, asynchronous full-duplex

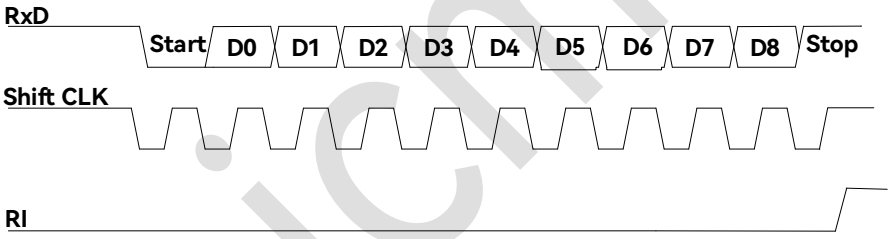
  Mode 2 supports 11-bit full-duplex asynchronous communication with a baud rate fixed at 1/16 of the system clock. A frame consists of a start bit (logic 0), 8 data bits (the lower bit is the first bit), a programmable 9th bit and a stop bit (logic 1). Mode 2 and Mode 3 support multi-processor communication.

  Any write operation with SBUF as the destination register will initiate transmission while loading TB8 into the bit 9 of the TX shift register. The start bit is first shifted out from the TX pin, followed by the 9 data bits being shifted. After all the data are transmitted, the stop bit is shifted out from the TX pin, and the TI flag is set to issue an interrupt request as the stop bit starts to be sent.



Transmit Timing of Mode 2

Reception is allowed only when REN is set. The serial port starts receiving serial data when a falling edge is detected on the RX pin. If the start bit is valid, it is shifted into the shift register, followed by other bits into the shift register. After the 9 data bits and 1 stop bit are shifted in, the contents of the shift register are loaded into SBUF and RB8 respectively, followed by RI being set. At this point, the receiver continues to detect the next falling edge on RX pin. The user needs to clear the RI by software before it can receive again.



Receive Timing of Mode 2

- Mode 3: 9-bit UART, variable baud rate, asynchronous full-duplex

  Mode 3 adopts the transmission protocol of Mode 2 and the baud rate generation method of Mode 1.

Note: The internal pull-up of serial IO shall be enabled correspondingly in the PxPUN register, otherwise the RX pin is floating and vulnerable to interference.

9.3 UART1 Operation Mode

Table 9-2: List of UART1 Operation Mode

SM	Mode	Description	Baud Rate
0	A	9-bit UART	Configurable
1	B	8-bit UART	Configurable

Refer to Mode 3 and Mode 1 of UART0 for Mode A and Mode B of UART1, respectively.

9.4 Multi-processor Communication

Mode 2 and Mode 3 of UART0 and Mode A of UART1 are provided with the function of multi-processor communication. In the case of multi-processor communication system being adopted, when the master is to send a block of data to one of several slaves, it first sends an address byte to address the target slave. The address byte and the data byte can be distinguished by the 9th data bit, which is 1 for the address byte and 0 for the data byte. The receiver judges whether to receive based on the information of the 9th bit, and the multi-processor communication process is as follows:

- The transmitting process of the master communication is set as follows:
  1. Set to the 9-bit mode, send the receiver address, and set TB8 = 1.
  2. Transmit the data according to the custom protocol and set TB8 = 0.
- The receiving process of the master communication is set as follows:
  1. Set SM2 = 0 (receive all data unconditionally).
  2. The master parses the data according to the custom protocol.
- The receiving process of the slave communication is set as follows:
  1. Set SM2 = 1 for the slave to be in the state of receiving only address frames, at this time only the address data with the 9th bit being 1 will be received.
  2. When the data is received, the software determines whether it matches the serial

address set by the local machine.

3. After receiving the address frame, each slave will compare the received address with the local address.
   - If they are matched, it is the target slave, clear SM2 = 0, prepare to receive the data frame to be sent by the master, and set SM2 = 1 again after receiving.
   - If they do not match, keep SM2 = 1 and ignore all the next data frames without generating interrupt requests until the address frame is received for comparison and confirmation again.

9.5 UART0 Register Description

Table 9-3: List of Registers

Address	Name	Description
98H	UART0_S0CON	Interrupt register
AAH	UART0_S0RELL	Baud rate configuration register
BAH	UART0_S0RELH (BAH)	Baud rate configuration register
99H	UART0_S0BUF	Data register
9EH	UARTEN	Enable register

9.5.1 UART0_S0CON Interrupt Register

98H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART0_S0CON	SM0	SM1	SM20	REN0	TB80	RB80	TI0	RI0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	SM0	UART0 mode selection bit:

6	SM1	SM0 SM1 T0MODE Description Baud Rate 0 0 0 Shift register SYSCLK / 12 0 1 1 8-bit UART Configurable 1 0 2 9-bit UART SYSCLK / 16 1 1 3 9-bit UART Configurable
5	SM20	UART0 multi-processor communication enable bit: 0: multi-processor communication disabled 1: multi-processor communication enabled to receive only the 9-bit data when RB80 = 1
4	REN0	UART0 receive enable bit: 0: UART0 disabled to receive 1: UART0 enabled to receive
3	TB80	The 9th TX data bit in Mode 2 and 3 of UART0
2	RB80	The 9th RX data bit in Mode 2 and 3 of UART0
1	TI0	UART0 TX interrupt flag bit: Upon completion of the data transmission, this bit shall be set by hardware and must be cleared by software.
0	RI0	UART0 RX interrupt flag bit: Upon completion of the data reception, this bit shall be set by hardware and must be cleared by software.

9.5.2 UART0_S0REL Baud Rate Configuration Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART0_S0RELL (AAH)	S0RELL.7	S0RELL.6	S0RELL.5	S0RELL.4	S0RELL.3	S0RELL.2	S0RELL.1	S0RELL.0
Reset Value	1	1	1	0	0	1	1	0
UART0_S0RELH (BAH)	-	-	-	-	-	-	S0RELH.1	S0RELH.0
Reset Value	-	-	-	-	-	-	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit No.	Bit Designator	Description
7-0 1-0	S0RELL[7:0] S0REH[1:0]	Baud rate configuration register for Mode 1 and 3 of UART0: S0RELH + S0RELL constitute the baud rate setting S0REL[9:0] Baud rate = $\frac{SYSCK}{16\times(1024-S0REL)}$

9.5.3 UART0_S0BUF Data Register

99H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART0_S0BUF	S0BUF.7	S0BUF.6	S0BUF.5	S0BUF.4	S0BUF.3	S0BUF.2	S0BUF.1	S0BUF.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	S0BUF[7:0]	UART0 data register: Read S0BUF to return the data received by UART0; write S0BUF to start UART0 data transmission.

9.5.4 UARTEN Enable Register

9EH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UARTEN	OUTSEL		OUTEN	FLH24MOUTEN	-	-	UART1EN	UART0EN
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	OUTSEL	11: XTH output 10: system clock output 01: RCL38K clock output 00: RCH_CLK_DIV clock output
5	OUTEN	CLK output enable: 1: CLKOUT enabled 0: CLKOUT disabled Note: After the output is enabled, P0_3 is used as the CLKOUT pin to output the clock signal.
4	FLH24MOUTEN	-
3-2	-	-

1	UART1EN	UART1 enable bit: 0: P1.4 and P1.5 configured as GPIO 1: P1.4 and P1.5 configured as RX1 and TX1 of UART1 Note: This bit is not required to be set when P1_1 and P1_0 are configured as TX1 and RX1 of UART1.
0	UART0EN	UART0 enable bit: 0: P2.6 and P2.7 configured as GPIO 1: P2.6 and P2.7 configured as TX0 and RX0 of UART0 Note: This bit is not required to be set when P1_3 and P1_2 are configured as TX0 and RX0 of UART0.

9.6 UART1 Register Description

Table 9-4: List of Registers

Address	Name	Description
9BH	UART1_S1CON	Interrupt register
9DH	UART1_S1RELL	Baud rate configuration register
BBH	UART1_S1RELH	Baud rate configuration register
9CH	UART1_S1BUF	Data register
9EH	UARTEN	Enable register

9.6.1 UART1_S1CON Interrupt Register

9BH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART1_S1CON	SM	-	SM21	REN1	TB81	RB81	TI1	RI1
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

<table>
<tr><th>Bit No.</th><th>Bit Designator</th><th>Description</th></tr>
<tr><td>7</td><td>SM</td><td>UART1 mode selection bit:
<table>
<tr><th>SM</th><th>Mode</th><th>Description</th><th>Baud Rate</th></tr>
<tr><td>0</td><td>A</td><td>9-bit UART</td><td>Configurable</td></tr>
<tr><td>1</td><td>B</td><td>8-bit UART</td><td>Configurable</td></tr>
</table></td></tr>
<tr><td>6</td><td>-</td><td>-</td></tr>
</table>

5	SM21	UART1 multi-processor communication enable bit: 0: multi-processor communication disabled 1: multi-processor communication enabled to receive only the 9-bit data with RB8 = 1
4	REN1	UART1 receive enable bit: 0: UART1 disabled to receive 1: UART1 enabled to receive
3	TB81	The 9th TX data bit of UART1
2	RB81	The 9th RX data bit of UART1
1	TI1	UART1 TX interrupt flag bit: Upon completion of the data transmission, this bit shall be set by hardware and must be cleared by software.
0	RI1	UART1 RX interrupt flag bit: Upon completion of the data reception, this bit shall be set by hardware and must be cleared by software.

9.6.2 UART1_S1REL Baud Rate Configuration Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART1_S1RELL (9DH)	S1RELL.7	S1RELL.6	S1RELL.5	S1RELL.4	S1RELL.3	S1RELL.2	S1RELL.1	S1RELL.0
Reset Value	1	1	1	0	0	1	1	0
UART1_S1RELH (BBH)	-	-	-	-	-	-	S1RELH.1	S1RELH.0
Reset Value	-	-	-	-	-	-	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit No.	Bit Designator	Description
7-0 1-0	S1RELL[7:0] S1REH[1:0]	UART0 baud rate configuration register: S1RELH + S1RELL constitute the baud rate setting S1REL[9:0] Baud rate = $\frac{\text{SYSCK}}{16\times(1024-\text{S1REL})}$

9.6.3 UART1_S1BUF Data Register

9CH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART1_S1BUF	S1BUF.7	S1BUF.6	S1BUF.5	S1BUF.4	S1BUF.3	S1BUF.2	S1BUF.1	S1BUF.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	S1BUF[7:0]	UART1 data register: Read S1BUF to return the data received by UART1; write S1BUF to start UART1 data transmission.

9.6.4 UARTEN Enable Register

Please refer to Chapter "UARTEN Enable Register" for details.

9.7 Baud Rate

- UART0 Mode 1 and 3

  Baud rate = $\frac{\text{SYSCK}}{16\times(1024-\text{S0REL})}$

- UART1

  Baud rate = $\frac{\text{SYSCK}}{16\times(1024-\text{S1REL})}$

SYSCLK is of 16 M, and the configuration values of the common baud rate SxREL and actual errors are as follows:

Table 9-5: Table of Baud Rate Error

Target Baud Rate	SxREL	Actual Baud Rate	Error
115200	1015	111111	3.5%
57600	1007	58824	-2.1%
38400	998	38462	-0.2%
19200	972	19231	-0.2%
9600	920	9615	-0.2%
4800	816	4808	-0.16%
2400	607	2398	0.08%

10 UART2/3 (Universal Asynchronous Receiver Transmitter)

Universal asynchronous receiver/transmitter (hereinafter referred to as UART) is a widely used serial communication interface that supports full duplex communication. UART is to send the data transmitted in parallel in memory or processor to the UART receiver of peripherals in series, or to receive the serial data of UART peripherals and convert them into parallel data for the processor. It supports serial communication with external interface devices.

10.1 Main Features

Functional features:

- Providing standard asynchronous communication bits (start bit, parity bit, stop bit):
  - 1 start bit
  - 1 parity bit (odd or even), or no parity bit
  - 1 stop bit
  - Bytes transmitted sequentially from LSB to MSB
- 8-bit 4-level RX FIFO
- Programmable baud rate (adjustable according to parameter F/D)
- Data communication and error handling interrupt:
  - Status bit can be accessed by either query or interrupt.
  - Flags of FIFO non-empty, half-full, full, overflow
  - Parity error flag
- Validity check of start bit
- 2 * 8-bit baud rate parameter registers
- Supporting transmission at common baud rates such as 9600 bps, 19200 bps and 115200 bps

10.2 Register Description

UART2 register base address: 0xCD00; UART3 register base address: 0xCE00.

Table 10-1: List of Registers

Offset	Name	Description
0x00	UART_ISR	UART interrupt status register
0x01	UART_IER	UART interrupt enable register
0x02	UART_CR	UART control register
0x03	UART_TDR	UART transmit data register
0x03	UART_RDR	UART receive data register
0x04	UART_BPR_L	UART baud rate parameter low-order register
0x05	UART_BPR_H	UART baud rate parameter high-order register

10.2.1 UART_ISR Interrupt Status Register

CD00H/CE00H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_ISR	RSV		FIFO_NE	FIFO_HF	FIFO_FU	FIFO_OV	TXEND	TRE
R/W	R		R/W	R/W	R/W	R	R/W	R
Reset value	0		0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	RSV	Reserved
5	FIFO_NE	FIFO non-empty flag: FIFO_NE = 0, FIFO empty; FIFO_NE = 1, FIFO not empty. This bit will be automatically cleared when FIFO is empty. Software can also clear this bit by writing 0.
4	FIFO_HF	FIFO half-full flag: FIFO_HF = 0, FIFO not half-full; FIFO_HF = 1, FIFO half-full. This bit will be automatically cleared when the data in FIFO is not full. Software can also clear this bit by writing it to 0.
3	FIFO_FU	FIFO full flag: FIFO_FU = 0, FIFO not full; FIFO_FU = 1, FIFO full. This bit will be automatically cleared when the data in FIFO

		is not full. Software can also clear this bit by writing it to 0.
2	FIFO_OV	RX FIFO RX overflow: FIFO_OV = 0, no RX overflow; FIFO_OV = 1, RX overflow occurred. Software can also clear this bit by writing 0.
1	TXEND	UART transmission complete flag: TXEND = 0, transmission not completed; TXEND = 1, transmission completed. This bit is set to 1 by hardware, and can be cleared by software via writing it to 0.
0	TRE	UART TX/RX parity error flag: TRE = 0, there is no parity error upon completion of UART transmitting/receiving. TRE = 1, there is parity error upon completion of UART transmitting/receiving. This bit is set to 1 by hardware, and can be cleared by software via writing it to 0.

10.2.2 UART_IER Interrupt Enable Register

CD01H/CE01H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_IER	RSV		FIFO_EN	FIFO_HFEn	FIFO_FUEn	FIFO_OVEn	TXENDEn	TREEn
R/W	R		R/W	R/W	R/W	R/W	R/W	R
Reset value	0		0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	RSV	Reserved
5	FIFO_EN	FIFO non-empty interrupt enable: FIFO_EN = 0, disabled; FIFO_EN = 1, enabled.
4	FIFO_HFEn	FIFO half-full interrupt enable: FIFO_HFEn = 0, disabled; FIFO_HFEn = 1, enabled.
3	FIFO_FUEn	FIFO full interrupt enable: FIFO_FUEn = 0, disabled; FIFO_FUEn = 1, enabled.
2	FIFO_OVEn	RX FIFO overflow interrupt enable: FIFO_OVEn = 0, disabled; FIFO_OVEn = 1, enabled.
1	TXENDEn	UART TX completion interrupt enable: TXENDEn = 0,

		disabled; TXENDEn = 1, enabled.
0	TREEn	UART TX/RX parity error interrupt enable: TREEn = 0, disabled; TREEn = 1, enabled.

10.2.3 UART_CR Control Register

CD02H/CE02H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_CR	RSV	TX_EN	TX_OEN	UART_LB	UART_P0	FLUSH	TRS	ODD_EN
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-5	RSV	Reserved
6	TX_EN	UART single-wire mode enable: TX_EN = 0, disabled; TX_EN = 1, enabled.
5	TX_OEN	UART TX pin data transmission direction control in single-wire mode: TX_OEN = 0, TX pin used as data output pin; TX_OEN = 1, TX pin used as data input pin.
4	UART_LB	UART self-test mode enable: UART_LB = 0, disabled; UART_LB = 1, enabled.
3	UART_P0	Parity enable: UART_PD = 0, with parity; UART_PD = 1, no parity.
2	FLUSH	Clear data and pointer in UART RX FIFO: FLUSH = 0, not cleared; FLUSH = 1, cleared.
1	TRS	UART data transmission flag: TRS = 0, data transmission disabled; TRS = 1, data transmission enabled.
0	ODD_EN	Selection of parity mode: ODD_EN = 0, even parity; ODD_EN = 1, odd parity.

10.2.4 UART_TDR TX Data Register

CD03H/CE03H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_TDR	UARTDATA							
R/W	W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	UARTDATA	Store the data to be transmitted.

10.2.5 UART_RDR RX Data Register

CD03H/CE03H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_RDR	UARTDATA							
R/W	R							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	UARTDATA	Store the data to be received.

10.2.6 UART_BPR_L Baud Rate Parameter Low-order Register

CD04H/CE04H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_BPR_L	UART_BPR_L							
R/W	R/W							
Reset value	0x74							

Bit No.	Bit Designator	Description
7-0	UART_BPR_L	The baud rate parameter registers UART_BPRH and UART_BPRL constitute a 16-bit frequency divider. **For example, to obtain a baud rate of 9600 kbps with the 40 MHz system clock,** the register shall be configured as UART_BPR = 40 × 1000000 ÷ 9600 = 1046H, i.e., UART_BPR_H = 10 H, UART_BPR_L = 46 H. **For example, to obtain a baud rate of 19200 kbps with the 40 MHz system clock,**

		the register shall be configured as UART_BPR = 0823H, i.e., UART_BPR_H = 08 H, UART_BPR_L = 23 H.

Note: The system clock shall be with more than 12 times the baud rate.

10.2.7 UART_BPR_H Baud Rate Parameter High-order Register

CD05H/CE05H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
UART_BPR_H	UART_BPR_H							
R/W	R/W							
Reset value	0x01							

Bit No.	Bit Designator	Description
7-0	UART_BPR_H	The baud rate parameter registers UART_BPRH and UART_BPRL constitute a 16-bit frequency divider. **For example, to obtain a baud rate of 9600 kbps with the 40 MHz system clock,** the register shall be configured as UART_BPR = 40 × 1000000 ÷ 9600 = 1046H, i.e., UART_BPR_H = 10 H, UART_BPR_L = 46 H. **For example, to obtain a baud rate of 19200 kbps with the 40 MHz system clock,** the register shall be configured as UART_BPR = 0823H, i.e., UART_BPR_H = 08 H, UART_BPR_L = 23 H.

Note: The system clock shall be with more than 12 times the baud rate.

10.3 Process Description

10.3.1 UART Transmission Process

1. Configure PCLK1 and PRESET1, enable UART, and reset release.
2. Multiplex IO into UART_RX and UART_TX according to the IO multiplexing relation.
3. Configure UART_CR, set whether there is parity, and enable data transmission.
4. According to the calculation, configure UART_BPR_H and UART_BPR_L, set the baud rate.
5. Enable UART interrupt and total interrupt, configure UART_IER, and enable transmission done interrupt.
6. Write data to UART_TDR.
7. Query the interrupt status of UART_ISR.
8. Complete the transmission.

10.3.2 UART Reception Process

1. Configure PCLK1 and PRESET1, enable UART, and reset release.
2. Multiplex IO into UART_RX and UART_TX according to the IO multiplexing relation.
3. Configure UART_CR, set whether there is parity, and enable data transmission.
4. According to the calculation, configure UART_BPR_H and UART_BPR_L, set the baud rate.
5. Enable UART interrupt and total interrupt, configure UART_IER, and enable RX FIFO non-empty interrupt.
6. Query the interrupt status of UART_ISR.
7. Read data from UART_TDR.
8. Complete the transmission.

Note: The internal pull-up of serial IO shall be enabled correspondingly in the PxPUN register, otherwise the RX pin is floating and vulnerable to interference.

11 SPI

11.1 Overview

The serial peripheral interface (SPI) is a serial synchronous communication means for external devices to exchange data over a single line. The chip provides an SPI module that can be configured as a master or slave device to enable SPI communication with the outside.

11.2 Main Features

- SPI standard protocol, master/slave mode configurable
- MISO, MOSI, single-wire transmission, half-duplex / full-duplex transmission, big-endian / little-endian configuration
- Programmable clock polarity and phase
- TX_ONLY mode transmission supported
- PCB board delay compensation function; combined logic filtering function of SSN/SCK/MOSI input signal in slave mode

11.3 Register Description

SPI register base address: 0xC400

Table 11-1: List of Registers

Address	Name	Description
0xC400	SPI_CR1	Control register 1
0xC401	SPI_CR2	Control register 2
0xC402	SPI_CR3	Control register 3
0xC403	SPI_CR4	Control register 4
0xC404	SPI_IE	Interrupt enable register
0xC405	SPI_SR	Status register
0xC406	SPI_TXBUF	Transmit data register
0xC407	SPI_RXBUF	Receive buffer register

11.3.1 SPI_CR1 Control Register

C400H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_CR1	BR[2:0]			SSN_MODE	LSBFIRST	MSTR	CPOL	CPHA
R/W	R/W			R/W	R/W	R/W	R/W	R/W
Reset value	1			0	0	1	0	0

Bit No.	Bit Designator	Description
7-5	BR[2:0]	Baud rate configuration bit in master mode: 000: f_{PCLK} / 2 001: f_{PCLK} / 4 010: f_{PCLK} / 8 011: f_{PCLK} / 16 100: f_{PCLK} / 32 101: f_{PCLK} / 64 110: f_{PCLK} / 128 111: f_{PCLK} / 256 These bits shall not be modified while communication is ongoing.
4	SSN_MODE	In Master mode, SSN_MODE indicates whether SSN will be pulled high after each 8-bit transmission: 0: when 8 bits have been transmitted in the case of TXBUF being non-empty, if WAIT_CNT is not 0, SSN is pulled high after waiting for 1 + WAIN_CNT SCL cycles. 1: SSN is pulled high when 8 bits have been transmitted in the case of TXBUF being non-empty.
3	LSBFIRST	Frame format: 0: MSB first (bit7) 1: LSB first (bit0) Note: This bit shall not be modified while communication is ongoing.
2	MSTR	Master/slave mode selection: 0: slave 1: master
1	CPOL	Clock polarity selection: 0: serial clock stops at low level 1: serial clock stops at high level Note: This bit shall not be modified while communication is ongoing.

0	CPHA	Clock phase selection: 0: the first clock transition is the first data capture edge. 1: the second clock transition is the first data capture edge. Note: This bit shall not be modified while communication is ongoing.

Note:

- When ERROR occurs, this register remains unchanged; if SPI is required to be restarted, the software writes SPI_en to 0 first and then to 1.
- If SPI is required to be restarted after changing CPOL and CPHA, the software writes SPI_en to 0 first and then to 1.

11.3.2 SPI_CR2 Control Register

C401H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_CR2	SAMPLE_P	TXONLY_AUTO_CLR	SPI_EN	SSN_MCU_EN	WAIT_CNT		RSV	RSV
R/W	R/W	R/W	R/W	R/W	R/W		R	R
Reset value	0	1	0	0	0		0	0

Bit No.	Bit Designator	Description
7	SAMPLE_P Sdin_sample_mode	Selection of sample position for MISO signal output by slave in Master mode: 0: the sample position is consistent with that in the protocol. 1: for the baud rate lower than f_{PCLK} / 2, the sample position will be delayed by at least one SPICLK cycle, or by half an SPISCK cycle. Note: This bit does not work when the baud rate is f_{PCLK} / 2.
6	TXONLY_AUTO_CLR (original TXONLY_EN)	Enabling of TXONLY hardware auto-clear function: 0: TXONLY hardware auto-clear function disabled 1: TXONLY hardware auto-clear function is enabled. After TXONLY of SPI_CR3 is enabled by software, it will be cleared by hardware upon completion of the transmission.
5	SPI_EN	SPI enable (disabled by turning off the clock): 0: SPI disabled. Reset the status and clear TXBUF and RXBUF. 1: SPI enabled
4	SSN_MCU_EN	Enabling of SSN port controlled by software in master mode: 1: SSN output controlled by software enabled 0: SSN output controlled by internal hardware

3-2	WAIT_CNT	In master mode, the waiting time for transmitting the next 8-bit data after each 8-bit transmission: 00: no wait 01: waiting for 2 SCK cycles 10: waiting for 3 SCK cycles 11: waiting for 4 SCK cycles
1-0	RSV	Reserved

Note: When ERROR occurs, this register remains unchanged; if SPI is required to be restarted, the software writes SPI_en to 0 first and then to 1.

11.3.3 SPI_CR3 Control Register

C402H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_CR3	SCK_EN	MOSI_EN	MISO_EN	CS_EN	TX_ONLY	SSN_MCU	Signal_filter	send_p
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	0	1	1	0

Bit No.	Bit Designator	Description
7	SCK_EN	P1_3 is enabled as SPI_SCK signal. When this bit is 1 and SPI_EN is 1, P1_3 acts as the SPI_SCK signal. Note: This bit is not required to be set when P0_1 / P0_4 / P2_3 are configured as SPI_SCK.
6	MOSI_EN	P1_4 is enabled as SPI_MOSI signal. When this bit is 1 and SPI_EN is 1, P1_4 acts as the SPI_MOSI signal. Note: This bit is not required to be set when P2_0 / P2_3 / P2_7 are configured as SPI_MOSI.
5	MISO_EN	P1_5 is enabled as SPI_MISO signal. When this bit is 1 and SPI_EN is 1, P1_5 acts as the SPI_MISO signal. Note: This bit is not required to be set when P1_1 / P2_2 / P2_6 are configured as SPI_MISO.
4	CS_EN	P0_3 is enabled as SPI_CSN signal. When this bit is 1 and SPI_EN is 1, P0_3 acts as the SPI_CSN signal. Note: This bit is not required to be set when P0_0 / P2_2 / P2_5 are configured as SPI_CSN.

3	TX_ONLY	Enable of TX_ONLY for SPI: 0: TX_ONLY mode disabled 1: TX_ONLY mode enabled
2	SSN_MCU	In Master mode, if SSN_MCU_EN = 1, MCU can control SSN output port by writing this bit: 0: SSN is written to 0 by software. 1: SSN is written to 1 by software.
1	Signal_filter	Whether to digitally filter glitches that may occur on SSN/SCK/MOSI: 0: no 1: yes
0	send_p Sdout_send_m ode	In Slave mode, the transmit start point clock is used for the MOSI signal output from slave: 0: transmit at the time in protocol 1: transmit half a cycle in advance

Note: When ERROR occurs, this register remains unchanged; if SPI is required to be restarted, the software writes SPI_en to 0 first and then to 1.

11.3.4 SPI_CR4 Control Register

C403H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_CR4	RSV	RSV	RSV	RSV	CLR_ TXBUF	CLR_RXBUF	RSV	RSV
R/W	R	R	R	R	W	W	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-4	RSV	Reserved
3	CLR_TXBUF	Writing 1 clears all the contents of TX_BUF and the TXBUF_EMPTY flag bit, and it will be automatically reset to 0 by hardware.
2	CLR_RXBUF	Writing 1 clears all the contents of RX_BUF and the RXBUF_FULL flag bit, and it will be automatically reset to 0 by hardware.
1-0	RSV	Reserved

Note: When ERROR occurs, the TXONLY bit will be automatically reset to 0 by hardware. Therefore, this register shall be reconfigured when SPI is restarted.

11.3.5 SPI_IE Interrupt Enable Register

Interrupts are generated in the interrupt enable register based on the enable of this register:

C404H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_IE	RSV	RSV	RSV	RSV	RSV	ERROR_IE	TX_E_IE	RX_NE_IE
R/W	R	R	R	R	R	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-3	RSV	Reserved
2	ERROR_IE	Error interrupt enable, including TXBUF/RXBUF overflow and master/slave error: 0: disabled 1: enabled
1	TX_E_IE	TXBUF empty interrupt enable: 0: disabled 1: enabled
0	RX_NE_IE	RXBUF non-empty interrupt enable: 0: disabled 1: enabled

11.3.6 SPI_SR Status Register

C405H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_SR	RSV	RSV	RSV	RXBUF_WCOL	TXBUF_WCOL	BUSY	TXBUF_EMPTY	RXBUF_FULL
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	1	0

Bit No.	Bit Designator	Description
7-5	RSV	Reserved
4	RXBUF_WCOL	RX buffer overflow, software clears this bit by writing 0.
3	TXBUF_WCOL	When TXBUF is full, MCU writes TXBUF: 1: conflict 0: no conflict Software writing 0 clears this bit.

2	BUSY	Indication: 0: TXBUF is empty and SPI is not transmitting data. 1: TXBUF is not empty, or SPI is transmitting data. Based on SSN and TXBUF in Slave mode, SSN is high, TXBUF is empty and BUSY signal is 0.
1	TXBUF_EMPTY	Writing TXBUF clears this flag bit: 0: there is data to be sent in TXBUF. 1: there is no data in TXBUF and it is writable.
0	RXBUF_FULL	Reading RXBUF clears this flag bit: 0: there is no data in RXBUF. 1: there is data in RXBUF.

11.3.7 SPI_TXBUF Transmit Data Register

MCU writes the data to be sent into this register to achieve the purpose of caching the sent data into SPI_TXBUF. SPI_TXBUF has no actual register and only supports write operation.

C406H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_TXBUF	WR_DATA							
R/W	W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	WR_DATA	When this register is written, if SPI_TXBUF is full, an overflow interrupt will be generated.

11.3.8 SPI_RXBUF Receive Buffer Register

The data received via SPI will be cached into SPI_RXBUF first, and one byte of data received by SPI will be written to SPI_RXBUF. By reading this register, MCU can obtain the data received by SPI.

C407H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SPI_RXBUF	D7-D0							
R/W	R							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	D7-D0	Based on the non-empty interrupt of SPI_RXBUF, MCU reads the data inside SPI_RXBUF by performing a read operation to this address.

11.4 Software Operation Process

11.4.1 Master Transmit

1. Configure SPI 4-wire pin alternate function (via related bits in SPI_CR3 register or IO_CFG register).
2. Write the MSTR bit in SPI_CR1 register to enable master mode.
3. If software control of SSN signal is required, write the SSN_MCU_EN bit in SPI_CR2 register.
4. Write the BR bit in SPI_CR1 register to configure the baud rate of SPI communication.
5. Write the CPOL and CPHA bits in SPI_CR1 register to configure SPI communication mode.
6. If interrupt is to be used, write the EAL and ESPI bits first, then write SPI_IE register to enable the TX_E_IE interrupt.
7. Write SPI_SR register to clear all interrupt flag bits. Write SPI_CR4 register to clear the contents of TX_BUF.
8. Write the SPI_EN bit in SPI_CR2 register to enable SPI.
9. Write the SSN_MCU bit in SPI_CR3 register to 0 to start the SPI communication.

10. When TXBUF_EMPTY in SPI_SR register is 1, put the data to be sent to the slave into SPI_TXBUF register.
11. If more than one byte of data is to be sent, repeat step 10. After transmitting all the data, wait for the BUSY bit in SPI_SR register to change from 1 to 0, and then end the transmission.
12. Write the SSN_MCU bit in SPI_CR3 register to 1 to end the SPI communication.

11.4.2 Master Receive

1. Configure SPI 4-wire pin alternate function (via related bits in SPI_CR3 register or IO_CFG register).
2. Write the MSTR bit in SPI_CR1 register to enable master mode.
3. If software control of SSN signal is required, write the SSN_MCU_EN bit in SPI_CR2 register.
4. Write the BR bit in SPI_CR1 register to configure the baud rate of SPI communication.
5. Write the CPOL and CPHA bits in SPI_CR1 register to configure SPI communication mode.
6. Write the EAL and ESPI bits first, then write the SPI_IE register to enable the RX_NE_IE interrupt.
7. Write SPI_SR register to clear all interrupt flag bits. Write SPI_CR4 register to clear the contents of TX_BUF.
8. Write the SPI_EN bit in SPI_CR2 register to enable SPI.
9. Write the SSN_MCU bit in SPI_CR3 register to 0 to start the SPI communication.
10. When TXBUF_EMPTY in SPI_SR register is 1, put the data to be sent to the slave into SPI_TXBUF register. Then wait for the trigger upon RX_BUF non-empty interrupt, read the RXBUF register for the received data sent from slave.
11. If more than one byte of data is to be received, repeat step 10. After receiving all the data, wait for the BUSY bit in SPI_SR register to change from 1 to 0, and then end the reception.
12. Write the SSN_MCU bit in SPI_CR3 register to 1 to end the SPI communication.

11.4.3 Slave Transmit

1. Configure SPI 4-wire pin alternate function (via related bits in SPI_CR3 register or IO_CFG

register).

2. Write the MSTR bit in SPI_CR1 register to enable slave mode.
3. Write the CPOL and CPHA bits in SPI_CR1 register to configure SPI communication mode.
4. If interrupt is to be used, write the EAL and ESPI bits first, then write the SPI_IE register to enable the TX_E_IE interrupt.
5. Write SPI_SR register to clear all interrupt flag bits. Write SPI_CR4 register to clear the contents of TX_BUF.
6. Write the SPI_EN bit in SPI_CR2 register to enable SPI.
7. Set the TX_ONLY bit in SPI_CR3 register to enable the single-transfer mode.
8. Upon trigger by TX_E_IE interrupt, or after querying the TXBUF_EMPTY status bit in SPI_SR register, write the data to be sent into RXBUF register, query the BUSY bit in SPI_SR register and wait for TXBUF to be empty.
9. Repeat step 8 if the slave is to transmit more than one byte of data.

11.4.4 Slave Receive

1. Configure SPI 4-wire pin alternate function (via related bits in SPI_CR3 register or IO_CFG register).
2. Write the MSTR bit in SPI_CR1 register to enable slave mode.
3. Write the CPOL and CPHA bits in SPI_CR1 register to configure SPI communication mode.
4. If interrupt is to be used, write the EAL and ESPI bits first, then write SPI_IE register to enable the RX_NE_IE interrupt.
5. Write SPI_SR register to clear all interrupt flag bits. Write SPI_CR4 register to clear the contents of TX_BUF.
6. Write the SPI_EN bit in SPI_CR2 register to enable SPI.
7. Upon trigger by RX_NE_IE interrupt, or after querying the RXBUF_FULL status bit in SPI_SR register, read the data in RXBUF register.

12 LPTIMER (Low-power Timer)

12.1 Overview

LPTIMER is a 16-bit low-power timer/counter module running in Always-on power domain. Thanks to its diversity of clock sources, LPTIMER is able to keep running in all low-power modes with extremely low power consumption. Given its capability to run even with no internal clock source, LPTIMER can be used as a “Pulse Counter” in low-power mode. Also, in combination with an external input trigger signal, LPTIMER is able to realize “Timeout function” regarding waking up the system from low-power modes.

12.2 Main Features

- 16-bit upcounter
- 3-bit prescaler with 8 possible dividing factors (1, 2, 4, 8, 16, 32, 64, 128)
- Selectable clock source:

  Internal clock sources: LSCLK, RCLP, system clock

  External clock source: LPTIN (with analog filter)
- 16-bit compare register
- 16-bit target value register
- Selectable software/hardware input trigger
- 2 × input capture
- Configurable input polarity
- External pulse counting with no clock source
- Externally triggered timeout wakeup from low-power modes
- 2 × 16-bit PWM

12.3 Structure Diagram

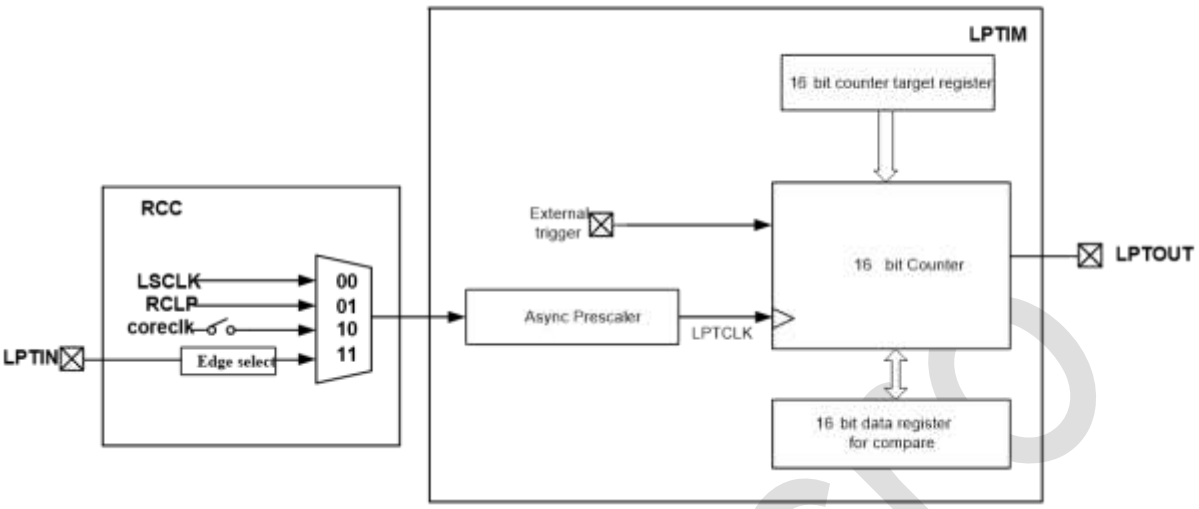


Figure 12-1: Structure Diagram

12.4 Operation Mode

12.4.1 General-purpose Timer

- Work with internal or external clock input
- There is a synchronization process of two counting clocks when enabled.
- Start working without trigger after enabled.

12.4.2 Pulse-trigger Counting

- Work with internal clock
- Internal clock samples the asynchronous trigger signal from the external input
- Possible to count the rising, falling or both edges of trigger
- There is a synchronization process of two counting clocks when enabled.

12.4.3 External Asynchronous Pulse Counting

- External input pulse directly used as the counting clock
- Input polarity configurable for rising-edge or falling-edge counting
- Trigger not required
- No synchronization process after enabled

12.4.4 Timeout Mode

- Work with internal or external clock input
- Sampling asynchronous trigger signal from the external input
- The counter is started at the first trigger, and is cleared and restarted when a trigger is sampled after the start.
- If there is no new trigger before the counter overflows, an overflow interrupt will be generated, the counting will be stopped, and the enable will be cleared.
- There is a synchronization process of two counting clocks when enabled.

12.4.5 Counting Mode

There are two counting modes for LPTIMER:

- Continuous counting mode: the counter keeps running after being triggered until it is turned off. After the counter reaches the target value, it returns to 0 to restart counting, and an overflow interrupt is generated.
- One-shot counting mode: after the counter is triggered, it counts to the target value, then returns to 0 and automatically stops, generating an overflow interrupt.

12.4.6 Externally Triggered Timeout Wakeup

LPTIMER can be enabled by an external trigger signal or by a software trigger. In timeout mode, the effective edge of the first external trigger input will start the counter, while the subsequent trigger signal will clear the counter. If there is no effective trigger signal before the counter reaches the comparison value, a timeout interrupt is generated to wake up MCU.

The effective edge of the external trigger signal that considered as an asynchronous input can be configured by register, so the effective edge has a latency of at least two counting clocks for sampling and judgment.

12.4.7 16-bit PWM

LPTIMER starts counting from 0x0000 after the PWM mode is enabled, and the output goes high when the count value is equal to the comparison value while goes low when the count value is equal to the final value. The PWM period is determined by the final value register and the duty cycle is determined by the compare value register.

12.5 Register Description

LPTIMER register base address: 0xC800

Table 12-1: List of Registers

Address	Name	Description
0xC800	LPTIMER_CFG0	LPTIMER configuration register 0
0xC801	LPTIMER_CFG1	LPTIMER configuration register 1
0xC802	LPTIMER_CNT_L	LPTIMER count value low-order register
0xC803	LPTIMER_CNT_H	LPTIMER count value high-order register
0xC804	LPTIMER_CMP1_L	LPTIMER capture / comparison value 1 low-order register
0xC805	LPTIMER_CMP1_H	LPTIMER capture / comparison value 1 high-order register
0xC806	LPTIMER_TARGET_L	LPTIMER target value low-order register
0xC807	LPTIMER_TARGET_H	LPTIMER target value high-order register
0xC808	LPTIMER_IE	LPTIMER interrupt enable register
0xC809	LPTIMER_IF	LPTIMER interrupt flag register
0xC80A	LPTIMER_CTRL	LPTIMER control register
0xC80B	LPTIMER_CCMCFG1	LPTIMER capture channel 1 control register
0xC80C	LPTIMER_CCMCFG2	LPTIMER capture channel 2 control register
0xC80D	LPTIMER_CMP2_L	LPTIMER capture / comparison value 2 low-order register
0xC80E	LPTIMER_CMP2_H	LPTIMER capture / comparison value 2 high-order register
0xC811	LPTIMER_LOAD	Auto-load register
0xC812	LPTIMER_BUFFER_L	Load buffer low-order register
0xC813	LPTIMER_BUFFER_H	Load buffer high-order register

12.5.1 LPTIMER_CFG0 Register

C800H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CFG0	TRIGCFG		EDGESEL	CLKSEL		DIVSEL		
R/W	R/W		R/W	R/W		R/W		
Reset value	0		0	1	0	0		

Bit No.	Bit Designator	Description
7-6	TRIGCFG	External trigger edge selection: 00: rising-edge trigger for external input signal 01: falling-edge trigger for external input signal 10/11: rising/falling-edge trigger for external input signal These bits shall not be modified while communication is ongoing.
5	EDGESEL	LPTIM input edge selection: 0: LPTIM rising edge count 1: LPTIM falling edge count
4-3	CLKSEL	Clock source selection: 00: LSCLK as counter clock (38-kHz RCL) 01: RCLP as counter clock (1-Hz clock divided from the 38-kHz RCL) 10: gating clock PCLK as counter clock 11: LPTIN as counter clock
2:0	DIVSEL	Counter clock division selection: 000: divided by 1 001: divided by 2 010: divided by 4 011: divided by 8 100: divided by 16 101: divided by 32 110: divided by 64 111: divided by 128

12.5.2 LPTIMER_CFG1 Register

C801H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CFG1	EXTRIGGER_IO_IEN	LPOUT_IO_IEN	LPTIN_IO_IEN	TMODE		MODE	PWM	POLARITY
R/W	R/W	R/W	R/W	R/W		R/W	R/W	R/W
Reset value	0	0	0	0		0	0	0

Bit No.	Bit Designator	Description
7	EXTRIGGER_IO_IEN	1: P1_1 configured as EXTRIGGER_IO_IEN Note: This bit is not required to be set when P1_4 is configured as EXTRIGGER_IO.
6	LPOUT_IO_IEN	1: P0_3 configured as LPOUT1 Note: This bit is not required to be set when P0_0 and P2_0 are configured as LPTOUT1.
5	LPTIN_IO_IEN	1: P1_0 configured as LPTIN Note: This bit is not required to be set when P1_3 is configured as LPTIN.
4-3	TMODE	Operation mode selection: 00: general timer mode with waveform output 01: pulse-trigger counting mode 10: external asynchronous pulse counting mode 11: timeout mode
2	MODE	Counting mode: 0: continuous counting mode: the counter keeps running after being triggered until it is turned off. After the counter reaches the target value, it returns to 0 to restart counting, and an overflow interrupt is generated. 1: one-shot counting mode: after being triggered, the counter counts to the target value, then returns to 0 and automatically stops, generating an overflow interrupt.
1	PWM	Pulse width modulation mode: 0: periodic square wave output 1: PWM output
0	POLARITY	Counter clock division selection: 0: positive waveform, that is, when the first count value = the comparison value, the rising edge of the output waveform is generated.

		1: negative waveform, that is, when the first count value = the comparison value, the falling edge of the output waveform is generated.

12.5.3 LPTIMER_CNT Count Value Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CNT_L (C802H)	CNT16[7:0]							
LPTIMER_CNT_H (C803H)	CNT16[15:8]							
R/W	R							
Reset value	0							

Bit No.	Bit Designator	Description
15-0	CNT16	Counter value

12.5.4 LPTIMER_CMP1 Compare Value Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CMP1_L (C804H)	COMPARE_REG[7:0]							
LPTIMER_CMP1_H (C805H)	COMPARE_REG[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
15-0	COMPARE_REG	Capture / compare value register 1 Reading the LPTIMER_CMP1_H register clears the COMPIF flag bit.

12.5.5 LPTM_TARGET Target Value Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_TARGET_L (C806H)	TARGET_REG[7:0]							
LPTIMER_TARGET_H (C807H)	TARGET_REG[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
15-0	TARGET_REG	Target value register

12.5.6 LPTIMER_IE Interrupt Enable Register

Interrupts are generated in the interrupt enable register based on the enable of this register:

C808H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_IE	RSV				COMP2IE	TRIGIE	OVIE	COMPIE
R/W	R				R/W	R/W	R/W	R/W
Reset value	0				0	0	0	0

Bit No.	Bit Designator	Description
7-4	RSV	Reserved
3	COMP2IE	Compare matching 2 interrupt enable bit: 1: interrupt enabled at matching 2 of the counter value and the compare value 0: interrupt disabled at matching 2 of the counter value and the compare value
2	TRIGIE	External trigger arrival interrupt enable bit: 1: enabled 0: disabled
1	OVIE	Counter overflow interrupt enable bit: 1: enabled 0: disabled

0	COMPIE	Compare matching 1 interrupt enable bit: 1: interrupt enabled at matching 1 of the counter value and the compare value 0: interrupt disabled at matching 1 of the counter value and the compare value

12.5.7 LPTIMER_IF Interrupt Flag Register

C809H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_IF	RSV				COMP2IF	TRIGIF	OVIF	COMPIF
R/W	R				R/W	R/W	R/W	R/W
Reset value	0				0	0	0	0

Bit No.	Bit Designator	Description
7-4	RSV	Reserved
3	COMP2IF	Capture / compare matching 2 interrupt enable bit (can be cleared by writing 1): 1: interrupt generated at matching 2 of the counter value and the compare value 0: no interrupt generated
2	TRIGIF	External trigger arrival interrupt enable bit (can be cleared by writing 1): 1: external trigger arrival interrupt generated 0: no interrupt generated
1	OVIF	Counter overflow interrupt enable bit (can be cleared by writing 1): 1: counter overflow interrupt generated 0: no interrupt generated
0	COMPIF	Capture / compare matching 1 interrupt enable bit (can be cleared by writing 1): 1: interrupt generated at matching 1 of the counter value and the compare value 0: no interrupt generated

12.5.8 LPTIMER_CTRL Control Register

C80AH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CTRL	RSV						CAP1SRSEL	LPTEN
R/W	R						R/W	R/W
Reset value	0						0	0

Bit No.	Bit Designator	Description
7-2	RSV	Reserved
1	CAP1SRSEL	Channel 1 capture source selection: 0: LPT_CAP1 input 1: RCLP (1-Hz clock divided from the 38-kHz RCL)
0	LPTEN	LPTIMER enable: 1: counter counting enabled 0: counter counting disabled

12.5.9 LPTIMER_CCMCFG1 Control Register

C80BH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CCMCFG1	RSV		CAP1EDGE		RSV		CC1S	CC1E
R/W	R		R/W		R		R/W	R/W
Reset value	0		0		0		0	0

Bit No.	Bit Designator	Description
7-6	RSV	Reserved
5-4	CAP1EDGE	Channel 1 capture edge selection: 00: rising edge capture 01: falling edge capture 10: rising or falling edge capture 11: undefined
3-2	-	-
1	CC1S	Channel 1 capture / compare selection: 0: channel 1 configured as output 1: channel 1 configured as input
0	CC1E	Channel 1 capture / compare enable: 0: channel 1 capture / compare enabled 1: channel 1 capture / compare disabled

12.5.10 LPTIMER_CCMCFG2 Control Register

C80CH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CCMCFG2	RSV		CAP2EDGE		RSV	CC2P	CC2S	CC2E
R/W	R		R/W		R	R/W	R/W	R/W
Reset value	0		0		0	0	0	0

Bit No.	Bit Designator	Description
7-6	RSV	Reserved
5-4	CAP2EDGE	Channel 2 capture edge selection: 00: rising edge capture 01: falling edge capture 10: rising or falling edge capture 11: undefined
3	RSV	Reserved
2	CC2P	Channel 2 output polarity selection: 0: low when CNT<= CCR2 and high when CNT > CCR2 1: high when CNT<= CCR2 and low when CNT > CCR2
1	CC2S	Channel 2 capture / compare selection: 0: channel 2 configured as output 1: channel 2 configured as input
0	CC2E	Channel 2 capture / compare enable: 0: channel 2 capture / compare enabled 1: channel 2 capture / compare disabled

12.5.11 LPTIMER_CMP2 Compare Value Register

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_CMP2_L (C80DH)	COMPARE2_REG[7:0]							
LPTIMER_CMP2_H (C80EH)	COMPARE2_REG[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
15-0	COMPARE2_REG	Capture / compare value register 2 Reading the LPTIMER_CMP2_H register clears the COMP2F flag bit.

12.5.12 LPTIMER_LOAD Auto-load Register

C811H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LPTIMER_LOAD	RSV							LPTEN
R/W	R							R/W
Reset value	0							0

Bit No.	Bit Designator	Description
7-1	RSV	Reserved
0	LPTIMER_LOAD	When the software writes LPTIMER_LOAD = 1, the hardware puts the current count value into the buffer register.

12.5.13 LPTIMER_BUFFER Count Value Load Register

	Bit7 Bit6 Bit5 Bit4 Bit3 Bit2 Bit1 Bit0
LPTIMER_BUFFER_L (C812H)	BUFFER[7:0]
LPTIMER_BUFFER_H (C813H)	BUFFER[15:8]
R/W	R/W
Reset value	0

Bit No.	Bit Designator	Description
15-0	BUFFER	When the software writes LPTIMER_LOAD register, the hardware puts the current count value into the buffer register.

12.6 Software Operation Process

1. Select the clock source, set the clock division value, and set the operation mode and counting mode.
2. Set the value of the high-low order compare value register.
3. Set the value of the high-low order target value register.
4. Enable the interrupt flag bit.
5. Enable the LPTEN bit to start the counter.

12.6.1 General-purpose Timer

1. Configure LPTIMER_CFG.CLKSEL to select the clock source.
2. Configure LPTIMER_CFG.DIV to set the clock division value.
3. Configure LPTIMER_CFG.MODE to set the counting mode.
4. Configure LPTIMER_CFG.TMODE to select the general timer mode.
5. Configure the target value register LPTIMER_TARGET.
6. Enable the interrupt register LPTIMER_IE to select overflow interrupt.
7. Enable the LPTIMER_CTRL.LPTEN bit to start the counter.

12.6.2 PWM Output

1. Configure LPTIMER_CFG.CLKSEL to select the clock source.
2. Configure LPTIMER_CFG.DIV to set the clock division value.
3. Configure LPTIMER_CFG.MODE to set the counting mode.
4. Configure LPTIMER_CFG.PWM to select the PWM output mode.
5. Configure LPTIMER_CFG.POLARITY to select the waveform polarity.
6. Configure LPTIMER_CFG.TMODE to select the general timer mode.

7. Configure LPTIMER_CMP to set the compare register value.
8. Configure LPTIMER_CCMCFG to configure the corresponding channel compare output to enable the compare function.
9. Configure the target value register LPTIMER_TARGET.
10. Enable the interrupt register LPTIMER_IE to activate the interrupt.
11. Enable the LPTIMER_CTRL.LPTEN bit to start the counter.

12.6.3 Pulse-trigger Counting Mode

1. Configure LPTIMER_CFG.CLKSEL to select the clock source.
2. Configure LPTIMER_CFG.DIV to set the clock division value.
3. Configure LPTIMER_CFG.MODE to set the counting mode.
4. Configure LPTIMER_CFG.TRIGCFG to set the external trigger edge.
5. Configure LPTIMER_CFG.TMODE to select the pulse-trigger counting mode.
6. Enable the LPTIMER_IE.TRIGIE interrupt register to activate the external trigger interrupt.
7. Enable the LPTIMER_CTRL.LPTEN bit to start the counter.

12.6.4 External Asynchronous Pulse Counting Mode

1. Configure LPTIMER_CFG.CLKSEL to select the clock source.
2. Configure LPTIMER_CFG.DIV to set the clock division value.
3. Configure LPTIMER_CFG.MODE to set the counting mode.
4. Configure LPTIMER_CFG.EDGESEL to set the LPTIN input edge.
5. Configure LPTIMER_CFG.TMODE to select the external asynchronous pulse counting mode.
6. Configure the target value register LPTIMER_TARGET.
7. Enable the interrupt register LPTIMER_IE to activate the interrupt.
8. Enable the LPTIMER_CTRL.LPTEN bit to start the counter.

12.6.5 Timeout Mode

1. Configure LPTIMER_CFG.CLKSEL to select the clock source.
2. Configure LPTIMER_CFG.DIV to set the clock division value.
3. Configure LPTIMER_CFG.MODE to set the counting mode.
4. Configure LPTIMER_CFG.TRIGCFG to set the external trigger edge.
5. Configure LPTIMER_CFG.TMODE to select the timeout mode.
6. Configure the target value register LPTIMER_TARGET.
7. Enable the interrupt register LPTIMER_IE to activate the overflow interrupt.
8. Enable the LPTIMER_CTRL.LPTEN bit to start the counter.

Note: If there is no new trigger before the counter overflows, an overflow interrupt will be generated, the counting will be stopped, and the enable will be cleared. If it is to be reused, the interrupt shall be enabled again.

12.6.6 Input Capture

1. Configure LPTIMER_CFG.CLKSEL to select the clock source.
2. Configure LPTIMER_CFG.DIV to set the clock division value.
3. Configure LPTIMER_CFG.MODE to set the counting mode.
4. Configure LPTIMER_CFG.POLARITY to select the waveform polarity.
5. Configure LPTIMER_CFG.TMODE to select the general timer mode.
6. Configure LPTIMER_CCMCFG to configure the corresponding channel capture input to enable the capture function.
7. Configure the target value register LPTIMER_TARGET.
8. Enable the interrupt register LPTIMER_IE to activate the interrupt.
9. Enable the LPTIMER_CTRL.LPTEN bit to start the counter.
10. The count value at the time of capture can be obtained by reading the value of compare register LPTIMER_CMP in capture interrupt.

13 GTIMER

13.1 Main Features

- 16-bit up, down, up/down auto-reload counter
- 16-bit programmable prescaler allowing real-time adjustment of the counter clock division
- Flexible selection of counter clock source
- Channels for input capture, output compare, PWM generation (edge- or center-aligned mode), and one-pulse output mode
- Supporting cascade between timers

13.2 Structure Diagram

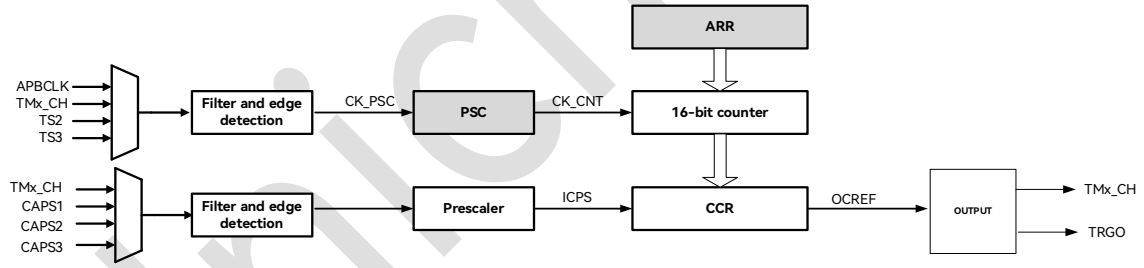


Figure 13-1: Structure Diagram

13.3 Register Description

- GTIMER0 register base address: 0xC900
- GTIMER1 register base address: 0xCA00
- GTIMER2 register base address: 0xCB00

Table 13-1: Register Description

Offset	Name	Description
0x00	GTIMER_CR0	GTIMER control register 0
0x01	GTIMER_CR1	GTIMER control register 1

Offset	Name	Description
0x02	GTIMER_CR2	GTIMER control register 2
0x03	GTIMER_CR3	GTIMER control register 3
0x04	GTIMER_IER	GTIMER interrupt enable register
0x05	GTIMER_SR	GTIMER status register
0x06	GTIMER_EGR	GTIMER event generation register
0x07	GTIMER_CCMR0	GTIMER capture/compare mode register 0
0x08	GTIMER_CCMR1	GTIMER capture/compare mode register 1
0x09	GTIMER_CCER	GTIMER capture/compare enable register
0x0A	GTIMER_CNT0	GTIMER counter register 0
0x0B	GTIMER_CNT1	GTIMER counter register 1
0x0C	GTIMER_PSC0	GTIMER prescaler register 0
0x0D	GTIMER_PSC1	GTIMER prescaler register 1
0x0E	GTIMER_ARR0	GTIMER auto-reload register 0
0x0F	GTIMER_ARR1	GTIMER auto-reload register 1
0x10	GTIMER_ARR2	GTIMER auto-reload register 2
0x11	GTIMER_ARR3	GTIMER auto-reload register 3
0x12	GTIMER_CCR0	GTIMER capture/compare register 0
0x13	GTIMER_CCR1	GTIMER capture/compare register 1
0x14	GTIMER_CCR2	GTIMER capture/compare register 2
0x15	GTIMER_CCR3	GTIMER capture/compare register 3

13.3.1 GTIMER_CR0 Control Register

C900H/CA00H/CB00H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CR0	MMS	ARPE	CMS		CEN_ALL_EN	DIR	OPM	CEN
R/W	R/W	R/W	R/W		R/W	R/W	R/W	R/W
Reset value	0	0	0		0	0	0	0

Bit No.	Bit Designator	Description
7	MMS	Master mode selection for configuring the source of synchronization trigger signal (TRGO) sent to slave in master mode: 1: UE (update event) signal used as TRGO 0: OCxREF used as TRGO
6	ARPE	Auto-preoad enable: 0: disabled 1: enabled

5-4	CMS	Counter alignment mode selection: 00: edge-aligned mode 01: center-aligned mode 1: output compare interrupt flag is set only when the counter is counting down. 10: center-aligned mode 2: output compare interrupt flag is set only when the counter is counting up. 11: center-aligned mode 3: output compare interrupt flag is set both when the counter is counting up or down.
3	CEN_ALL_EN	CEN_ALL enable: 1: current GTIMER can be controlled by CEN_ALL. 0: current GTIMER is not available to CEN_ALL signal.
2	DIR	Counting direction register: 0: count up 1: count down Note: This register is read-only when the timer is configured in center-aligned mode.
1	OPM	One-pulse mode output: 0: the counter does not stop at the occurrence of update event. 1: the counter stops at the occurrence of update event (CEN, clear automatically).
0	CEN	Counter enable: 0: disabled 1: enabled

13.3.2 GTIMER_CR1 Control Register

C901H/CA01H/CB01H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CR1	PWMS_B_S		SOFT_BK	CEN_ALL	BKE_POL	BKE	PWM_DEAD	PWM_INV
R/W	R/W		R/W	W	R/W	R/W	R/W	R/W
Reset value	0		0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-6	PWMS_B_S	Setting bit of PWM positive level status after PWM break trigger: 00: low level 01: high level 10/11: high-impedance state
5	SOFT_BK	Setting bit for software-triggered break function:

		1: software triggers the break function. 0: software does not trigger the break function.
4	CEN_ALL	1: GTIMER0/1/2 are enabled simultaneously, and after writing this bit the CEN bits of GTIMER0/1/2 are 1 at the same time. 0: no operation This bit is always read as 0.
3	BKE_POL	Configuration of break signal polarity: 1: break signal active low 0: break signal active high
2	BKE	Break function enable: 1: enabled 0: disabled
1	PWM_DEAD	PWM dead-time insertion enable: 0: disabled 1: enabled
0	PWM_INV	Difference enable for complementary PWM and original PWM: 0: in the same phase 1: in opposite phases

13.3.3 GTIMER_CR2 Control Register

C902H/CA02H/CB02H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CR2	-	BREAK2_SEL	BREAK1_SEL	PWMN_IDLE	PWMP_IDLE	MOE	PWMN_B_S	
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	
Reset value	0	0	0	0	0	0	0	

Bit No.	Bit Designator	Description
7	-	RSV: not implemented, read as 0
6	BREAK2_SEL	**GTIMER0** Selection of IO pin break source: 1: IO configured by GTIMER2 selected as the break source 0: IO configured by GTIMER2 not selected as the break source **GTIMER1** Selection of IO pin break source: 1: IO configured by GTIMER2 selected as the break source 0: IO configured by GTIMER2 not selected as the break source

		GTIMER2 Selection of IO pin break source: 1: IO configured by GTIMER1 selected as the break source 0: IO configured by GTIMER1 not selected as the break source
5	BREAK1_SEL	**GTIMER0** Selection of IO pin break source: 1: IO configured by GTIMER1 selected as the break source 0: IO configured by GTIMER1 not selected as the break source **GTIMER1** Selection of IO pin break source: 1: IO configured by GTIMER0 selected as the break source 0: IO configured by GTIMER0 not selected as the break source **GTIMER2** Selection of IO pin break source: 1: IO configured by GTIMER0 selected as the break source 0: IO configured by GTIMER0 not selected as the break source
4	PWMN_IDLE	PWM output in negative-level idle state: 1: output high level 0: output low level
3	PWMP_IDLE	PWM output in positive-level idle state: 1: output high level 0: output low level
2	MOE	Output enable: 1: always enabled 0: disabled
1-0	PWMN_B_S	Setting bit of PWM complementary level status after PWM break trigger: 00: low level 01: high level 10/11: high-impedance state

13.3.4 GTIMER_CR3 Control Register

C903H/CA03H/CB03H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CR3	-					COMP2_BKEN	COMP1_BKEN	COMP0_BKEN
R/W	R					R/W	R/W	R/W
Reset value	0					0	0	0

Bit No.	Bit Designator	Description
7-3	--	RSV: not implemented, read as 0
2	COMP2_BKEN	Enable COMP2 as break source: 1: enabled 0: disabled
1	COMP1_BKEN	Enable COMP1 as break source: 1: enabled 0: disabled
0	COMP0_BKEN	Enable COMP0 as break source: 1: enabled 0: disabled

13.3.5 GTIMER_IER Interrupt Enable Register

C904H/CA04H/CB04H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_IER	-					BKE_IE	CC1IE	UIE
R/W	R					R/W	R/W	R/W
Reset value	0					0	0	0

Bit No.	Bit Designator	Description
7-3	--	RSV: not implemented, read as 0
2	BKE_IE	Break interrupt enable: 1: enabled 0: disabled
1	CCIE	Capture/compare channel interrupt enable: 0: capture/compare interrupt disabled 1: capture/compare interrupt enabled
0	UIE	Update event interrupt enable: 0: disabled 1: enabled

13.3.6 GTIMER_SR Status Register

C905H/CA05H/CB05H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_SR	-					BKE_F	CCIF	UIF
R/W	R					R/W	R/W	R/W
Reset value	0					0	0	0

Bit No.	Bit Designator	Description
7-3	--	RSV: not implemented, read as 0
2	BKE_F	Break interrupt flag: 1: in break state 0: not in break state Write 1 to clear this bit.
1	CCIF	Capture/compare channel interrupt flag: If the capture/compare channel is configured as output: CCIF is set when the count value is equal to the compare value (CCR) and is cleared by software writing it to 1. If the capture/compare channel is configured as input: CCIF is set when a capture event occurs, and is cleared by software writing it to 1 or automatically cleared by software reading GTIMER_CCR.
0	UIF	Update event interrupt flag is set by hardware and cleared by software via writing it to 1. When an update event occurs, UIF is set to 1 and the shadow register is updated.

13.3.7 GTIMER_EGR Event Generation Register

C906H/CA06H/CB06H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_EGR	-							UG
R/W	R							W
Reset value	0							0

Bit No.	Bit Designator	Description
7-1	--	RSV: not implemented, read as 0

0	UG	This bit can be set by software to generate an update event, and is automatically cleared by hardware. When the UG bit is set by software, the counter is reinitialized, the shadow register is updated, and the prescaler counter is cleared.

13.3.8 GTIMER_CCMR0 Capture/Compare Mode Register

C907H/CA07H/CB07H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCMR0	TFLT	TEDGE	TSSEL		-			CCS
R/W	R/W	R/W	R/W		R/W			R/W
Reset value	0	0	0		0			0

Bit No.	Bit Designator	Description
7	TFLT	External count source filter enable: 0: no filter function 1: with filter function
6	TEDGE	Counting source edge selection: 0: rising-edge count 1: falling-edge count
5-4	TSSEL	Counting source selection bit: **GTIMER0:** 00: PCLK 01: GTIMER2_TRG0 (synchronization trigger signal of GTIMER2) 10: CLK38K_GTIMER0 (38 kHz clock) 11: GTIMER0_CH (GTIMER0 capture input) **GTIMER1:** 00: PCLK 01: GTIMER2_TRG0 (synchronization trigger signal of GTIMER0) 10: CLK38K_GTIMER1 (38 kHz clock) 11: GTIMER1_CH (GTIMER1 capture input) **GTIMER2** 00: PCLK 01: GTIMER2_TRG0 (synchronization trigger signal of GTIMER1) 10: CLK38K_GTIMER1 (38 kHz clock) 11: GTIMER2_CH (GTIMER2 capture input)
3-1	--	RSV: not implemented, read as 0

0	CCS	Capture/compare 1 channel selection: 0: CC channel configured as output 1: CC channel configured as input Note: CCS bit is writable only when the channel is OFF (i.e. CCE = 0).

13.3.9 GTIMER_CCMR1 Capture/Compare Mode Register

C908H/CA08H/CB08H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCMR1	CAPCLR	ICPSC		CAPFLT	CAPEDGE		CAPSSEL	
R/W	R/W	R/W		R/W	R/W		R/W	
Reset value	0	0		0	0		0	

Bit No.	Bit Designator	Description
7	CAPCLR	First-capture clear control bit: 0: the counter starts counting immediately after it is enabled. 1: the counter remains at 0 after it is enabled and does not start counting until the first edge is captured.
6-5	ICPSC	Capture source prescaler bit: 00: divided by 1 01: divided by 2 10: divided by 4 11: divided by 8
4	CAPFLT	Input capture signal filter enable: 0: no input filter function 1: with input filter function
3-2	CAPEDGE	Control bit of capture edge trigger: 00: rising-edge trigger 01: falling-edge trigger 10/11: rising- or falling-edge trigger
1-0	CAPSSEL	Capture source selection bit: **GTIMER0:** 00: GTIMER0_CH 01: UART0_RX 10: CLK38K_GTIMER0 11: LPTIMER_LPOUT0 **GTIMER1:** 00: GTIMER1_CH

		01: UART1_RX 10: CLK38K_GTIMER1 11: LPTIMER_LPOUT1 **GTIMER2:** 00: GTIMER2_CH 01: UART2_RX 10: CLK38K_GTIMER2 11: UART3_RX

13.3.10 GTIMER_CCER Capture/Compare Enable Register

C909H/CA09H/CB09H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCER	-						CCP	CCE
R/W	R						R/W	R/W
Reset value	0						0	0

Bit No.	Bit Designator	Description
7-3	--	RSV: not implemented, read as 0.
1	CCP	CC channel polarity when it is configured as output: 0: outputs high level if CNT < CCR (CC channel output in phase with OCxREF) 1: outputs high level if CNT > CCR (CC channel output in reversed phase with OCxREF)
0	CCE	Capture/compare output enable: **In the case of CC channel being configured as output:** 0: OC no output 1: OC with output **In the case of CC channel being configured as input:** 0: capture function disabled 1: capture function enabled

13.3.11 GTIMER_CNT0 Counter Register

C90AH/CA0AH/CB0AH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CNT0	CNT[7:0]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	CNT	Count value CNT [7:0]

13.3.12 GTIMER_CNT1 Counter Register

C90BH/CA0BH/CB0BH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CNT1	CNT[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	CNT	Count value CNT [15:8]

13.3.13 GTIMER_PSC0 Prescaler Register

C90CH/CA0CH/CB0CH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_PSC0	PSC[7:0]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	PSC	Counter clock (CK_CNT) prescaler value: $f_{CK\_CNT} = f_{CK\_PSC} / (PSC[15:0] + 1)$

Note: If preload is not enabled, the psc value cannot be loaded into the shadow register until there is an update event.

13.3.14 GTIMER_PSC1 Prescaler Register

C90DH/CA0DH/CB0DH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_PSC1	PSC[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	PSC	Counter clock (CK_CNT) prescaler value: $f_{CK\_CNT} = f_{CK\_PSC} / (PSC[15:0] + 1)$

Note: If preload is not enabled, the psc value cannot be loaded into the shadow register until there is an update event.

13.3.15 GTIMER_ARR0 Auto-reload Register

C90EH/CA0EH/CB0EH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_ARR0	ARR[7:0]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ARR	Auto-reload value in the case of counter overflow This is a preload register whose content are transferred into the shadow register at each update event.

13.3.16 GTIMER_ARR1 Auto-reload Register

C90FH/CA0FH/CB0FH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_ARR1	ARR[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ARR	Auto-reload value in the case of counter overflow This is a preload register whose content are transferred into the shadow register at each update event.

13.3.17 GTIMER_ARR2 Auto-reload Register

C910H/CA10H/CB10H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_ARR2	ARRN[7:0]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ARRN	Auto-reload value in the case of counter overflow, complementary counter This is a preload register whose content are transferred into the shadow register at each update event.

13.3.18 GTIMER_ARR3 Auto-reload Register

C911H/CA11H/CB11H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_ARR3	ARRN[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ARRN	Auto-reload value in the case of counter overflow, complementary counter This is a preload register whose content are transferred into the shadow register at each update event.

13.3.19 GTIMER_CCR0 Capture/Compare Register

C912H/CA12H/CB12H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCR0	CCR[7:0]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	CCR	Capture/compare channel register **If the channel is configured as output:**

		This is a preload register whose content are loaded into the shadow register for comparison with the counter to generate an OC output. **In the case of channel being configured as input:** CCR is the counter value transferred by the last input capture event, at this point the CCR register is read-only.

13.3.20 GTIMER_CCR1 Capture/Compare Register

C913H/CA13H/CB13H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCR1	CCR[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	CCR	Capture/compare channel register **If the channel is configured as output:** This is a preload register whose content are loaded into the shadow register for comparison with the counter to generate an OC output. **In the case of channel being configured as input:** CCR is the counter value transferred by the last input capture event, at this point the CCR register is read-only.

13.3.21 GTIMER_CCR2 Capture/Compare Register

C914H/CA14H/CB14H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCR2	CCRN[7:0]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	CCRM	Capture/compare channel register, complementary counter **If the channel is configured as output:** This is a preload register whose content are loaded into the shadow register for comparison with the counter to generate an OC output.

13.3.22 GTIMER_CCR3 Capture/Compare Register

C915H/CA15H/CB15H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
GTIMER_CCR3	CCRN[15:8]							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	CCRM	Capture/compare channel register, complementary counter **If the channel is configured as output:** This is a preload register whose content are loaded into the shadow register for comparison with the counter to generate an OC output.

13.4 Usage Instructions

13.4.1 Counter Operation Mode

Counting direction:

1. Up-counting mode

   In up-counting mode, the counter counts from 0 to the auto-reload value, then restarts from 0 and generates an interrupt. At this time, the update event (UEV) occurs, and only then will the internal load register of the chip be updated.

2. Down-counting mode

   In down-counting mode, the counter counts from the auto-reload value down to 0, then restarts from the auto-reload value and generates an interrupt. At this time, the update event (UEV) occurs, and only then will the internal load register of the chip be updated.

3. Center-aligned mode (up-/down-counting)

- In center-aligned mode, the counter counts from 0 to the “auto-reload value – 1”, generating an interrupt; next counts from the auto-reload value down to 1, generating an interrupt; and then it restarts counting from 0.
- In this mode, the DIR direction bit in the register cannot be written.
- At each counter overflow and underflow, the UEV occurs, and only then will the internal load register of the chip be updated.

13.4.2 Input Capture Mode

In input capture mode, the capture register (CCR) is used to latch the value of the counter after a transition detected by the corresponding ICx signal. When a capture occurs, the corresponding interrupt flag is set and a capture interrupt can be generated if it is enabled. CCxIF is cleared by software. The transition edge can be controlled by the register as a rising edge or a falling edge. The capture source can be filtered or not filtered.

13.4.3 PWM Mode

PWM mode allows to generate a waveform with a frequency determined by the value of ARR and PSC, as well as a duty cycle determined by the value of CCR.

In up-counting mode, OCxREF is set high when CNT < CCR, otherwise it is set low; in down-counting mode, OCxREF is set low when CNT > CCR, otherwise it is set high.

- PWM edge-aligned mode

  In up-counting mode, when it is configured in PWM mode 1 and CCP is configured to 0, the OCxREF signal is high when CNT < CCR, otherwise it is low. And OCxREF will be held at 1 if CCR > ARR and held at 0 if CCR is 0.

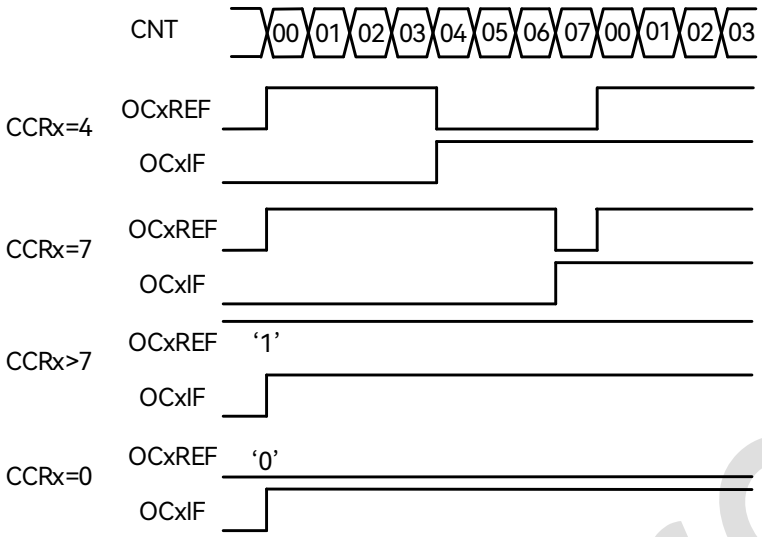


Figure 13-2: Edge-aligned PWM Waveform (ARR = 7)

- PWM center-aligned mode

  The definition of OCxREF level is the same as that in edge-aligned mode. The figure below is an example.

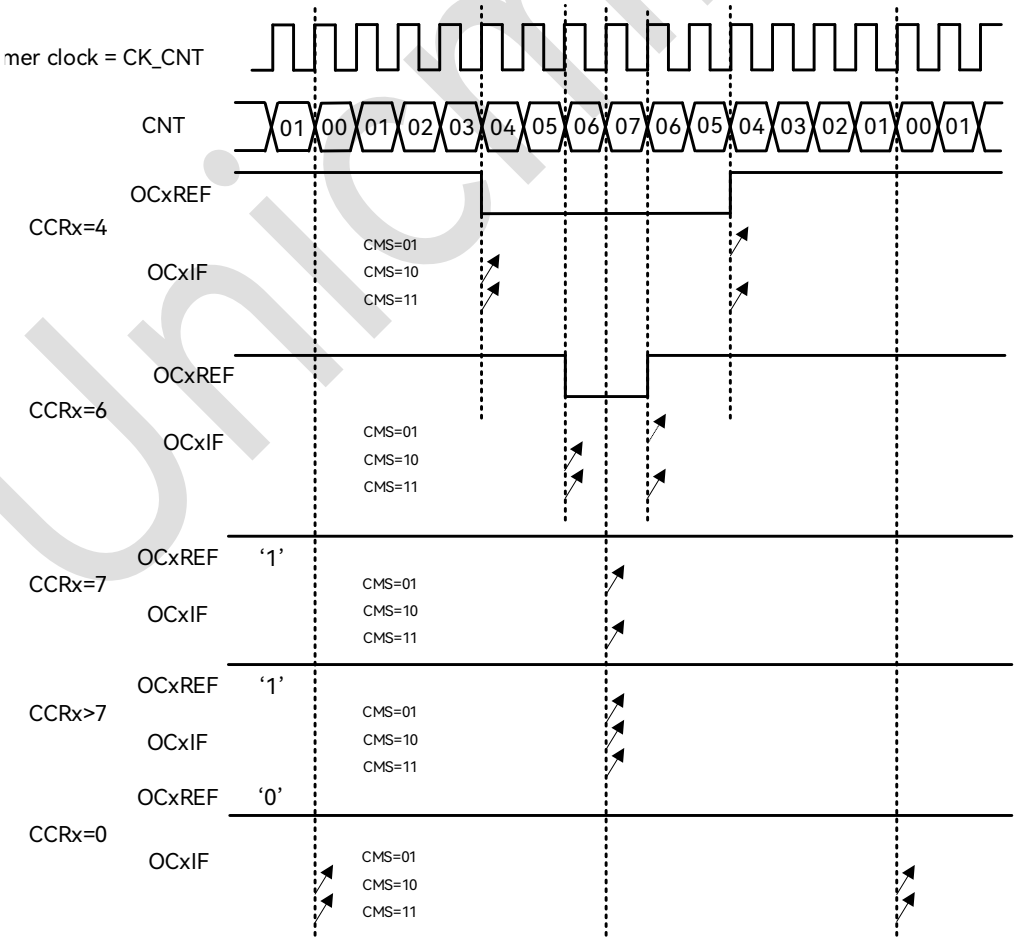


Figure 13-3: Center-aligned PWM Waveform (ARR = 7)

As the function of complementary PWM with dead-time insertion is enabled, the PWM output signal frequency depends on the ARR register and the PSC register. The greater of the ARR and ARRN values in ARR register determines the maximum count value. And the duty cycle depends on the CCR register.

The OCxREF output signal is as follows:

1. In edge-aligned down-counting mode (DIR = 1), OCxREF outputs high level when (MAX (ARR, ARRN) - ARR) <= CNT <= CCR, otherwise it outputs low level.
2. In edge-aligned up-counting mode (DIR = 0), OCxREF outputs high level when CNT < CCR or CNT > ARR or software writing UG, otherwise it outputs low level.
3. In center-aligned mode, OCxREF outputs high level when CNT <= CCR in down-counting (DIR = 1) or when CNT < CCR in up-counting (DIR = 0), otherwise it outputs low level.
4. In the case of MIN (CCR, CCRN) = MAX (ARR, ARRN), OCxREF outputs low level when CNT = ARR in edge-aligned up-counting mode, otherwise it outputs high level.
5. In the case of MIN (CCR, CCRN) > MAX (ARR, ARRN), OCxREF always outputs high level.
6. In the case of MIN (CCR, CCRN) = 0 in edge-aligned down-counting mode, the output condition of OCxREF is the same as indicated in item 1 above.

The OCxREFn output signal is as follows:

1. In edge-aligned down-counting mode (DIR = 1), OCxREFn outputs high level when (MAX (ARRN, ARRN) - ARR) <= CNT <= CCRN, otherwise it outputs low level.
2. In edge-aligned up-counting mode (DIR = 0), OCxREFn outputs high level when CNT < CCRN or CNT > ARRN or software writing UG, otherwise it outputs low level.
3. In center-aligned mode, OCxREFn outputs high level when CNT <= CCRN in down-counting (DIR = 1) or when CNT < CCRN in up-counting (DIR = 0), otherwise it outputs low level.
4. In the case of MIN (CCR, CCRN) = MAX (ARRN, ARRN), OCxREFn outputs low level when CNT = ARRN in edge-aligned up-counting mode, otherwise it outputs high level.

5. In the case of MIN (CCR, CCRN) > MAX (ARR, ARRN), OCxREFn always outputs high level.
6. In the case of MIN (CCR, CCRN) = 0 in edge-aligned down-counting mode, the output condition of OCxREFn is the same as indicated in item 1 above.

13.4.4 Break Function

The PWM waveform output can be triggered to stop (break) by software writing SOFT_BK bit or by hardware trigger via external IO. After the PWM output is stopped by the break signal, it is required to clear the BKE_F bit first and then enable the MOE bit to resume PWM output.

13.5 Use Process

13.5.1 General-purpose Timer

1. Configure PCLK1 and PRESET1, enable GTIMER, and reset release.
2. Configure GTIMER_PSC to set the reload value.
3. Configure GTIMER_ARR to set the prescaler value.
4. Configure GTIMER_EGR to generate an update event to immediately load the PSC value into the shadow register.
5. Clear the interrupt flag bit generated by UE.
6. Configure GTIMER_CRO to enable GTIMER counting.

13.5.2 PWM Output

1. Multiplex IO into GTIMER_CH and GTIMER_CHN according to the IO multiplexing relation.
2. Configure PCLK1 and PRESET1, enable GTIMER, and reset release.
3. Configure GTIMER_PSC to set the reload value.
4. Configure GTIMER_ARR to set the prescaler value.

5. Configure GTIMER_CCR to set the compare value.
6. To output complementary PWM, set GTIMER_ARRN and GTIMER_CCRN.
7. Configure GTIMER_EGR to generate an update event to immediately load the PSC value into the shadow register.
8. Clear the interrupt flag bit generated by UE.
9. Configure GTIMER_CCER to have OC output.
10. Configure GTIMER_CR2 to enable total output.
11. Configure GTIMER_CRO to enable GTIMER counting.

13.5.3 Input Capture

1. Multiplex IO into GTIMER_CH according to the IO multiplexing relation.
2. Configure PCLK1 and PRESET1, enable GTIMER, and reset release.
3. Configure GTIMER_PSC to set the reload value.
4. Configure GTIMER_ARR to set the prescaler value.
5. Configure GTIMER_CCMR1 to configure CC channel for input, capture source selection, capture source prescaling, and capture edge trigger control.
6. To adopt capture interrupt, configure GTIMER0_IER as compare/capture interrupt.
7. Configure GTIMER_CCER to enable the capture function.
8. Configure GTIMER_CRO to enable GTIMER counting.

13.5.4 Break Function

1. Multiplex IO into GTIMER_BKE according to the IO multiplexing relation.
2. Configure PCLK1 and PRESET1, enable GTIMER, and reset release.
3. Configure GTIMER_CR1 to set the positive level status of PWM after it is triggered to break.
4. Configure GTIMER0_CR1 to select the break signal polarity.

5. To adopt break interrupt, configure GTIMER_IER as break interrupt.
6. Configure GTIMER0_CR1 to enable break function.
7. Configure GTIMER_CRO to enable GTIMER counting.

14 I2C

14.1 Overview

The I2C bus interface handles communications between the microcontroller and the serial I2C bus. The I2C module receives and transmits data, and converts data from serial to parallel, or vice versa. It is connected to the I2C bus via the data pin SDA and the clock pin SCL to control all I2C bus-specific sequencing. This module supports master and slave modes.

14.2 Main Features

- I2C as master
- I2C as slave
- Programmable I2C slave device address
- Number of SDA transitions detected when SCL is low
- Programmable NACK/ACK
- Input SCL bus filter function
- Standard/Fast/HS mode available
- 7-bit device address supported
- Multi-master mode

14.3 Register Description

I2C register base address: 0xCC00

Table 14-1: List of Registers

Offset	Name	Description
0x00	I2C_SLAVE_ADDR1	I2C slave address register 1
0x01	I2C_CLK_DIV	SCL frequency division of I2C
0x02	I2C_CR0	Control register 0
0x03	I2C_CR1	Control register 1
0x04	I2C_SR0	Status register 0
0x05	I2C_SR1	Status register 1
0x06	I2C_DR	Data register
0x07	I2C_SLAVE_ADDR2	I2C slave address register 2

14.3.1 I2C_SLAVE_ADDR1 Slave Address Register 1

CC00H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_SLAVE_ADDR1	ADD1[7:1]							RSV
R/W	R/W							R
Reset value	0							0

Bit No.	Bit Designator	Description
7-1	ADD1[7:1]	7-1 bits of the address
0	RSV	Reserved

14.3.2 I2C_CLK_DIV Clock Division Register

CC01H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_CLK_DIV	I2C_CLK_DIV							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	I2C_CLK_DIV	SCL frequency division value; configure this register to set the transmission rate of I2C: $F_{scl} = (F_{sysclk}) / (4 * (DIV\ register\ value + 1))$

14.3.3 I2C_CR0 Control Register 0

CC02H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_CR0	MAAS1_INT_En	MIEN	RSTA	TACK	MTX	MSTA	RSV	MEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	MAAS1_INT_En	MAAS1 (received device address matches slave device address register 1) interrupt enable: 0: MAAS1 interrupt disabled 1: MAAS1 interrupt enabled
6	MIEN	MTF (byte data (including address) being transferred) interrupt enable: 0: MTF interrupt disabled 1: MTF interrupt enabled
5	RSTA	RSTA generation bit: 0: no condition for repeating start is generated 1: after writing this bit, a start is generated upon completion of transmitting or receiving a byte of data.
4	TACK	Transmission acknowledge bit / Stop condition bit: **For master mode:** 0: an ACK is generated on the acknowledge cycle after receiving a byte. 1: the master will generate a STOP after transmitting the current byte. **For slave mode:** 0: an ACK is generated on the acknowledge cycle after receiving a byte. 1: a NACK is generated on the acknowledge cycle after receiving a byte.
3	MTX	0: device serving as receiver 1: device serving as transmitter When acting as a slave, the processor shall query the SRW bit of I2C_SR to determine whether it is a transmitter or a receiver, and then set the matching MTX bit.
2	MSTA	Master-slave device selection bit, START bit: 0: slave mode 1: master mode The module generates a START condition if this bit changes from 0 to 1. When the STOP condition is generated, MSTA is cleared. MSTA is also cleared when the slave address matches (MAAS1 or MAAS2 is 1).

1	HOLD_EN	1: enable this bit, after receiving the address match interrupt, wait for the mtf signal to be 1 before writing data.
0	MEN	0: device disabled 1: device enabled

14.3.4 I2C_CR1 Control Register 1

CC03H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_CR1	RSV	MTX_ANTO_EN	OD_MODE	RSV	MAAS2_INT_EN	WBT_INT_EN	RXNE_INT_EN	TXE_INT_EN
R/W	R	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	RSV	Reserved
6	MTX_ANTO_EN	● When this module is in slave mode, the transmission data and address SDA lines are automatically switched. ● When this module is in slave mode, this bit is set to 1. After the master transmits the address information, it automatically switches the SDA transmission direction according to the RW bit on the bus. 0: auto-switching disabled 1: auto-switching enabled
5	OD_MODE	SCL and SDA output mode selection: 0: output in push-pull mode 1: output in open-drain mode
4	RSV	Reserved
3	MAAS2_INT_EN	MAAS2 (received device address matches slave device address register 2) interrupt enable: 0: MAAS2 interrupt disabled 1: MAAS2 interrupt enabled
2	WBT_INT_EN	WBT (byte has been transferred and TXE or RXNE is 1) interrupt enable: 0: WBT interrupt disabled 1: WBT interrupt enabled
1	RXNE_INT_EN	RXNE (data register being non-empty when receiving) interrupt enable: 0: RXNE interrupt disabled 1: RXNE interrupt enabled

0	TXE_INT_EN	TXE (data register being empty when transmitting) interrupt enable: 0: TXE interrupt disabled 1: TXE interrupt enabled

14.3.5 I2C_SR0 Status Register 0

CC04H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_SR0	MAAS2	MTF	MAAS1	MBB	RSV	SRW	MTF_H	RACK
R/W	R/W	R/W	R/W	R	R	R	R/W	R
Reset value	0	0	0	0	0	0	0	1

Bit No.	Bit Designator	Description
7	MAAS2	0: device address 2 does not match the received address 1: device address 2 matches the received address Write 1 to clear this bit.
6	MTF	0: byte transmission not completed 1: byte transmission completed This bit is 0 when a byte of data (including address) is being transmitted, and is set to 1 at the 9th SCL clock falling edge (ACK cycle) after the transmission of one byte. Write 1 at half an SCL cycle later than MTF_H to clear this bit.
5	MAAS1	0: device address 1 does not match the received address 1: device address 1 matches the received address Write 1 to clear this bit.
4	MBB	0: no data transmission on the bus (this bit is cleared when the STOP flag on the bus is detected) 1: data transmission being in progress on the bus (this bit is set to 1 when the START flag on the bus is detected)
3	RSV	Reserved
2	SRW	0: not serving as slave transmitter 1: serving as slave transmitter ● When the address matches, SRW indicates the R/W bit in the address call command, which is valid only under the following conditions: a complete transmission has occurred, no other transmissions are initialized, and I2C is configured in slave mode with the slave address matched.

		● This bit is automatically cleared when a STOP condition or a new START condition is received.
1	MTF_H	Completely done flag of fast byte transmission: 0: fast byte transmission not completed (half SCL cycle earlier) 1: fast byte transmission completed; MTF_H is generated at the 9th rising edge of SCL clock (ACK cycle), half a cycle before MTF. Clear by writing this bit or MTF to 1.
0	RACK	Acknowledgment receive status bit: 0: ACK received in the latest transmitting acknowledge cycle 1: no ACK received in the latest transmitting acknowledge cycle Only the START condition will clear the RACK bit.

14.3.6 I2C_SR1 Status Register 1

CC05H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_SR1	RSV					WBT	RXNE	TXE
R/W	R					R/W	R/W	R/W
Reset value	0					0	0	0

Bit No.	Bit Designator	Description
7-3	RSV	Reserved
2	WBT	0: byte transmission not completed or byte transmission completed with TXE and RXNE not being 1 1: byte transmission completed and TXE or RXNE is 1 This bit can be cleared by reading and writing the data register I2C_DR, or by writing 1.
1	RXNE	0: data register being empty when receiving 1: data register being non-empty when receiving This bit can be set by hardware, and can be cleared by reading the data register I2C_DR or by writing 1.
0	TXE	0: data register being non-empty when transmitting 1: data register being empty when transmitting This bit can be set by hardware, and can be cleared by writing the data register I2C_DR or by writing 1.

14.3.7 I2C_DR Data Register

CC06H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_DR	I2CDR							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	I2CDR	I2C data register value

14.3.8 I2C_SLAVE_ADDR2 Slave Address Register 2

CC07H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
I2C_SLAVE_ADDR2	ADD2[7:1]							ADDR2_EN
R/W	R/W							R/W
Reset value	0							0

Bit No.	Bit Designator	Description
7-1	ADD2[7:1]	Bits 7–1 of the address
0	ADDR2_EN	0: SLAVE_ADDR2 address matching disabled 1: SLAVE_ADDR2 address matching enabled

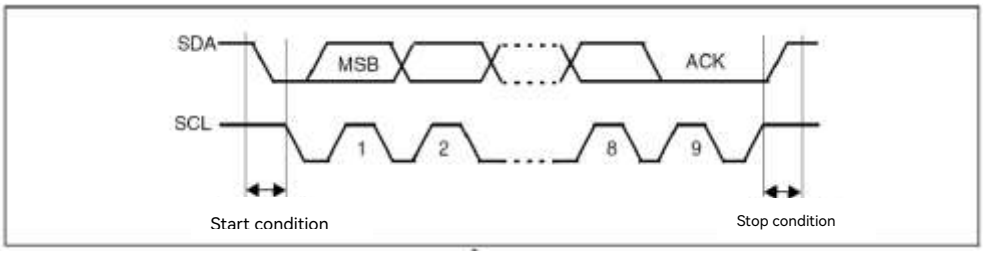
14.4 Functional Description

14.4.1 Mode Selection

The module is in slave mode by default and switches to master mode when the software initiates a START operation. In master mode, I2C starts data transmission and generates a clock signal, and can send out a STOP signal to stop the transmission. In slave mode, I2C is able to recognize its own address (7 bits). Data and address are transmitted in 8 bits/byte, with the high bit first.

The start condition is followed by address, which is transmitted only in the master mode. The receiver must return an ACK to the transmitter during the 9th clock cycle after 8 clock cycles of

a byte transmission. See the figure below:



14.4.2 I2C Slave Mode

Once the start condition is detected, the address received on the SDA line is sent to the shift register. It is then compared to the own device address of the chip, and if the address does not match I2C, ignore it and wait for another start condition. If the address matches, the controller will detect whether the current operation is a transmission or a reception (SRW register), and I2C will perform the following operations:

- **Slave transmitter:**

  The transmitter loads the byte from the data register to the SDA line via the internal shift register, generating the data register empty flag TXE, which shall be cleared by software updating the data register.

  When the acknowledge pulse is received, if the new data is still not written into the data register before the next data transmission ends, i.e., TXE is still 1, the byte wait flag bit (WBT) is set to 1, and the I2C interface keeps SCL low to wait for the new data to be written into the data register.

S	Address	A	Data 1	A	Data 2	A	.....	Data N	NA	P

Figure 14-1: Transmission Diagram of 7-bit Slave Transmitter

Note: S = Start (start condition), P = Stop (stop condition), A = ACK, NA = NACK.

: From master to slave

: From slave to master

- **Slave receiver:**

  After receiving the data, the slave receiver latches the bytes received from the SDA line to the data register via the internal shift register, generating the data register non-empty flag RXNE, which shall be cleared by software reading the value of the data register.

  The I2C interface generates an ACK pulse at the reception of every byte.

  If the value of the data register is not read out before the new data is received, i.e., RXNE is still 1, the byte wait flag bit (WBT) is set to 1, and the I2C interface keeps SCL low to wait for the value of the data register to be read.

Figure 14-2: Transmission Diagram of 7-bit Slave Receiver

Note: S = Start (start condition), P = Stop (stop condition), A = ACK, NA = NACK.

- **Close slave communication**

  After the last byte is transferred, the master issues a stop operation, and the I2C interface releases the SCL and SDA lines at the detection of this condition.

14.4.3 I2C Master Mode

In master mode, the I2C interface initiates data transmission and generates clock signal. The serial data transmission always starts with a start condition and ends with a stop condition. When a start operation is initiated on the bus via the START bit, the device enters the master mode. Operations in the master mode are as follows:

1. Configure the clock control register.
2. Configure the data register (with the address and read/write control bits of the slave).
3. Set the MSTA bit of the control register to 1 for generating the start condition.

- **Master transmitter:**

  After transmitting the address, the master loads the byte from the data register to the SDA

line via the internal shift register, generating the data register empty flag TXE, which shall be cleared by software updating the data register.

Confirm whether the new data has been transmitted to the data register when the acknowledge pulse is received. If the new data is still not written into the data register before the next data transmission ends, i.e., TXE is still 1, the byte wait flag bit (WBT) is set to 1, and the I2C interface keeps SCL low to wait for the new data to be written into the data register.

The master sends out a STOP signal to generate a stop condition.

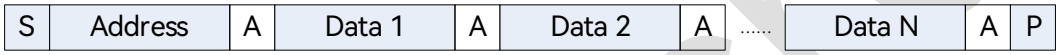


Figure 14-3: Transmission Diagram of 7-bit Master Transmitter

Note: S = Start (start condition), P = Stop (stop condition), A = ACK, NA = NACK

- **Master receiver:**

After transmitting the address, the I2C interface latches the bytes received from the SDA line to the data register via the internal shift register, generating the data register non-empty flag RXNE, which shall be cleared by software reading the value of the data register. If the value of the data register is not read out before the new data is received, i.e., RXNE is still 1, the byte wait flag bit (WBT) is set to 1, and the I2C interface keeps SCL low to wait for the value of the data register to be read.

Send an acknowledge pulse at the reception of each byte, and confirm that the value in the data register has been shifted.

The master sends a NACK after the last byte is received by the slave. Upon reception of NACK, the slave releases the control of the SCL and SDA lines. Then the master can send a stop/repeat start condition.

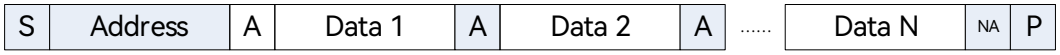


Figure 14-4: Transmission Diagram of 7-bit Master Receiver

Clock stretching:

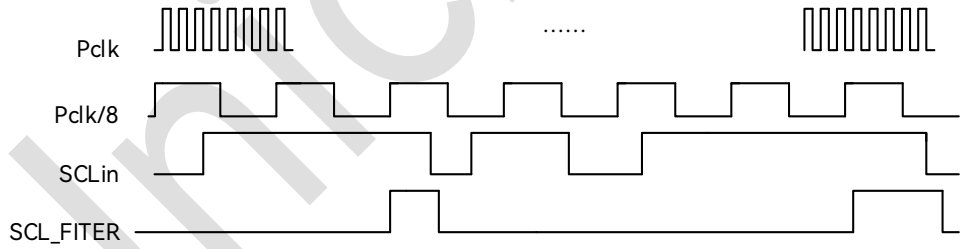
- TX mode: When the data in the data register is not updated, pull SCL low to wait for the new data to be written.
- RX mode: When the data in the data register is not read, pull SCL low to wait for the old data to be read.

14.4.4 SCL Bus Filter Algorithm 0

When SCL_FILTER_SEL is 0 and the value of I2C_FILTER register is 0, SCL is not provided with the filter function. When it is not 0, the filtering time is t_{cntc}* I2C_FILTER. Wherein, t_{cntc} is the 8-division clock cycle of pclk.

Example:

In the case of I2C_FILTER = 2, SCL can output high level only when a high level with the width of two consecutive t_{cntc}s is sampled, as pulse with width less than two t_{cntc}s are considered as interference glitches and filtered out：



Note:

- During use, set SDA_IN_DELAY to the same value as SCL_FILTER, and the filtered SDA and SCL signals will be kept at the input phase.
- In the process of communication, the SCL filtering function only filters the high level with the time below t_{cntc} * I2C_FILTER, and does not filter the low level. During non-communication process, the SCL filtering function only filters the low level with the time below t_{cntc} * I2C_FILTER, and does not filter the high level. The level of SCL filter function can be automatically switched according to START bit.

14.4.5 SCL Bus Filter Algorithm 1

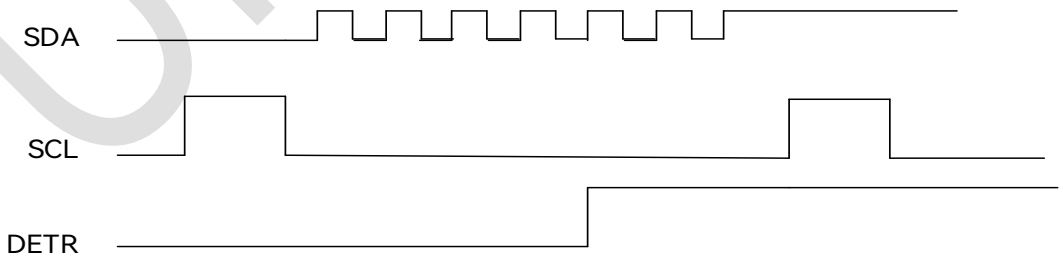
When SCL_FILTER_SEL is 1, SCL filter 1 function is selected. The maximum width of the filtered glitches is Tfclk x SWIDTH_THOLD. FREEZE_THOLD is the upper limit of the filter count, and in the case of adopting SCL filter algorithm 1, be sure that SWIDTH_THOLD < FREEZE_THOLD. As with algorithm 0, SDA_IN_DELAY can be adopted to delay the SDA signal.

14.4.6 Detecting SDA Transition at Low SCL

If the value of I2C_DET register is 0, the detection function is not enabled; while if it is other value, the detection function is enabled. When SCL is slow and the number of detected SDA rising-edge transitions is greater than the value of I2C_DET register, set the DETR bit of I2C_SR register to 1.

Example:

When I2C_DET = 5 and SCL is low, the internal counter starts to work, and when the accumulated number of SDA rising-edge transitions is greater than or equal to 5, DETR = 1. When SCL goes high, the internal counter is cleared and will start counting again at the arrival of the next low level of SCL. As shown in the figure below:



14.5 Use Process

14.5.1 Initialization Process

1. Configure PCLK0 and PRESET0, enable I2C, and reset release.
2. Multiplex IO into I2C_SDA and I2C_SCL according to the IO multiplexing relation, and configure PxPUN to enable the internal pull-up resistor.
3. Configure I2C_CR1 to enable the open-drain mode.
4. Configure the value of I2C_CLK_DIV register to determine the I2C transmission rate.
5. In the case of serving as slave, configure REG_I2C_CR1 to switch SDA transmission direction automatically in slave mode, configure REG_I2C_CR0, and enable hold_en.
6. Configure I2C_CR0 to enable I2C.

14.5.2 Master Transmitter

1. Write the 7-bit slave address to be accessed by I2C to the I2C_DR register.
2. Write MTX, MEN and MSTA of I2C_CR0 register to 1 to initiate START condition.
3. When TXE is read as 1, the first byte to be sent can be written to the I2C_DR register and the hardware clears the TXE bit.
4. Wait for the MTF flag and determine whether ACK is received. If ACK is received, it means the slave is correct. If NAK is received, the hardware will automatically issue the STOP signal and release the bus, and the software will wait for MBB to be 0 and then exit.
5. Repeat steps 3 and 4.
6. After writing the last byte to I2C_DR, wait until the penultimate byte is sent (MTF == 1), write TACK of I2C_CR register to 1 to indicate that the transmission is about to end. If the Restart operation is required to be generated, write RSTA to 1 here.
7. Upon completion of the last byte transmission, if RSTA is 1, initiate the Restart flag and

continue to transmit data; if RSTA is 0, the hardware will automatically generate a STOP condition, and the software will wait for MBB to be 0 and then exit.

Note: The above is the standard transmission process of the master transmitter, i.e., there is no clock stretching. If there is clock stretching (WBT == 1), the software shall deal with it immediately, otherwise the SCL will stop.

14.5.3 Master Receiver

1. Write the 7-bit slave address and the read bit of the 1-bit R/W bit to be accessed by I2C to the I2C_DR register, indicating that it is a receiver.
2. Write MTX, MEN and MSTA of I2C_CR register to 1 to initiate START condition.
3. Wait for the MTF flag and determine whether ACK is received. If ACK is received, it means the slave is correct. If NAK is received, the master will automatically issue the STOP bit and release the bus, and the software will wait for MBB to be 0 and then exit.
4. When RXNE is read as 1, the processor shall read the bytes received in the I2C_DR register, and the hardware clears the RXNE bit. Wait for the next byte to be received.
5. Wait for the MTF flag and clear it by software (the software process can continue even if it is not cleared).
6. Repeat steps 4 and 5.
7. When the penultimate byte is received and the ACK signal (i.e., the penultimate MTF flag) is transmitted, the master writes TACK of I2C_CR register to 1 to indicate that the next byte to be received is the last byte.
8. After the last byte is received, the hardware will automatically issue a NACK signal and generate STOP condition, and the software will wait for MBB to be 0 and then exit.

Note: The above is the standard reception process of the master receiver, i.e., there is no clock stretching. If there is clock stretching (WBT == 1), the software shall deal with it immediately, otherwise the SCL will stop.

14.5.4 Slave Transmitter

1. Write a 7-bit address to the I2C_SLAVE_ADDR1 register or the I2C_SLAVE_ADDR2 register as the address to be addressed in the slave mode.
2. Write the MEN bit of I2C_CR register to 1 to enable the I2C module.
3. Wait for the MAAS1 or MASS2 (ADDR2_EN = 1) flag to be valid. Repeat step 3 if the address match is invalid.
4. If the address match is valid, determine if the SRW bit is 1. 0 indicates slave receiver while 1 indicates slave transmitter.
5. Wait for the byte transmission completion flag (MTF).
6. Write the first byte of data to be sent into I2C_DR, and write MTX of I2C_CR register to 1 to indicate that it serves as a slave transmitter.
7. When TXE is read as 1, the second byte to be sent can be written to the I2C_DR register and the hardware clears the TXE bit.
8. Wait for the MTF flag and determine whether ACK is received. If ACK is received, it means the master is correct.
9. Repeat steps 7 and 8. After receiving the NACK or STOP signal from the master, the I2C module releases the bus. The software waits for MBB to be 0 and then exits.

14.5.5 Slave Receiver

1. Write a 7-bit address to the I2C_SLAVE_ADDR1 register or the I2C_SLAVE_ADDR2 register as the address to be addressed in the slave mode.
2. Write the MEN bit of I2C_CR register to 1 to enable the I2C module.
3. Wait for the MAAS1 or MASS2 (ADDR2_EN = 1) flag to be valid. Repeat step 3 if the address match is invalid.
4. If the address match is valid, determine if the SRW bit is 1. 0 indicates slave receiver while

1 indicates slave transmitter.

5. Wait for the byte transmission completion flag (MTF).
6. When RXNE is read as 1, the processor shall read the bytes received in the I2C_DR register, and the hardware clears the RXNE bit. Wait for the next byte to be received.
7. Wait for the MTF flag and clear it by software (the software process can continue even if it is not cleared).
8. Repeat steps 6 and 7 and stop after receiving the stop signal.
9. I2C can also receive and transmit in one byte (i.e., MTF == 1). When TACK of the I2C_CR register is written to 1, then a NACK signal will be transmitted to the master upon completion of the next byte reception. The software waits for MBB to be 0 and then exits.

Note: The default IOs for I2C are P04 and P10, which shall be configured to 0 first before multiplexing other IOs for I2C function.

15 CAN

15.1 Overview

The CAN controller being compliant with CAN2.0A/B protocol can be used in the fields of automotive electronics and industrial control.

15.2 Main Features

Functional features:

- Being compliant with CAN2.0A/B protocol (11/29-bit ID)
- Data transmission rate: 1 Mbps (max.), 5 kbps (min.)
- Hardware data filter (single/double-filter optional)
- 32-byte RX FIFO (available for storing ID in standard format)
- 16-byte TX buffer
- Overload frame generated at FIFO overflow
- Normal & listening mode
- Transmission can be aborted
- Error counter value visible

15.3 Register Description

CAN register base address: 0xCF00

Table 15-1: Register Description

Offset	Name	Description
0x00	CAN_MR	Mode register
0x01	CAN_CMR	Command register

Offset	Name	Description
0x02	CAN_SR	Control register
0x03	CAN_ISR	Interrupt status register
0x04	CAN_IMR	Interrupt enable register
0x05	CAN_RMC	RX FIFO data bit count register
0x06	CAN_BTR0	Bus timing register 0
0x07	CAN_BTR1	Bus timing register 1
0x08	CAN_TXBUF0	TX buffer register 0
0x09	CAN_TXBUF1	TX buffer register 1
0x0A	CAN_TXBUF2	TX buffer register 2
0x0B	CAN_TXBUF3	TX buffer register 3
0x0C	CAN_RXBUF0	RX buffer register 0
0x0D	CAN_RXBUF1	RX buffer register 1
0x0E	CAN_RXBUF2	RX buffer register 2
0x0F	CAN_RXBUF3	RX buffer register 3
0x10	CAN_ACR0	RX filter match register 0
0x11	CAN_ACR1	RX filter match register 1
0x12	CAN_ACR2	RX filter match register 2
0x13	CAN_ACR3	RX filter match register 3
0x14	CAN_AMR0	RX filter mask register 0
0x15	CAN_AMR1	RX filter mask register 1
0x16	CAN_AMR2	RX filter mask register 2
0x17	CAN_AMR3	RX filter mask register 3
0x18	CAN_ECC	Error code capture register
0x19	CAN_RXERR	RX error count register
0x1A	CAN_TXERR	TX error count register
0x1B	CAN_ALC	Arbitration lost capture register

15.3.1 CAN_MR Mode Register

CF00H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_MR	-	-	-	-	-	RM	LOM	AFM
R/W	R	R	R	R	R	R/W	R/W	R/W
Reset value	0	0	0	0	0	1	0	0

Bit No.	Bit Designator	Description
7-3	RSV	Reserved

2	RM	Reset mode set bit: 1: CAN works in reset mode. 0: CAN works in other modes. No data transmission and reception are performed in reset mode, which is used for some hardware configuration (some registers can only be written in reset mode). After reset mode, it can enter listening mode or normal mode.
1	LOM	Listening mode set bit: 1: if RM = 0, CAN enters listening mode*. 0: if RM = 0, CAN enters normal mode. This bit can only be set in reset mode.
0	AFM	Hardware matching data selection bit: 1: use single filter 0: use double filters This bit can only be set in reset mode.

*: In listening mode, the CAN controller will not answer to the CAN bus (no ACK response will be sent) even if the message is successfully received. The error counter will stop at the current value. Listening mode is mainly used for bit rate detection without interfering with network traffic, and can also be used for the CAN bus analyzer.

15.3.2 CAN_CMR Command Register

CF01H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_CMR	-	-	-	-	-	TR	AT	-
R/W	R	R	R	R	R	W	W	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-3	RSV	Reserved
2	TR	Transmit request setting bit: 1: start transmitting in frame format 0: transmitting prohibited
1	AT	Enable bit for aborting transmission: 1: transmission allowed to be aborte 0: transmission prohibited to be aborted

		Setting TR and AT at the same time can initiate a single-transmission, and frame retransmission will not be performed in the case of bus error or arbitration lost. Aborting only works for frames that are about to be transmitted, frames that have already been sent cannot be aborted. If the transmission is started by setting TR to 1 in the previous command, it cannot be canceled by setting TR bit to 0. In this case, the transmission can be canceled by setting AT to 1.
0	RSV	Reserved

15.3.3 CAN_SR Status Register

CF02H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_SR	RBS	DSO	TBS	-	RS	TS	ES	BS
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	1	0	0	0	0	0

Bit No.	Bit Designator	Description
7	RBS	RX FIFIO status: 1: at least one message in FIFO 0: no message in FIFO
6	DSO	Data overflow status: 1: RX FIFO overflow triggers an interrupt (if enabled). 0: no overflow has occurred since the last data overflow clearing.
5	TBS	TX buffer status: 1: TX buffer can be written by CPU. 0: TX buffer is locked. A message is being sent or waiting to be sent. If the CPU tries to write to the TX buffer in the locked state (TBS = 0), the written data is not accepted.
4	RSV	Reserved
3	RS	RX status bit: 1: CAN is receiving. 0: CAN is not in receive state.
2	TS	TX status bit: 1: CAN is transmitting. 0: CAN is not in transmit state.

1	ES	Error status bit: 1: at least one CAN error counter reaches the error warning limit of 96. 0: normal state
0	BS	Bus status bit: **1: off-line state** The CAN controller is in reset mode and the error warning interrupt is triggered (if enabled). The TX error counter is set to 127, and the RX error counter is set to 0. CAN will remain in reset mode until CPU clears the RM bit. After this operation, CAN will wait for the occurrence of 128 bus idle signals (11 consecutive recessive bits), and the TX error counter will count down. Then the BS bit is cleared, the error counter is reset, and the error warning interrupt is triggered (if enabled). **0: normal state** Frame transmission and reception can be performed.

15.3.4 CAN_ISR Interrupt Status / Acknowledgment Register

CF03H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ISR	-	ALI	EWI	EPI	RI	TI	BEI	DOI
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	RSV	Reserved
6	ALI	Arbitration lost interrupt status bit: When an arbitration lost is detected during message transmission and CAN becomes the receiver, this bit can be set to read ALC register to check which bit in the arbitration segment is lost, and writing 1 clears this interrupt.
5	EWI	Error warning interrupt status bit: This error warning interrupt bit is set when the ES or BS bit in SR register changes. Therefore, it can be used to detect if the CAN enters or exits the bus-off state. Writing 1 clears this interrupt.
4	EPI	Error passive interrupt status bit: This bit is set when the CAN bus controller reaches or exits the error passive level (i.e., the status changes from active to passive or vice

		versa). Writing 1 clears this interrupt.
3	RI	RX interrupt status bit: CAN sets this bit to 1 when there is at least one CAN frame data in RX FIFO. After reading the message, CPU must write the RI bit to 1 (message read acknowledgment) to decrement the count of RX message counter (RMC), which does not automatically decrement.
2	TI	TX interrupt status bit: The TX interrupt bit is set upon successful transmission. The write pointer can be reset to TX RAM by clearing the TI bit (via writing 1) before writing a new data frame.
1	BEI	Bus error interrupt status bit: Set BEI when CAN encounters a bus error in the course of sending or receiving messages. Writing 1 clears this interrupt.
0	DOI	RX data overflow interrupt status bit: DOI is set when an RX FIFO overflow occurs. Writing 1 clears this interrupt.

15.3.5 CAN_IMR Interrupt Enable Register

CF04H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_IMR	-	ALIM	EWIM	EPIM	RIM	TIM	BEIM	DOIM
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	RSV	Reserved
6	ALIM	Arbitration lost interrupt enable bit: enable the CAN transmitter to trigger an interrupt when an arbitration lost is detected during transmission and it becomes a CAN receiver: 1: ALI interrupt enabled 0: ALI interrupt disabled
5	EWIM	Error warning interrupt enable bit: enable to trigger an interrupt when the status of the BS or ES bit of CAN_SR register changes: 1: EWI interrupt enabled 0: EWI interrupt disabled
4	EPIM	Error passive interrupt enable bit: enable to trigger an interrupt when the CAN controller enters or exits error passive mode:

		1: EPI interrupt enabled 0: EPI interrupt disabled
3	RIM	RX interrupt enable bit: 1: RI interrupt enabled 0: RI interrupt disabled
2	TIM	TX interrupt enable bit: 1: TI interrupt enabled 0: TI interrupt disabled
1	BEIM	Bus error interrupt enable bit: enable to trigger an interrupt when a bus error occurs during CAN transmission or reception: 1: BEI interrupt enabled 0: BEI interrupt disabled
0	DOIM	RX data overflow interrupt enable bit: 1: DOI interrupt enabled 0: DOI interrupt disabled

15.3.6 CAN_RMC RX Data Count Register

CF05H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_RMC	-	-	-	EPIM				
R/W	R	R	R	R				
Reset value	0	0	0	0				

Bit No.	Bit Designator	Description
7-5	RSV	Reserved
4-0	RMC	Number of CAN frames in RX FIFO, which can store up to 8 messages. The following equation allows calculating the maximum number of messages to be stored in RX FIFO: $n = \frac{32}{3 + data\_length\_code}$ Note: Here data_length_code is at least 1. If the length of the CAN data segment is 0, then data_length_code = 1.

15.3.7 CAN_BTR0 Bus Timing Register 0

This register can only be written in reset mode and can be read in any mode.

CF06H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_BTR0	SJW		BRP					
R/W	R/W		R/W					
Reset value	0		0					

Bit No.	Bit Designator	Description
7-6	SJW	Synchronization jump width: $t_{SJW} = t_{SCLK} \times (2 \times SJW.1 + SJW.0 + 1)$ In order to compensate the phase shift between clock oscillators of different CAN bus controllers, the bit period must be shortened or extended accordingly. SJW defines the maximum number of clock cycles for a resynchronization to change a bit period. During synchronization, the hardware synchronizes with the RX signal by increasing "1 + SJW" t_{SCLK} in PBS1 segment or decreasing "1− (1 + SJW)" t_{SCLK} in PBS2 segment.
5-0	BRP	Baud rate prescaler value: $t_{SCLK} = 2 \times t_{CLK} \times (32 \times BRP.5 + 16 \times BRP.4 + 8 \times BRP.3 + 4 \times BRP.2 + 2 \times BRP.1 + BRP.0 + 1)$ Wherein, $t_{CLK} = 1 / f_{PCLK}$.

15.3.8 CAN_BTR1 Bus Timing Register 1

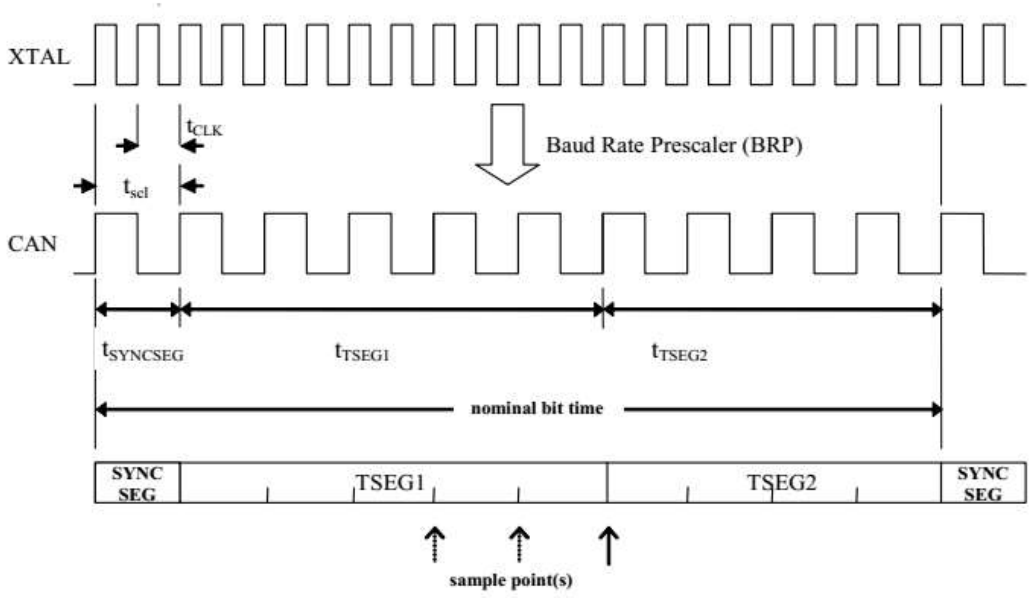
This register can only be written in reset mode and can be read in any mode.

CF07H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_BTR1	SAM	TSEG2			TSEG1			
R/W	R/W	R/W			R/W			
Reset value	0	0			0			

Bit No.	Bit Designator	Description
7	SAM	Bus level sampling number selection bit: 1: sample the bus level three times (for medium/low speed buses) 0: sample the bus level once (for high speed buses)

6-4	TSEG2	Number of clock cycles for time segment 2: $t_{TSEG2} = t_{SCLK}$ x (4 x TSEG2.2 + 2 x TSEG2.1 + TSEG2.0 + 1)
3-0	TSEG1	Number of clock cycles for time segment 1: $t_{TSEG1} = t_{SCLK}$ x (8 x TSEG1.3 + 4 x TSEG1.2 + 2 x TSEG1.1 + TSEG1.0 + 1)

The bit period structure of CAN is shown in the following diagram. Wherein, the synchronization segment (SYNC SEG) is 1 x t_{SCLK}, and the lengths of phase buffer segments 1 and 2 are determined by TSEG1 and TSEG2.



15.3.9 CAN_TXBUF0 Transmit Buffer Register 0

CF08H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_TXBUF0	TXBUF[7:0]							
R/W	W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	TXBUF	The TX buffer register is used to write the CAN frames to be transmitted over the CAN network. Writing to this register performs automatic increment of the internal write pointer. By writing TI bit in ISR register, the write pointer can be reset to the address 0h of the TX memory.

15.3.10 CAN_TXBUF1 Transmit Buffer Register 1

CF09H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_TXBUF1	TXBUF[15:8]							
R/W	W							
Reset value	0							

Bit No.	Bit Designator	Description
7:0	TXBUF	The TX buffer register is used to write the CAN frames to be transmitted over the CAN network. Writing to this register performs automatic increment of the internal write pointer. By writing TI bit in ISR register, the write pointer can be reset to the address 0h of the TX memory.

15.3.11 CAN_TXBUF2 Transmit Buffer Register 2

CF0AH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_TXBUF2	TXBUF[23:16]							
R/W	W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	TXBUF	The TX buffer register is used to write the CAN frames to be transmitted over the CAN network. Writing to this register performs automatic increment of the internal write pointer. By writing TI bit in ISR register, the write pointer can be reset to the address 0h of the TX memory.

15.3.12 CAN_TXBUF3 Transmit Buffer Register 3

CF0BH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_TXBUF3	TXBUF[31:24]							
R/W	W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	TXBUF	The TX buffer register is used to write the CAN frames to be transmitted over the CAN network. Writing to this register performs automatic increment of the internal write pointer. By writing TI bit in ISR register, the write pointer can be reset to the address 0h of the TX memory.

15.3.13 CAN_RXBUF0 Receive Buffer Register 0

CF0CH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_RXBUF0	RXBUF[7:0]							
R/W	R							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	RXBUF	The RX buffer register is used to read CAN frames received from the CAN network. Reading this register will automatically increment the read address pointer of the internal FIFO (increment after reading).

15.3.14 CAN_RXBUF1 Receive Buffer Register 1

CF0DH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_RXBUF1	RXBUF[15:8]							
R/W	R							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	RXBUF	The RX buffer register is used to read CAN frames received from the CAN network. Reading this register will automatically increment the read address pointer of the internal FIFO (increment after reading).

15.3.15 CAN_RXBUF2 Receive Buffer Register 2

CF0EH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_RXBUF2	RXBUF[23:16]							
R/W	R							
Reset value	0							

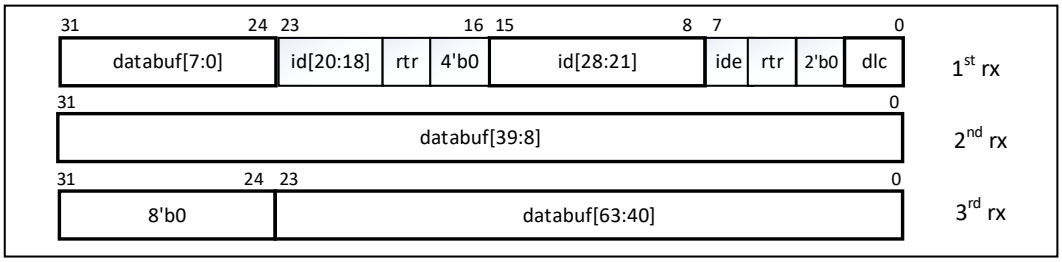
Bit No.	Bit Designator	Description
7-0	RXBUF	The RX buffer register is used to read CAN frames received from the CAN network. Reading this register will automatically increment the read address pointer of the internal FIFO (increment after reading).

15.3.16 CAN_RXBUF3 Receive Buffer Register 3

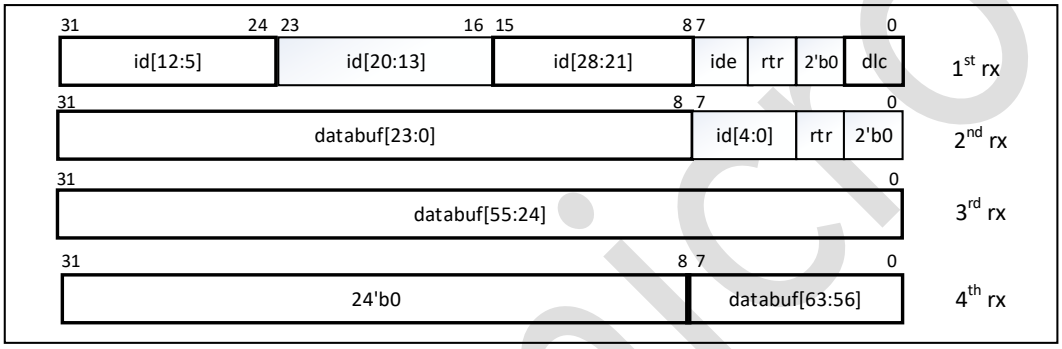
CF0FH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_RXBUF3	RXBUF[31:24]							
R/W	R							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	RXBUF	The RX buffer register is used to read CAN frames received from the CAN network. Reading this register will automatically increment the read address pointer of the internal FIFO (increment after reading).

After receiving a frame of CAN data, the format of data read by RXBUF register is as follows (the length of databuf segment is determined by DLC segment, and the number is 0-8 bytes):



CAN RX_FIFO for 11bits ID



CAN RX_FIFO for 29bits ID

After receiving a frame of CAN data, the RMC register count is increased by 1, and the CAN controller will write data into RX FIFO one by one. RBS will be set when a 32-bit data is written. After a frame of data is written, the RI flag bit is set.

15.3.17 CAN_ACR RX Filter Match Register

Only when the identifier bit of the received message is equal to the predefined bit in the RX filter can the RX filter in CAN controller pass the received message to RX FIFO. The RX filter consists of an RX filter match register (ACR3: ACR0) and an RX filter mask register (AMR3: AMR0). The AFM bit of the mode register can set single/double filters. In the configuration of single filter, the filter is 4 bytes long. If the received data is in standard frame mode, the first 2 bytes of arbitration bit, RTR bit and data bit can be received (the data byte does not have to be received). All individual-bit comparisons must be signaled to indicate successful reception of the data; if the data is received in expanded frame format, the data of arbitration bit and RTR bit can be

received. For bits in undefined format, the filter will not perform a comparison.

The double-filter configuration defines two filters of shorter lengths. The received data will be compared with the two filters to decide whether the data shall be stored in RX FIFO. If at least one of the RX filters successfully matches, the received data will be stored in FIFO. If a standard frame is received, the first filter will compare the arbitration, the RTR bit and the first data byte in standard format. The second filter only compares the arbitration and the RTR bit in the standard format. If the data byte is not filtered in filter 1, the lower four bits of AMR1 and AMR3 shall be set to logic 1 (this bit is not compared).

15.3.18 CAN_ACR0 RX Filter Match Register 0

CF10H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ACR0	ACR0							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ACR0	The RX filter match register contains the arbitration bit of the message to be received, and the corresponding RX filter mask register defines the bits to be compared and the irrelevant bits.

15.3.19 CAN_ACR1 RX Filter Match Register 1

CF11H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ACR1	ACR1							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ACR1	The RX filter match register contains the arbitration bit of the message to be received, and the corresponding RX filter mask register defines the bits to be compared and the irrelevant bits.

15.3.20 CAN_ACR2 RX Filter Match Register 2

CF12H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ACR2	ACR2							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ACR2	The RX filter match register contains the arbitration bit of the message to be received, and the corresponding RX filter mask register defines the bits to be compared and the irrelevant bits.

15.3.21 CAN_ACR3 RX Filter Match Register 3

CF13H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ACR3	ACR3							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	ACR3	The RX filter match register contains the arbitration bit of the message to be received, and the corresponding RX filter mask register defines the bits to be compared and the irrelevant bits.

15.3.22 CAN_AMR RX Filter Mask Register

Only when the identifier bit of the received message is equal to the predefined bit in RX filter can the RX filter in CAN controller pass the received message to RX FIFO. The RX filter is defined by the RX filter match register (ACR3: ACR0) and the RX filter mask register (AMR3: AMR0).

15.3.23 CAN_AMR0 RX Filter Mask Register 0

CF14H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_AMR0	AMR0							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	AMR0	The RX filter mask register defines the bits to be compared and the irrelevant bits. Set the corresponding bit to 1 indicating not to compare the corresponding bits in ACR register.

15.3.24 CAN_AMR1 RX Filter Mask Register 1

CF15H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_AMR1	AMR1							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	AMR1	The RX filter mask register defines the bits to be compared and the irrelevant bits. Set the corresponding bit to 1 indicating not to compare the corresponding bit in ACR register.

15.3.25 CAN_AMR2 RX Filter Mask Register 2

CF16H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_AMR2	AMR2							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	AMR2	The RX filter mask register defines the bits to be compared and the irrelevant bits. Set the corresponding bit to 1 indicating not to compare the corresponding bit in ACR register.

15.3.26 CAN_AMR3 RX Filter Mask Register 3

CF17H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_AMR3	AMR3							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	AMR3	The RX filter mask register defines the bits to be compared and the irrelevant bits. Set the corresponding bit to 1 indicating not to compare the corresponding bit in ACR register.

The bit formats corresponding to different filter settings and different arbitration lengths (11 bits in standard frame / 29 bits in extended frame) are as follows:

- **Single filter:**

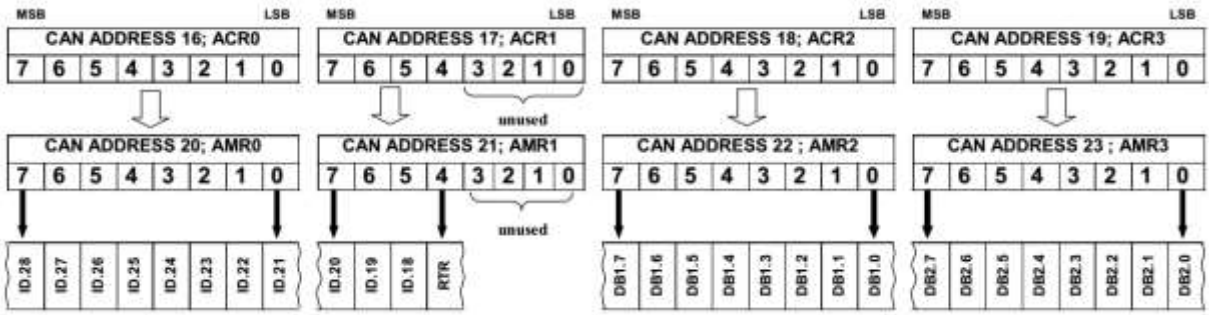


Figure 15-1: Single Filter for Standard Frame Message

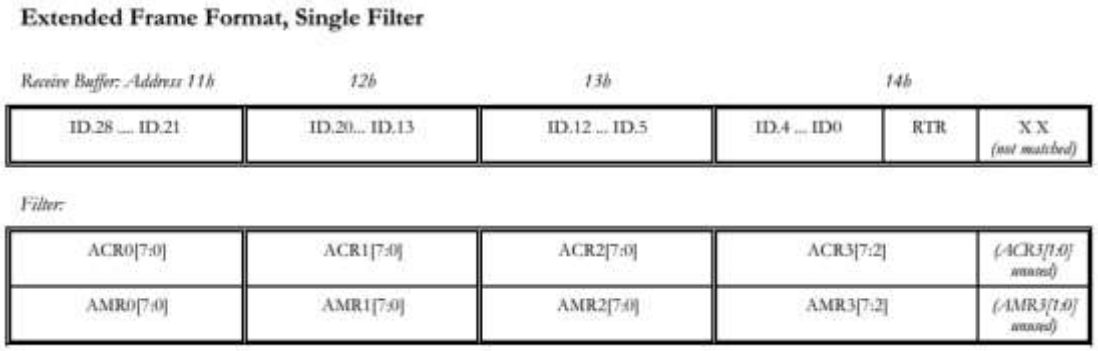


Figure 15-2: Single Filter for Extended Frame Message

- **Double filters:**

In standard mode, when data is received, it will be compared with the first filter ID including RTR bit, and the first received data byte, or with the second filter ID including RTR bit.

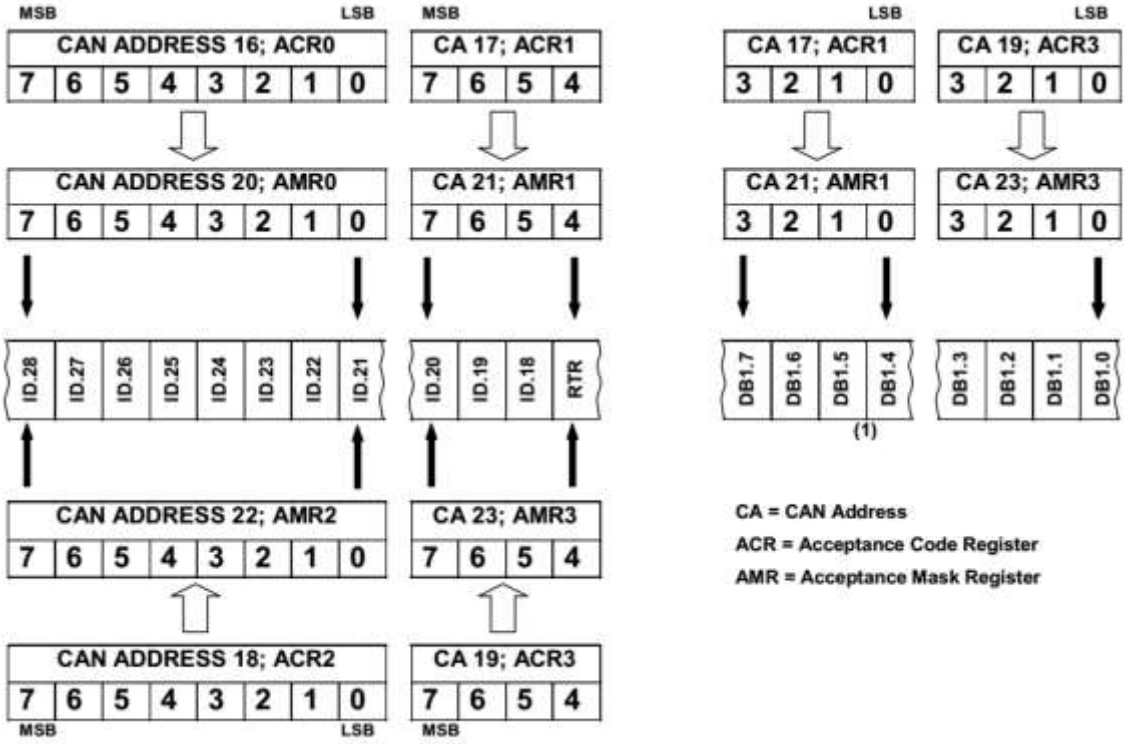


Figure 15-3: Double Filters for Standard Frame Message

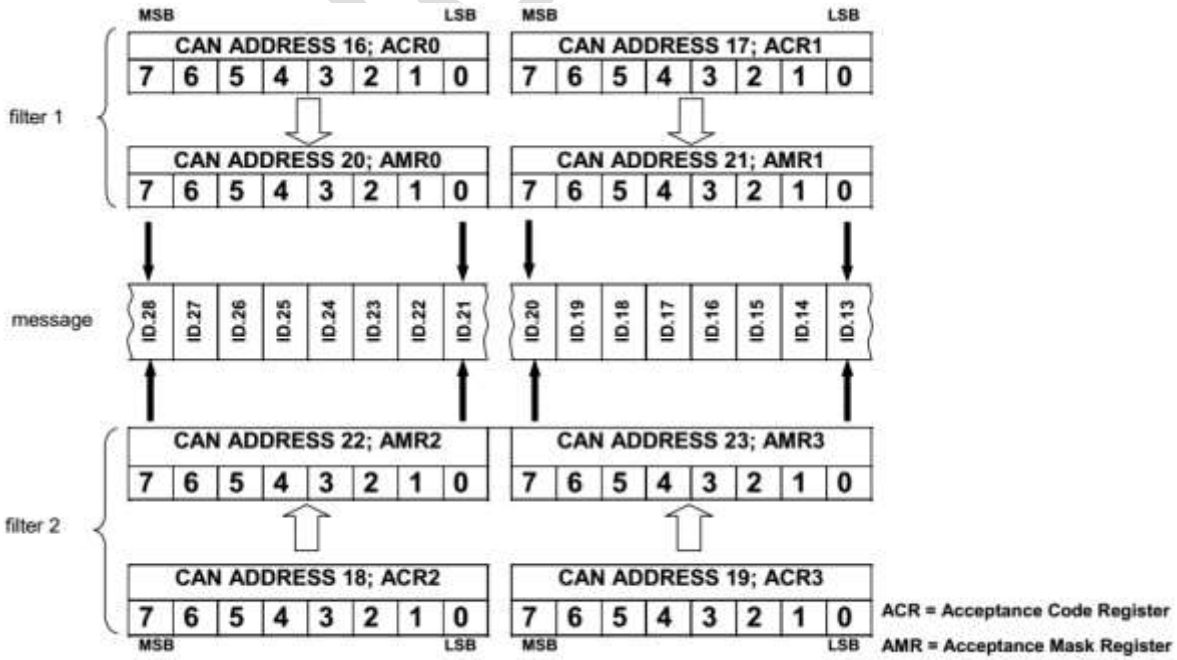


Figure 15-4: Double Filters for Extended Frame Message

15.3.27 CAN_ECC Error Code Capture Register

The ECC read-only register holds the error code regarding the last bus error that occurred on the CAN network. This register is read-only.

The CAN core will not update this register until the previous bus error is acknowledged (by acknowledging the bus error interrupt).

CF18H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ECC	RXWRN	TXWRN	EDIR	ACKER	FRMER	CRCER	STFER	BER
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	RXWRN	Set this bit to 1 when the RXERR counter value is greater than or equal to 96.
6	TXWRN	Set this bit to 1 when the TXERR counter value is greater than or equal to 96.
5	EDIR	Direction of data transmission in the occurrence of an error: 0: transmitting 1: receiving
4	ACKER	Set this bit to 1 when an ACK error occurs.
3	FRMER	Set this bit to 1 when a frame format error occurs.
2	CRCER	Set this bit to 1 when an CRC error occurs.
1	STFER	Set this bit to 1 when a padding error occurs.
0	BER	Set this bit to 1 when a bit error occurs.

15.3.28 CAN_RXERR RX Error Count Register

CF19H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_RXERR	RXERR							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	RXERR	Current value of the RX error counter. If a bus-off event occurs, the RX error counter will be initialized to 0.

15.3.29 CAN_TXERR TX Error Count Register

CF1AH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_TXERR	TXERR							
R/W	R/W							
Reset value	0							

Bit No.	Bit Designator	Description
7-0	TXERR	Current value of TX error counter. If a bus-off event occurs, the TX error counter will be initialized to 127 to calculate the minimum protocol-defined time (128 occurrences of the bus idle signal). Read TXERR during this time to obtain information about the bus recovery state.

15.3.30 CAN_ALC Arbitration Lost Capture Register

The CAN controller is able to determine the exact in-frame location of the arbitration lost. Immediately thereafter an “arbitration lost interrupt” will be generated. In addition, the number of bits is captured in the arbitration lost capture register. Once the master controller reads the contents of this register, the capture function will be activated for the next arbitration lost. This feature allows CAN to monitor each CAN bus access. For diagnostics or during system configuration, each case of unsuccessful arbitration can be determined.

CF1BH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAN_ALC	-	-	-	ALC				
R/W	R	R	R	R				
Reset value	0	0	0	0				

Bit No.	Bit Designator	Description
7-5	RSV	Reserved
4-0	ALC	Arbitration lost position

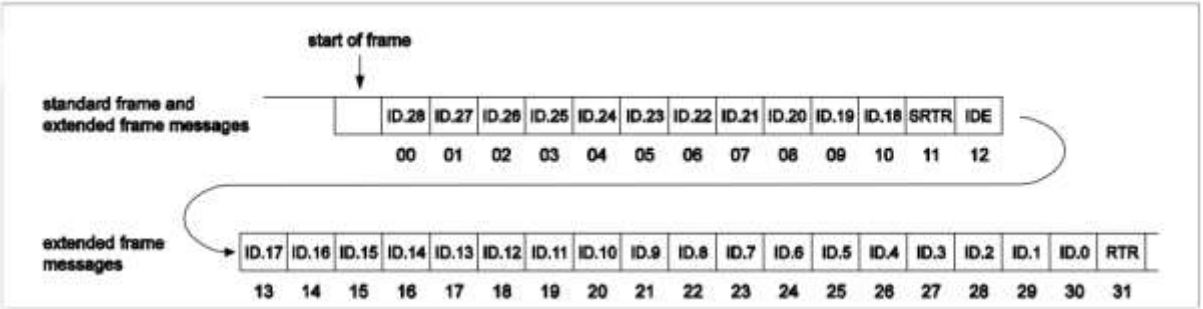


Figure 15-5: Arbitration Lost Bit Number Interpretation

Table 15-2: ALC Register Contents Description

Bits					Decimal Value	Description
ALC4	ALC3	ALC2	ALC1	ALC0		
0	0	0	0	0	00	Arbitration lost in **ID28 / 10**
0	0	0	0	1	01	Arbitration lost in **ID27 / 9**
0	0	0	1	0	02	Arbitration lost in **ID26 / 8**
0	0	0	1	1	03	Arbitration lost in **ID25 / 7**
0	0	1	0	0	04	Arbitration lost in **ID24 / 6**
0	0	1	0	1	05	Arbitration lost in **ID23 / 5**
0	0	1	1	0	06	Arbitration lost in **ID22 / 4**
0	0	1	1	1	07	Arbitration lost in **ID21 / 3**
0	1	0	0	0	08	Arbitration lost in **ID20 / 2**
0	1	0	0	1	09	Arbitration lost in **ID19 / 1**
0	1	0	1	0	10	Arbitration lost in **ID18 / 0**
0	1	0	1	1	11	Arbitration lost in **SRTR / RTR**
0	1	1	0	0	12	Arbitration lost in **IDE** bit
0	1	1	0	1	13	Arbitration lost in **ID17***
0	1	1	1	0	14	Arbitration lost in **ID16***
0	1	1	1	1	15	Arbitration lost in **ID15***
1	0	0	0	0	16	Arbitration lost in **ID14***
1	0	0	0	1	17	Arbitration lost in **ID13***
1	0	0	1	0	18	Arbitration lost in **ID12***
1	0	0	1	1	19	Arbitration lost in **ID11***
1	0	1	0	0	20	Arbitration lost in **ID10***
1	0	1	0	1	21	Arbitration lost in **ID9***
1	0	1	1	0	22	Arbitration lost in **ID8***
1	0	1	1	1	23	Arbitration lost in **ID7***
1	1	0	0	0	24	Arbitration lost in **ID6***
1	1	0	0	1	25	Arbitration lost in **ID5***
1	1	0	1	0	26	Arbitration lost in **ID4***
1	1	0	1	1	27	Arbitration lost in **ID3***
1	1	1	0	0	28	Arbitration lost in **ID2***
1	1	1	0	1	29	Arbitration lost in **ID1***
1	1	1	1	0	30	Arbitration lost in **ID0***
1	1	1	1	1	31	Arbitration lost in **RTR**

15.4 Process Description

15.4.1 Transmit CAN Data Frame

1. Turn on the CAN clock, release the reset, and multiplex the CAN function pins.
2. Configure the bus timing registers CAN_BTR0/CAN_BTR1.
3. Clear the error flag bit/interrupt flag bit in CAN_ISR register.
4. Configure the interrupt enable register IMR to enable TI interrupt (optional).
5. Configure the LOM bit of the mode register MR to enter normal mode.
6. Configure the TX buffer register CAN_TXBUF, write the contents of CAN data frames according to the defined format, and write them in the order of transmitting, with 32 bits of data at a time.
7. Configure the TR bit of the command register CMR to start transmitting.
8. Wait for the TBS bit of the status register being set to 1 (if TI interrupt is enabled, here the trigger of TI interrupt can be waited), then the data is transmitted.

15.4.2 Receive CAN Data Frame

1. Turn on the CAN clock, release the reset, and multiplex the CAN function pins.
2. Configure the bus timing registers CAN_BTR0/CAN_BTR1.
3. Clear the error flag bit/interrupt flag bit in CAN_ISR register.
4. Configure the interrupt enable register IMR to enable RI interrupt (optional).
5. Set the configuration of the RX filter. If a single filter is used, set the AFM bit of CAN_MR register to 1. The CAN_ACR register configures what the user needs to filter, and the CAN_AMR register selects the bits that need to be compared with those of ACR register. If no comparison is required, all bits of AMR register shall be set to 1.
6. Configure the LOM bit of the mode register MR to enter normal mode.
7. Wait for the RBS bit of the status register being set to 1 (if RI interrupt is enabled, here the trigger of RI interrupt can be waited), read the data in RX buffer register CAN_RXBUF several times until all data are retrieved.

16 Analog-to-digital Converter (ADC)

16.1 Main Features

- 12-bit resolution
- Reference voltage: V_{DDA} or external V_{REF} optional
- Up to 8 analog channel inputs, with channels 0–6 as external inputs and channel 7 input source fixed as internal LDO
- Sampling rate: 1 Msps
- ADC voltage reference can be V_{DDH} or external V_{REF} as ADC power reference source (when configured to 1, port P2_5 will be used as the external reference input).

16.2 Register Description

Table 16-1: List of Registers

Address	Name	Description
93H	ADC_IER	Interrupt enable register
ACH	ADC_GCR0	Control enable register
ADH	ADC_GCR1	Power-down enable register
B4H	ADC_GCR2	Configuration register
B5H	ADC_GCR3	Sampling register
B6H	ADC_DR0	Data low-order register
B7H	ADC_DR1	Data high-order register
F5H	ADC_HL	Channel setting register
F6H	ADC_CSTAT	Start register
F7H	ADC_SPW	Sampling clock pulse width configuration register
FDH	ADC_VREF	Voltage reference source selection register
FEH	ADC_CDR0	Clock division register
FFH	ADC_CDR1	Clock division register

16.2.1 ADC_IER Interrupt Enable Register

93H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_IER	-	-	-	-	-	-	-	RXINTEN
R/W	R	R	R	R	R	R	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-1	-	-
0	RXINTEN	RX BUF with valid data interrupt enable bit: 1: interrupt enabled 0: interrupt disabled

16.2.2 ADC_GCR0 Control Register

ACH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_GCR0	-	ADCCLKSEL	ADCRCEN	-	-	-	ADCCUNSET	ADCEN
R/W	R	R/W	R/W	R	R	R	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	-	-
6	ADCCLKSEL	ADC clock source selection: 0: clock generated by internal clock divider 1: clock generated by system clock generator (with inverted phase)
5	ADCRCEN	Enable clearing of ADC data register (ADCDRx) after reading: 0: disable clearing of ADC data register after reading 1: enable clearing of ADC data register after reading
4-2	-	-
1	ADCCUNSET	Continuous mode setting bit: 1: ADC operates in continuous mode. 0: ADC operates in single-shot mode.
0	ADCEN	ADC controller enable bit: 0: ADC module disabled 1: ADC module enabled

16.2.3 ADC_GCR1 Power-down Enable Register

ADH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_GCR1	-	-	-	-	-	ADCSTEN	ADCRST	ADCPDEN
R/W	R	R	R	R	R	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	1	1

Bit No.	Bit Designator	Description
7-3	-	-
2	ADCSTEN	ADC conversion start enable bit: When the signal is converted from low to high, ADC conversion starts. When the signal is converted from high to low, ADC conversion is completed. In continuous mode, this bit is automatically set. When ADC_EN = 0, it will be cleared. Default value: 0
1	ADCRST	ADC internal digital logic reset bit: 1: SAR ADC is reset. 0: SAR ADC is released.
0	ADCPDEN	SAR SAR power-down enable bit: 0: power on SAR ADC 1: power down SAR ADC

16.2.4 ADC_GCR2 Configuration Register

B4H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_GCR2	-	-	ADC_PS	P1_2_SEL	CHEN			
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-4	-	-
5	ADC_PS	Analog ADC parameter configuration
4	P1_2_SEL	When this bit is 1, P1_2 is selected as the sampling channel for channel 0; when this bit is 0, P1_4 is selected as the sampling channel for channel 0.
3-0	CHEN	Enable relevant ADC channels for analog-to-digital conversion. Default value: 0 0001: channel 0

		0010: channel 1 0011: channel 2 0100: channel 3 0101: channel 4 0110: channel 5 0111: channel 6 1000: channel 7 0000: reserved

16.2.5 ADC_GCR3 Sampling Register

B5H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_GCR3	-	-	-	-	-	-	-	SAMPNEG
R/W	R	R	R	R	R	R	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-1	-	-
0	SAMPNEG	Selection of EOC signal edge for sampling ADC data: 0: ADC data sampled at the rising edge of EOC 1: ADC data sampled at the falling edge of EOC Note: This bit can only be set to 0 in this chip design.

16.2.6 ADC_DR0 Data Low-order Register

B6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_DR0	CHDATAL							
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	CHDATAL	A/D channel RX data low-order register

16.2.7 ADC_DR1 Data High-order Register

B7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_DR1	-	-	-	-	CHDATAH			
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	-	-
6-4	-	-
3-0	CHDATAH	A/D channel RX data high-order register

16.2.8 ADC_HL Channel Setting Register

F5H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_HL	-	ADCHL[6:0]						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	-	-
6-0	ADCHL	**ADCH[6:0] channel configuration bit** ADCH[0]: 0: P1_4 configured as GPIO 1: P1_4 configured as ADC input ADCH[1]: 0: P1_5 configured as GPIO 1: P1_5 configured as ADC input ADCH[2]: 0: P2_0 configured as GPIO 1: P2_0 configured as ADC input ADCH[3]: 0: P2_2 configured as GPIO 1: P2_2 configured as ADC input

		ADCH[4]: 0: P2_3 configured as GPIO 1: P2_3 configured as ADC input ADCH[5]: 0: P2_6 configured as GPIO 1: P2_6 configured as ADC input ADCH[6]: 0: P2_7 configured as GPIO 1: P2_7 configured as ADC input

16.2.9 ADC_CSTAT Start Register

F6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_CSTAT	-	-	-	-	-	-	-	RXAVL
R/W	R	R	R	R	R	R	R	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-1	-	-
0	RXAVL	● This signal indicates that BUF has received data. This bit is 1 when RX BUF is not empty. 1: RX BUF with available data 0: RX BUF empty Write 1 to clear this bit. ● This signal facilitates CPU polling operation.

16.2.10 ADC_SPW Sampling Clock Pulse Width Configuration Register

F7H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_SPW	-	-	-	-	-	SAMPW		
R/W	R	R	R	R	R	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	1	1

Bit No.	Bit Designator	Description
7-3	-	-
2-0	SAMPW	Sampling clock pulse width configuration. **Note:** In this chip design, this register shall be set to a value greater than or equal to 3. The legal value range of this register is 3–5, beyond which the ADC may not work properly. 3: SMAPCLK width is 4 ADC_CLK pulse signals. 4: SMAPCLK width is 5 ADC_CLK pulse signals. 5: SMAPCLK width is 6 ADC_CLK pulse signals.

16.2.11 ADC_VREF Voltage Reference Source Selection Register

FDH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_VREF	-	-	ADCSPEED		-	-	-	VREFSEL
R/W	-	-	R/W	R/W	-	-	-	R/W
Reset value	-	-	0	0	-	-	-	0

Bit No.	Bit Designator	Description
7-6	-	-
5-4	ADCSPEED	ADCSPEED selection bit, reserved to the default value of 0.
3-1	-	-
0	VREFSEL	ADC voltage reference selection bit: 0: V_{DDH} as ADC voltage reference 1: external V_{REF} as ADC power reference source (when configured to 1, port P2_5 will be used as the external reference input).

16.2.12 ADC_CDR0 Clock Division Register

FEH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_CDR0	CLKDIV0							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	1	1	1	1	1	1	1	1

Bit No.	Bit Designator	Description
7-0	CLKDIV0	ADC internal clock frequency divider The ADC clock frequency formula is: $f_{adc\_clk} = f_{pclk}$ / {clkdiv1, clkdiv0} Wherein, f_{adc_clk} is the internal clock frequency of ADC, f_{pclk} is the APB clock frequency, and clkdiv is the frequency divider. Note: Do not set clkdiv to 0 or 1, otherwise, it is also regarded as dividing the frequency by 2. If it is required to be divided by 1, an external clock is recommended.

16.2.13 ADC_CDR1 Clock Division Register

FFH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_CDR1	CLKDIV1							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7-0	CLKDIV1	ADC internal clock frequency divider The ADC clock frequency formula is: $f_{adc\_clk} = f_{pclk}$ / {clkdiv1, clkdiv0} Wherein, f_{adc_clk} is the internal clock frequency of ADC, f_{pclk} is the APB clock frequency, and clkdiv is the frequency divider. Note: Do not set clkdiv to 0 or 1, otherwise, it is also regarded as dividing the frequency by 2. If it is required to be divided by 1, an external clock is recommended.

16.3 Process Description

Initialization:

1. Configure ADCCEN of PCLK0 to 1 to enable ADC clock.
2. Configure ADCREN of PRESET0 to 1 to enable ADC reset release.
3. Configure ADCCLKSEL of ADCGCR0 to select the ADC clock source. If the clock generated by the internal clock divider is selected, configure ADCCDR0 and ADCCDR1 to set the division value of the divider.
4. Configure VREFSEL of ADCVREF to select the ADC voltage reference.
5. Configure SAMPW of ADCSPW to set the sampling clock pulse width.
6. Configure ADCHL to configure the GPIO corresponding to the desired ADC channel as ADC input; if channel 0 is used as the sampling channel, it is required to configure P1_2_SEL of ADCGCR2 and select P1_2 or P1_4 as channel 0.
7. Configure ADCRCEN of ADCGCR0 to disable or enable the clearing of ADC data register after reading.
8. Configure ADCCUNSET of ADCGCR0 to select the operation mode of ADC.
9. Configure SAMPNEG of ADCGCR3 to 0 to configure EOC rising edge sampling.
10. Configure ADCPDEN of ADCGCR1 to 0 to power on the SAR ADC.
11. Configure ADCRST of ADCGCR1 to 0 to enable SAR release.
12. Configure ADCEN of ADCGCR0 to 1 to enable ADC controller.
13. If ADC interrupt is required, configure IP0 and IP1 of IP and set the priority of ADC interrupt.
14. Configure RXINTEN of ADCIER to 1 to enable interrupt of ADC RX BUF with valid data.
15. Configure EADC and EA of IEN0 to 1 to enable the ADC interrupt and the total interrupt.

ADC sampling in single-shot mode:

1. Configure CHEN of ADCGCR2 to enable relevant ADC channels for analog-to-digital conversion.

2. Configure ADCSTEN of ADCGCR1, set the signal to 0 first and then to 1 to generate a low-to-high signal conversion, thus the ADC conversion starts.
3. Read the status of ADCSTEN of ADCGCR1. When ADCSTEN is set to 0, the conversion is completed.
4. Read the status of RXAVL of ADCCSTAT. When RXAVL is set to 1, it means that there is data in RXBUF. After setting to 1, it shall be cleared by writing 1.
5. After the conversion is completed and the RX BUF has data, read the channel data in ADCDR0 and ADCDR1.
6. In continuous mode, it is required to configure ADCEN of ADCGCR0 to 0 to stop the conversion.

Notes:

1. The sampling rate of ADC is set as 1 Msps, ADC sampling rate = f(ADCCLK) / (sampling time + conversion time) = 16 MHz / (4clk + 12clk) = 1 Msps. When the clock generated by the system clock generator is selected as the clock source, the frequency division of ADCCDR0 and ADCCDR1 is invalid.
2. Channel 7 is the internal LDO channel.

17 LVD

17.1 Overview

LVD refers to the low-voltage detection function that can filter the detection results and enhance the system stability.

17.2 Register Description

Table 17-1: List of Registers

Address	Name	Description
DDH	LVD_CON	Enable register
A2H	OINTEN	Interrupt enable register
A3H	OINTUS	Interrupt status register
A6H	LVD_OSTATUS	Status register
D8H	LVD_RSTSTAT	Reset register
C004H	LVD_LV	Filter enable register

17.2.1 LVD_CON Enable Register

DDH	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LVD_CON	LVDF	-	LVDREN	LVDEN	LVDS[3:1]			-
RW	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0	0	0	0	0

Bit No.	Bit Designator	Description
7	LVDF	LVD flag bit: 0: low voltage not detected 1: low voltage detected
6	-	-
5	LVDREN	LVD reset enable bit: 0: disable reset at detection of low voltage 1: enable reset at detection of low voltage, before which LVDEN

<table>
<tr><td></td><td></td><td>must be enabled first</td></tr>
<tr><td>4</td><td>LVDEN</td><td>LVD module enable bit:
0: LVD module enabled
1: LVD module disabled</td></tr>
<tr><td>3-1</td><td>LVDS[3:0]</td><td>Setting of LVD point voltage:<table><tr><th>LVDS</th><th>LVD point</th></tr><tr><td>000</td><td>4.39 V</td></tr><tr><td>001</td><td>3.95 V</td></tr><tr><td>010</td><td>3.59 V</td></tr><tr><td>011</td><td>3.29 V</td></tr><tr><td>100</td><td>3.04 V</td></tr><tr><td>101</td><td>2.82 V</td></tr><tr><td>110</td><td>2.63 V</td></tr><tr><td>111</td><td>2.46 V</td></tr></table></td></tr>
<tr><td>0</td><td>-</td><td>-</td></tr>
</table>

17.2.2 OINTEN Interrupt Enable Register

Please refer to Chapter “OINTEN Interrupt Enable Register” for details.

17.2.3 OINTUS Interrupt Status Register

Please refer to Chapter “OINTEN Interrupt Enable Register” for details.

17.2.4 LVD_OSTATUS Status Register

A6H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LVD_OSTATUS	-		BOOTLOCK	C_NVRLOCK	LVDLOW	NVR2LOCK	NVR1LOCK	EFCREADY
R/W	R	R	R	R	R	R	R	R
Reset value	0	0	0	0	0	0	0	1

Bit No.	Bit Designator	Description
7-6	-	-
5	BOOTLOCK	Whether the BOOT area is locked: 1: BOOT area locked

		0: BOOT area not locked
4	C_NVRLOCK	Whether the C_NVR area is locked: 1: C_NVR area locked 0: C_NVR area not locked
3	LVDLOW	0: voltage detected being normal 1: voltage detected being too low This bit reflects the real-time status of the current low-voltage detection.
2	NVR1LOCK	Whether the NVR1 area is locked: 1: NVR1 area locked 0: NVR1 area not locked
1	NVR0LOCK	Whether the NVR0 area is locked: 1: NVR0 area locked 0: NVR0 area not locked
0	EFCREADY	EFlash status indicator, indicating the working status of EFlash: 1: EFlash being idle 0: EFlash being busy

17.2.5 lVD_RSTSTAT Reset Register

D8H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LVD_RSTSTAT	WDOF	WDEN	LVDRF	PORF	ERSTF	WDT[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value (POR)	0	0	x	1	x	0	0	0
Reset value (WDT)	1	0	x	x	x	0	0	0
Reset value (PIN)	x	0	x	x	1	0	0	0
Reset value (LVD)	x	0	1	x	x	0	0	0

Bit No.	Bit Designator	Description
7	WDOF	Watchdog overflow flag bit: set to 1 by hardware when the watchdog overflows, and can be cleared by software or by power-on reset. 0: no WDT overflow 1: WDT overflow occurred
6	WDEN	Watchdog enable bit: 0: watchdog function disabled

		1: watchdog function enabled
5	LVDRF	LVD reset flag bit: set to 1 by hardware after LVD reset, and cleared by software. 0: no low-voltage reset occurred 1: low-voltage reset occurred
4	PORF	Power on reset flag bit: set to 1 by hardware after power-on reset, and cleared by software. 0: no power-on reset occurred 1: power-on reset occurred
3	ERSTF	Pin reset flag bit: set to 1 by hardware after pin reset, and cleared by software. 0: no pin reset occurred 1: pin reset occurred
2-0	WDT[2:0]	WDT overflow period control bit: 000: minimum overflow period = 4096 ms 001: minimum overflow period = 1024 ms 010: minimum overflow period = 256 ms 011: minimum overflow period = 128 ms 100: minimum overflow period = 64 ms 101: minimum overflow period = 16 ms 110: minimum overflow period = 4 ms 111: minimum overflow period = 1 ms

17.2.6 LVD_LV Filter Enable Register

C004H	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
LVD_LV	-	-	-	-	-	LVD_TSET		LVDLVEN
R/W	R	R	R	R	R	R/W	R/W	R/W
Reset value	0	0	0	0	0	1	1	1

Bit No.	Bit Designator	Description
7-3	-	-
2-1	LVD_TSET	Filter time setting bit: 11: filter time of one RC38K cycle 10: filter time of 8 RC38K cycles 01: filter time of 16 RC38K cycles 00: filter time of 29 RC38K cycles
0	LVDLVEN	LVD filter enable bit: 1: RC38K filter function enabled 0: RC38K filter function disabled

18 Interrupt

18.1 Main Features

- 7 interrupt sources: EX0, ES1, ES0, EPWM, EADC, EFC, SPI
- 4 interrupt priorities configurable

18.2 Interrupt Summary

Interrupt Source	Entry Address	Enable Bit	Flag Bit	Polling Priority	Interrupt Number (C Language)
Reset	0000H	-	-	0 (top priority)	-
INT0	0003H	EX0 + PxIENy	PxIRQy	1	0
UART1	0013H	ES1	RI1 + TI1	8	2
UART0	0023H	ES0	RI0 + TI0	12	4
PWM	002BH	EPWM + PWMxIE	PWMxIF	15	5
ADC	0033H	EADC + ADCIER	RXAVL	2	6
SPI	003BH	ESPI + SPIIE	SPI_SR	6	7
CAN	0043H	CANINTEN	CAN_ISR	4	8
EFC	005BH	EFCINTEN + OINTEN	OINTUS	11	11
LPTIMER	0063H	LPTIMINTEN + LPTIE	LPTIMER_IF	14	12
I2C	006BH	I2CINTEN + I2CCR	-	17	13
UART2	0083H	UART2INTEN	UART_ISR	3	16
UART3	008BH	UART3INTEN	UART_ISR	7	17
GTIMER2	009BH	GTIMER2INTEN	GTIMER_SR	10	19
GTIMER1	00A3H	GTIMER1INTEN	GTIMER_SR	13	20
GTIMER0	00ABH	GTIMER0INTEN	GTIMER_SR	16	21

19 Instruction Set

The machine cycle is one clock cycle and most instructions take only one machine cycle to execute.

19.1 Instruction Operands Description

Rn	Working registers R0–R7
direct	256 internal RAM locations, any special function register
@Ri	Indirect internal or external RAM location addressed by register R0 or R1
#data	8-bit constant included in instruction
#data 16	16-bit constant included as bytes 2 and 3 of instruction
bit	256 software flags, any bit-addressable I/O pin, control or status bit
A	Accumulator
addr16	Destination address for LCALL and LJMP may be anywhere within the 64K bytes of program memory address space
addr11	Destination address for ACALL and AJMP will be within the same 2K bytes page of program memory as the first byte of the following instruction
rel	SJMP and all conditional jumps include an 8-bit offset byte. Range is +127/–128 bytes relative to the first byte of the following instruction

19.2 Arithmetic Operation Instruction

Mnemonic	Description	Code	Byte	Cycle
ADD A, Rn	Add register to accumulator	28-2F	1	1
ADD A, direct	Add direct byte to accumulator	25	2	2
ADD A, @Ri	Add indirect RAM to accumulator	26-27	1	2
ADD A, #data	Add immediate data to accumulator	24	2	2
ADDC A, Rn	Add register to accumulator with carry flag	38-3F	1	1
ADDC A, direct	Add direct byte to A with carry flag	35	2	2
ADDC A, @Ri	Add indirect RAM to A with carry flag	36-37	1	2
ADDC A, #data	Add immediate data to A with carry flag	34	2	2
SUBB A, Rn	Subtract register from A with borrow	98-9F	1	1
SUBB A, direct	Subtract direct byte from A with borrow	95	2	2
SUBB A, @Ri	Subtract indirect RAM from A with borrow	96-97	1	2

SUBB A, #data	Subtract immediate data from A with borrow	94	2	2
INC A	Increment accumulator	04	1	1
INC Rn	Increment register	08-0F	1	2
INC direct	Increment direct byte	05	2	3
INC @Ri	Increment indirect RAM	06-07	1	3
INC DPTR	Increment data pointer	A3	1	1
DEC A	Decrement accumulator	14	1	1
DEC Rn	Decrement register	18-1F	1	2
DEC direct	Decrement direct byte	15	2	3
DEC @Ri	Decrement indirect RAM	16-17	1	3
MUL AB	Multiply A and B	A4	1	5
DIV	Divide A by B	84	1	5
DA A	Decimal adjust accumulator	D4	1	1

19.3 Logic Operation Instruction

Mnemonic	Description	Code	Byte	Cycle
ANL A, Rn	AND register to accumulator	58-5F	1	1
ANL A, direct	AND direct byte to accumulator	55	2	2
ANL A, @Ri	AND indirect RAM to accumulator	56-57	1	2
ANL A, #data	AND immediate data to accumulator	54	2	2
ANL direct, A	AND accumulator to direct byte	52	2	3
ANL direct, #data	AND immediate data to direct byte	53	3	4
ORL A, Rn	OR register to accumulator	48-4F	1	1
ORL A, direct	OR direct byte to accumulator	45	2	2
ORL A, @Ri	OR indirect RAM to accumulator	46-47	1	2
ORL A, #data	OR immediate data to accumulator	44	2	2
ORL direct, A	OR accumulator to direct byte	42	2	3
ORL direct, #data	OR immediate data to direct byte	43	3	4
XRL A, Rn	Exclusive OR register to accumulator	68-6F	1	1
XRL A, direct	Exclusive OR direct byte to accumulator	65	2	2
XRL A, @Ri	Exclusive OR indirect RAM to accumulator	66-67	1	2
XRL A, #data	Exclusive OR immediate data to accumulator	64	2	2
XRL direct, A	Exclusive OR accumulator to direct byte	62	2	3
XRL direct, #data	Exclusive OR immediate data to direct byte	63	3	4
CLR A	Clear accumulator	E4	1	1
CPL A	Complement accumulator	F4	1	1

RL A	Rotate accumulator left	23	1	1
RL A	Rotate accumulator left through carry	33	1	1
RR A	Rotate accumulator right	03	1	1
RRC A	Rotate accumulator right through carry	13	1	1
SWAP A	Swap nibbles within the accumulator	C4	1	1

19.4 Data Transfer Instruction

Mnemonic	Description	Code	Byte	Cycle
MOV A, Rn	Move register to accumulator	E8-EF	1	1
MOV A, direct	Move direct byte to accumulator	E5	2	2
MOV A, @Ri	Move indirect RAM to accumulator	E6-E7	1	2
MOV A, #data	Move immediate data to accumulator	74	2	2
MOV Rn, A	Move accumulator to register	F8-FF	1	2
MOV Rn, direct	Move direct byte to register	A8-AF	2	4
MOV Rn, #data	Move immediate data to register	78-7F	2	2
MOV direct, A	Move accumulator to direct byte	F5	2	3
MOV direct, Rn	Move register to direct byte	88-8F	2	3
MOV direct1, direct2	Move direct byte to direct byte	85	3	4
MOV direct, @Ri	Move indirect RAM to direct byte	86-87	2	4
MOV direct, #data	Move immediate data to direct byte	75	3	3
MOV @Ri, A	Move accumulator to indirect RAM	F6-F7	1	3
MOV @Ri, direct	Move direct byte to indirect RAM	A6-A7	2	5
MOV @Ri, #data	Move immediate data to indirect RAM	76-77	2	3
MOV DPTR, #data16	Load data pointer with a 16-bit constant	90	3	3
MOVC A, @A+DPTR	Move code byte relative to DPTR to accumulator	93	1	3
MOVC A, @A+PC	Move code byte relative to PC to accumulator	83	1	3
MOVX A, @Ri	Move external RAM (8-bit addr.) to A	E2-E3	1	3-10
MOVX A, @DPTR	Move external RAM (16-bit addr.) to A	E0	1	3-10
MOVX @Ri, A	Move A to extern RAM (8-bit addr.)	F2-F3	1	4-11
MOVX @DPTR, A	Move A to extern RAM (16-bit addr.)	F0	1	4-11
PUSH direct	Push direct byte onto stack	C0	2	4
POP direct	Pop direct byte from stack	D0	2	3
XCH A, Rn	Exchange register with accumulator	C8-CF	1	2
XCH A, direct	Exchange direct byte with accumulator	C5	2	3
XCH A, @Ri	Exchange indirect RAM with accumulator	C6-C7	1	3
XCHD A, @Ri	Exchange low-order nibble indir.RAM with A	D6-D7	1	3

19.5 Control Program Jump Instruction

Mnemonic	Description	Code	Byte	Cycle
ACALL addr11	Absolute subroutine call	xxx11	2	6
LCALL addr16	Long subroutine call	12	3	6
RET	from subroutine	22	1	4
RETI	from interrupt	32	1	4
AJMP addr11	Absolute jump	xxx01	2	3
LIMP addr16	Long jump	02	3	4
SJMP rel	Short jump (relative addr.)	80	2	3
JMP @A + DPTR	Jump indirect relative to the DPTR	73	1	2
JZ rel	Jump if accumulator is zero	60	2	3
JNZ rel	Jump if accumulator is not zero	70	2	3
JC rel	Jump if carry flag is set	40	2	3
JNC	Jump if carry flag is not set	50	2	3
JB bit, rel	Jump if direct bit is set	20	3	4
JNB bit, rel	Jump if direct bit is not set	30	3	4
JBC bit, direct rel	Jump if direct bit is set and clear bit	10	3	4
CJNE A, direct rel	Compare direct byte to A and jump if not equal	B5	3	4
CJNE A, #data rel	Compare immediate to A and jump if not equal	B4	3	4
CJNE Rn, #data rel	Compare immed. to reg. and jump if not equal	B8-BF	3	4
CJNE @Ri, #data rel	Compare immed. to ind. and jump if not equal	B6-B7	3	4
DJNE Rn, rel	Decrement register and jump if not zero	D8-DF	2	3
DJNZ direct, rel	Decrement direct byte and jump if not zero	D5	3	4
NOP	No operation	00	1	1

19.6 Bitwise Operation Instruction

Mnemonic	Description	Code	Byte	Cycle
CLR C	Clear carry flag	C3	1	1
CLR bit	Clear direct bit	C2	2	3
SETB C	Set carry flag	D3	1	1
SETB bit	Set direct bit	D2	2	3
CPL C	Complement carry flag	B3	1	1
CPL bit	Complement direct bit	B2	2	3
ANL C, bit	AND direct bit to carry flag	82	2	2

ANL C, /bit	AND complement of direct bit to carry	B0	2	2
ORL C, bit	OR direct bit to carry flag	72	2	2
ORL C, /bit	OR complement of direct bit to carry	A0	2	2
MOV C, bit	Move direct bit to carry flag	A2	2	2
MOV bit, C	Move carry flag to direct bit	92	2	3

20 Power Supply Scheme

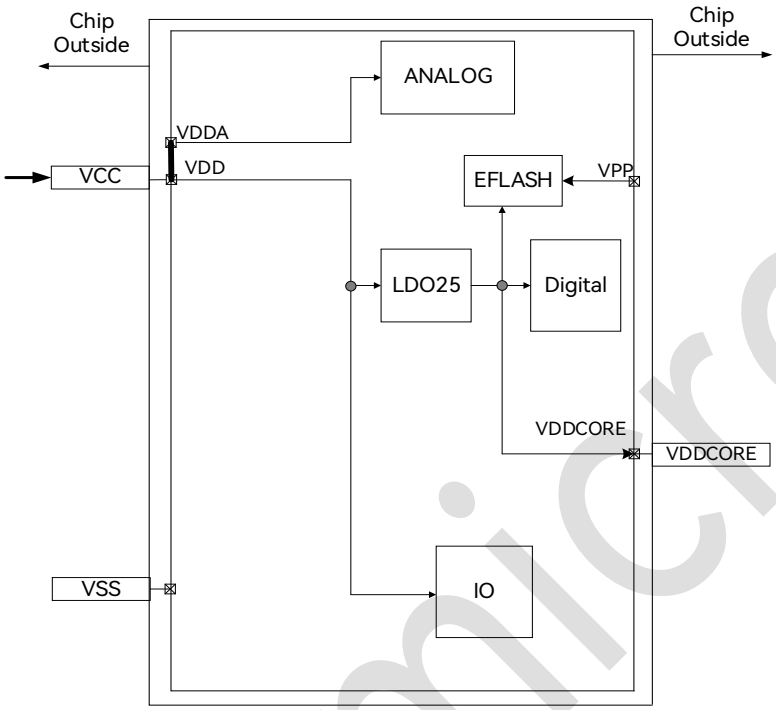


Figure 20-1: Power Supply Scheme Diagram

21 Revision History

Version	Date	Changes
V1.0	Jan-12-2022	Initial release.
V1.1	Feb-10-2022	1. Unified the register naming style, and modified some register parameters. 2. Updated QFN20 package outline drawing. 3. Updated pin descriptions. 4. Updated TSSOP20 package outline drawing. 5. Updated DC parameters. 6. Deleted descriptions regarding T0/T1. 7. Updated parameters of the internal RCL oscillator.
V1.2	Mar-11-2022	1. Updated the pinout diagram. 2. Added the chapter of Alternate Function. 3. Updated pin descriptions. 4. Updated parameters of the internal RCH oscillator. 5. Deleted the chapter of Address Mapping. 6. Moved registers in the chapter of System Configuration to the chapters of corresponding modules, unified the register naming style, and added the corresponding register list. 7. Updated the clock architecture diagram. 8. Updated descriptions regarding address mapping after remapping. 9. Deleted descriptions regarding address mapping before remapping.
V1.3	Mar-28-2022	1. Added the chapter of Operating Condition at Power-up and Power-down. 2. Updated parameters of the internal RCH oscillator. 3. Updated the operating temperature range. 4. Updated the chapter on EFC (deleted the NVR related descriptions and the chapter of EFC Prefetch at Power-up).
V1.4	Apr-22-2022	1. Updated the SOP8 package outline drawing. 2. Modified the value of 1.5 V into 2.5 V for the whole document. 3. Modified the setting of LVD detection point voltage. 4. Deleted the setting of LVR detection point voltage. 5. Modified the description for bit 7 of P13_CFG[2:0] into PWM2.

		6. Updated parameters of Pxx_CFG register. 7. Added Chapter 6.3 Parameter Address.
V1.5	Jul-14-2022	1. Deleted the chapter of port alternate function. 2. Updated some chapter descriptions. 3. Adjusted the structure of the electrical characteristics chapter.
V1.6	Jan-05-2023	Deleted OSEC register and descriptions thereof.
V1.7	May-06-2024	1. Deleted chapters of “Pin Description”, “Package Information” and “Electrical Characteristics”. 2. Updated descriptions in the chapter of “Watchdog Reset”.
V1.7.1	Aug-26-2024	Updated the LVD point values in “17.2.1 LVD_CON Enable Register”.

