1. BF7512CMXX-SJLX MCU General Description

1.1. Features

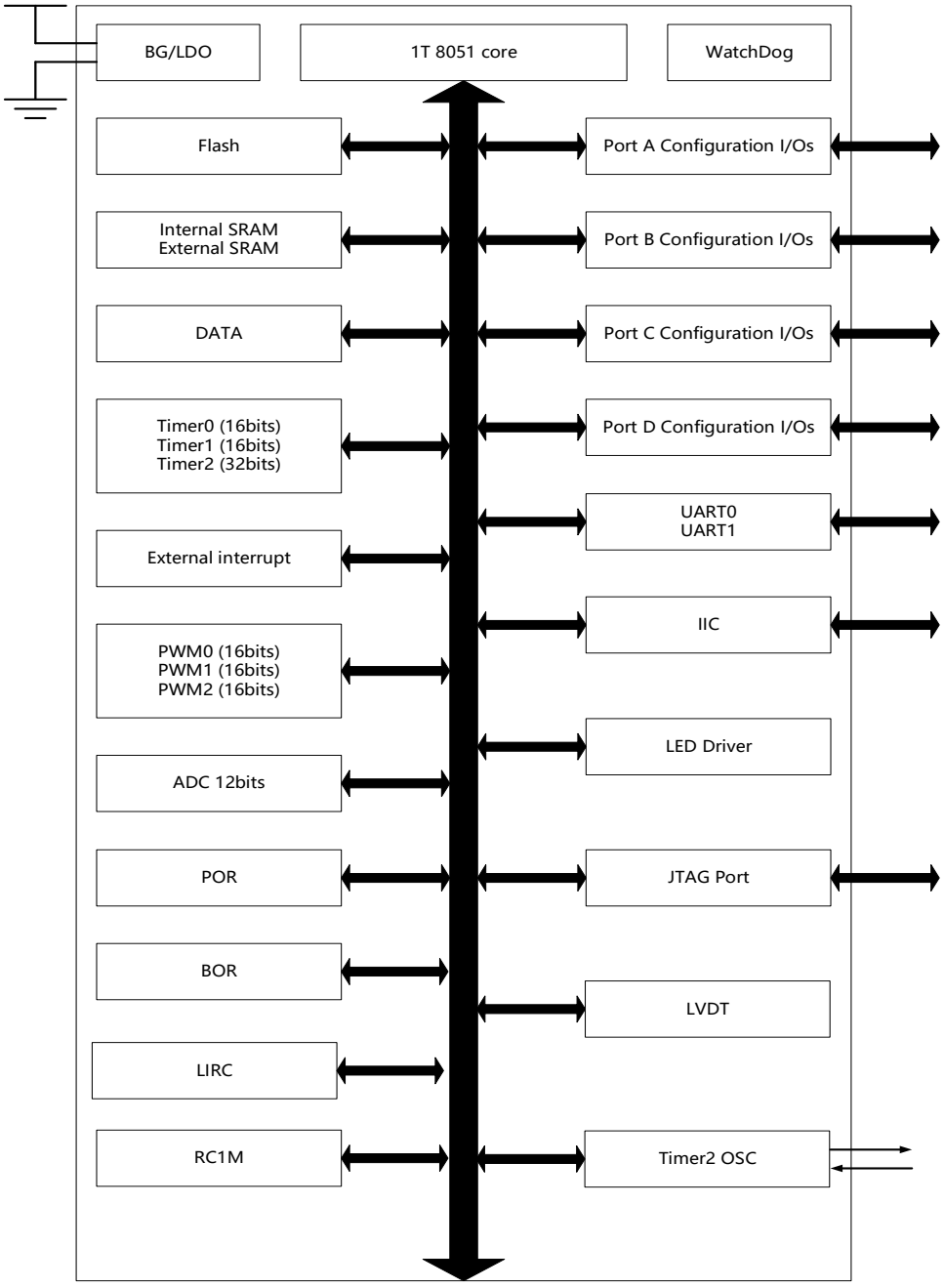
- **Core: High-speed 8051, 1T instruction cycle**
  - Operating Frequency: 12MHz, 6MHz, 4MHz
  - Clock error: ±1% @27℃, 5V
    ±3% @-40℃~105℃, 5V
- **Memory(FLASH)**
  - CODE: 15K Bytes
  - DATA: 1K Bytes +2*512 Bytes
  - SRAM: 256Bytes(data)+512 Bytes(xdata)
- **Clock source, reset and power management**
  - Internal low-speed clock LIRC: 32.768kHz
    Clock error: ±15% @27℃, 5V
    ±35% @-40℃~105℃, 5V
  - Internal high-speed RC oscillator: 1MHz
  - External crystal oscillator: 32768Hz/4MHz
  - 7 resets, including brown-out reset voltage 2.1V
  - Low voltage detection: 2.4V/3.0V/3.6V/4.2V
- **IO**
  - Support built-in pull up resister 4.7k
  - High current sink port (PB0~PB7)
  - Support device peripheral function multiplexing
  - Support external interrupt function, INT0~2 external interrupt (rising edge, falling edge, double edge)
- **Communication Module**
  - UART communication (UART0/1), support IO mapping
  - IIC slave mode, support 100/400kHz
- **16-bit PWM**
  - PWM0 supports 4 channels, same frequency and different duty cycle
  - PWM1 and PWM 2 support 1 channel
- **Operating Voltage: 2.5V~5.5V**
- **Operating Temperature: -40℃~105℃**
  - Enhanced industrial grade, in line with JESD industrial grade reliability certification standards
- **12-bit High-speed ADC**
- Up to 26 analog input channels
- **Interrupt**
  - Two-level interrupt priority capablity
  - ADC, LED, INT0/1/2, LVDT, Timer0~2, WDT, UART0/1, IIC
- **Timer**
  - 16-bit Timer0/1, 32-bit Timer2
  - Timer2 clock source is internal low-speed clock LIRC 32k or XTAL 32768Hz /4MHz
  - Watchdog timer, overflow time 18ms to 2.304s
- **LED Driver**
  - 8*8, 7*8, 7*7, 6*7, 6*6, 5*5, 4*4 serial dot matrix driver
- **Low power mode**
  - Idle mode and sleep mode
  - Deep sleep, power consumption 26μA @5V typical
- **With JTAG debug simulation interface**
- **Package**
  - SOP16/SOP20/SOP28

1.2. Overview

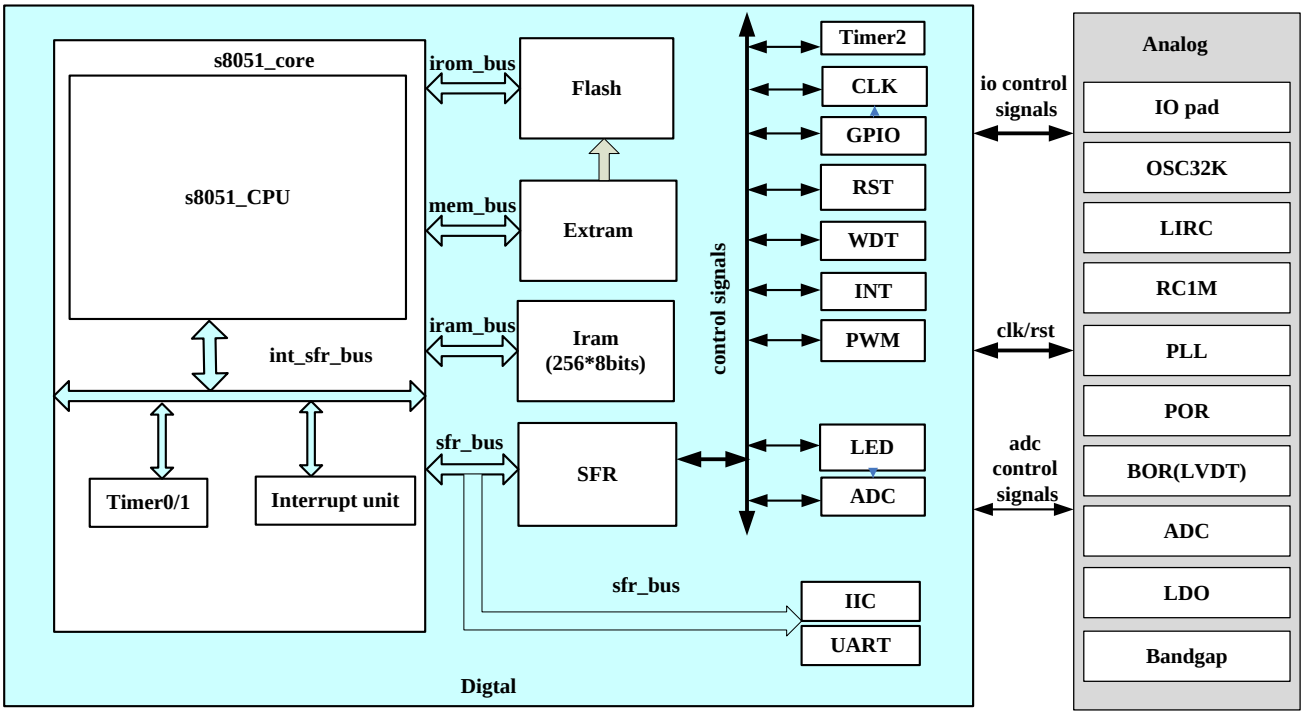
BF7512CMXX-SJLX uses the high speed 8051 core with 1T instruction cycle, compared to the standard 8051 (12T) instruction cycle, has the quicker running speed, compatibility standard 8051 instruction.

BF7512CMXX-SJLX includes a watchdog, LED serial dot matrix driver, IIC, UART, low voltage detection, power down reset, 3-channe l6bit PWM, Timer0, Timer1, Timer2, 12bit successive approximation ADC, low power mode.

1.3. System Architecture

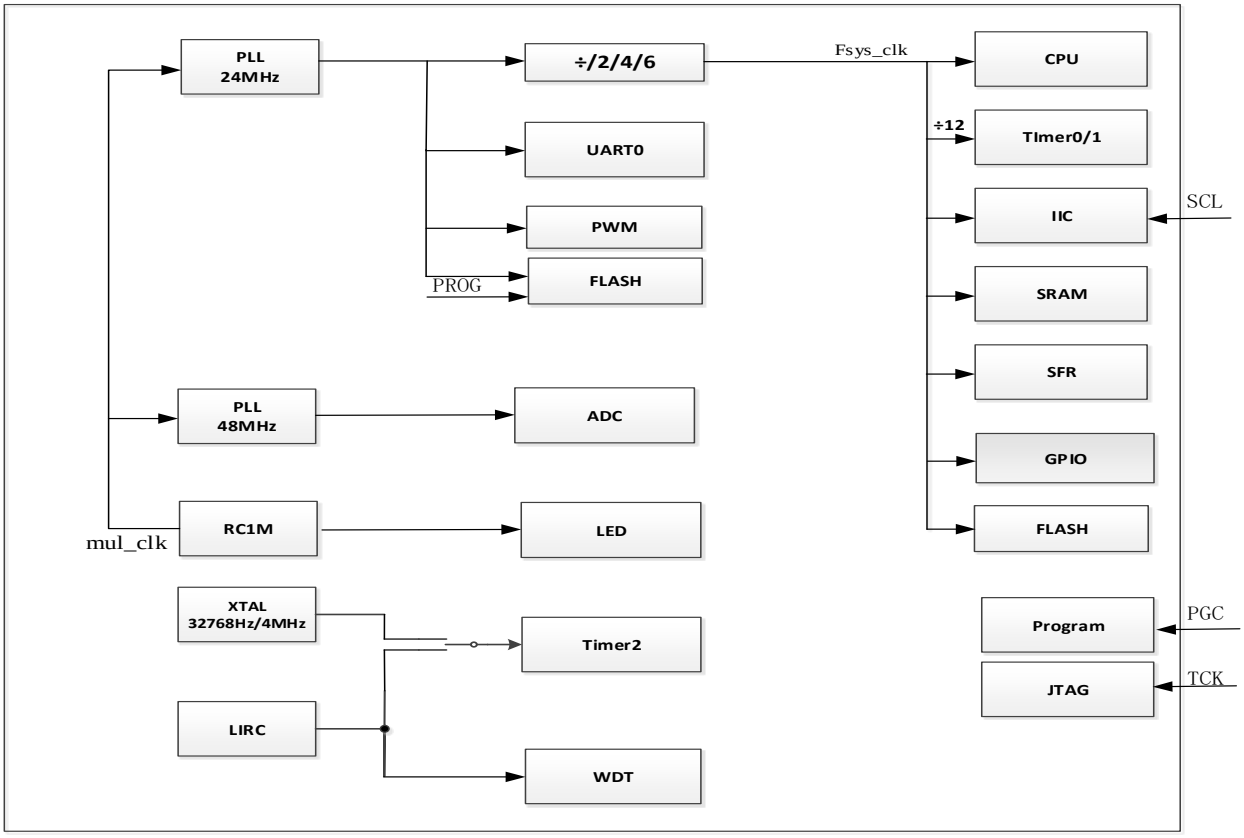


System Architecture



System bus frame diagram

1.4. Clock Diagram



Clock Diagram

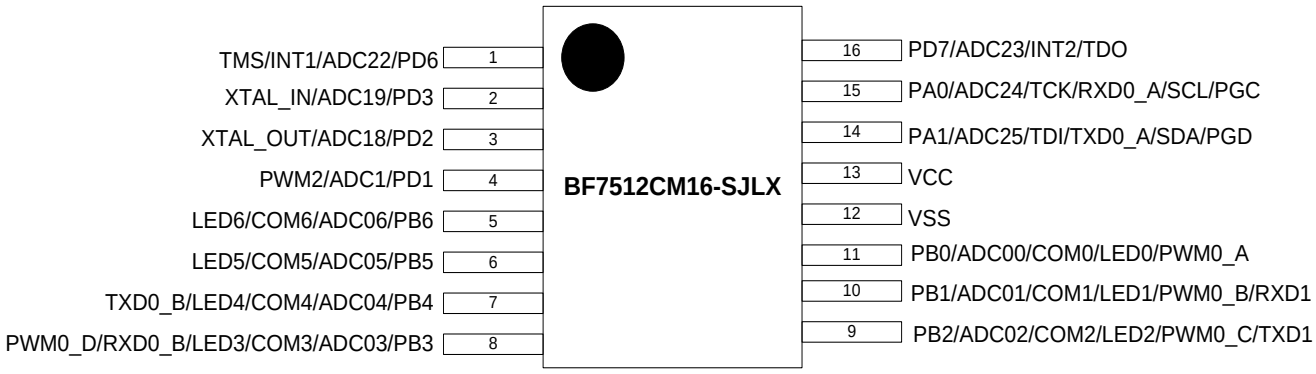
1.5. Selection List

Type		BF7512CM16-SJLX	BF7512CM20-SJLX	BF7512CM28-SJLX
Operating voltage (V)		2.5~5.5	2.5~5.5	2.5~5.5
Operating frequency (Hz)		12M	12M	12M
Core		1T 8051	1T 8051	1T 8051
Storage module (Bytes)	CODE	15K	15K	15K
	DATA	1K + 2*512	1K + 2*512	1K + 2*512
	SRAM	256+512	256+512	256+512
Timer	WDT	1	1	1
	Timer0*16bit	1	1	1
	Timer1*16bit	1	1	1
	Timer2*32bit	1	1	1
Communication module	IIC	1	1	1
	UART	2	2	2
GPIO		14	18	26
INT		2	3	3
COM		7	8	8
Analog module	ADC*12bit	14	18	26
Display module	LED serial	6*7	7*8	8*8
PWM	PWM0*16bit	4	4	4
	PWM1*16bit	0	1	1
	PWM2*16bit	1	0	1
Package		SOP16	SOP20	SOP28

Selection list

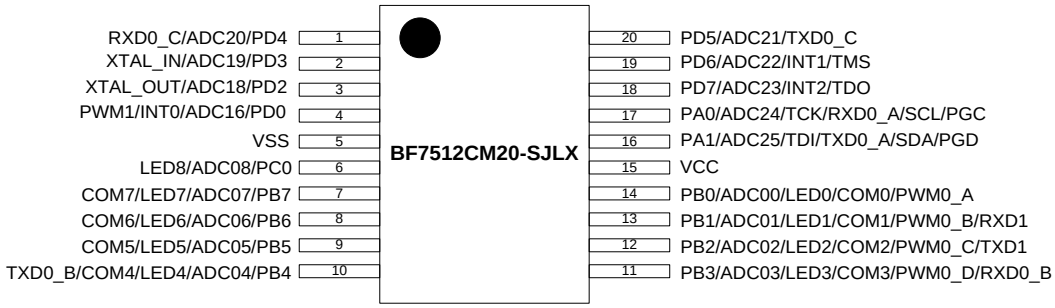
1.6. Pin Assignment

1.6.1. BF7512CM16-SJLX



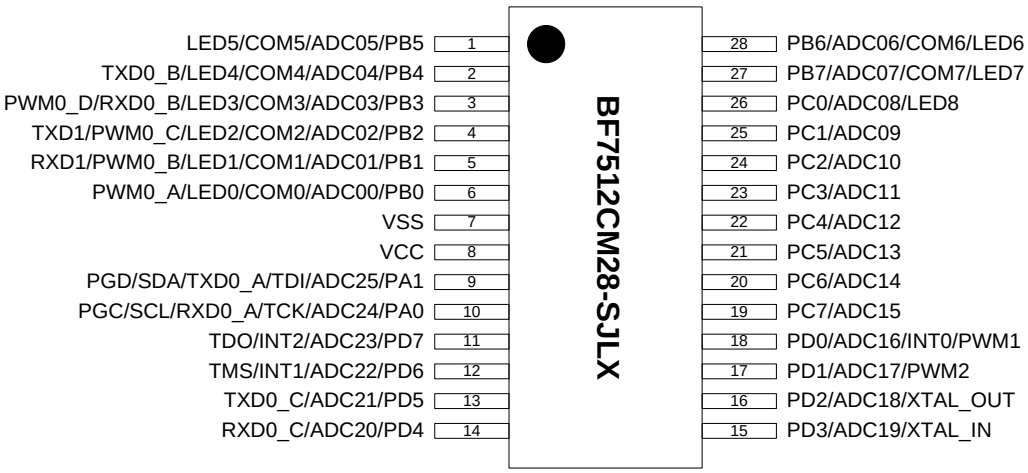
BF7512CM16-SJLX SOP16 package pin diagram

1.6.2. BF7512CM20-SJLX



BF7512CM20-SJLX SOP20 package pin diagram

1.6.3. BF7512CM28-SJLX



BF7512CM28-SJLX SOP28 package pin diagram

1.7. Pin Description

BF7512CM28-SJLX	BF7512CM20-SJLX	BF7512CM16-SJLX	Function description
14	1	-	Default function: GPIO <PD4> Other function: ADCXX: ADC channel RXDXX: serial pot reception
15	2	2	Default function: GPIO <PD3> Other function: ADCXX: ADC channel XTAL0_IN: External crystal input
16	3	3	Default function: GPIO <PD2> Other function: ADCXX: ADC channel XTAL0_OUT: External crystal output
17	-	4	Default function: GPIO <PD1> Other function: ADCXX: ADC channel PWMXX: PWM output port
18	4	-	Default function: GPIO <PD0> Other function: ADCXX: ADC channel INTXX: External Interrupt PWMXX: PWM output port
25	-	-	Default function: GPIO <PC1> Other function: ADCXX: ADC channel
26	6	-	Default function: GPIO <PC0> Other function: ADCXX: ADC channel LEDX: LED serial dot matrix
27	7	-	Default function: GPIO <PB7> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix
28	8	5	Default function: GPIO <PB6> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix
1	9	6	Default function: GPIO <PB5> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix

2	10	7	Default function: GPIO <PB4> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix TXDXX: serial pot transmission
3	11	8	Default function: GPIO <PB3> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix RXDXX: serial pot reception PWMXX: PWM output port
4	12	9	Default function: GPIO <PB2> Other function: ADCXX: ADCchannel COMX: Large current sink LEDX: LED serial dot matrix PWMXX: PWM output port TXDXX: serial pot transmission
5	13	10	Default function: GPIO <PB1> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix PWMXX: PWM output port RXDXX: serial pot reception
6	14	11	Default function: GPIO <PB0> Other function: ADCXX: ADC channel COMX: Large current sink LEDX: LED serial dot matrix PWMXX: PWM output port
7	5	12	Default function: GND <VSS>
8	15	13	Default function: Power supply <VCC>
9	16	14	Default function: GPIO <PA1> Other function: ADCXX: ADC channel TDI: JTAG emulation test data serial input TXDXX: serial pot transmission SDAXX: IIC serial data line PGD: Burning port PGD
10	17	15	Default function: GPIO <PA0> Other function: ADCXX: ADC channel TCK: JTAG simulation test clock RXDXX: serial pot reception

			SCLXX: Serial clock line of IIC PGC: Burning port PGC
11	18	16	Default function: GPIO <PD7> Other function: ADCXX: ADC channel INTXX: External Interrupt TDO: JTAG emulation test data serial output
12	19	1	Default function: GPIO <PD6> Other function: ADCXX: ADC channel INTXX: External Interrupt TMS: JTAG simulation test mode selection
13	20	-	Default function: GPIO <PD5> Other function: ADCXX: ADC channel TXDXX: serial pot transmission
19	–	–	Default function: GPIO <PC7> Other function: ADCXX: ADC channel
20	–	–	Default function: GPIO <PC6> Other function: ADCXX: ADC channel
21	–	–	Default function: GPIO <PC5> Other function: ADCXX: ADC channel
22	–	–	Default function: GPIO <PC4> Other function: ADCXX: ADC channel
23	–		Default function: GPIO <PC3> Other function: ADCXX: ADC channel
24	–	–	Default function: GPIO <PC2> Other function: ADCXX: ADC channel

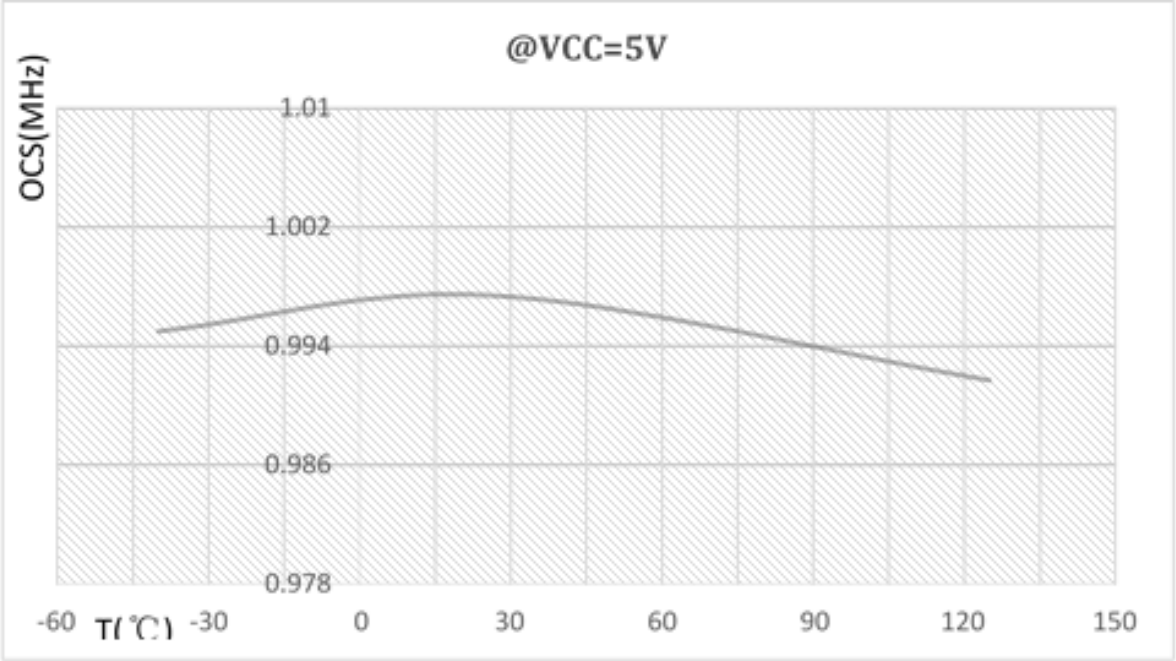
Package pin correspondence diagram

2. Electrical Characteristics

2.1. AC Characteristics

Parameter	Symbol	Conditions	Clock error	Units
RC	RC1M	Ambient temperature 27℃	±1%	MHz
		Ambient temperature -40℃~105℃	±3%	
System clock	Fsys_clk	Ambient temperature 27℃	±1%	
		Ambient temperature -40℃~105℃	±3%	
WDT clock	LIRC	Ambient temperature 27℃	±15%	KHz
		Ambient temperature -40℃~105℃	±35%	

AC characteristics parameters table



OSC temperature characteristic curve

2.2. DC Characteristics

Unless otherwise stated, typical values are test values at 27℃

Parameter	Symbol	Conditions	Min	Typical	Max	Units
Operating voltage	VCC		2.5	-	5.5	V
Working mode	Active	@5V, System clock 12M, no load, turn off other functiuons.	-	2.1	2.7	mA
		@5V, System clock 6M, no load, turn off other functiuons.	-	1.7	2.2	mA
		@5V, System clock 4M, no load, turn off other functiuons.	-	1.5	2.0	mA
		@3.3V, System clock 12M, no load, turn off other functiuons.	-	2.0	2.6	mA
		@3.3V, System clock 6M, no load, turn off other functiuons.	-	1.6	2.0	mA
		@3.3V, System clock 4M, no load, turn off other functiuons.	-	1.4	1.8	mA
	Idle	@5V,WDT_CTRL=7, wake up, 2ms work time, IO set to low, turn off other functions.	-	29	38	μA
		@5V, Timer2 external crystal 2s wake up, 2ms work time, IO set to low, turn off other functions.	-	29	38	μA
		@3.3V, WDT_CTRL=7, WDT wake up, 2ms work time, IO set to low, turn off other functions.	-	28	39	μA
		@3.3V, Timer2 external crystal 2s wake up, 2ms work time, IO set to low, turn off other functions.	-	28	39	μA
	Sleep	@5V PCON = 0x01, turn off BOR, IO set to low, turn off other functions.	-	26	35	μA
		@3.3V PCON = 0x01, turn off BOR, IO set to low, turn off other fu nctions.	-	27	36	μA
Input Low level	V_{IL}	VCC=3.3~5.5V	-	-	0.3*VCC	V
Input High level	V_{IH}	VCC=3.3~5.5V	0.7*VCC	-	-	V
INT0/1/2	V_{INTL}	VCC=3.3~5.5V	-	-	0.3*VCC	V

Input Low level						
INT0/1/2 Input High level	V_{INTH}	VCC=3.3~5.5V	0.7*VCC	-	-	V
I/O Output Low level	V_{OL}	I_{OL}=4mA@VCC=3.3V, I_{OL}=10mA@VCC=5V	-	-	0.1*VCC	V
I/O Output High level	V_{OH}	I_{OH}=4mA@VCC=3.3V, I_{OH}=10mA@VCC=5V	0.9VCC	-	-	V
IO Sink current	IOL	V_{OL} =0.1VCC, @VCC=5V	-	60	-	mA
IO Source current	IOH	V_{OH} =0.9VCC, @VCC=5V	-	17	-	mA
PB0~PB7 large Sink current	I_{com}	V_{OL} =0.1VCC, @VCC=5V	-	120	-	mA
Input leakage current	I_{IH}&&I_{IL}	VCC=5V	-	1	5	μA
IO Pull_up	Pull_up Res	VCC=5V	-	4.7	-	K

DC characteristics parameters table

2.3. ADC Characteristics

Unless otherwise stated, typical values are test values at 27℃

Parameter	Symbol	Min	Typical	Max	Units
Operating voltage	VDD	2.5	-	5.5	V
ADC input voltage range	VADCIN	0	-	VDD	V
resolution	ADCRESO	12			Bit
ADC sampling period	TAD	0.5	-	-	us
Input channel	-	-	-	26	Channel
precision	-	-	9	10	Bit
No missing code	-	-	9	10	Bit
ADC conversion time	TCON	2.625	-	-	us
Integral nonlinearity error	EINL	-	±2	±3	LSB
Differential nonlinearity error	EDNL	-	±1	±2	LSB

ADC characteristics parameters table

2.4. Limit Parameters

Parameter	Symbol	Min	Typ	Max	Units
Supply voltage when working	VCC	VSS+2.5	-	VSS+5.5	V
Non-working storage temperature	Tstg	-40	-	125	℃
Operating temperature	Totg	-40	-	105	℃
I/O input voltage	Vin	VSS-0.5	-	VCC+0.5	V
IOL total current	IOLA	130			mA
IOH total current	IOHA	-130			mA
Port electrostatic discharge voltage	ESD(HBM)	-8	-	8	kV

Limit parameters characteristics parameters table

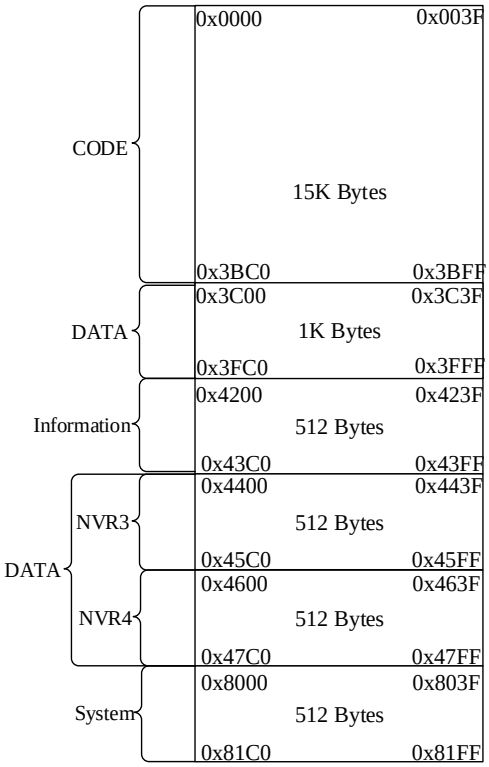
Notes: Exceed the limit parameters may cause damage to the chip, unable to expect the chip work outside the above indicated range. If you work under conditions outside the marked range for a long time, it may affect the reliability of the chip.

3. Memory and SFR

3.1. Flash

FLASH features are as follows:

- CODE area: ICP programming supports block erase, page erase, byte write
- DATA area: page erase, byte write
- Program/erase times: CODE area: at least 20000 times @27℃
  DATA area: at least 20000 times @27℃
- Data storage period: 100 Years@27℃
  20 Years@85℃



Flash Storage Architecture

Module	Address range	Space size (Byte)	Page
CODE	0x0000~0x3BFF	15K	30
DATA	0x3C00~0x3FFF	1K	1
	NVR3：0x4400~0x45FF	512	1
	NVR4：0x4600~0x47FF	512	1
Information	0x4200~0x43FF	512	1
System	0x8000~0x81FF	512	1

Steps to read the unique identification code (UID) of the chip:

1. Turn off the interrupt;
2. The read CODE absolute address 0x43A8~0x43AF corresponds to product ID1~ID8.
3. Restore interrupt settings.

Note:

1. **It is recommended that BOR must be turned on at the first initialization of the program to reduce the risk of errors.**
2. **It is not recommended to store the DATA area (0x3C00~0x3FFF) as the user CODE.**

3.2. RAM

There are 256 Bytes internal, the address is 00H~FFH, including working registers group, bit addressing areas, buffers and SFR, the buffer contain the stack area.

Internal low 128 Bytes, 00H~7FH has 128 Bytes. Read and write data by immediate addressing or indirect addressing.

Internal high 128 Bytes, 80H~FFH has 128 Bytes. Read and write data only by immediate addressing or indirect addressing.

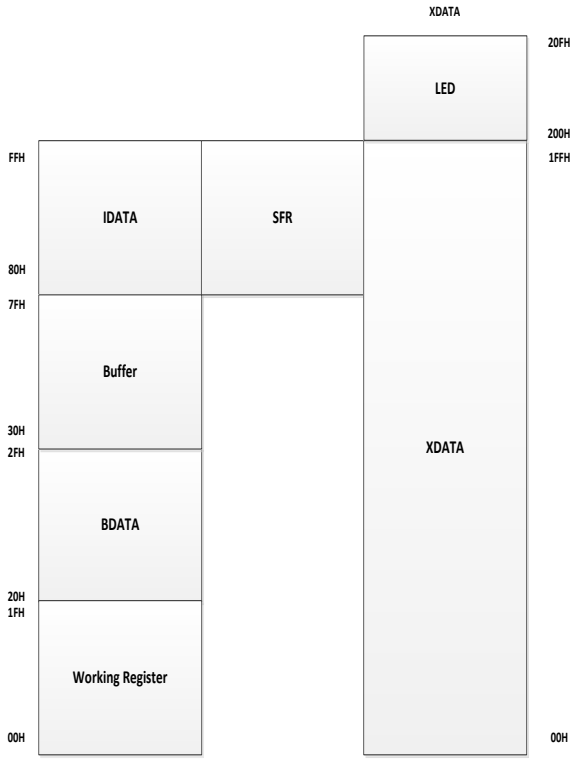
Special function register SFR: The address is 80H~FFH, Read and write data only by direct addressing.

Xdata have 512 Bytes, the address is 0000H~01FFH, users can use this area completely. To read and write data through the data pointer or working registers group addressing mode.

LED storage RAM occupies XRAM, address 200~20FH. This area is the LED display buffer, and the display content is modified by changing the area data

Note reserved stack space when writing a program, in order to avoid stack overflow and program goes wrong. Stack first address automatically assigned by program, when programming with C language, but it must be stored in data or idata. KEIL stack can be set in the first address in STARTUP.A51.

RAM address space allocation map:



RAM	(MSB)							(LSB)	
7FH									Buffer
2FH	7F	7E	7D	7C	7B	7A	79	78	Bit addressing area
2EH	77	76	75	74	73	72	71	70	
2DH	6F	6E	6D	6C	6B	6A	69	68	
2CH	67	66	65	64	63	62	61	60	
2BH	5F	5E	5D	5C	5B	5A	59	58	
2AH	57	56	55	54	53	52	51	50	
29H	4F	4E	4D	4C	4B	4A	49	48	
28H	47	46	45	44	43	42	41	40	
27H	3F	3E	3D	3C	3B	3A	39	38	
26H	37	36	35	34	33	32	31	30	
25H	2F	2E	2D	2C	2B	2A	29	28	
24H	27	26	25	24	23	22	21	20	
23H	1F	1E	1D	1C	1B	1A	19	18	
22H	17	16	15	14	13	12	11	10	
21H	0F	0E	0D	0C	0B	0A	09	08	
20H	07	06	05	04	03	02	01	00	
1FH	3 groups								Working register area
18H									
17H	2 groups								
10H									
0FH	1 groups								
08H									
07H	0 groups								
00H									

address allocation map for each bit addressing area

The following table lists the methods to get value in the three parts of RAM:

DATA	MOV A, direct
	MOV direct, A
	MOV direct, #data
	MOV direct1, direct2
	MOV Rn, direct
	MOV direct, Rn

IDATA	MOV A, @Ri
	MOV @Ri, A
	MOV direct, @Ri
	MOV @Ri, direct
	MOV @Ri, #data

XDATA	MOVX @DPTR, A MOVX A, @DPTR

RAM value instruction set

Notes: n: 0~7, i: 0~1.

3.3. SFR Table

Address	Name	W/R	Default/Reset	Function description
0x80	DATAB	RW	0xFF	PB data register
0x81	SP	RW	0x07	Stack pointer register
0x82	DPL	RW	0x00	Data pointer register0 low 8-bit
0x83	DPH	RW	0x00	Data pointer register0 high 8-bit
0x84	SYS_CLK_CFG	RW	0x01	Clock control register
0x85	INT_PE_STAT	RW	0x00	WDT/Timer2 interrupt status register
0x86	INT_POBO_STAT	RW	0x00	LVDT boost/LVDT buck interrupt status register
0x87	PCON	RW	0x00	Low-power mode select register
0x88	TCON	RW	0x05	Timer control register
0x89	TMOD	RW	0x00	Timer mode register
0x8A	TL0	RW	0x00	Timer 0 counter low 8-bit
0x8B	TL1	RW	0x00	Timer 1 counter low 8-bit
0x8C	TH0	RW	0x00	Timer 0 counter high 8-bit
0x8D	TH1	RW	0x00	Timer 1 counter high 8-bit
0x8E	SOFT_RST	RW	0x00	Soft reset register
0x90	DATAC	RW	0xFF	PC port data register
0x91	WDT_CTRL	RW	0x00	WDT timing overflow control register
0x92	WDT_EN	RW	0x00	WDT timing enable register
0x93	TIMER2_CFG	RW	0x00	TIMER2 CFG register
0x94	TIMER2_SET_H	RW	0x00	TIMER2 count value configuration register, high 8 bits
0x95	TIMER2_SET_L	RW	0x00	TIMER2 count value configuration register, low 8 bits
0x96	REG_ADDR	RW	0x00	Second address bus register
0x97	REG_DATA	RW	0x00	Second data read and write bus register
0x98	DATAD	RW	0xFF	PD port data register
0x99	PWM1_L_L	RW	0x00	PWM1 low level control register(low 8-bit)
0x9A	PWM1_L_H	RW	0x00	PWM1 low level control register(high 8-bit)
0x9B	PWM1_H_L	RW	0x00	PWM1 high level control register(low 8-bit)
0x9C	PWM1_H_H	RW	0x00	PWM1 high level control register(high 8-bit)
0x9D	PWM2_L_L	RW	0x00	PWM2 low level control register(low

				8-bit)
0x9E	PWM2_L_H	RW	0x00	PWM2 low level control register(high 8-bit)
0x9F	PWM2_H_L	RW	0x00	PWM2 high level control register(low 8-bit)
0xA0	P2_XH	RW	0xFF	MOVX @Ri,A operation xdata address high 8 bits
0xA1	PWM2_H_H	RW	0x00	PWM2 high level control register(high 8-bit)
0xA2	PWM_EN	RW	0x00	PWM control register
0xA3	PWM0_CH_CTRL	RW	0x00	PWM0 control register
0xA4	PWM0_CH0_CNT_L	RW	0x00	PWM0 channel 0 count value configuration register low 8 bits
0xA5	PWM0_CH0_CNT_H	RW	0x00	PWM0 channel 0 count value configuration register high 8 bits
0xA6	PWM0_CH1_CNT_L	RW	0x00	PWM0 channel 1 count value configuration register low 8 bits
0xA7	PWM0_CH1_CNT_H	RW	0x00	PWM0 channel 1 count value configuration register high 8 bits
0xA8	IEN0	RW	0x00	Interrupt enable register
0xA9	PWM0_CH2_CNT_L	RW	0x00	PWM0 channel 2 count value configuration register low 8 bits
0xAA	PWM0_CH2_CNT_H	RW	0x00	PWM0 channel 2 count value configuration register high 8 bits
0xAB	PWM0_CH3_CNT_L	RW	0x00	PWM0 channel 3 count value configuration register low 8 bits
0xAC	PWM0_CH3_CNT_H	RW	0x00	PWM0 channel 3 count value configuration register high 8 bits
0xAD	PWM0_MOD_L	RW	0x00	PWM0 cycle configuration register low 8 bits
0xAE	PWM0_MOD_H	RW	0x00	PWM0 cycle configuration register high 8 bits
0xAF	SCAN_START	RW	0x00	LED scan open register
0xB0	DP_CON	RW	0x00	LED scan control register
0xB1	SCAN_WIDTH	RW	0x00	LED scan on time 1 control register
0xB2	LED2_WIDTH	RW	0x00	LED scan on time 2 control register
0xB3	LED_DRIVE	RW	0x00	LED drive capability configuration register
0xB4	ADC_SPT	RW	0x00	ADC sample time configure register
0xB5	ADC_SCAN_CFG	RW	0x00	ADC scan control register

0xB6	ADCCKC	RW	0x00	ADC clock control register
0xB8	IPL0	RW	0x00	Interrupt priority register 0
0xB9	ADC_RDATAH	R	0x00	ADC scan result register high 4 bits
0xBA	ADC_RDATAL	R	0x00	ADC scan result register low 8 bits
0xBB	ADC_CFG1	RW	0x00	ADC sampling timing control register 1
0xBC	ADC_CFG2	RW	0x02	ADC sampling timing control register 2
0xBD	UART0_BDL	RW	0x00	UART0 Baudrate control registe
0xBE	UART0_CON1	RW	0x00	UART0 control register 1
0xBF	UART0_CON2	RW	0x0C	UART0 control register 2
0xC0	UART0_STATE	RW	0x00	UART0 status flag register
0xC1	UART0_BUF	RW	0xFF	UART0 data register
0xC2	SCI_BDH	RW	0x00	UART1 baudrate control register
0xC3	SCI_BDL	RW	0x00	UART1 baudrate control register
0xC4	SCI_C1	RW	0x00	UART1 control register 1
0xC5	SCI_C2	RW	0x00	UART1 control register 2
0xC6	SCI_C3	RO/RW	0x00	UART1 control register 3
0xC7	SCI_S2	RW	0x00	UART1 sync segment control register
0xC8	SCI_S1	RO	0x00	UART1 interrupt status flag register
0xC9	SCI_D	RW	0xFF	UART1 data register
0xD0	PSW	R/RW	0x00	Program status register
0xD7	RST_STAT	RW	rst_state	Reset flag register
0xD8	SCI_INT_CLR	RW	0x00	UART1 interrupt flag clear register
0xD9	ADC_IO_SEL1	RW	0x00	ADC function selection register 1
0xDA	ADC_IO_SEL2	RW	0x00	ADC function selection register 2
0xDB	ADC_IO_SEL3	RW	0x00	ADC function selection register 3
0xDC	ADC_IO_SEL4	RW	0x00	ADC function selection register 4
0xDD	PU_PA	RW	0x00	PA port pull-up resistor selection register
0xDE	PU_PB	RW	0x00	PB port pull-up resistor selection register
0xDF	PU_PC	RW	0x00	PC port pull-up resistor selection register
0xE0	ACC	RW	0x00	Accumulator
0xE1	IRCON2	RW	0x00	Interrupt flag register 2
0xE2	PU_PD	RW	0x00	PD port pull-up resistor selection register
0xE3	IICADD	RW	0x00	IIC address register
0xE4	IICBUF	RW	0x00	IIC transmit and receive data register

0xE5	IICCON	RW	0x10	IIC configuration register
0xE6	IEN1	RW	0x00	Interrupt enable register 1
0xE7	IEN2	RW	0x00	Interrupt enable register 2
0xE8	IICSTAT	RO/RW	0x44	IIC status register
0xE9	IICBUFFER	RW	0x00	IIC transmit and receive data buffer register
0xEA	TRISA	RW	0x03	PA port direction register
0xEB	TRISB	RW	0xFF	PB port direction register
0xEC	TRISC	RW	0xFF	PC port direction register
0xED	TRISD	RW	0xFF	PD port direction register
0xEE	COM_IO_SEL	RW	0x00	COM large sink current selection register
0xEF	ODRAIN_EN	RW	20x00	PA open drain enable register
0xF0	B	RW	0x00	B register
0xF1	IRCON1	RW	0x00	Interrupt flag register 1
0xF2	PERIPH_IO_SEL	RW	0x40	IIC/UART0/INT function control register
0xF4	IPL2	RW	0x00	Interrupt priority register 2
0xF6	IPL1	RW	0x00	Interrupt priority register 1
0xF7	EXT_INT_CON	RW	0x15	External interrupt polarity control register
0xF8	DATAA	RW	0x03	PA data register
0xF9	SPROG_ADDR_H	RW	0x00	Address control register
0xFA	SPROG_ADDR_L	RW	0x00	Address register,lower 8 bits
0xFB	SPROG_DATA	RW	0x00	Data register
0xFC	SPROG_CMD	RW	0x00	Command register
0xFD	SPROG_TIM	RW	0x5A	Erase Time Control Register
0xFE	PD_ANA	RW	0x1F	Module switch control register
0xFF	SEL_LVDT_VTH	RW	0x00	LVDT threshold select register

SFR register summary

Note: 1. Registers whose addresses end with 8 or 0 can be bit-operated, such as register addresses 0x80, 0x88.

2. Reset value: reset value in different modes (WDT overflow reset, power on reset, power down reset, program reset, debug reset, PC pointer overflow reset, software reset);

Power-on reset: rst_state is 0x02;

Reset in other modes: The reset flag bit corresponding to rst_state is 1, and other reset flags remain in their original state.

3. R: only read. RW: read and write.

3.4. Secondary Bus Register Table

The BF7512CMCMXX series supports expanded secondary bus registers for expanding more register functions. Just write the address of the secondary bus register to be accessed into REG_ADDR, and then access the corresponding secondary bus register through the REG_DATA register. It is recommended that when reading and writing secondary bus registers, first EA = 0, and then EA = 1 after the operation is completed. It is to prevent other interrupts or operations from modifying the address or data of the secondary bus register.

Addr	Name	Bit	R/W	Por	Function description
0x96	REG_ADDR	<5:0>	RW	0x00	Secondary bus address configuration register
0x97	REG_DATA	<7:0>	RW	0x00	Secondary bus data read and write register

Addr	Name	R/W	Por	Function description
0x00	CFG0_REG	R	0xFF①	Configuration word register0
0x01	CFG1_REG	R	0xFF①	Configuration word register1
0x02	CFG2_REG	R	0xFF①	Configuration word register2
0x03	CFG3_REG	R	0xFF①	Configuration word register3
0x04	CFG4_REG	R	0xFF①	Configuration word register4
0x05	CFG5_REG	R	0xFF①	Configuration word register5
0x06	CFG6_REG	R	0xFF①	Configuration word register6
0x07	CFG7_REG	R	0xFF①	Configuration word register7
0x08	CFG8_REG	R	0xFF①	Configuration word register8
0x09	CFG9_REG	R	0xFF①	Configuration word register9
0x0A	CFG10_REG	R	0xFF①	Configuration word register10
0x0B	CFG11_REG	R	0xFF①	Configuration word register11
0x0C	CFG12_REG	R	0xFF①	Configuration word register12
0x0D	CFG13_REG	R	0xFF①	Configuration word register13
0x0E	CFG14_REG	R	0xFF①	Configuration word register14
0x0F	CFG15_REG	R	0xFF①	Configuration word register15
0x10	CFG16_REG	R	0xFF①	Configuration word register16
0x11	CFG17_REG	R	0xFF①	Configuration word register17
0x12	CFG18_REG	R	0xFF①	Configuration word register18
0x13	CFG19_REG	R	0xFF①	Configuration word register19
0x14	CFG20_REG	R	0xFF①	Configuration word register20
0x15	CFG21_REG	R	0xFF①	Configuration word register21
0x16	CFG22_REG	R	0xFF①	Configuration word register22
0x17	CFG23_REG	R	0xFF①	Configuration word register23
0x18	CFG24_REG	R	0xFF①	Configuration word register24
0x19	CFG25_REG	R	0xFF①	Configuration word register25
0x1A	CFG30_REG	R	0xFF①	Configuration word register30

0x1F	DUMMY_REG	RW	0x00	RTC crystal oscillator circuit selection register
0x20	EEP_SELECT	RW	0x00	EEP NVR/main block selection register

Note:

1. '①': The reset value is the default value after power-on reset, and the value after the global reset is completed is the factory calibration value;
2. 'x': indeterminate state;
3. 'R': Read only;
4. 'RW': Read and write.

4. Register Summary

4.1. SFR Register details

DATAB(80H)PB port data register

Bit number	7	6	5	4	3	2	1	0
Symbol	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
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 number	Bit symbol	Description
7~0	--	The output level of the PB group can be configured as the GPIO port. The read value is the level state of the current IO port or the configured output value.

SP(81H) Stack pointer register

Bit number	7	6	5	4	3	2	1	0
Symbol	SP[7:0]							
R/W	R/W							
Reset value	7							

DPL(82H) Data pointer register0 low 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	DPL[7:0]							
R/W	R/W							
Reset value	0							

DPH(83H) Data pointer register0 high 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	DPH[7:0]							
R/W	R/W							
Reset value	0							

SYS_CLK_CFG(84H) Clock control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	PLL_CLK_SEL[1:0]	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	1

Bit number	Bit symbol	Description
7~2	--	Reserved
1~0	PLL_CLK_SEL	PLL clock divided selection register 00: 12Mhz; 01: 6Mhz; 10: 4Mhz; 11: Reserved

INT_PE_STAT(85H)WDT/Timer2 interrupt status register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	INT_WDT_STAT	INT_TIMER2_STAT
R/W	-	-	-	-	-	-	R/W	R/W
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1	INT_WDT_STAT	WDT interrupt status, set 0, write WDT_CTRL can set 0. 1: interrupt effective 0: invalid interrupt
0	INT_TIMER2_STAT	TIMER2 interrupt status, set 0, write TIMER2_CFG can set 0. 1: interrupt effective 0: invalid interrupt

INT_POBO_STAT (86H) LVDT boost/LVDT buck interrupt status register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	INT_PO_STAT	INT_BO_STAT
R/W	-	-	-	-	-	-	R/W	R/W
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1	INT_PO_STAT	Lvdt boost interrupt status 1: boost interrupt is valid 0: boost interrupt is invaild
0	INT_BO_STAT	Lvdt buck interrupt state 1: buck interrupt is valid 0: buck interrupt is invalid

PCON(87H) Low-power mode select register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-		LPM
R/W	-	-	-	-	-	-	-	R/W
Reset value	-	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
0	LPM	Low-power mode control 1: Low-power mode; 0: Normal mode, automatically cleared after wake-up

TCON(88H) Timer control register

Bit number	7	6	5	4	3	2	1	0
Symbol	TF1	TR1	TF0	TR0	IE1	-	IE0	-

R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	-
Reset value	0	0	0	0	0	-	0	-

Bit number	Bit symbol	Description
7	TF1	Timer1 overflow flag. Set to 1 when Timer1 overflows, or Timer0's TH0 overflows in mode three.
6	TR1	Timer1 start enable, When set to 1, start Timer1, or start Timer0 mode three TH0 counte.
5	TF0	Timer0 overflow flag, the hardware set 1 when Timer0 overflows.
4	TR0	Timer0 start enable, when set to 1, start Timer0 count.
3	IE1	External interrupt 1. The hardware set 1, the software is cleared.
2	--	Reserved
1	IE0	External interrupt 0. The hardware set 1, the software is cleared
0	--	Reserved

TMOD(89H) Timer mode register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	M1[1:0]		-	-	M0[1:0]	
R/W	-	-	R/W		-	-	R/W	
Reset value	-	-	0	0	-	-	0	0

Bit number	Bit symbol	Description
7~6	--	Reserved
5~4	M1[1:0]	M1-Timer1: Bit 1, M1-Timer1: Bit 0. 00=mode0 – 13bit Timer 01=mode1 – 16bit Timer 10=mode2 – manual reload mode 8bit Timer 11=mode3 – 2*8bit Timer
3~2	--	Reserved
1~0	M0[1:0]	M0-Timer0: Bit 1, M0-Timer0: Bit 0. 00=mode0 – 8bit Timer 01=mode1 – 16 bit Timer 10=mode2 – automatic reload mode 8bit Timer 11=mode3 – 2*8bit Timer

TL0(8AH) Timer 0 counter low 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TL0[7:0]							

R/W	R/W
Reset value	0

TL1(8BH) Timer 1 counter low 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TL1[7:0]							
R/W	R/W							
Reset value	0							

TH0(8CH) Timer 0 counter high 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TH0[7:0]							
R/W	R/W							
Reset value	0							

TH1(8DH) Timer 1 counter high 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TH1[7:0]							
R/W	R/W							
Reset value	0							

SOFT_RST(8EH) Soft reset register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Software reset register. Software reset is only generated when the register value is 0x55.

DATAC(90H) PC port data register

Bit number	7	6	5	4	3	2	1	0
Symbol	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PC data register. The output level of the PC group can be configured as the GPIO port. The read value is the level state of the current IO port or the configured output value.

WDT_CTRL(91H) WDT timing overflow control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	WDT_TIME_SEL		

R/W	-	-	-	-	-	R/W		
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
2~0	WDT_TIME_SEL	WDT overflow timer register. Timing length is as follows: 0x00: 18ms; 0x01: 36ms; 0x02: 72ms; 0x03: 144ms; 0x04: 288ms; 0x05: 576ms; 0x06: 1152ms; 0x07: 2304ms

WDT_EN(92H) WDT timing enable register

Bit number	7	6	5	4	3	2	1	0
Symbol	WDT_EN							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	WDT_EN	WDT timing enable configuration register. WDT is turned off when the configuration value is 0x55.

TIMER2_CFG (93H) TIMER2 CFG register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	TIMER2_CNT_MOD	TIMER2_CLK_SEL	TIMER2_RLD	TIMER2_EN
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
3	TIMER2_CNT_MOD	TIMER2 count step mode select register 1: count step is 65536 clock. 0: count step is 1 clock.
2	TIMER2_CLK_SEL	TIMER2 clock select register 1: select clk_xtal 0: select clk_rc
1	TIMER2_RLD	TIMER2 reload enable control register 1: automatic reload mode 0: manual reload mode
0	TIMER2_EN	TIMER2 count enable register 1: turn on timing; 0: stop timing; In manual reload mode, the hardware automatically clears this register after timing is completed, stop count. In manual reload mode, will maintain the enable register after the count is completed. Automatically re-counting from 0, no matter which mode, configuring this register

		to 1 during counting will start counting from 0.

TIMER2_SET_H(94H) TIMER2 count value configuration register, high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	TIMER2 count configuration register, high 8 bit, Configuring this register during the scan will recount.

TIMER2_SET_L(95H) TIMER2 count value configuration register, low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	TIMER2 count configuration register, low 8 bit, Configuring this register during the scan will recount.

REG_ADDR (96H) Second address bus register

Bit number	7	6	5	4	3	2	1	0
Symbol	-		REG_ADDR					
R/W	-		R/W					
Reset value	-		0	0	0	0	0	0

Bit number	Bit symbol	Description
5~0	REG_ADDR	Secondary bus address configuration register. When operating the secondary bus, it is recommended to read and write the secondary bus register, EA = 0 first, EA = 1 after the operation is completed, to prevent other interruptions or operations from modifying the secondary bus register address or data.

REG_DATA (97H) Second bus data read and write register

Bit number	7	6	5	4	3	2	1	0
Symbol	REG_DATA							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	REG_DATA	Secondary bus read and write registers. It is recommended to

		read and write the secondary bus register, EA = 0 first, EA = 1 after the operation is completed, to prevent other interruptions or operations from modifying the secondary bus register address or data.

DATAD(98H) PD port data register

Bit number	7	6	5	4	3	2	1	0
Symbol	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PD data register. The output level of the PD group can be configured as the GPIO port. The read value is the level state of the current IO port or the configured output value.

PWM1_L_L (99H) PWM1 low level control register(low 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM1_L_H (9AH) PWM1 PWM1 low level control register(high 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM1_H_L (9BH) PWM1 high level control register(low 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM1_H_H (9CH) PWM1 high level control register(high 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM2_L_L (9DH) PWM2 low level control register(low 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM2_L_H (9EH) PWM2 low level control register(high 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM2_H_L (9FH) PWM2 high level control register(low 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

P2_XH (A0H) MOVX @Ri,A operation xdata address high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	P2_XH	When using the MOVX @Ri, A instruction, when operating the pdata area, P2_XH need to be clear to 0.

PWM2_H_H (A1H) PWM2 high level control register(high 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM_EN (A2H) PWM control register

Bit number	7	6	5	4
Symbol	-	-	PWM0_CH3_CMOD	PWM0_CH2_CMOD
R/W	-	-	R/W	R/W
Reset value	-	-	0	0
Bit number	3	2	1	0
Symbol	PWM0_CH1_CMOD	PWM2_EN	PWM1_EN	PWM0_EN
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
5	PWM0_CH3_CMOD	PWM0 channel 3 duty cycle mode select register 1: select channel 0 duty cycle 0: select its own channel duty cycle
4	PWM0_CH2_CMOD	PWM0 channel 2 duty cycle mode select register

		1: select channel 0 duty cycle 0: select its own channel duty cycle
3	PWM0_CH1_CMOD	PWM0 channel 1 duty cycle mode select registe 1: select channel 0 duty cycle 0: select its own channel duty cycle
2	PWM2_EN	PWM2 module enable register 1: enable 0: not enable
1	PWM1_EN	PWM1 module enable register 1: enable 0: not enable
0	PWM0_EN	PWM0 module enable register 1: enable 0: not enable

PWM0_CH_CTRL (A3H) PWM0 control register

Bit number	7	6	5	4
Symbol	PWM0_CH3_POLA_SEL	PWM0_CH2_POLA_SEL	PWM0_CH1_POLA_SEL	PWM0_CH0_POLA_SEL
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	PWM0_CH3_EN	PWM0_CH2_EN	PWM0_CH1_EN	PWM0_CH0_EN
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	PWM0_CH3_POLA_SEL	Channel 3 polarity selection ch3_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
6	PWM0_CH2_POLA_SEL	Channel 2 polarity selection ch2_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
5	PWM0_CH1_POLA_SEL	Channel 1 polarity selection ch1_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
4	PWM0_CH0_POLA_SEL	Channel 0 polarity selection ch0_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
3	PWM0_CH3_EN	Channel 3 enable ch3_en 1: enable

		0: not enable
2	PWM0_CH2_EN	Channel 2 enable ch2_en 1: enable 0: not enable
1	PWM0_CH1_EN	Channel 1 enable ch1_en 1: enable 0: not enable
0	PWM0_CH0_EN	Channel 0 enable ch0_en 1: enable 0: not enable

PWM0_CH0_CNT_L (A4H) PWM0 channel 0 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH0_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH0_CNT_L	Channel 0 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH0_CNT_H (A5H) PWM0 channel 0 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH0_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH0_CNT_H	Channel 0 count configuration register high 8 bits. Configure PWM output duty cycle.

PWM0_CH1_CNT_L (A6H) PWM0 channel 1 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH1_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH1_CNT_L	Channel 1 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH1_CNT_H (A7H) PWM0 channel 1 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH1_CNT_H							

R/W	R/W
Reset value	0

Bit number	Bit symbol	Description
7~0	PWM0_CH1_CNT_H	Channel 1 count configuration register high 8 bits. Configure PWM output duty cycle.

IEN0(A8H) Interrupt enable register

Bit number	7	6	5	4	3	2	1	0
Symbol	EA	-			ET1	EX1	ET0	EX0
R/W	R/W	-			R/W	R/W	R/W	R/W
Reset value	0	-			0	0	0	0

Bit number	Bit symbol	Description
7	EA	EA- Interrupt enable bit. EA=0 block all interrupts (EA takes precedence over the interrupt enable bits of the interrupt source). EA=1, open interrupts. Whether the interrupt request of each interrupt source is allowed or disable, and also needs to be determined by the respective enable bits.
6~4	--	Reserved
3	ET1	ET1-Timer1 overflow interrupt allow bit. ET1=0, disable Timer1 (TF1) to apply for interrupt. ET1=1, allow TF1 to apply for interrupt.
2	EX1	EX1-INT_EXT1 allow bit. EX1=0, disable INT_EXT1 to apply for interrupt. Allow INT_EXT1 to apply for interrupt.
1	ET0	ET0- Timer0 overflow interrupt allow bit. ET0=0, disable Timer1 (TF0) to apply for interrupt. ET0=1, allow Timer1 (TF0) to apply for interrupt.
0	EX0	EX0-INT_EXT0 allow bit. EX0=0, disable INT_EXT0 to apply for interrupt. EX0=1, allow INT_EXT0 to apply for interrupt.

PWM0_CH2_CNT_L (A9H) PWM0 channel 2 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH2_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH2_CNT_L	Channel 2 count configuration register low 8 bits.

		Configure PWM output duty cycle.

PWM0_CH2_CNT_H (AAH) PWM0 channel 2 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH2_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH2_CNT_H	Channel 2 count configuration register high 8 bits. Configure PWM output duty cycle.

PWM0_CH3_CNT_L (ABH) PWM0 channel 3 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH3_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH3_CNT_L	Channel 3 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH3_CNT_H (ACH) PWM0 channel 3 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH3_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH3_CNT_H	Channel 3 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_MOD_L (ADH) PWM0 cycle configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_MOD_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_MOD_L	PWM0 count cycle configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_MOD_H (AEH) PWM0 cycle configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0

Symbol	PWM0_MOD_H
R/W	R/W
Reset value	0

Bit number	Bit symbol	Description
7~0	PWM0_MOD_H	PWM0 count cycle configuration register high 8 bits. Configure PWM output duty cycle.

SCAN_START(AFH) LED scan open register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	-
R/W	-	-	-	-	-	-	-	R/W
Reset value	-	-	-	-	-	-	-	0

DP_CON (B0H) LED scan control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	DUTY_SEL			SCAN_MODE	COM_MOD
R/W	-	-	-	R/W			R/W	R/W
Reset value	-	-	-	0	0	0	O	0

Bit number	Bit symbol	Description
4~2	DUTY_SEL	LED port drive mode matrix selection configuration register. 0: no matrix 1: 4x4 matrix (LED0~LED4) 2: 5x5 matrix (LED0~LED5) 3: 6x6 matrix (LED0~LED6) 4: 6x7 matrix (LED0~LED7) 5: 7x7 matrix (LED0~LED7) 6: 7x8 matrix (LED0~LED7) 7: 8x8 matrix (LED0~LED8)
1	SCAN_MODE	LED scan mode. 1: cycle scan mode 0: interrupt scan mode
0	COM_MOD	Large sink current ports drive enable. 1: COM port function lock, work as a large current IO port. 0: COM port function is not locked and can be configured as other functions. When the COM port locks the large sink current IO port, by configuring GPIO registers output drive timing, it is vaild when all of the following LED scan configurations are invalid.

SCAN_WIDTH (B1H) LED scan on time 1 control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	LED dot matrix drive mode, corresponding to a signal lamp lighting time configuration register——on time 1 configuration. period=(scan_width+1)*16us, support configuration range 0.016~4.096ms.

LED2_WIDTH (B2H) LED scan on time 2 control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	LED dot matrix drive mode, corresponding to a signal lamp lighting time configuration register——on time 2 configuration period=(led2_width+1)*16us, support configuration range 0.016~4.096ms.

LED2_DRIVE (B3H) LED drive capability configuration register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-			
R/W	-	-	-	-	R/W			
Reset value	-	-	-	-	0			

ADC_SPT (B4H) ADC sample time configure register

Bit number	7	6	5	4	3	2	1	0
Symbol	ADC_SPT							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	ADC_SPT	ADC sample time configure register sample time: sample_Timer = (ADC_SPT+1)*4Tadc_clk

ADC_SCAN_CFG (B5H) ADC scan control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	ADC_ADDR					ADC_START
R/W	-	-	R/W					R/W
Reset value	-	-	0					0

Bit number	Bit symbol	Description
5~1	ADC_ADDR	ADC channel address select register. 0~25: corresponding to ADC0~ADC25; 26: internal input channel of the chip
0	ADC_START	ADC scan open register ADC_START=0→1( ) turn to conversion, ADC_START configuration is not allowed during scanning. ADC_START is set from 0 to 1, ADC start to scan, after scanning once, ADC_START hardware is automatically set to 0, corresponding to the interrupt flag set to 1, ADC interrupt flag bit needs to be cleared by software.

ADCCKC (B6H) ADC clock control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	ADCCKV		ADCK	
R/W	-	-	-	-	R/W		R/W	
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
3~2	ADCCKV	ADC comparator offset cancellation analog input clock. 0: 12MHz 1: 8MHz 2: 4MHz 3: 2MHz
1~0	ADCK	ADC_CLK frequency division selection. 0: 8MHz 1: 6MHz 2: 4MHz 3: 3MHz

IPL0 (B8H) Interrupt priority register 0

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	PT1	PX2	PT0	PX0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
7~4	–	Reserved
3	PT1	PT1-TF1(Timer1 interrupt) priority selection bit. PT1=0: TF1(Timer1 interrupt) is low priority. PT1=1: TF1(Timer1 interrupt) is high priority.
2	PX2	PX2- INT_EXT1 interrupt priority selection bit.

		PX2=0: INT_EXT1 is low priority. PX2=1: INT_EXT1 is high priority.
1	PT0	PT0-TF0(Timer0 interrupt) priority selection bit. PT0=0: TF0(Timer0 interrupt) is low priority. PT0=1: TF0(Timer0 interrupt) is high priority.
0	PX0	PX0- INT_EXT0 interrupt priority selection bit. PX0=0: INT_EXT0 is low priority. PX0=1: INT_EXT0 is high priority.

ADC_RDATAH (B9H) ADC scan result register high 4 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	ADC_RAWDATA<11:8>			
R/W	-	-	-	-	R			
Reset value	-	-	-	-	0			

Bit number	Bit symbol	Description
3~0	ADC_RAWDATA<11:8>	ADC scan result register

ADC_RDATAL(BAH) ADC scan result register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	ADC_RAWDATA<7:0>							
R/W	R							
Reset value	0							

Bit number	Bit symbol	Description
7~0	ADC_RAWDATA<7:0>	ADC scan result register

ADC_CFG1(BBH) ADC sampling timing control register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	ADCWNUM					SAMBG	SAMDEL	
R/W	R/W					R/W	R/W	
Reset value	0					0	0	

Bit number	Bit symbol	Description
7~3	ADCWNUM	Distance conversion interval selection after sampling. (3+ADCWNUM)*ADC_CLK
2	SAMBG	Sampling timing and comparison timing interval selection. 0: interval 0; 1: interval 1*ADC_CLK.
1~0	SAMDEL	Sampling delay time selection. 0: 0*ADC_CLK; 1: 2*ADC_CLK; 2: 4*ADC_CLK;

		3: 8*ADC_CLK.

ADC_CFG2 (BCH) ADC sampling timing control register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	FILTER_R_SEL	VREF_IN_ADC_SEL		ADC_I_SEL[1:0]		CTRL_SEL[1:0]	
R/W	-	R/W	R/W		R/W		R/W	
Reset value	-	0	0		0		10	

Bit number	Bit symbol	Description
6	FILTER_R_SEL	Input signal filtering selection, 0 means no RC filtering, 1 means RC filtering.
5~4	VREF_IN_ADC_SEL	Input to ADC26 reference voltage selection 01: 2.253V; other: reserved; Need to read the calibration voltagevalue from the chip flash when using. VREF_IN_ADC_SEL voltage = { CBYTE[0x43C6], CBYTE[0x43C7]}mV.
3~2	ADC_I_SEL[1:0]	ADC bias current size selection register. ADC_I_SEL[0]: 0 is the comparator bias current is 4uA; 1 is the comparator bias current is 5uA; ADC_I_SEL[1]: 0 is the op amp bias current is 4uA; 1 is the op amp bias current is 5uA;
1~0	CTRL_SEL[1:0]	ADC comparator offset cancellation selection signal, the default is 10. CTRL_SEL[1:0]: 00/01: sampling first in offset cancellation; 10: all switches are disconnected together; 11: the switch is disconnected in turn.

UART0_BDL (BDH) UART0 Baudrate control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description

7~0	--	Baud rate control register. Baud rate modules divisor register lower 8 bits, bandrate={UART0_BDH[1:0], UART0_BDL}, bandrate=0, does not generate baud rate clock. bandrate=1~1023, SCI bandrate = BUSCLK/(16xbandrate)

UART0_CON1 (BEH) UART0 control register 1

Bit number	7	6	5	4
Symbol	-	uart0_enable	receive_enable	multi_mode
R/W	-	R/W	R/W	R/W
Reset value	-	0	0	0
Bit number	3	2	1	0
Symbol	stop_mode	data_mode	parity_en	parity_sel
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
6	uart0_enable	Module enable. 1: module enable; 0: module off.
5	receive_enable	Receiver enable. 1: receiver open; 0: receiver off.
4	multi_mode	Multiprocessor communication mode. 1: mode enable; 0: mode disable.
3	stop_mode	stop bit width selection. 1: 2 bit; 0: 1 bit.
2	data_mode	Data mode select. 1: 9bit mode; 0: 8bit mode.
1	parity_en	Parity enable. 1: parity enable; 0: parity disable.
0	parity_sel	Parity select. 1: odd parity; 0: even parity.

UART0_CON2 (BFH) UART0 control register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	tx_empty_ie	rx_full_ie	UART0_BDH	
R/W	-	-	-	-	R/W		R/W	
Reset value	-	-	-	-	1	1	0	0

Bit number	Bit symbol	Description
3	tx_empty_ie	Send interrupt enable. 1: interrupt enable; 0: interrupt disable(used in polling mode)

2	rx_full_ie	Received interrupt enable 1: interrupt enable; 0: interrupt disable(used in polling mode)
1~0	UART0_BDH	Baud rate modulus divisor register high 2bit.

UART0_STATE (C0H) UART0 status flag register

Bit number	7	6	5	4
Symbol	-	r8	t8	tx_empty_if
R/W	-	R	R	R/W
Reset value	-	0	0	0
Bit number	3	2	1	0
Symbol	frx_full_i	rx_overflow_if	frame_err_if	parity_err_if
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
6	r8	Receiver’s ninth data, read only.
5	t8	Transmitter’s ninth data, read only when parity is enabled.
4	tx_empty_if	Send interrupt flag. 1: send buffer is empty; 0: send buffer is full, software write 0 clear 0, write 1 invalid.
3	frx_full_i	Receive interrupt flag,. 1: receive buffer is full; 0: receive buffer is empty, software write 0 clear 0, write 1 invalid.
2	rx_overflow_if	Receive overflow flag; 1: receive overflow(lost new data); 0: no overflow, software write 0 clear 0, write 1 invalid.
1	frame_err_if	Framing error flag. 1: framing error flag; 0: no framing error flag, software write 0 clear 0, write 1 invalid.
0	parity_err_if	Parity error flag. 1: receiver parity error; 0: parity is correct, software write 0 clear 0, write 1 invalid.

UART0_BUF (C1H) UART0 data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	Data register Read returns read-only receive data buffer contents, write into write-only send data buffer.

SCI_BDH (C2H) UART1 baudrate control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7	break_check_ie	Interval detection interrupt enable. 1: interrupt enable; 0: interrupt disable.
6	rx_edge_ie	RxD pin active edge interrupt enable. 1: interrupt enable; 0: interrupt disable.
5	--	Reserved
4~0	--	Baud rate modules divisor register high 5 bits.

SCI_BDL (C3H) UART1 baudrate control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Baud rate control register. Baud rate modules divisor register lower 8 bits, Baud_Mod ={UART0_BDH[1:0], UART0_BDL}, Baud_Mod =0, does not generate baud rate clock. Baud_Mod =1~1023, SCI bandrate = BUSCLK/(16x Baud_Mod)

SCI_C1 (C4H) UART1 control register 1

Bit number	7	6	5	4
Symbol	cycle_mode	stop_mode	single_txd	data_mode
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	parity_en	parity_sel	-	sci_enable
R/W	R/W	R/W	-	R/W
Reset value	0	0	-	0

Bit number	Bit symbol	Description
7	cycle_mode	Cycle mode enable. 1: cycle mode or signal mode, txd connection rxd; 0: normal two-wire mode.
6	stop_mode	stop bit selection. 1: 2bits; 0: 1bit.
5	single_txd	Signal line mode enable. 1: cycle_mode=1, select line mode, txd pin is valid; 0: internal cycle mode, txd pin is invalid.
4	data_mode	Transmission data mode selection. 1: 9 bit mode (the ninth bit is parity bit); 0: 8 bit mode.
3	parity_en	Parity enable. 1: parity enable; 0: parity disable.
2	parity_sel	Parity select. 1: odd parity; 0: even parity
1	--	Reserved
0	sci_enable	Clock gating enable when the module is working, and writing 1 indicates that the enable is valid. Open the module working clock, write 0 will close the module working clock, and reset the function module.

SCI_C2 (C5H) UART1 control register 2

Bit number	7	6	5	4
Symbol	tx_empty_ie	tx_finish_ie	rx_full_ie	idle_ie
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	trans_enable	receive_enable	rwu	break_trans_start
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	tx_empty_ie	Send buffer empty interrupt enable. 1: interrupt enable; 0: interrupt disable.
6	tx_finish_ie	Send complete interrupt enable. 1: interrupt enable; 0: interrupt disable.
5	rx_full_ie	Accept full interrupt enable. 1: interrupt enable;

		0: interrupt disable.
4	idle_ie	Idle line interrupt enable. 1: interrupt enable; 0: interrupt disable
3	trans_enable	Transmitter enable. 1: transmitter open,; 0: transmitter close
2	receive_enable	Receiver enable. 1: receiver open; 0: receiver close.
1	rwu	Receiver wake-up control. 1: receiver is in standby and waiting for the wake condition. 0: receiver is running normally.
0	break_trans_start	Send interval, write 1 and 0 to this bit, that is, a gap is placed in the data stream.

SCI_C2(C6H) UART1 control register 3

Bit number	7	6	5	4
Symbol	r8	t8	txd_direct	txd_inv
R/W	R	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	rxd_inv	rwu_idlesel	idle_sel	wake_sel
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	r8	Receiver’s ninth data, read only.
6	t8	Transmitter’s ninth data.
5	txd_direct	txd pin direction selection in signal line mode. 1: txd pin is the output in signal line mode; 0: txd pin is the input in signal line mode.
4	txd_inv	txd data inversion. 1: send data is reversed; 0: send data is not reserved.
3	rxd_inv	rxd data inversion . 1: receive data is reversed; 0: receive data is not reserved.
2	rwu_idlesel	Receive wake idle detection. 1: during the receive standby state (RWU=1), the idle bit is set when an IDLE character is detected; 0: during the receive standby state (RWU=1), the idle bit is

		not set when an IDLE character is detected.
1	idle_sel	Idle line type selection. 1: idle character bit count starts after stop bit; 0: idle character bit count starts after start bit, and the 10-bit time is counted (if data_mode=1 or stop_mode=1, then add one time separately).
0	wake_sel	Receiver wake-up mode selection. 1: address mark wake up; 0: idle line wake up.

SCI_S2(C7H) UART1 sync segment control register

Bit number	7	6	5	4
Symbol	break_check_if	rx_edge_if	rx_active_flag	-
R/W	R/W	R/W	R/W	-
Reset value	0	0	0	-
Bit number	3	2	1	0
Symbol	-	-	break_trans_size	break_check_en
R/W	-	-	R/W	R/W
Reset value	-	-	0	0

Bit number	Bit symbol	Description
7	break_check_if	Interval detection interrupt flag. 1: interval detected; 0: no interval detected, this bit writes 1 clear, write 0 is invalid.
6	rx_edge_if	RxD pin active edge interrupt flag. 1: active edge on the receive pin; 0: active edge does not appear on the receive pin; this bit writes 1 clear, write 0 is invalid.
5	rx_active_flag	Receiver activity tag, read only. 1: receiver activity; 0: receiver idle.
4~2	--	Reserved
1	break_trans_size	Interval generation bit length. 1: send by 13-bit time (if data_mode=1 or stop_mode=1, add 1 bit length respectively); 0: send by 10-bit time (if data_mode=1 or stop_mode=1, add 1 bit length respectively).
0	break_check_en	Interval detection enable. 1: detected over 11 bit lengths (if data_mode=1 or stop_mode=1, add 1 bit length respectively); 0: not detecting.

SCI_S1(C8H) UART1 interrupt status flag register

Bit number	7	6	5	4
Symbol	tx_empty_if	tx_finish_if	rx_full_if	idle_if
R/W	R	R	R	R
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	rx_overflow_if	noise_err_if	frame_err_if	Parity_err_if
R/W	R	R	R	R
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	tx_empty_if	Send buffer empty interrupt flag. 1: send buffer is empty; 0: send buffer is full, read only.
6	tx_finish_if	Send completion interrupt flag. 1: send completed, transmitter idle; 0: the transmitter is working, read only.
5	rx_full_if	Accept full interrupt flag. 1: receiver buffer is full; 0: receiver buffer is empty, read only.
4	idle_if	Idle line break flag. 1: idle line detected; 0: no idle line detected, read only.
3	rx_overflow_if	Receive overflow mark. 1: receive overflow (new data loss); \ 0: no overflow, read only.
2	noise_err_if	Noise marker. 1: noise detected; 0: no noise detected, read only.
1	frame_err_if	Frame error flag. 1: framing error detected; 0: no framing error detected, read only.
0	parity_err_if	Parity error flag. 1: receiver parity error; 0: parity is correct, read only.

SCI_D(C9H) UART1 data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description

7~0	-	SCI data register. Read returns the contents of the read-only receive data buffer, writes to the write-only send data buffer.

PSW(D0H) Program status register

Bit number	7	6	5	4	3	2	1	0
Symbol	CY	AC	F0	RS[1:0]		OV	F1	P
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 number	Bit symbol	Description
7	CY	Carry flag. Set when the addition generates a carry or subtracts a borrow, otherwise clears. Set when the first operand of CJNE is less than the second operand, cleared by MUL or DIV instruction. Also affected by mouse commands (RLC, RRC) and bitwise instructions.
6	AC	Auxiliary carry flag Set when the addition is borrowed from the third to fourth bits of the accumulator, or when the subtraction is borrowed from the third to fourth bits, otherwise cleared.
5	F0	0 flag bit. Universal label for users.
4~3	RS[1:0]	Working register group: Select a valid working register group: RS[1:0] Bank IRAM Area 00 0 0x00-0x07; 01 1 0x08-0x0F; 10 2 0x10-0x17; 11 3 0x18-0x1F
2	OV	Overflow flag bit When the addition produces a different carry of accumulator bits 6 and 7, or subtraction produces a borrow of accumulator bits 6 and 7, otherwise cleared. The OV flag indicates that the signed 8-bit result is out of bounds (greater than 127 or less than -128). The overflow flag is also set when the multiplication result is greater than 255 or an attempt is made to divide by 0.
1	F1	1 flag bit. Universal label for users.
0	P	Parity flag. Always contains the sum of Form 2 of all the bits in the accumulator.

RST_STAT (D7H) Reset flag register

Bit number	7	6	5	4	3	2	1	0

Symbol	-	DEBUG_F	SOFT_F	PROG_F	ADDROF_F	BO_F	PO_F	WDTRST_F
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	rst_state							

Bit number	Bit symbol	Description
7	-	Reserved
6	DEBUG_F	0: No effect; 1: Trim configuration reset occurred
5	SOFT_F	0: No effect; 1: Software reset occurred
4	PROG_F	0: No effect; 1: A programming reset occurred
3	ADDROF_F	0: No effect; 1: PC pointer overflow reset occurred
2	BO_F	0: No effect; 1: Brown-out reset occurred
1	PO_F	0: No effect; 1: Power-on reset occurred
0	WDTRST_F	0: No effect; 1: Watchdog timer overflow reset occurred

SCI_INT_CLR (D8H) UART1 interrupt flag clear register

Bit number	7	6	5	4
Symbol	clr_tx_empty_if	clr_tx_finish_if	clr_rx_full_if	clr_idle_if
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	clr_rx_overflow_if	clr_noise_err_if	clr_frame_err_if	clr_parity_err_if
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	clr_tx_empty_if	Transmit buffer empty interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
6	clr_tx_finish_if	Transmit complete interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
5	clr_rx_full_if	Receive full interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
4	clr_idle_if	Idle line interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.

3	clr_rx_overflow_if	Receive overflow flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
2	clr_noise_err_if	Noise flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
1	clr_frame_err_if	Frame flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
0	clr_parity_err_if	Parity error flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.

ADC_IO_SEL1 (D9H) ADC function selection register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	SEL_ADC[7:0]							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	SEL_ADC[7:0]	ADC function selection. 1: select ADC function; 0: do not select ADC function.

ADC_IO_SEL2(DAH) ADC function selection register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	SEL_ADC[15:8]							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	SEL_ADC[15:8]	ADC function selection. 1: select ADC function; 0: do not select ADC function

ADC_IO_SEL3(DBH) ADC function selection register 3

Bit number	7	6	5	4	3	2	1	0
Symbol	SEL_ADC[23:16]							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	SEL_ADC[23:16]	ADC function selection. 1: select ADC function; 0: do not select ADC function

ADC_IO_SEL4(DCH) ADC function selection register 4

Bit number	7	6	5	4	3	2	1	0

Symbol	-	-	-	-	-	-	SEL_ADC[25:24]
R/W	-	-	-	-	-	-	R/W
Reset value	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
1~0	SEL_ADC[25:24]	ADC function selection. 1: select ADC function; 0: do not select ADC function

PU_PA (DDH) PA port pull-up resistor selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	

Bit number	Bit symbol	Description
1~0	--	PA port pull-up resisor control register. Set PU_PA to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

PU_PB(DEH) PB port pull-up resistor selection register

Bit number		6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	PB port pull-up resisor control register. Set PU_PB to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

PU_PC(DFH) PC port pull-up resistor selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	PC port pull-up resisor control register. Set PU_PC to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up

		resistor, the pull-up resistor is 4.7K.

ACC(E0H) Accumulator

Bit number	7	6	5	4	3	2	1	0
Symbol	ACC							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	ACC	Accumulator The targe register is suitable for all arithmetic and logic operations.

IRCON2 (E1H) Interrupt flag register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	IE10	IE9	IE8
R/W	-	-	-	-	-	R/W	R/W	R/W
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
7~3	--	Reserved
2	IE10	UART1 interrupt flag
1	IE9	UART0 interrupt flag
0	IE8	LVDT interrupt flag

PU_PD (E2H) PD port pull-up resistor selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	PD port pull-up resisor control register. Set PU_PD to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

IICADD (E3H) IIC address register

Bit number	7	6	5	4	3	2	1	0
Symbol	IICADD[7:1]							-
R/W	R/W							-
Reset value	0							-

IICBUF (E4H) IIC transmit and receive data register

Bit number	7	6	5	4	3	2	1	0
Symbol	IICBUF							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	IICBUF	IIC transmit receive data buffer

IICCON (E5H) IIC configuration register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	IIC_RST	RD_SCL_EN	WR_SCL_EN	SCLEN	SR	IIC_EN
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	-	-	0	1	0	0	0	0

Bit number	Bit symbol	Description
7~6	--	Reserved
5	IIC_RST	IIC module reset signal 1: IIC module reset operation 0: IIC module works properly
4	RD_SCL_EN	Host read pull low clock line control bit. 1: enable the host to read and pull the low clock line function; 0: disable the host to read and pull the low clock line function.
3	WR_SCL_EN	Host write pull low clock line control bit. 1: enable the host to write and pull the low clock line function; 0: disable the host to write and pull the low clock line function.
2	SCLEN	IIC clock enable bit 1= clock work properly 0= pull down the clock line.
1	SR	IIC conversion rate control bit 1: Conversion rate control is turned off to adapt to the standard speed mode (100K); 0: Conversion rate control is enabled to adapt to fast speed mode (400K)
0	IIC_EN	IIC work enable bit 1= IIC normal work; 0= IIC not work

IEN1 (E6H) Interrupt enable register 1

Bit number	7	6	5	4	3	2	1	0

Symbol	EX7	-	EX5	EX4	EX3	EX2	-	-
R/W	R/W	-	R/W	R/W	R/W	R/W	-	-
Reset value	0	-	0	0	0	0	-	-

Bit number	Bit symbol	Description
7	EX7	WDT/Timer2 interrupt enable
6	EX6	LED interrupt enable
5	--	Reserved
4	EX4	ADC interrupt enable
3	EX3	IIC interrupt enable
2	EX2	External interrupt 2 interrupt enable
1~0	-	Reserved

IEN2(E7H) Interrupt enable register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	EX10	EX9	EX8
R/W	-	-	-	-	-	R/W	R/W	R/W
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
7~3	-	Reserved
2	EX10	UART1 interrupt enable
1	EX9	UART0 interrupt enable
0	EX8	LVDT interrupt enable

IICSTAT (E8H) IIC status register

Bit number	7	6	5	4
Symbol	IIC_START	IIC_STOP	IIC_RW	IIC_AD
R/W	R	R	R	R
Reset value	0	1	0	0
Bit number	3	2	1	0
Symbol	IIC_BF	IIC_ACK	IIC_ACK	IIC_RECOV
R/W	R	R	R/W	R/W
Reset value	0	1	0	0

Bit number	Bit symbol	Description
7	IIC_START	Start signal flag 1: boot bit detected; 0: no boot bit detected
6	IIC_STOP	Stop signal flag 1: stop status detected;

		0: no stop status detected
5	IIC_RW	Read and write flag. Record the read/write information obtained from the address byte after the last address match. 1: read; 0: write.
4	IIC_AD	Address data flag bit. 1: indicates that the most recently received or sent byte is data; 0: indicates that the most recently received or sent byte is address.
3	IIC_BF	IICBUF full flag. Received in IIC bus mode: 1: received successfully, buffer is full; 0: received successfully, buffer is empty. Send in IIC bus mode 1: data transmission is in progress(does not include the acknowledge bit and the stop bit), buffer is full; 0: data transmission has been completed(does not include the acknowledge bit and the stop bit), buffer is empty.
2	IIC_ACK	Answer flag 1: invalid response signal; 0: effective response signal.
1	IIC_WCOL	Write conflict flag. 1: when the IIC is transmitting the current data, the new data is attempted to be written to the transmit buffer; new data cannot be written to the buffer. 0: no write conflict
0	IIC_RECOV	Receive overflow flag bit 1: When the previous data received by the IIC has not been taken, new data is received, the new data cannot be received by the buffer. 0: no receive overflow.

IICBUFFER (E9H) IIC transmit and receive data buffer register

Bit number	7	6	5	4	3	2	1	0
Symbol	IICBUFFER							
R/W	R/W							
Reset value	0							

TRISA (EAH) PA port direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	

Reset value	-	-	-	-	-	-	1	1

Bit number	Bit symbol	Description
1~0	--	PA direction register, 0: output; 1: input

TRISB(EBH) PB port direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PB direction register, 0: output; 1: input

TRISC(ECH) PC port direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PC direction register, 0: output; 1: input

TRISD(EDH) PD port direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PD direction register, 0: output; 1: input

COM_IO_SEL (EEH) COM large sink current selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description

7~0	--	COM port select configure register, corresponding PB port. 1: select COM port mode; 0: select IO port mode.

ODRAIN_EN (EFH) PA open drain enable register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1~0	--	PA0/1 port open drain output enable register 1: open drain output 0: CMOS output

B (F0H) B register

Bit number	7	6	5	4	3	2	1	0
Symbol	B							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	B	B register. Source and destination registers formultiplication and division.

IRCON1 (F1H) Interrupt flag register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	IE7	-	IE5	IE4	IE3	IE2	-	-
R/W	R/W	-	R/W	R/W	R/W	R/W	-	-
Reset value	0	-	0	0	0	0	-	-

Bit number	Bit symbol	Description
7	IE7	WDT/Timer2 interrupt flag
6	IE6	LED interrupt flag
5	--	Reserved
4	IE4	ADC interrupt flag
3	IE3	IIC interrupt flag
2	IE2	External interrupt 2 interrupt flag
1~0	–	Reserved

PERIPH_IO_SEL (F2H) IIC/UART0/INT function control register

Bit number	7	6	5	4	3
Symbol	-	IIC_AFIL_SEL	IIC_DFIL_SEL	UART0_IO_SEL	

R/W	-	R/W	R/W	R/W	R/W
Reset value	-	1	0	0	0
Bit number	2	1	0	/	
Symbol	INT2_IO_SEL	INT1_IO_SEL	INT0_IO_SEL	/	
R/W	R/W	R/W	R/W		
Reset value	0	0	0		

Bit number	Bit symbol	Description
6	IIC_AFIL_SEL	IIC port analog filter selection enable 1: select analog filter function; 0: do not select analog filter function.
5	IIC_DFIL_SEL	IIC port digital filter selection enable. 1: select digital filter function; 0: do not select digital filter function.
4~3	UART0_IO_SEL	UART0 select enable. 00: select UART0(RXD0_A/TXD0_A) function 01: select UART0(RXD0_B/TXD0_B) function 1x: select UART0(RXD0_C/TXD0_C) function
2	INT2_IO_SEL	INT2 select enable, correspond PD7 1: select INT2 function 0: not select INT2 function
1	INT1_IO_SEL	INT1 select enable, correspond PD6 1: select INT1 function 0: not select INT1 function
0	INT0_IO_SEL	INT0 select enable, correspond PD0 1: select INT0 function 0: not select INT0 function

IPL2 (F4H) Interrupt priority register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	IPL2.2	IPL2.1	IPL2.0
R/W	-	-	-	-	-	R/W	R/W	R/W
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
7~3	--	Reserved
2	IPL2.2	UART1 interrupt priority. 1: high; 0: low.
1	IPL2.1	UART0 interrupt priority. 1: high; 0: low.
0	IPL2.0	LVDT interrupt priority.

		1: high; 0: low.

IPL1 (F6H) Interrupt priority register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	IPL1.7	IPL1.6	-	IPL1.4	IPL1.3	IPL1.2	-	-
R/W	R/W	R/W	-	R/W	R/W	R/W	-	-
Reset value	0	0	-	0	0	0	-	-

Bit number	Bit symbol	Description
7	IPL1.7	WDT/Timer 2 interrupt priority. 1: high; 0: low.
6	IPL1.6	LED interrupt priority. 1: high; 0: low.
5	--	Reserved
4	IPL1.4	ADC interrupt priority. 1: high; 0: low.
3	IPL1.3	IIC interrupt priority. 1: high; 0: low.
2	IPL1.2	External interrupt priority. 1: high; 0: low.
1~0	--	Reserved

EXT_INT_CON (F7H) External interrupt polarity control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	INT2_POLARITY		INT1_POLARITY		INT0_POLARITY	
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	-	-	0	1	0	1	0	1

Bit number	Bit symbol	Description
5~4	INT2_POLARITY	External interrupt 2 trigger polarity selection: INT2_POLARITY=01: falling edge (Low wake-up in low power mode) INT2_POLARITY=10: rising edge (Low wake-up in high power mode) INT2_POLARITY=00/11: double edge (Low wake-up in low power mode).
3~2	INT1_POLARITY	External interrupt 1 trigger polarity selection: INT1_POLARITY=01: falling edge (Low wake-up in low power mode) INT1_POLARITY=10: rising edge (Low wake-up in high power mode)

		INT1_POLARITY=00/11: double edge (Low wake-up in low power mode).
1~0	INT0_POLARITY	External interrupt 0 trigger polarity selection: INT0_POLARITY=01: falling edge (Low wake-up in low power mode) INT0_POLARITY=10: rising edge (Low wake-up in high power mode) INT0_POLARITY=00/11: double edge (Low wake-up in low power mode).

DATAA (F8H) PA data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	PA1	PA0
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	1	1

Bit number	Bit symbol	Description
1~0	--	PA data register. The output level of the PA group can be configured as the GPIO port. The read value is the level state of the current IO port (input) or the configured output (output) value.

SPROG_ADDR_H (F9H) Address control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-		
R/W	-	-	-	-	-	R/W		
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
2~0	--	When EEP_SELECT = 0, Bit[2]: DATA area selection enable, 0: select 0x3C00~0x3FFF; 1: reserved. {SPROG_ADDR_H[1:0], SPROG_ADDR_L[7:0]} means 0x3C00~0x3FFF address When EEP_SELECT = 1, Bit[2] = 0, select NVR3 (512Bytes); Bit[2] = 1, select NVR4 (512Bytes) {SPROG_ADDR_H[0], SPROG_ADDR_L[7:0]} represents the byte address within the page Bit[1]: reserved.

SPROG_ADDR_L(FAH) Address register,lower 8 bits

Bit number	7	6	5	4	3	2	1	0

Symbol	-
R/W	R/W
Reset value	0

Bit number	Bit symbol	Description
7~0	--	Lower 8 bits of address

SPROG_DATA(FBH) Data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Data to be written

SPROG_CMD(FCH) Command register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Write 0x96: page erase; Write 0x69: byte burning.

SPROG_TIM(FDH) Erase time control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	1	0	1	1	0	1	0

Bit number	Bit symbol	Description
7~5	--	Byte write time is fixed at 23.5us
4~0	--	When EEP_SELECT=0, bit[4:0]: 0~9 corresponds to the erasing time (1~10ms) + 0.13ms (step 1ms), >9 is 10.13ms. When EEP_SELECT=1, bit[4:0]: 0~9 corresponds to erasing time (0.5~5ms) + 0.065ms (step 0.5ms), and when >9, it is 5.065ms.SPROG_TIM[4:0] > 9: erasing time =5.07(ms)

PD_ANA (FEH) Module switch control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	PD_LVDT	PD_BOR	PD_XTAL_32K	-	PD_ADC
R/W	-	-	-	R/W	R/W	R/W	-	R/W
Reset value	-	-	-	1	1	1	-	1

Bit number	Bit symbol	Description
4	PD_LVDT	LVDT control register. 1: close; 0: open; default close.
3	PD_BOR	BOR control register. 1: close; 0: open; VBOR=2.1V , default close.
2	PD_XTAL_32K	RTC crystal oscillator circuit (32768Hz/4MHz) control register. 1: close; 0: open; default close.
1	--	Reserved
0	PD_ADC	Analog ADC shutdown control register: PD_ADC=0 ADC module work properly; PD_ADC=1 ADC module don’t work.

SEL_LVDT_VTH (FFH) LVDT threshold select register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1~0	--	LVDT threshold selection; 00=2.4V; 01=3.0V; 10=3.6V; 11=4.2V.

SFR register detail table

Note: For reserved registers and reserved bits of registers, write operations are prohibited, otherwise it may cause chip exceptions

4.2. Secondary Bus Register Details

CFG0_REG(00H) Configuration word register 0

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG1_REG(01H) Configuration word register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							

Reset value	FF

CFG2_REG(02H) Configuration word register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG3_REG(03H) Configuration word register 3

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG4_REG(04H) Configuration word register 4

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG5_REG(05H) Configuration word register 5

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG6_REG(06H) Configuration word register 6

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG7_REG(07H) Configuration word register 7

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG8_REG(08H) Configuration word register 8

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG9_REG(09H) Configuration word register 9

Bit number	7	6	5	4	3	2	1	0
Symbol	-							

R/W	R
Reset value	FF

CFG10_REG(0AH) Configuration word register 10

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG11_REG(0BH) Configuration word register 11

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG12_REG(0CH) Configuration word register 12

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG13_REG(0DH) Configuration word register 13

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG14_REG(0EH) Configuration word register 14

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG15_REG(0FH) Configuration word register 15

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG16_REG(10H) Configuration word register 16

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG17_REG(11H) Configuration word register 17

Bit number	7	6	5	4	3	2	1	0

Symbol	-
R/W	R
Reset value	FF

CFG18_REG(12H) Configuration word register 18

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG19_REG(13H) Configuration word register 19

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG20_REG(14H) Configuration word register 20

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG21_REG(15H) Configuration word register 21

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG22_REG(16H) Configuration word register 22

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG23_REG(17H) Configuration word register 23

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG24_REG(18H) Configuration word register 24

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG25_REG(19H) Configuration word register 25

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

CFG30_REG(1AH) Configuration word register 30

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R							
Reset value	FF							

DUMMY_REG(1FH) RTC crystal oscillator circuit selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	XTAL_CLK_SEL
R/W	-	-	-	-	-	-	-	RW
Reset value	-	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
7~1	--	Reserved
0	XTAL_CLK_SEL	1: XTAL4MHz 0: XTAL32768Hz

EEP_SELECT (20H) DATA area selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	-
R/W	-	-	-	-	-	-	-	RW
Reset value	-	-	-	-	-	-	-	0

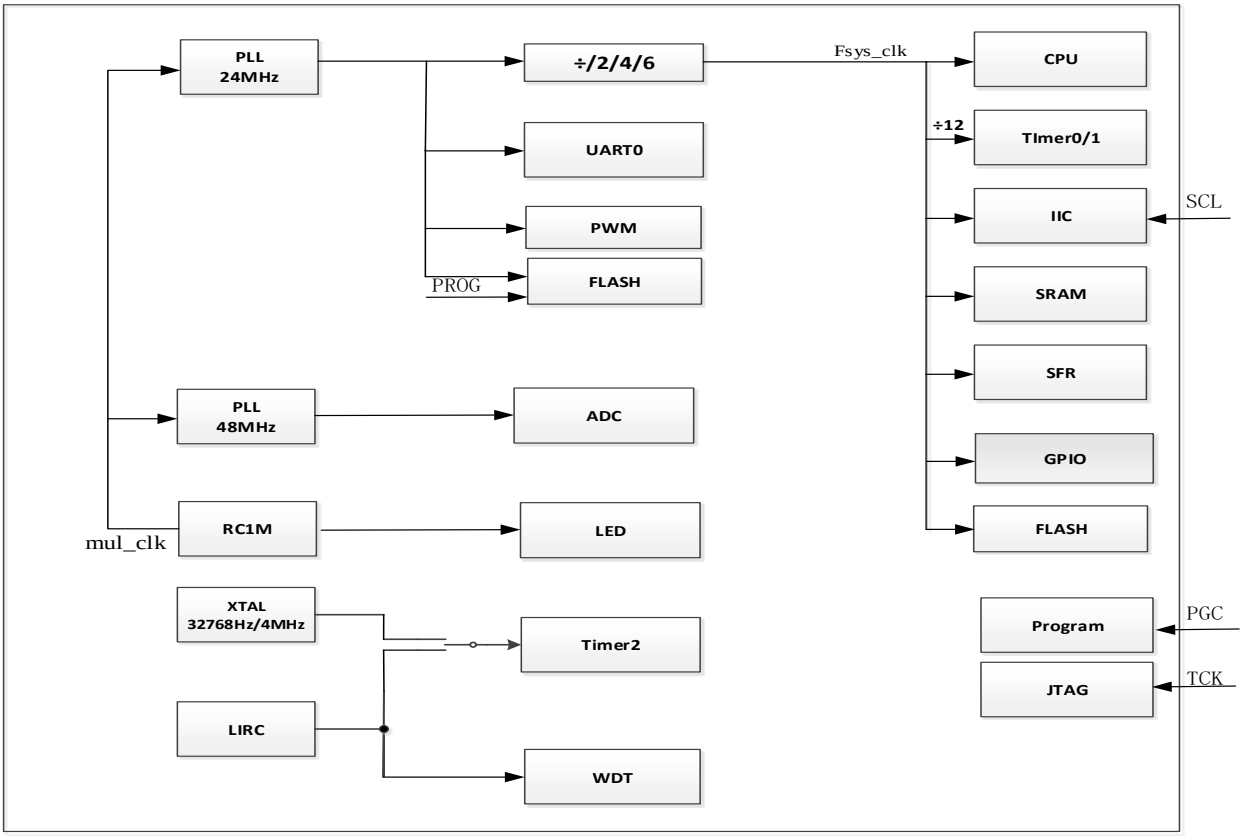
Bit number	Bit symbol	Description
7~1	--	Reserved
0	--	1: Select NVR3/4 as DATA area. NVR3, 1 page, 512 Bytes; NVR4, 1 page, 512 Bytes 0: Select DATA area (0x3C00~0x3FFF), 1 page, 1024 Bytes

5. Clock, Reset, Work Mode, WDT

5.1. Clock Definition

Clock source:

- Internal high-speed RC oscillator: RC1M
- Internal low-speed RC oscillator: LIRC32k
- External crystal oscillator: 32768 Hz/4 MHz
- RC1M multiplier to get PLL clock: PLL48M/ PLL24M



Clock block diagram

BF7512CMXX-SJLX clock definition:

PLL_24MHz: Frequency 24 MHz, gating module enabled, used as UART, Flash, PWM clock. The F_sys_clk clock is obtained by dividing by 2/4/6.

F_sys_clk: The system clock is 12MHz/6MHz/4MHz, which can be used as the core-related module clock.

XTAL32768Hz/4MHz: External 32768Hz/4MHz precision clock, can be used as Timer2 clock.

RC1MHz: Built-in RC oscillator with a frequency of 1MHz, as the LED drive clock.

LIRC: Internal low-speed clock 32kHz, this clock is used as watchdog clock and Timer2 clock.

PLL_48MHz: The 48MHz clock generated by the phase-locked loop is directly used for ADC and Flash control, and the system clock is obtained after frequency division.
SCL: IIC master clock, frequency 100kHz/400kHz, sent by IIC master as IIC communication clock.
PGC: Programming clock, frequency 400kHz~5MHz, download clock when programming and burning programs.
TCK: Test clock.

5.2. System Clock Selection Register Detailed Description

SYS_CLK_CFG(84H) System clock configuration register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	PLL_CLK_SEL[1:0]	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	1

Bit number	Bit symbol	Description
7~2	--	Reserved
1~0	PLL_CLK_SEL	PLL clock divider selection register: 00: 12Mhz; 01: 6Mhz; 10: 4Mhz; 11: Reserved

PD_ANA (FEH) Module switch control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	PD_LVDT	PD_BOR	PD_XTAL_32K	-	PD_ADC
R/W	-	-	-	R/W	R/W	R/W	-	R/W
Reset value	-	-	-	1	1	1	-	1

Bit number	Bit symbol	Description
2	PD_XTAL_32K	RTC crystal oscillator circuit (32768Hz/4MHz) control register. 1: close; 0: open; default close.

Secondary bus register:

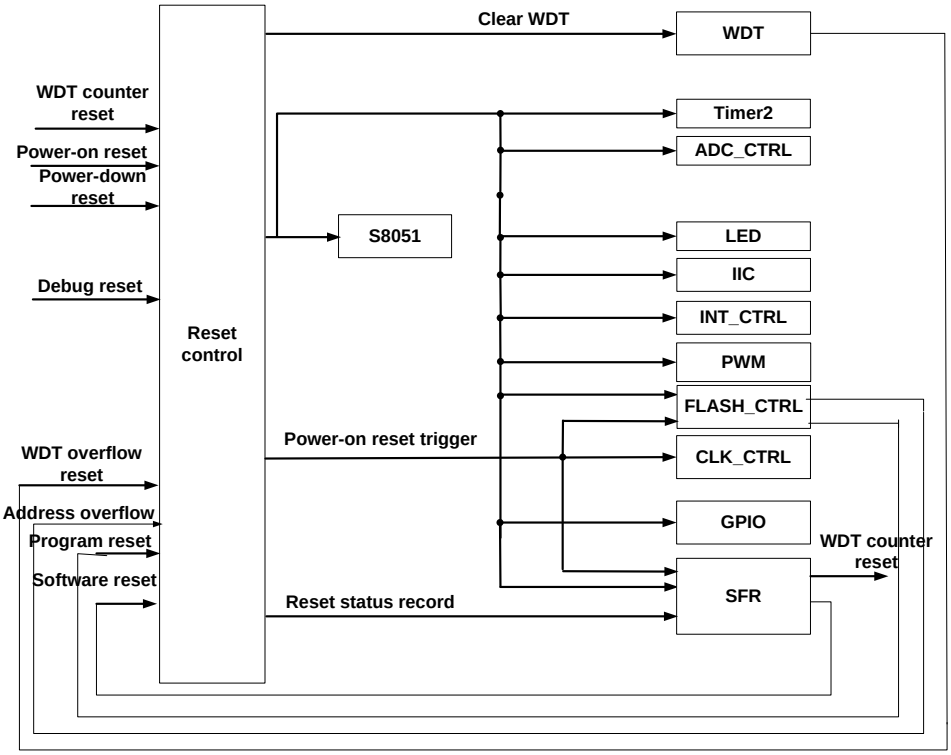
DUMMY_REG(1FH) RTC crystal oscillator circuit selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	XTAL_CLK_SEL
R/W	-	-	-	-	-	-	-	RW
Reset value	-	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
7~1	--	Reserved
0	XTAL_CLK_SEL	1: XTAL4MHz 0: XTAL32768Hz

5.3. Reset

There are seven reset mode in BF7512CMXX-SJLX: WDT overflow reset (WDTRST_F), power on reset (PO_F), power down reset (BO_F), program reset (PROG_F), debug reset (DEBUG_F), PC pointer overflow reset (ADDROF_F), software reset (SOFT_F). Any one of above reset, global will make chip reset. We can judge the reset flag register which reset happen, the reset must be cleared by software.



Reset block diagram

5.3.1. Reset Sequence

po_n: Power-on reset. After the system is powered on, the analog module generates a low-level signal and lasts for 93ms. When the power-on reset is low, the entire chip is in the reset state, and after the global reset signal continues to be effective 20ms after the power-on reset is high, the system exits the reset mode.

bo_n: Brown-out reset, the analog module generates a low-level signal after the system has a power-down reset. When the power-down reset signal is low, the entire chip is in the reset state. After the global reset signal becomes high, the system exits the reset mode after the global reset signal continues to be valid for 20ms.

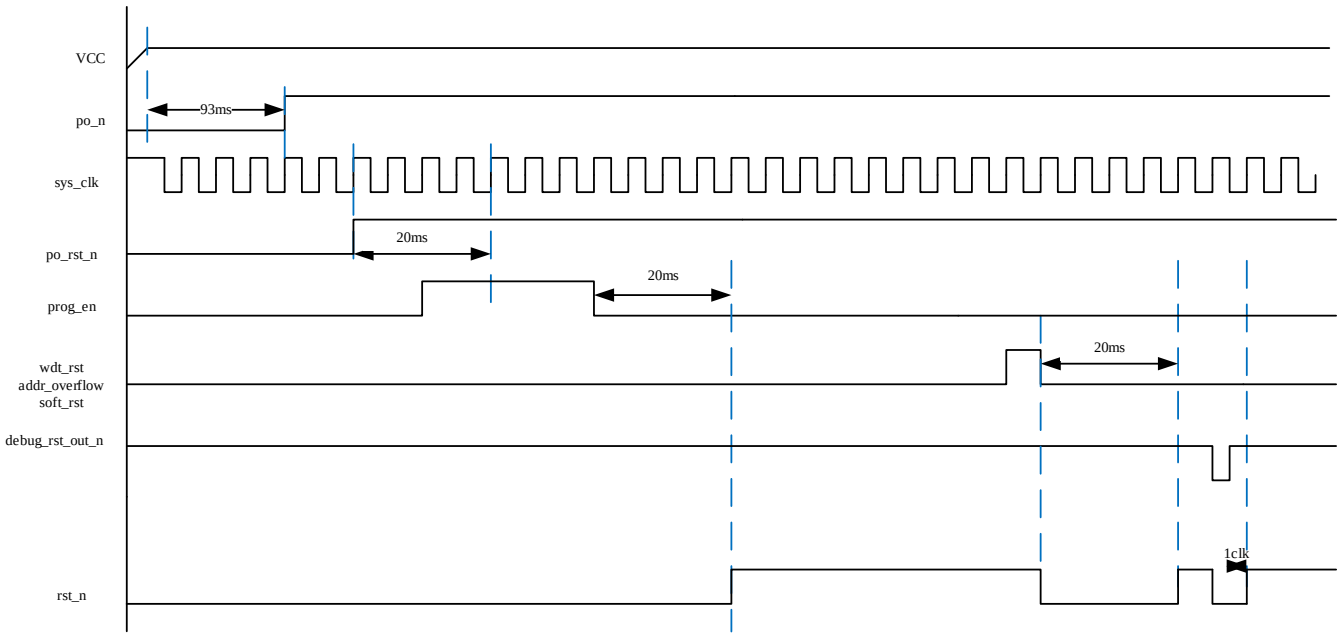
prog_en: FLASH Programming reset. When prog_en is high, it is the programming mode of FLASH. At this time, the global reset signal is valid. After it goes low, the global reset signal continues to be valid for 20ms.

WDT Overflow Reset, reset the global for 20ms after the WDT overflow. After 20ms, the system exits the reset mode.

addr_overflow: PC pointer overflow reset. If the PC pointer exceeds the valid address range of the flash when the MCU addresses the program memory, the addr_overflow signal becomes high, and the sys_clk clock rising edge detects the high level of addr_overflow (requires 1 clock cycle) and resets the global 20ms, the reset signal will clear the addr_overflow signal to zero. After 20ms, the system exits the reset mode.

Software reset, make the soft reset signal vaild by writing the SFR, so that the global reset signal is active 20ms. After 20ms, the system exits the reset mode.

Debug Reset, for the core repair module output reset signal, low means reset is valid. Chip global reset, there will be no 20ms initialization process, only one system clock reset low.

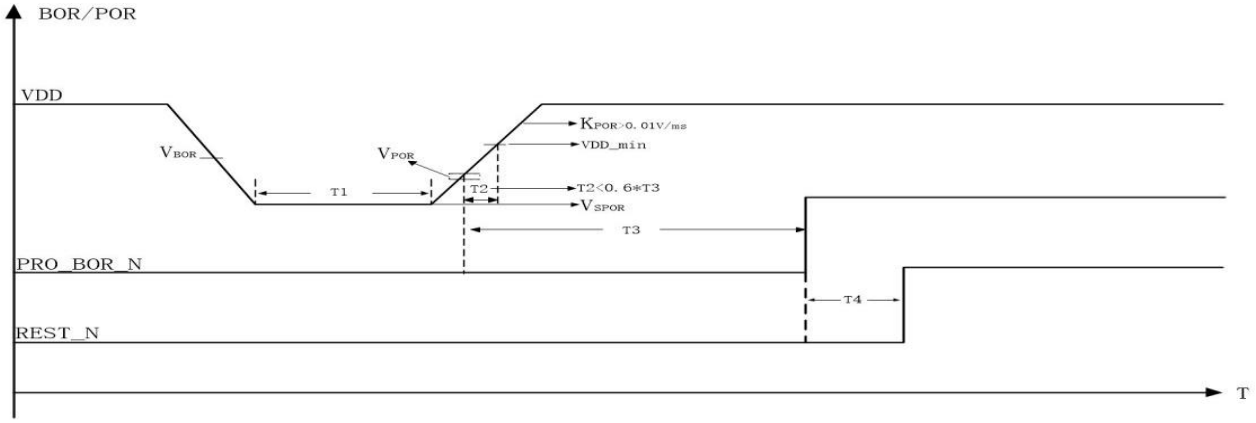


Reset timing diagram

Reset sequence description:

1. The chip has a power-on reset, and the analog POR module delays for 93ms, and po_n is pulled high.
2. The programmer sends instructions to make the chip enter the programming mode (prog_en is pulled high), and the system is in a global reset state in the programming mode. After the programming is completed, the programming mode is exited. After a delay of 20ms, rst_n is pulled high and the chip enters normal operation.
3. During normal operation, any one of watchdog reset, address overflow reset, and soft reset occurs, rst_n is pulled low, after a delay of 20ms, rst_n is pulled high, and the chip enters normal operation.
4. In debug mode, configure debug reset, pull down rst_n, pull up 1 system clock in debug_rst_out_n, pull up rst_n, and the chip enters normal operation.

Power-up/power-down sequence:



Power-on reset diagram

BOR/POR Parameters:

Symbol	Parameter	Min	Typ	Max	Unit
VSPOR	Power on reset start voltage	-	-	300	mV
KPRO	Power on reset voltage rate	0.01	-	-	V/ms
VPOR	Power on reset voltage	1.1	1.5	2.2	V
VBOR	Brownout reset voltage (±10%), hysteresis 0.2V	-	2.1	-	V
VDD_min	Minimum operating voltage	2.5	-	-	V
T1	VDD keep VSPOR time	0.1	-	-	ms
T2	VPOR from VDD_min time	-	-	0.6*T3	ms
T3	Reset POR_BOR_N duration	55	93	131	ms
T4	Global reset effective time	-	20	-	ms

Power on reset parameter characteristic table

When VDD is affected by the load or seriously interfered, if the voltage drops into the voltage dead zone and the chip is not within the working voltage range, it may cause the system to work abnormally, such as data loss in the DATA area. The function of power-down reset (BOR) is to monitor that when VDD drops to the BOR voltage, the MCU can generate a power-down reset in advance to avoid system errors.

Suggestions to prevent entering the voltage dead zone and reduce the probability of system

error:

- When the program is first initialized, open BOR without delay
- Increase the voltage drop slope

5.3.2. Registers

SFR				
Address	**Name**	**RW**	**Reset value**	**Description**
0x8E	SOFT_RST	RW	0x00	Soft reset register
0xD7	RST_STAT	RW	rst_state	Reset flag register
0xFE	PD_ANA	RW	0x1F	Module switch control register

SOFT_RST(8EH) Software reset register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Software reset register, software reset is only generated when the register value is 0x55.

RST_STAT (D7H) Reset flag register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	DEBUG_F	SOFT_F	PROG_F	ADDROF_F	BO_F	PO_F	WDTRST_F
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	rst_state							

Bit number	Bit symbol	Description
7	-	Reserved
6	DEBUG_F	0: No effect; 1: Trim configuration reset occurred
5	SOFT_F	0: No effect; 1: Software reset occurred
4	PROG_F	0: No effect; 1: A programming reset occurred
3	ADDROF_F	0: No effect; 1: PC pointer overflow reset occurred
2	BO_F	0: No effect; 1: Brown-out reset occurred
1	PO_F	0: No effect;

		1: Power-on reset occurred
0	WDTRST_F	0: No effect; 1: Watchdog timer overflow reset occurred

Power-on reset: the reset value of the RST_STAT register is 0x02; other mode resets: the reset flag bit corresponding to the RST_STAT register is 1, and the other reset flag bits remain in their original state.

PD_ANA (FEH) Module switch control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	PD_LVDT	PD_BOR	PD_XTAL_32K	-	PD_ADC
R/W	-	-	-	R/W	R/W	R/W	-	R/W
Reset value	-	-	-	1	1	1	-	1

Bit number	Bit symbol	Description
3	PD_BOR	BOR control register. 1: close; 0: open; VBOR=2.1V , default close. Note: It is recommended to turn on BOR when the program is first initialized

5.4. Working Mode

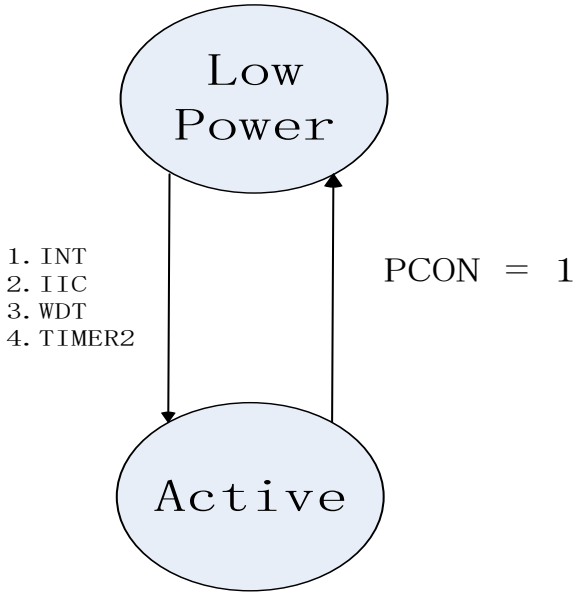
The BF7512CMXX-SJLX series has 2 working modes, which can be selected according to different situations.

- Active Mode
  Normal working mode, the module remains working properly. The module keeps working normally, and each module function is controlled by software configuration.
- Low_power mode
  PCON=1, close RC1M and PLL, LIRC keeps working, WDT/TIMER2 can be configured to work, CPU and remaining digital modules are not working.

PCON(87H) low power mode selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	LPM
R/W	-	-	-	-	-	-	-	R/W
Reset value	-	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
0	LPM	Low power mode control 1: Low power consumption mode; 0: Normal mode, automatically cleared after wake-up



Working mode conversion diagram

Otherwise, all modules can be individually configured to turn off the gate to reduce power consumption. As in active mode, configurable to turn off CPU and system module clock, and only enable modules such as LED/ADC modules; In low power mode, you can also configure to turn off the LIRC clock. This stops all clock sources and achieves the lowest power consumption. At this

time, only the external interrupt can be used to wake up the system.

Exit low_power mode method:

Enable IIC, External Interrupt0, External Interrupt1, External Interrupt2, WDT, Timer2, Any one of the interrupt generations can wake up the chip, exit low_power mode. After the interrupt response is generated, the CPU executes the interrupt vector related interrupt serive routine, and returns to the next instruction that causes the CPU to enter the low_power mode after the RETI return instruction is executed to continue the program.

Note: PCON = 0x01, BOR off can obtain lower power consumption, but the chip needs to ensure that it is in the normal operating voltage range (2.5V~5.5V), if the chip power supply is unstable, resulting in less than 2.5V, it is strongly recommended that BOR be turned on.

Mode	Contions for entering this mode	Effect on the clock	
Active	PCON=0;	LIRC	Depends on software configuration
		RC1M	work
		PLL	work
Low Power	PCON=1;	LIRC	Depends on software configuration
		RC1M	close
		PLL	close

Working status table of clock source in each mode

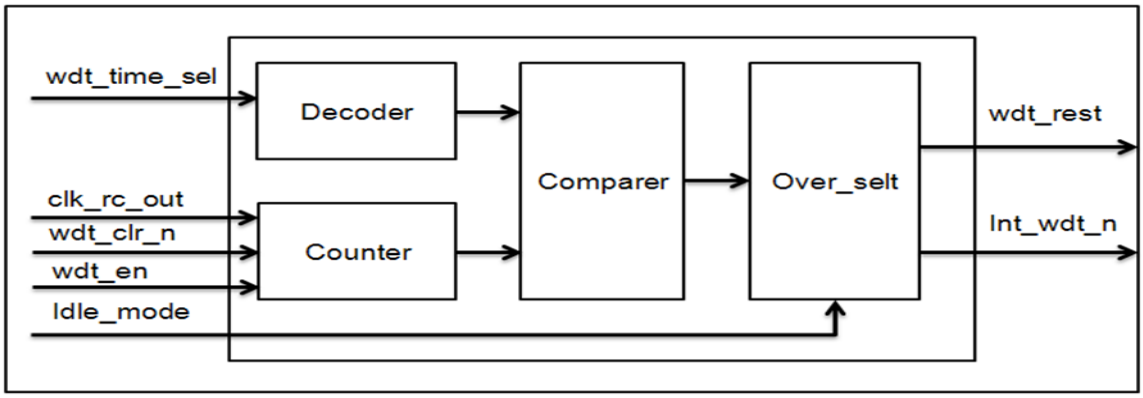
NO	Module Name	Clock source	Work status	
			Active	Low Power
1	s8051	PLL24M	√	×
2	UART0	PLL24M	Configured according to the program	×
3	PWM0~2	PLL24M	Configured according to the program	×
4	Internal Timer0	PLL24M	Configured according to the program	×
5	Internal Timer1	PLL24M	Configured according to the program	×
6	External Timer2	LIRC/XTAL	Configured according to the program	Configured according to the program
7	LED	RC1M	Configured according to the program	×
8	UART1	clk_sys12M(contain RC1M, PLL)	Configured according to the program	Configured according to the program
9	WDT	LIRC	Configured according	Configured according

			to the program	to the program
10	Adc_ctrl	PLL48M	Configured according to the program	×
11	IIC(S)	PLL24M	Configured according to the program	Configured according to the program

Status table for each digital module in different modes

5.5. WDT

The WDT timing counting circuit uses the internal low-speed clock LIRC for timing, and the configurable timing time is 2^n*18ms (n=0,1,2,3,4,5,6,7),-----here n The configuration value of the timing configuration register.



Classification of WDT overflow signals due to the particularity of the system applications:

In normal mode, if the WDT overflow occurs, the overflow signal is the WDT overflow reset signal, the WDT overflow reset affects the global reset. At this point, the system implements a global reset action and reloads the configuration information.

In IDLE mode, if the WDT overflow, the overflow signal is the WDT interrupt signal. Interrupt wake-up chip exits IDLE mode and executes WDT interrupt service function.

The watchdog module is a timing counting module. Its count clock is the internal low-speed clock LIRC. Its timing clear signal is composed of global reset and configuration clear. This signal is synchronously released by the watchdog timing clock in the reset module; The clearing action is generated every time the CPU configures the watchdog timer configuration register (WDT_CTRL), and the watchdog restarts timing; at the same time, the watchdog counter has the watchdog count enable control, when the count enable is valid , After the watchdog generates a timing overflow (reset or interrupt), as long as the watchdog counting enable is not turned off, the watchdog counter will restart counting.

5.5.1. WDT Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description**
0x91	WDT_CTRL	RW	0x00	WDT timeout configugration register
0x92	WDT_EN	RW	0x00	WDT timing enable configuration register

WDT SFR register list

Watchdog clock register:

The watchdog uses the internal low-speed clock LIRC to complete the timing function and can achieve timing from 18ms to 2.3s. The timing length is controlled by SFR (WDT_CTRL), as shown in the following table:

Watchdog Clock register WDT_CTRL – 0x91h

WDT_CTRL<2:0>	**Interval**
000	18ms
001	36ms
010	72ms
011	144ms
100	288ms
101	576ms
110	1152ms
111	2304ms

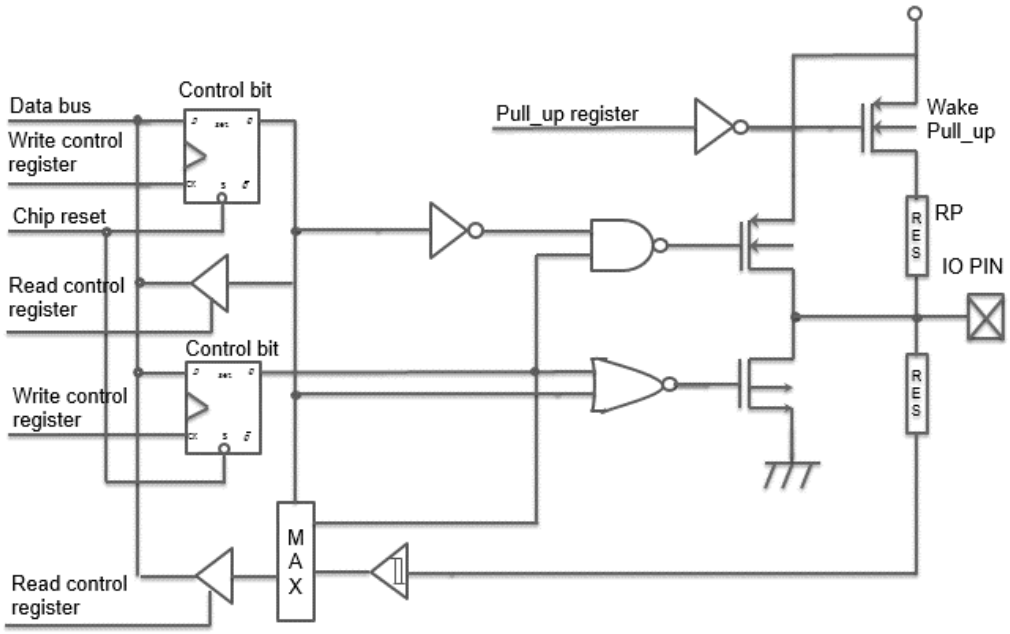
Watchdog Enable Register WDT_EN– 0x92h

SFR	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
WDT_EN	WDT_EN<7:0>							
SFR	0x00							

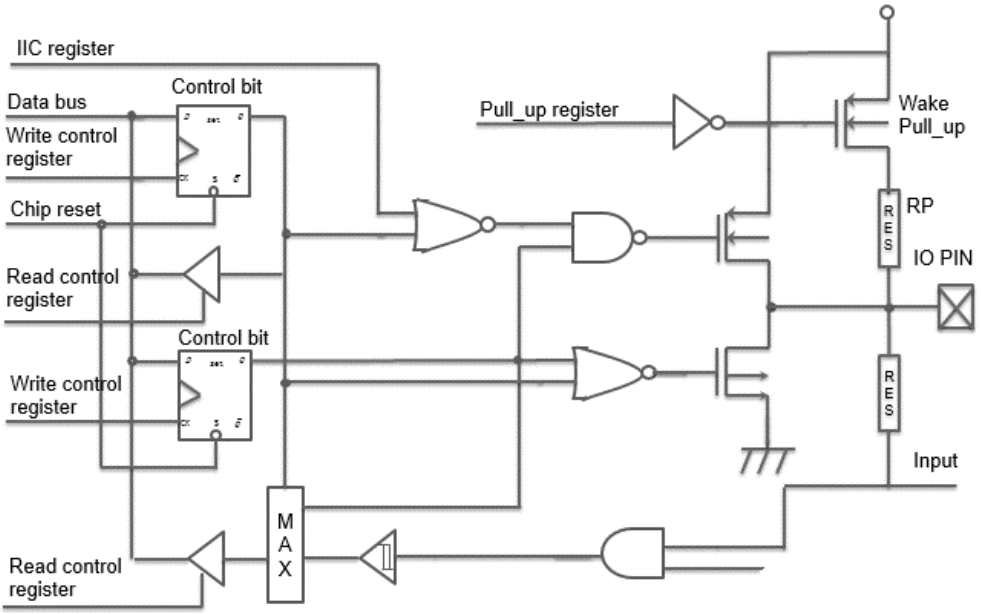
Turn off WDT when writing 0x55, write other values to enable WDT, the WDT always works after the reset is over. Clearing the WDT is done by writing to the WDT_CTRL register. Whichever values is written to this register will clear the WDT.

6. GPIO

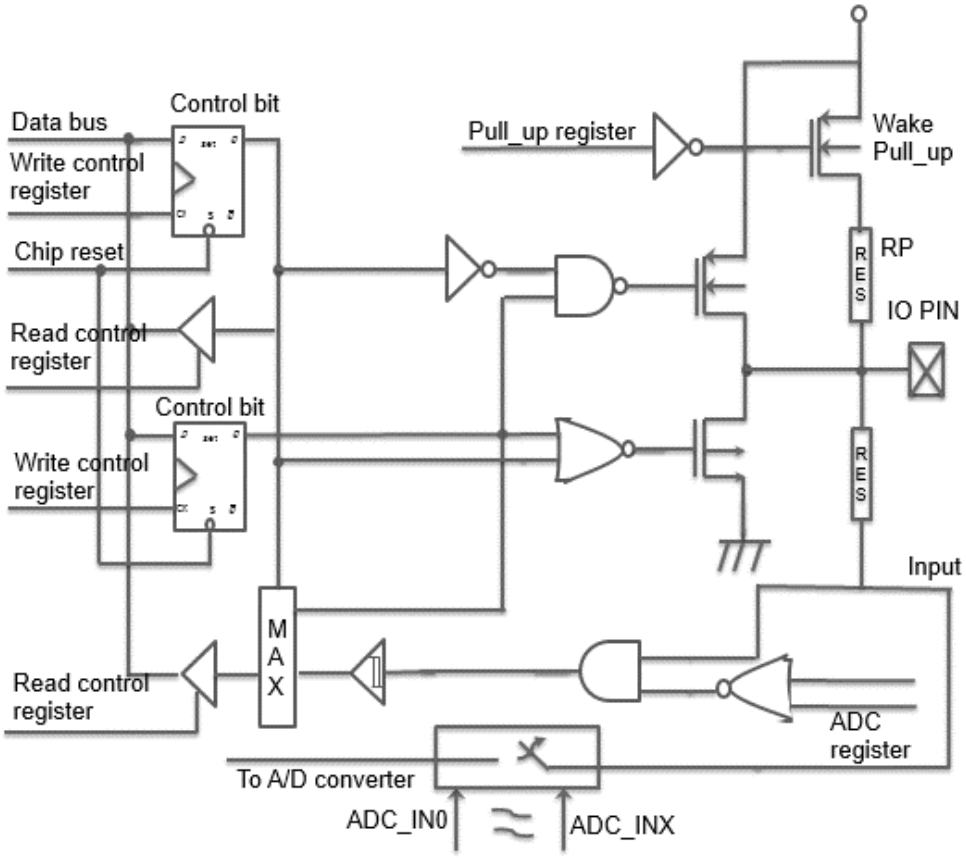
Some pins of the GPIO port are multiplexed with device peripheral functions, and cannot be configured as multiple clock functions at the same time, otherwise it will cause malfunction. IIC communication port, open-drain output, pull-up resister required.



General IO structure



Open-drain output structure



ADC IO structure

TRISX register (Direction Register): TRISX set to 1 can be configured as input pin, set to 0 can be configured as output pin.

DATAX register (Data Register): DATAX set to 1 the data in DATAX will be configured as high, set to 0 the data in DATAX will be configured as low.

PU_PX register (Pull-up resistance enable register): PU_PX set to 1 corresponding pin pull-up resistor enable, clear the corresponding pin does not enable pull-up resistor, and pull_up resistor 4.7k.

ODRAIN_EN register: ODRAIN_EN set to 1 corresponding pin will enable open drain output, set to 0 corresponding pin corresponding pin output disenabled, automatically turn on open-drain after enabling IIC function. IIC/UART recommends using external pull-up resistors.

Supports 8 GPIO ports for high current drive functions, software IO analog LED row and column drive timing.

6.1. GPIO Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description**
0xF8	DATAA	RW	0x03	PA data register
0x80	DATAB	RW	0xFF	PB data register
0x90	DATAC	RW	0xFF	PC data register
0x98	DATAD	RW	0xFF	PD data register
0xDD	PU_PA	RW	0x00	PA port pull-up resistor selection register
0xDE	PU_PB	RW	0x00	PB port pull-up resistor selection register
0xDF	PU_PC	RW	0x00	PC port pull-up resistor selection register
0xE2	PU_PD	RW	0x00	PD port pull-up resistor selection register
0xEA	TRISA	RW	0x03	PA direction register
0xEB	TRISB	RW	0xFF	PB direction register
0xEC	TRISC	RW	0xFF	PC direction register
0xED	TRISD	RW	0xFF	PD direction register
0xEE	COM_IO_SEL	RW	0x00	COM large sink current selection register
0xEF	ODRAIN_EN	RW	0x00	PA open drain enable register

Port configuration SFR list

6.2. GPIO Register Details

6.2.1. Data Registers

DATAA (F8H) PA data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	PA1	PA0
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	1	1

Bit number	Bit symbol	Description
1~0	--	PA data register. The output level of the PA group can be configured as the GPIO port. The read value is the level state of the current IO port (input) or the configured output (output) value.

DATAB(80H)PB data register

Bit number	7	6	5	4	3	2	1	0
Symbol	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
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 number	Bit symbol	Description
7~0	--	PBdata register. The output level of the PB group can be configured as the GPIO port. The read value is the level state of the current IO port (input) or the configured output (output) value.

DATAC(90H) PC data register

Bit number	7	6	5	4	3	2	1	0
Symbol	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PC data register. The output level of the PC group can be configured as the GPIO port. The read value is the level state of the current IO port (input) or the configured output (output) value.

DATAD(98H) PD data register

Bit number	7	6	5	4	3	2	1	0
Symbol	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PD data register. The output level of the PD group can be configured as the GPIO port. The read value is the level state of the current IO port (input) or the configured output (output) value.

6.2.2. Pull-up Resistor Selection Registers

PU_PA (DDH) PA port pull-up resisor control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	

Bit number	Bit symbol	Description
1~0	--	PA port pull-up resisor control register. Set PU_PA to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

PU_PB(DEH) PB port pull-up resisor control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	PB port pull-up resisor control register. Set PU_PB to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

PU_PC(DFH) PC port pull-up resisor control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							

Reset value	0

Bit number	Bit symbol	Description
7~0	--	PCport pull-up resisor control register. Set PU_PC to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

PU_PD (E2H) PD port pull-up resisor control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	PD port pull-up resisor control register. Set PU_PD to 1 to enable the corresponding pin pull-up resistor, clear the corresponding pin to disable the pull-up resistor, the pull-up resistor is 4.7K.

6.2.3. Direction Registers

TRISA (EAH) PA direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	1	1

Bit number	Bit symbol	Description
1~0	--	PA direction register 0: output 1: input

TRISB(EBH) PB direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PB direction register 0: output 1: input

TRISC(ECH) PC direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PCdirection register 0: output 1: input

TRISD(EDH) PD direction register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

Bit number	Bit symbol	Description
7~0	--	PD direction register 0: output 1: input

6.2.4. Large Current Sink

COM_IO_SEL (EEH) COM large sink current selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	COM port select configure register, corresponding PB port. 1: select COM port mode; 0: select IO port mode.

6.2.5. Open Drain Output Enable Register

ODRAIN_EN (EFH) PA open drain enable register

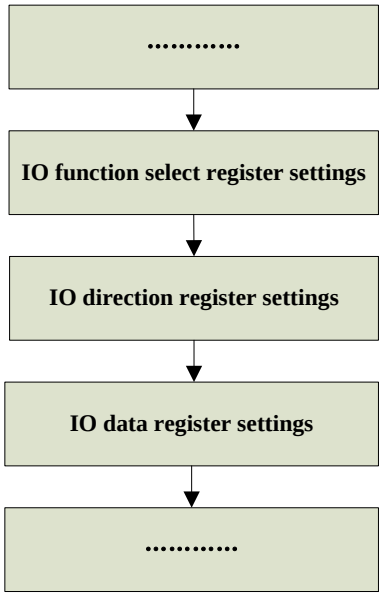
Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description

1~0	--	PA0/1 port open drain output enable register 1: open drain output 0: CMOS output

6.3. GPIO Configuration Process

When setting the port to GPIO, the following three sets of registers need to be set accordingly.



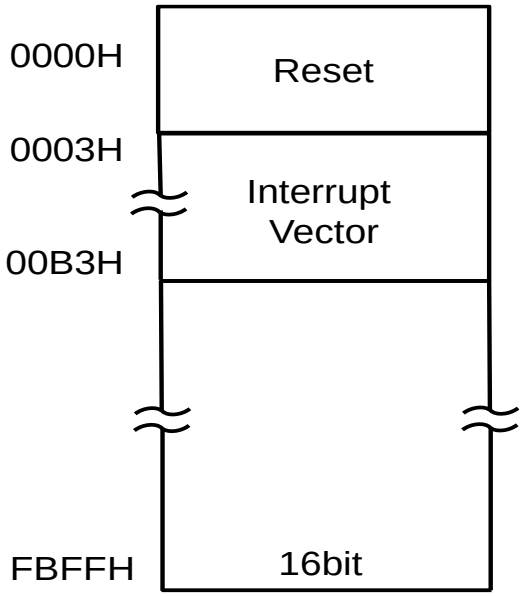
IO configuration flow chart

Notes:The default source current drive capability of the IO port is typically 17mA, the sink current drive capability typically 60mA @5V 0.9VCC. When using IO to drive the LED/digital tube, you need to pay attention to the Ifp current of the LED lamp. It is recommended to add a current limiting resistor to limit the IO drive peak current to the LED/digital tube Ifp current. If you want to save the resistance due to cost factors, it is recommended to use our unique LED serial dot matrix module to drive the LED/ digital tube.

7. Interrupt

7.1. Interrupt Sources And Entry Address

Interrupt source	Condition	Sign	Enable control	Priority control	Interrupt vector	Query priority	Interrupt number	Flag removal method	wake up low power
INT0	External interrupt 0 condition is met	IE0	IEN0[0]	IPL0[0]	0x0003	1	0	User must clear	Yes
Timer0	Timer0 overflow	TF0	IEN0[1]	IPL0[1]	0x000B	2	1	User must clear	No
INT1	External interrupt 1 condition is met	IE1	IEN0[2]	IPL0[2]	0x0013	3	2	User must clear	Yes
Timer1	Timer1 overflow	TF1	IEN0[3]	IPL0[3]	0x001B	4	3	User must clear	No
INT2	External interrupt 2 condition is met	IE2	IEN1[2]	IPL1[2]	0x004B	5	9	User must clear	Yes
IIC	receiveor send completed	IE3	IEN1[3]	IPL1[3]	0x0053	6	10	User must clear	Yes
ADC	ADC conversion completed	IE4	IEN1[4]	IPL1[4]	0x005B	7	11	User must clear	No
LED	Scan complete	IE6	IEN1[6]	IPL1[6]	0X006B	9	13	User must clear	No
WDT/Timer2	WDT/Timer2 overflow	IE7	IEN1[7]	IPL1[7]	0x0073	10	14	User must clear	Yes
LVDT	Voltage conditions meet	IE8	IEN2[0]	IPL2[0]	0x007B	11	15	User must clear	No
UART0	receiveor send completed	IE9	IEN2[1]	IPL2[1]	0x0083	12	16	User must clear	No
UART1	receiveor send completed	IE10	IEN2[2]	IPL2[2]	0x008B	13	17	User must clear	No



When the chip generates a reset signal, the program starts from the 0x0000 address. When an interrupt signal occurs, the program will jump to the interrupt vector program address to execute the interrupt service routine.

7.2. Interrupt Function

7.2.1. Interrupt Response

When an interrupt request, CPU according to the interrupt vectors determine the type of interrupt service routine (ISR) to run. CPU complete execution ISR, unless a higher priority interrupt source applying for a break. After each ISR has RETI (return from interrupt) instruction. After RETI instruction, CPU continues to execute the program before the interrupt did not happen.

ISR can only be a higher priority interrupt request interrupt. That is, the low-priority ISR can be interrupted by a high-priority interrupt request. High priority interrupt ISR can be interrupted by a power down interrupt.

BF7512CMXX-SJLX response interrupt request until the current instruction finished. If the RETI instruction is being executed or read IP, IE, EIP, EIE register, after an additional instruction then respond the interrupt request.

7.2.2. Interrupt Priority

BF7512CMXX-SJLX have two interrupt priority levels: interrupt level and the default priority. Interrupt level (top, high and low) override the default priority. The priority set to high is the first to respond. When the priority is set to the same level, the response will be queued by default. Power-down interrupt is the only high-level interrupt source if allowed. All interrupt sources can be

set to high priority or low priority.

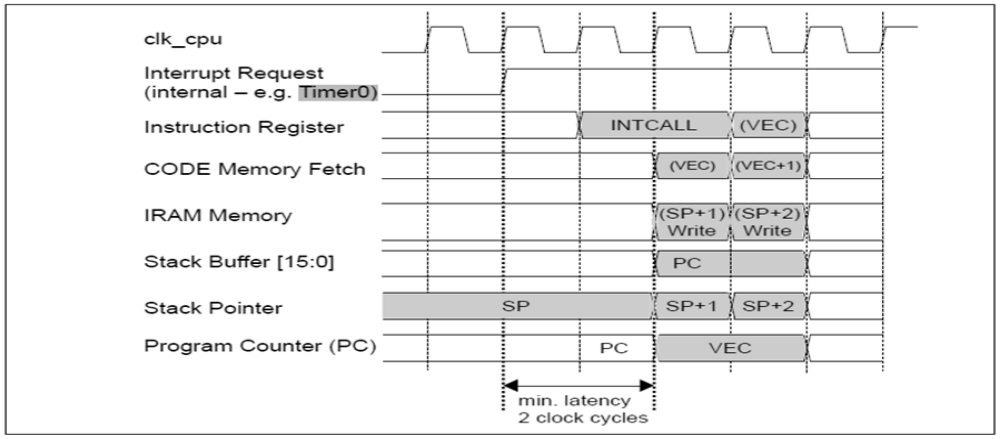
Each interrupt source can be assigned a priority level (high or low), and the default priority. The same level of interrupt sources (such as both high priority) the priority is the default priority decision. Interrupt service routine in progress can only be a high-priority interrupt request interrupt.

7.2.3. Interrupt Sampling

Internal modules such as internal timers and serial ports generate interrupt requests through interrupt flag bits in their respective SFRs. At the end of first clock per instruction cycle (C1), at the rising edge of the external interrupt system clock sampling.

The port external interrupt is active low and can be set to select edge trigger or level trigger via the IT0 bit in TCON SFR. For example, IT0=0, INT_EXT is the trigger for the edge. When the level change from high to low occurs on the INT_EXT pin, the external interrupt flag is set to 1.

To ensure edge-triggered interrupt is detected, the corresponding port must maintain high level for two clocks and maintain low for level two clock.



Interrupt sampling timing diagram

7.2.4. Interrupt Wait

Interrupt response time is determined by current state. Fastest response time is five instruction cycles: one cycle to detect the interrupt request, the other 4 used to execute long call (LCALL) to ISR.

When the system is executing a RETI instruction and is followed by a MUL or DIV instruction, the interrupt waits for the longest time (13 instruction cycles). This 13 instruction cycles are as follows: one cycle to detect the interrupt request, three to complete the RETI, five used to execute DIV or MUL instruction, 4 used to execute long call (LCALL) to ISR. In this case, the response time is 13 clock cycles.

7.3. Interrupt Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description**
0x85	INT_PE_STAT	RW	0x00	WDT/Timer2 interrupt status flag
0x86	INT_POBO_STAT	RW	0x00	LVDT interrupt status flag
0xA8	IEN0	RW	0x00	Interrupt enable register
0xB8	IPL0	RW	0x00	Interrupt priority register 0
0xE1	IRCON2	RW	0x00	Interrupt flag register 2
0xE6	IEN1	RW	0x00	Interrupt enable register 1
0xE7	IEN2	RW	0x00	Interrupt enable register 2
0xF1	IRCON1	RW	0x00	Interrupt flag register 1
0xF2	PERIPH_IO_SEL	RW	0x40	IIC/UART0/INT function control register
0xF4	IPL2	RW	0x00	Interrupt priority register 2
0xF6	IPL1	RW	0x00	Interrupt priority register 1
0xF7	EXT_INT_CON	RW	0x15	External interrupt trigger polarity select register

Interrupt SFR list

7.4. Interrupt SFR Detailed Description

INT_PE_STAT(85H)WDT/Timer2 interrupt status flag

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	INT_WDT_STAT	INT_TIMER2_STAT
R/W	-	-	-	-	-	-	R/W	R/W
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1	INT_WDT_STAT	WDT interrupt status flag. Write 0 to clear this bit, write WDT_CTRL can also clear 0. 1: interrupt effective; 0: invalid interrupt.
0	INT_TIMER2_STAT	TIMER2 interrupt status flag. Write 0 to clear this bit, write TIMER2_CFG can also clear 0. 1: interrupt effective; 0: invalid interrupt.

INT_POBO_STAT (86H) LVDT boost/LVDT buck interrupt status register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	INT_PO_STAT	INT_BO_STAT
R/W	-	-	-	-	-	-	R/W	R/W

Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1	INT_PO_STAT	LVDT boost interrupt status. 1: boost interrupt is valid; 0: boost interrupt is invalid.
0	INT_BO_STAT	LVDT buck interrupt status. 1: buck interrupt is valid; 0: buck interrupt is invalid.

IEN0(A8H) Interrupt enable register

Bit number	7	6	5	4	3	2	1	0
Symbol	EA	-			ET1	EX1	ET0	EX0
R/W	R/W	-			R/W	R/W	R/W	R/W
Reset value	0	-			0	0	0	0

Bit number	Bit symbol	Description
7	EA	EA- Interrupt enable bit. EA=0 block all interrupts (EA takes precedence over the interrupt enable bits of the interrupt source). EA=1, open interrupts. Whether the interrupt request of each interrupt source is allowed or disable, and also needs to be determined by the respective enable bits.
6~4	--	Reserved
3	ET1	ET1-Timer1 overflow interrupt allow bit. ET1=0, disable Timer1 (TF1) to apply for interrupt. ET1=1, allow TF1 to apply for interrupt.
2	EX1	EX1-INT_EXT1 allow bit. EX1=0, disable INT_EXT1 to apply for interrupt. Allow INT_EXT1 to apply for interrupt.
1	ET0	ET0- Timer0 overflow interrupt allow bit. ET0=0, disable Timer1 (TF0) to apply for interrupt. ET0=1, allow Timer1 (TF0) to apply for interrupt.
0	EX0	EX0-INT_EXT0 allow bit. EX0=0, disable INT_EXT0 to apply for interrupt. EX0=1, allow INT_EXT0 to apply for interrupt.

IPL0 (B8H) Interrupt priority register 0

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	PT1	PX2	PT0	PX0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
7~4	–	Reserved
3	PT1	PT1-TF1(Timer1 interrupt) priority selection bit. PT1=0: TF1(Timer1 interrupt) is low priority. PT1=1: TF1(Timer1 interrupt) is high priority.
2	PX2	PX2- INT_EXT1 interrupt priority selection bit. PX2=0: INT_EXT1 is low priority. PX2=1: INT_EXT1 is high priority.
1	PT0	PT0-TF0(Timer0 interrupt) priority selection bit. PT0=0: TF0(Timer0 interrupt) is low priority. PT0=1: TF0(Timer0 interrupt) is high priority.
0	PX0	PX0- INT_EXT0 interrupt priority selection bit. PX0=0: INT_EXT0 is low priority. PX0=1: INT_EXT0 is high priority.

IRCON2 (E1H) Interrupt flag register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	IE10	IE9	IE8
R/W	-	-	-	-	-	R/W	R/W	R/W
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
7~3	--	Reserved
2	IE10	UART1 interrupt flag
1	IE9	UART0 interrupt flag
0	IE8	LVDT interrupt flag

IEN1 (E6H) Interrupt enable register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	EX7	-	EX5	EX4	EX3	EX2	-	-
R/W	R/W	-	R/W	R/W	R/W	R/W	-	-
Reset value	0	-	0	0	0	0	-	-

Bit number	Bit symbol	Description
7	EX7	WDT/Timer2 interrupt enable
6	EX6	LED interrupt enable
5	--	Reserved
4	EX4	ADC interrupt enable
3	EX3	IIC interrupt enable
2	EX2	External interrupt 2 interrupt enable
1~0	-	Reserved

IEN2(E7H) Interrupt enable register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	EX10	EX9	EX8
R/W	-	-	-	-	-	R/W	R/W	R/W
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
7~3	-	Reserved
2	EX10	UART1 interrupt flag
1	EX9	UART0 interrupt flag
0	EX8	LVDT interrupt flag

IRCON1 (F1H) Interrupt flag register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	IE7	-	IE5	IE4	IE3	IE2	-	-
R/W	R/W	-	R/W	R/W	R/W	R/W	-	-
Reset value	0	-	0	0	0	0	-	-

Bit number	Bit symbol	Description
7	IE7	WDT/Timer2 interrupt flag
6	IE6	LED interrupt flag
5	--	Reserved
4	IE4	ADC interrupt flag
3	IE3	IIC interrupt flag
2	IE2	External interrupt 2 interrupt flag
1~0	–	Reserved

PERIPH_IO_SEL (F2H) IIC/UART0/INT function control register

Bit number	7	6	5	4	3
Symbol	-	IIC_AFIL_SEL	IIC_DFIL_SEL	UART0_IO_SEL	
R/W	-	R/W	R/W	R/W	R/W
Reset value	-	1	0	0	0
Bit number	2	1	0	/	
Symbol	INT2_IO_SEL	INT1_IO_SEL	INT0_IO_SEL	/	
R/W	R/W	R/W	R/W		
Reset value	0	0	0		

Bit number	Bit symbol	Description
2	INT2_IO_SEL	INT2 select enable, correspond PD7 1: select INT2 function 0: not select INT2 function

1	INT1_IO_SEL	INT1 select enable, correspond PD6 1: select INT1 function 0: not select INT1 function
0	INT0_IO_SEL	INT0 select enable, correspond PD0 1: select INT0 function 0: not select INT0 function

IPL2 (F4H) Interrupt priority register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	IPL2.2	IPL2.1	IPL2.0
R/W	-	-	-	-	-	R/W	R/W	R/W
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
7~3	--	Reserved
2	IPL2.2	UART1 interrupt priority. 1: high; 0: low.
1	IPL2.1	UART0 interrupt priority. 1: high; 0: low.
0	IPL2.0	LVDT interrupt priority. 1: high; 0: low.

IPL1 (F6H) Interrupt priority register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	IPL1.7	IPL1.6	-	IPL1.4	IPL1.3	IPL1.2	-	-
R/W	R/W	R/W	-	R/W	R/W	R/W	-	-
Reset value	0	0	-	0	0	0	-	-

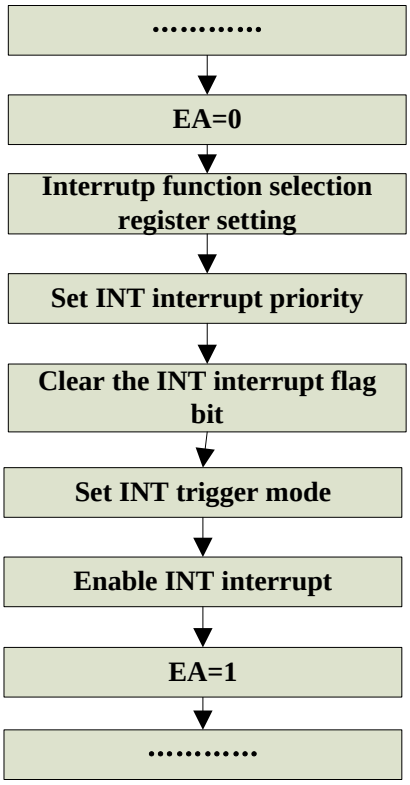
Bit number	Bit symbol	Description
7	IPL1.7	WDT/Timer 2 interrupt priority. 1: high; 0: low.
6	IPL1.6	LED interrupt priority. 1: high; 0: low.
5	--	Reserved
4	IPL1.4	ADC interrupt priority. 1: high; 0: low.
3	IPL1.3	IIC interrupt priority. 1: high; 0: low.
2	IPL1.2	External interrupt priority. 1: high; 0: low.
1~0	--	Reserved

EXT_INT_CON (F7H) External interrupt polarity control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	INT2_POLARITY		INT1_POLARITY		INT0_POLARITY	
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	-	-	0	1	0	1	0	1

Bit number	Bit symbol	Description
5~4	INT2_POLARITY	External interrupt 2 trigger polarity selection: INT2_POLARITY=01: falling edge (Low wake-up in low power mode) INT2_POLARITY=10: rising edge (Low wake-up in high power mode) INT2_POLARITY=00/11: double edge (Low wake-up in low power mode).
3~2	INT1_POLARITY	External interrupt 1 trigger polarity selection: INT1_POLARITY=01: falling edge (Low wake-up in low power mode) INT1_POLARITY=10: rising edge (Low wake-up in high power mode) INT1_POLARITY=00/11: double edge (Low wake-up in low power mode).
1~0	INT0_POLARITY	External interrupt 0 trigger polarity selection: INT0_POLARITY=01: falling edge (Low wake-up in low power mode) INT0_POLARITY=10: rising edge (Low wake-up in high power mode) INT0_POLARITY=00/11: double edge (Low wake-up in low power mode).

7.5. External Interrupt Configuration Process



INT0/1/2 configuration process chart

8. Timer

BF7512CMXX-SJLX contains three Timers(Timer0/Timer1/Timer2). Each Timer contains a 16-bit register that appears as two bytes when accessed: a low byte(TL0 or TL1)and a high byte(TH0 or TH1). Timer2 register is low byte TIMER2_SET_L, high byte TIMER2_SET_H.

Timer features:

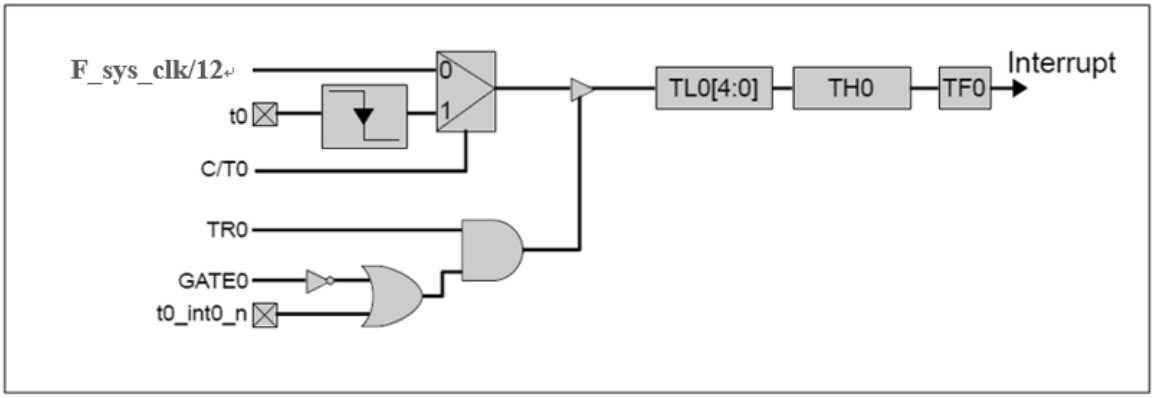
- Timer0 connection system clock, F_sys_clk;
- Timer1 connection system clock, F_sys_clk;
- Timer2 optional internal RC or external crystal clock, frequency 32768Hz/4MHz;
- Timer0 support 8bits automatic reload timing, 16bits manual reload timing function;
- Timer1 support 8bits automatic reload timing, 16bits manual reload timing function;
- Timer2 support 32bits automatic reload timing and manual reload timing, support interrupt wake-up function.

8.1. Timer0 and Timer1

The Timer 0/1 has four operating modes, controlled by TMOD SFR and TCON SFR.
Timer0/1 four modes of operation as follows:

- 13 bit timer (mode 0)
- 16 bit timer (mode 1)
- Automatic overload 8-bit timer (mode 2)
- Two 8-bit timer (Mode 3, only for timer 0)

Mode 0:13 bit timer



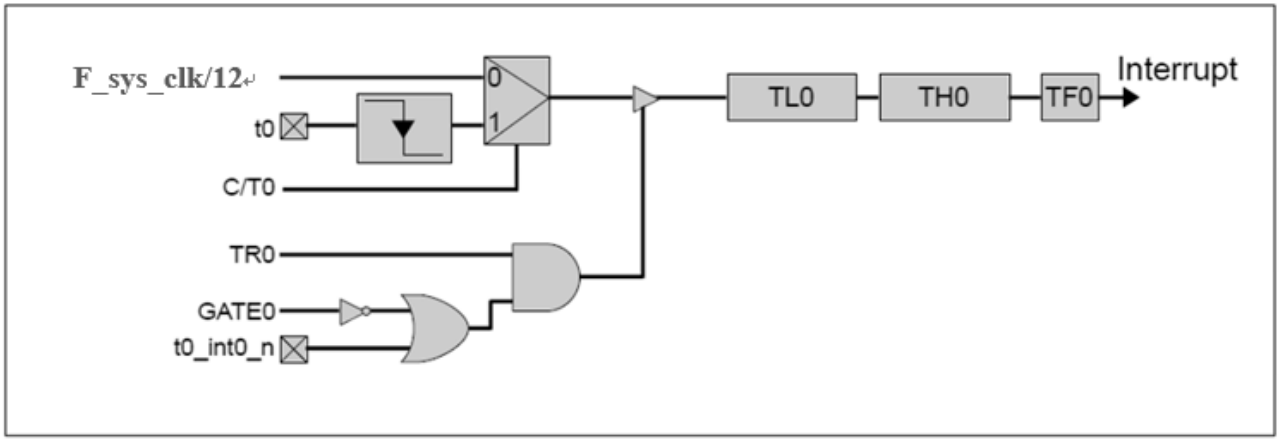
Mode 0 logical structure diagram

In mode 0, timer 0 and timer 1 work the same process, at the picture shows. In mode 0, Timer is 13bit counter, bit0-4 is TL0 (or TL1), other 8 bits is TH0 (or TH1). In TCON register (TR0/TR1) to control timer0/1 start or stop.

The Timer counts the selected clock source(clk/12); When the 13bit counter counts up to all 1,

the counter is cleared to 0(all 0) and TF0(or TF1)is set. In mode 0, TL0 (or TL1) high 3bit is not sure. These 3bit should be masked or ignored when reading the count value. t0/t1, C/T0/CT1 all 0, t0_int0_n/t1_int1_n all 1, count enable is only TR0/1.

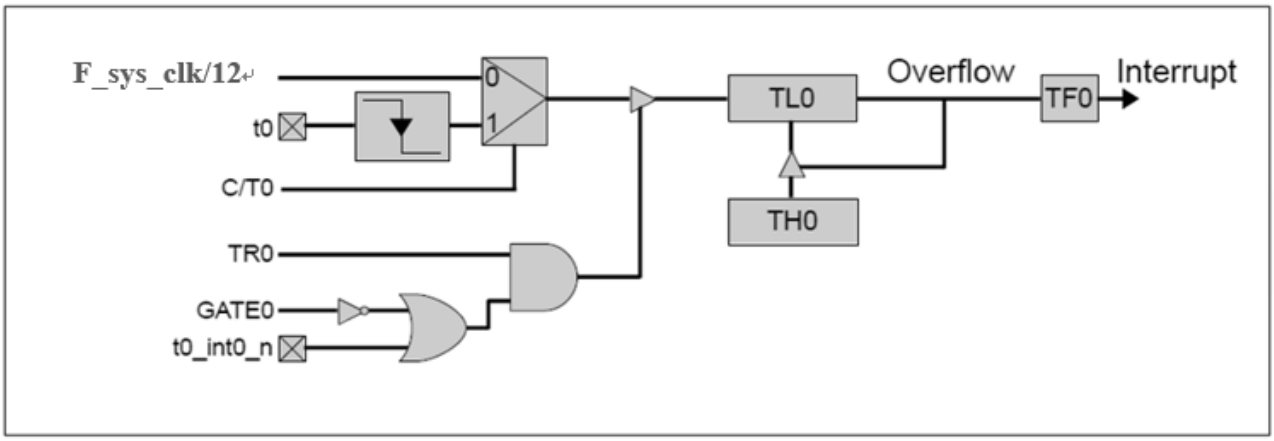
Mode 1: 16 bit timer



Mode 1 logical structure diagram

In mode 1, timer 0 and timer 1 work the same process. At the picture shows, in mode 1, Timer is 16bit counter, all 8 bits of the LSB register (TL0 or TL1) are used. When the counter count is accumulated to 0xFFFF, the counter is cleared to 0. In addition, mode 1 and mode 2 are the same. t0/t1, C/T0/CT1 all 0, t0_int0_n/t1_int1_n all 1, count enable is only determined by TR0/1.

Mode2: manual reload mode 8bit timer



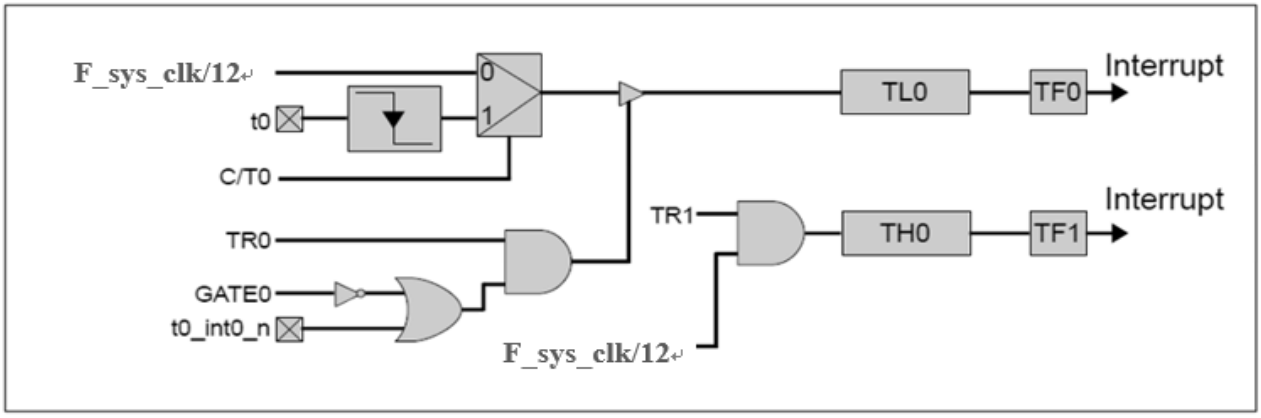
Mode 2 logical structure diagram

The modes of Timer0 and Timer1 are the same. In mode 2, the Timer is an 8bit counter with an automatic reload initial value. This counter is the LSB register (TL0 or TL1). The initial value that needs to be reloaded is saved in the MSB register (TL0 or TL1).

At the picture shows, mode 2 counter control is the same as mode 0 and mode 1. But in mode 2, When the TLn count is accumulated to FFh, the value stored in THn is overloaded to TLn. t0/t1,

C/T0/CT1 all 0, t0_int0_n/t1_int1_n all 1, count enable is only determined by TR0/1.

Mode3: Two 8bit timer



Mode 3 logical structure diagram

In Mode 3, Timer 0 is two 8-bit counters, while Timer 1 stops counting and saves its value. As shown in Figure 5, TL0 is an 8-bit register controlled by the control bits of Timer 0. The counter uses GATE as the enable terminal to control the reception of the INT_EXT signal.

TH0 is a separate 8-bit counter. TH0 can only be used to calculate the clock period (divide by 12). The control bits and flag bits (TR1 and TF1) of timer 1 are used as the control bits and flag bits of TH0.

When Timer 0 works in Mode 3, the use of Timer 1 is limited because Timer 0 uses the control bit (TR1) and interrupt flag bit (TF1) of Timer 1. Timer 1 can still be used to generate the baud rate, and the value of Timer 1 in the TL1 and TH1 registers is still valid.

When timer 0 works in mode 3, timer 1 is controlled by the mode bit of timer 1. To turn on Timer 1, you need to set Timer 1 to Mode 0, 1, or 2. To turn off Timer 1, set Timer 1's mode to 3. Timer 1 can be used as a timer (clock is clk/12), but because TR1 and TF1 are borrowed, overflow interrupt cannot be generated. When Timer 0 operates in Mode 3, the GATE of Timer 1 is valid. t0/t1, C/T0, C/T1 are all 0, t0_int0_n/t1_int1_n are all 1, and the count enable is only determined by TR0/1.

8.1.1. Timer0/1 Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0x88	TCON	RW	0x05	Timer control register
0x89	TMOD	RW	0x00	Timer mode register
0x8A	TL0	RW	0x00	Timer 0 timer lower 8 bits
0x8B	TL1	RW	0x00	Timer 1 timer lower 8 bits
0x8C	TH0	RW	0x00	Timer 0 timer high 8 bits
0x8D	TH1	RW	0x00	Timer 1 timer high 8 bits

Timer0/1 SFR list

8.1.2. Timer0/1 Register Detailed Description

TCON(88H) Timer control register

Bit number	7	6	5	4	3	2	1	0
Symbol	TF1	TR1	TF0	TR0	IE1	-	IE0	-
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	-
Reset value	0	0	0	0	0	-	0	-

Bit number	Bit symbol	Description
7	TF1	Timer1 overflow flag. Set to 1 when Timer1 overflows, or Timer0's TH0 overflows in mode three.
6	TR1	Timer1 start enable, When set to 1, start Timer1, or start Timer0 mode three TH0 count.
5	TF0	Timer0 overflow flag, the hardware set 1 when Timer0 overflows.
4	TR0	Timer0 start enable, when set to 1, start Timer0 count.
3	IE1	External interrupt 1. The hardware set 1, the software is cleared.
2	--	Reserved
1	IE0	External interrupt 0. The hardware set 1, the software is cleared.
0	--	Reserved

TMOD(89H) Timer mode register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	M1[1:0]		-	-	M0[1:0]	
R/W	-	-	R/W		-	-	R/W	
Reset value	-	-	0	0	-	-	0	0

Bit number	Bit symbol	Description
7~6	--	Reserved
5~4	M1[1:0]	M1-Timer1: Bit 1, M1-Timer1: Bit 0. 00=mode0 – 13bit Timer 01=mode1 – 16bit Timer 10=mode2 – manual reload mode 8bit Timer 11=mode3 – 2*8bit Timer
3~2	--	Reserved
1~0	M0[1:0]	M0-Timer0: Bit 1, M0-Timer0: Bit 0. 00=mode0 – 8bit Timer

		01=mode1 – 16 bit Timer 10=mode2 – automatic reload mode 8bit Timer 11=mode3 – 2*8bit Timer

TL0(8AH) Timer 0 Timer is low 8 bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TL0[7:0]							
R/W	R/W							
Reset value	0							

TL1(8BH) Timer 1 counter low 8-bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TL1[7:0]							
R/W	R/W							
Reset value	0							

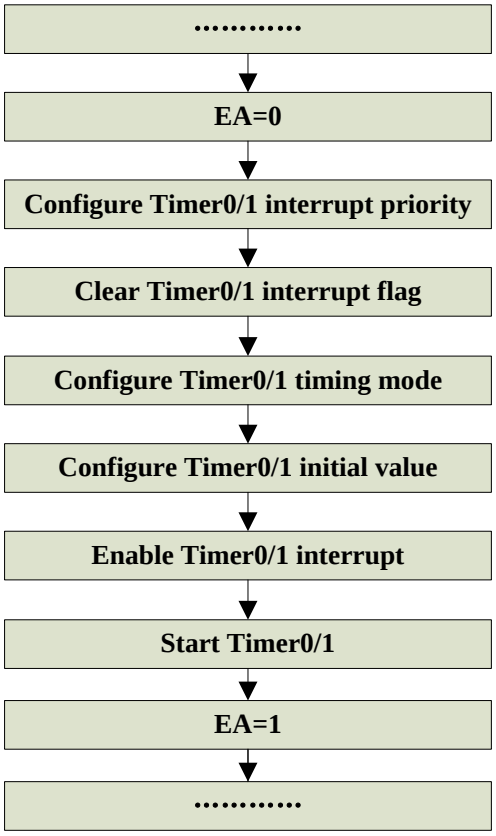
TH0(8CH) Timer 0 Timer is high 8 bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TH0[7:0]							
R/W	R/W							
Reset value	0							

TH1(8DH) Timer 1 Timer is high 8 bit

Bit number	7	6	5	4	3	2	1	0
Symbol	TH1[7:0]							
R/W	R/W							
Reset value	0							

8.1.3. Timer0/1 Configure Process



Timer0/1 configure process

8.2. Timer2

TIMER2 module plays a timing role, the internal structure of the Timer2 module is a 32bit counter. Timed function by counting the input clock, the counting principle of Timer2 is the accumulation counts to the set value. TIMER2's count clock can be selected from the external XTAL clock and the internal RC clock. TIMER2 have two working modes: signal time mode and automatic reload mode, regardless of the mode, the timing is completed and an interruption occurs.

TIMER2_EN configuration Timer2 function enable, TIMER2_RLD configuration automatic reload mode and manual reload mode. Timing time is determined by registers TIMER2_SET_L and TIMER2_SET_H. Timing clock optional internal RC or external crystal clock, frequency 32.768KHz, determined by the clock select register. Timer2 support interrupt wake up low power mode, software clear interrupt flag is required in the interrupt handler.

Timer2 timing duration formula:

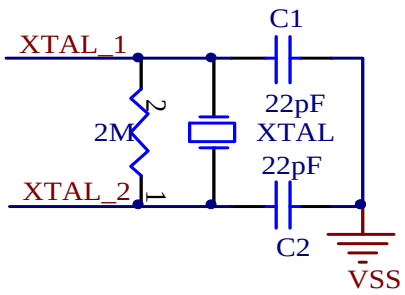
TIMER2_CNT_MOD=0:

$T_{TIMER2}=T_{TIMER2\_CLK}*(\{TIMER2\_SET\_H, TIMER2\_SET\_L\}+1)$

TIMER2_CNT_MOD=1:

$T_{TIMER2}=65536*T_{TIMER2\_CLK}*(\{TIMER2\_SET\_H, TIMER2\_SET\_L\}+1)$

Note: T_{TIMER2_CLK} = 1/32768 (s) or T_{TIMER2_CLK} = 1/4M (s)



External crystal oscillator circuit reference

Notes:

1. **Arbitrary configuration TIMER2_SET_H, TIMER2_SET_L, TIMER2_CFG will clear counter;**
2. **External crystal oscillator circuit is for reference only, actual reference cryatal specifications.**

8.2.1. Timer2 Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0x93	TIMER2_CFG	RW	0x00	TIMER2 configuration register
0x94	TIMER2_SET_H	RW	0x00	TIMER2 count value configuration register, high 8 bits
0x95	TIMER2_SET_L	RW	0x00	TIMER2 count value configuration register, lower 8 bits
0xFE	PD_ANA	RW	0x1F	Module switch control register

Timer2 SFR register list

Secondary bus register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0x1F	DUMMY_REG	RW	0x00	RTC crystal oscillator circuit selection register

8.2.2. Timer2 Register Detailed Description

TIMER2_CFG (93H) TIMER2 CFG register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	TIMER2_CNT_MOD	TIMER2_CLK_SEL	TIMER2_RLD	TIMER2_EN
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
3	TIMER2_CNT_MOD	TIMER2 count step mode select register 1: count step is 65536 clock. 0: count step is 1 clock.
2	TIMER2_CLK_SEL	TIMER2 clock select register 1: select clk_xtal 0: select clk_rc
1	TIMER2_RLD	TIMER2 reload enable control register 1: automatic reload mode 0: manual reload mode
0	TIMER2_EN	TIMER2 count enable register 1: turn on timing; 0: stop timing; In manual reload mode, the hardware automatically clears this register after timing is completed, stop count.

		In manual reload mode, will maintain the enable register after the count is completed. Automatically re-counting from 0, no matter which mode, configuring this register to 1 during counting will start counting from 0.

TIMER2_SET_H(94H) TIMER2 count value configuration register, high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	TIMER2 count configuration register, high 8 bit, Configuring this register during the scan will recount.

TIMER2_SET_L(95H) TIMER2 count value configuration register, low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	TIMER2 count configuration register, low 8 bit, Configuring this register during the scan will recount.

PD_ANA (FEH) Module switch control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	PD_LVDT	PD_BOR	PD_XTAL_32K	-	PD_ADC
R/W	-	-	-	R/W	R/W	R/W	-	R/W
Reset value	-	-	-	1	1	1	-	1

Bit number	Bit symbol	Description
2	PD_XTAL_32K	RTC crystal oscillator circuit (32768Hz/4MHz) control register 1: close; 0: open; default close.

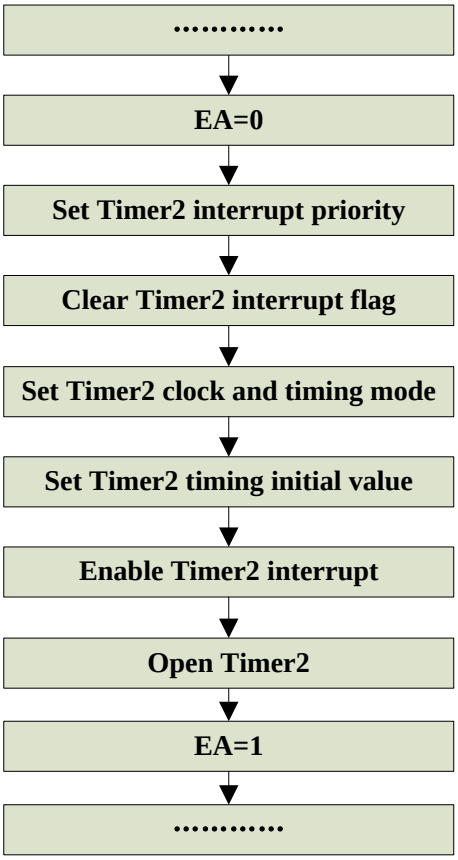
Secondary bus register:

DUMMY_REG(1FH) RTC crystal oscillator circuit selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	XTAL_CLK_SEL
R/W	-	-	-	-	-	-	-	RW
Reset value	-	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
7~1	--	Reserved
0	XTAL_CLK_SEL	1: XTAL4MHz 0: XTAL32768Hz

8.2.3. Timer2 Configure Process



Timer2 configure process table

In the configuration process:

1. First configure the timing set value register TIMER2_SET_H/TIMER2_SET_L and step configuration TIMER2_CNT_MOD;
2. Then automatically reload the enable register TIMER2_RLD according to the configuration, set to 1 if automatic loop count is required, otherwise configure 0;
3. Final configuration timing enable register TIMER2_EN, enable timing configuration TIMER2_EN=1;
4. Stop timing: TIMER2_EN=0.

Notes:

1. TIMER2_EN=0x1 to be placed at the end of all configurations;

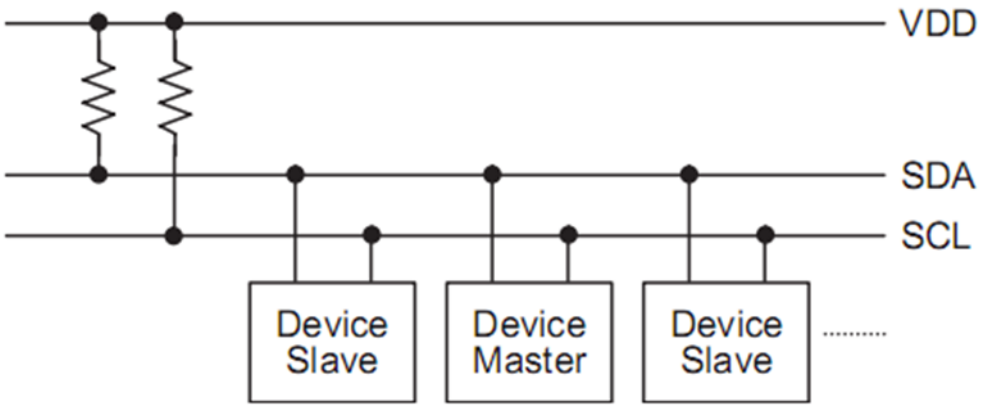
2. During the TIMER2 timing, it is forbidden to change the configuration of Timer2. To modify, you need to stop timing first;

3. For precise timing, in the auto-reload mode, it is not allowed to configure three registers of TIMER2 in interrupt processing.

9. IIC

BF7512CMXX-SJLX supports standard and fast IIC communication, and has the following characteristics:

- Two serial interfaces: serial data line SDA and serial clock line SCL;
- Meet philips's standard communication protocol;
- Transmission rate: 100Kbps, 400Kbps;
- Support for 7-bit address addressing;
- Has the function of extending the clock low level;
- Wake-up core can be interrupted by IIC in low_power mode;
- Detect write conflicts and cache BUF overflow exceptions;



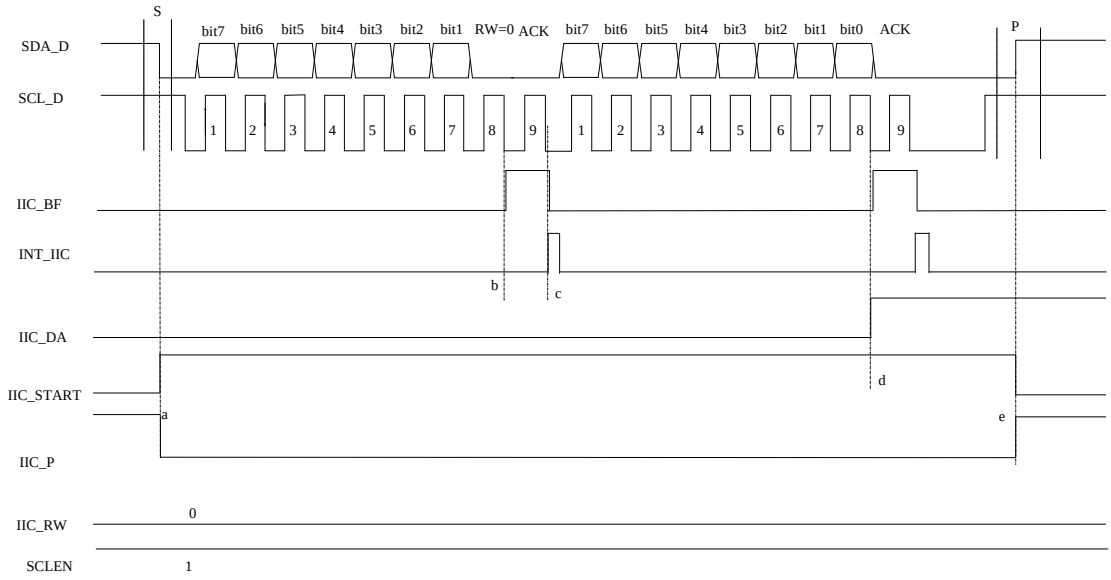
IIC host-slave

The host and slave from the SCL (serial clock) line, SDA (serial data) wire connection, in the communication mode, the PA0/1 is open drain, SCL, SDA must be connected to the pull resistor(suggest 4.7K~10K).

9.1. Communication Timing

BF7512CMXX-SJLX uses hardware slave. When host read /write data, after the slave receives the address, if the address matches, an interrupt is generated and a valid response signal is sent. And an interrupt is generated after the host computer writes the eighth clock of the data, and the host will not generate an interrupt signal when sending the stop signal. IIC timing diagram as follows:

IIC host write timing diagram



IIC write not pull down clock line diagram

As shown in the above figure, the schematic diagram of the clock line is not pulled down during the host write operation. From this, you can see the changes of the IIC bus and some internal signal changes.

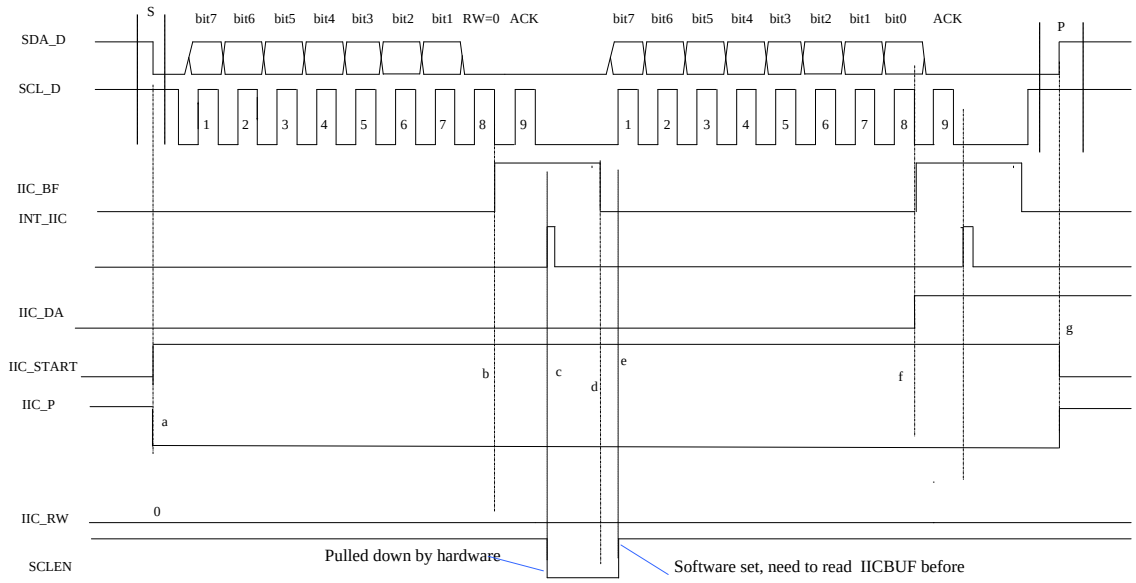
First the host sends a start signal IIC_START, and the slave sets the IIC_START status bit after detecting the IIC_START signal, as shown by the dotted line a in the figure.

Then the host sends the address bytes and write flag bit, and the slave automatically compares with its own address after receiving the address byte. Set IIC_BF after the falling edge of the eighth clock if the address matches, as shown by the dotted line b in the figure. An interrupt signal INT_IIC is generated after the falling edge of the ninth clock, as shown by the dotted line c .The MCU executes interrupt subroutine device needs to read IICBUF. Even if this data is not useful, it needs to be operated.Reading the IICBUF operation will indirectly clear the START_BF. The host continues to send messages. The IIC_BF is also set after the falling edge of the 8th clock of the 2nd byte, and the IIC_AD flag is also set. The currently received byte of the flag is data, and the stop signal has no effect on the IIC_STOP flag. That is, the stop signal IIC_STOP is detected, as shown

by the dotted line d. And the IIC_AD flag will not be cleared; The interrupt is generated after the falling edge of the ninth clock, and the interrupt subroutine requires the same operation. If the host wants to send multiple bytes, it can continue to send. The figure above only shows the case where the host sends a data.

Finally, the host sends a stop signal IIC_STOP after sending all the data, indicating the end of the communication, releasing the IIC bus, and the bus enters the idle state.

IIC host write pull low timing diagram



IIC write low clock line diagram

As shown in the above figure, it is a schematic diagram of pulling down the clock line during the host write operation, from which you can see the changes of the IIC bus and some internal signal changes.

First the host sends a start signal IIC_START, and the slave sets the IIC_START status bit after detecting the IIC_START signal, as shown by the dotted line a.

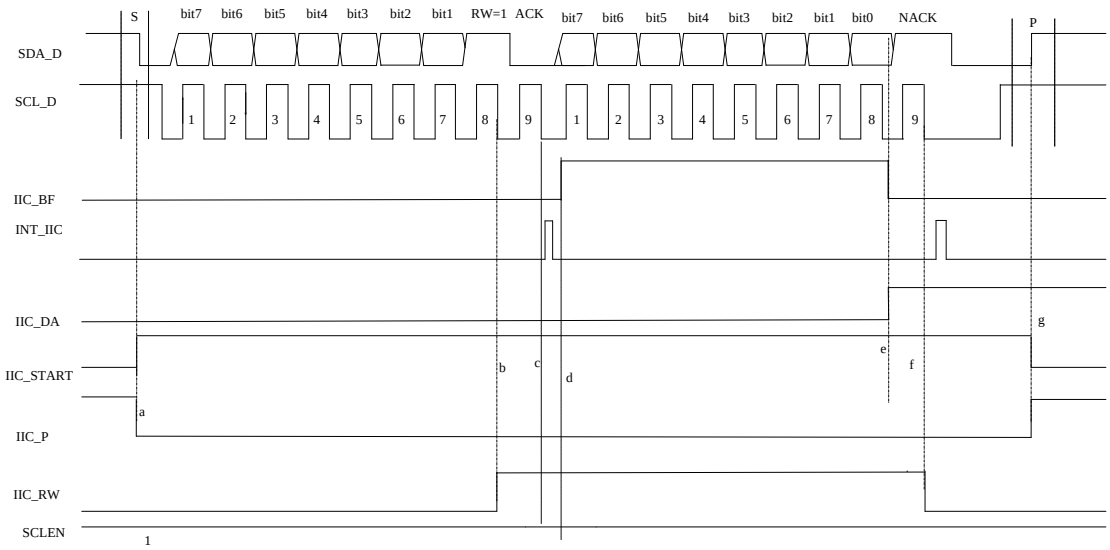
Then the host sends the address bytes and write flag bit, and the slave automatically compares with its own address after receiving the address byte. Set IIC_BF after the falling edge of the eighth clock if the address matches, as shown by the dotted line b.

An interrupt signal INT_IIC is generated after the falling edge of the ninth clock, as shown by the dotted line c. SCLEN will be cleared by hardware. This process is used to process or read data from the slave. Even if this data is not useful, reading IICBUF will cause IIC_BUF to be cleared indirectly, as shown by the dotted line d. Software sets SCLEN to release the clock line.As shown by the dotted line e.

After the master detects that the slave releases the SCL, it continues to send the synchronous clock. The IIC_BF is also set after the falling edge of the 8th clock of the 2nd byte, and the IIC_AD flag is also set. And the IIC_AD flag is also set. The currently received byte of the flag is data, as shown by the dotted line f, and the stop signal has no effect on the IIC_STOP flag. That is, the stop signal IIC_STOP is detected, and the IIC_AD flag will not be cleared; The interrupt is generated after the falling edge of the ninth clock, and the interrupt subroutine requires the same operation. If the host wants to send multiple bytes, it can continue to send. The figure above only shows the case where the host sends a data.

Finally, the host sends a stop signal IIC_STOP after sending all the data, indicating the end of the communication, releasing the IIC bus, and the bus enters the idle state.

IIC host read timing diagram



IIC host does not pull low clock line diagram

As shown in the above figure, the schematic diagram of the clock line is not pulled when the host reads.

First the host sends a start signal IIC_START, marking the beginning of communication. As shown by the dotted line a. The internal circuit detects the IIC_START signal timing and sets the status flag IIC_START.

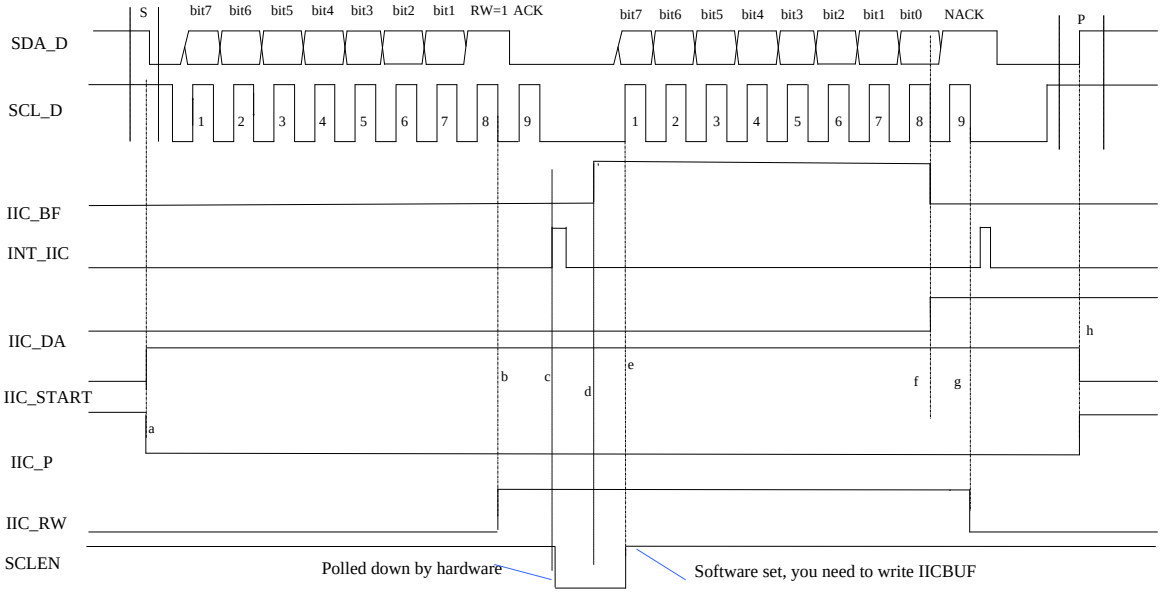
Then the host sends the address bytes and write flag bit, IIC_RW = 1, indicates that the host reads the slave. The slave automatically compares with its own address after receiving the address byte.Status bit IIC_RW set. As shown by the dotted line b. Set IIC_RW after the falling edge of the ninth clock if the address matches.

An interrupt signal INT_IIC is generated after the falling edge of the ninth clock. As shown by the dotted line c. Ballast the data in IICBUFFER to IICBUF, IIC is set to clear, as shown by the dotted line d, and the highest bit is sent to the bus. After the eighth clock, one byte of data is sent, IIC_BF is set to clear; At the same time, the address data flag will also be set, indicating the currently transmitted byte data.

As shown by the dotted line e. An interrupt signal INT_IIC is generated after the falling edge of the ninth clock.If the host needs to read the slave, the host replies with a valid acknowledge bit ACK and continues to communicate. If the data require by the host has been read, the host replies with an invalid response NACK, and then sends a stop signal IIC_STOP to stop the communication. This should be noted in the application. When the NACK is detected, the read/write flag IIC_RW is cleared by hardware. As shown by the dotted line f. If the host sends a NACK, the slave SCLEN will not be automatically pulled low.

Finally, the host sends a stop signal IIC_STOP after reading all the data, indicating the end of the communication. When the IIC_STOP signal is detected the status bit IIC_STOP is set and IIC_START is cleared. Release IIC bus. As shown by the dotted line g. The bus enters the idle state.

IIC host read pull low timing diagram



IIC host read pull low clock line diagram

As shown in the above figure, the schematic diagram of the clock line is not pulled when the host reads.

First the host sends a start signal IIC_START, marking the beginning of communication. As shown by the dotted line a. The internal circuit detects the IIC_START signal timing and sets the status flag IIC_START.

Then the host sends the address byte after the IIC_START signal. IIC_RW = 1, indicates that the host reads the slave. Status bit IIC_RW set. As shown by the dotted line b. Will not be set if the addresses do not match.

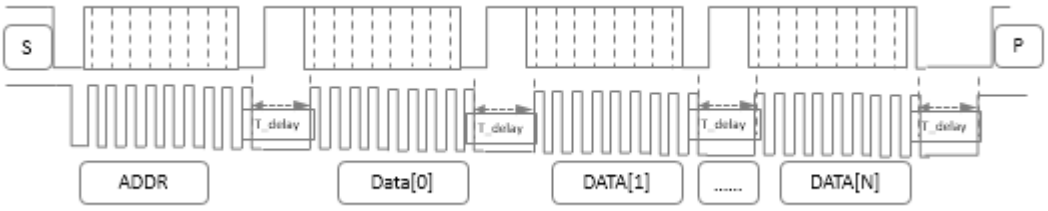
An interrupt signal INT_IIC is generated after the falling edge of the ninth clock. As shown by the dotted line c. SCLEN will also be automatically pulled low by the hardware after the falling edge of the ninth clock. This period is used to process or prepare data from the slave, then write the prepared data to IICBUF, set SCLEN in software, and release the clock line. As shown by the dotted line d. In writing the data to the IIC, the IIC will be set, indicating that the IIC is full at this time. As shown by the dotted line e.

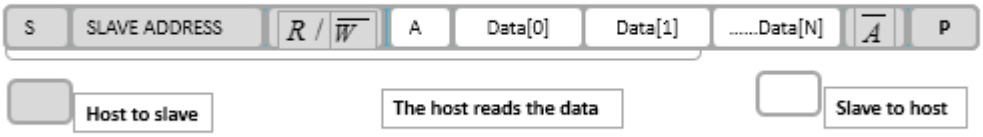
After the master detects that the slave releases the SCL, it continues to send the synchronous clock and read the slave data. After the falling edge of the 8th clock, one byte of data has been sent and IIC_BF cleared; At the same time, the address data flag will also be set, indicating the currently transmitted byte data. As shown by the dotted line f.

An interrupt signal INT_IIC is generated after the falling edge of the ninth clock.If the host needs to continue to read the slave, the host replies with a valid acknowledge bit ACK and continues to communicate; If the data require by the host has been read, the host replies with an invalid response NACK, and then sends a stop signal IIC_STOP to stop the communication. When the NACK is detected, the read/write flag IIC_RW is cleared by hardware. As shown by the dotted line g.

Finally, the host sends a stop signal IIC_STOP after reading all the data, indicating the end of the communication. When the IIC_STOP signal is detected the status bit IIC_STOP is set and IIC_START is cleared. Release IIC bus. As shown by the dotted line h. The bus enters the idle state.

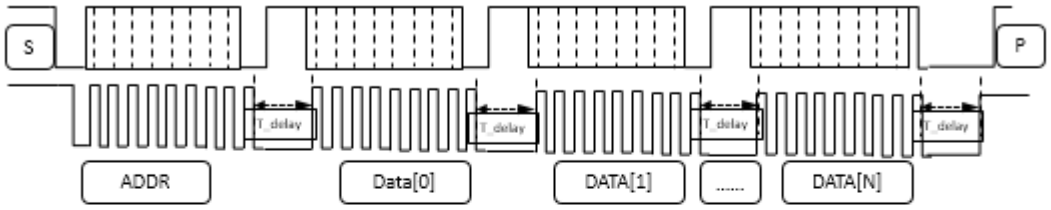
IIC host read data diagram

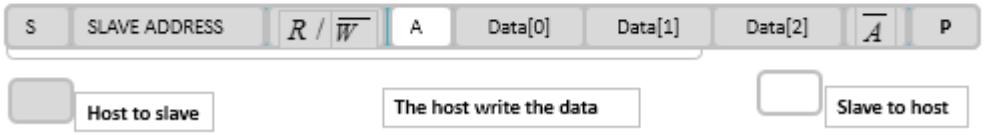




PS: T_delay: Reserve slave interrupt time, generally 60us~300us, if the slave IIC interrupts the service processing time at100us, suggest T_delay>200us.

IIC host write data diagram





PS: T_delay: Reserve slave interrupt time, generally 60us~300us, if the slave IIC interrupts the service processing time at100us, suggest T_delay>200us .

At the eighth clock slave send ack, IIC interrupt occurs at the ninth clock fulling edge.It is recommended that the host delay 60us~300us when the ninth clock fulling edge is sent. Reserve the slave IIC interrupt service data preparation time, and then send the clock signal.

Note: When IIC communication>=100K, it is recommended that the system clock>=6MHz.

9.2. IIC Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description function**
0xE3	IICADD	RW	0x00	IIC address register
0xE4	IICBUF	RW	0x00	IIC transmit receive data register
0xE5	IICCON	RW	0x10	IIC configuration register
0xE8	IICSTAT	RO/RW	0x44	IIC status register
0xE9	IICBUFFER	RW	0x00	IIC transmit receive data buffer register
0xF2	PERIPH_IO_SEL	RW	0x40	IIC filter selection enable

IIC SFR register list

9.3. IIC Register Details

IICADD (E3H) IIC address register

Bit number	7	6	5	4	3	2	1	0
Symbol	IICADD[7:1]							-
R/W	R/W							-
Reset value	0							-

IICBUF (E4H) IIC transmit and receive data register

Bit number	7	6	5	4	3	2	1	0
Symbol	IICBUF							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	IICBUF	IIC transmit receive data buffer

IICCON (E5H) IIC configuration register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	IIC_RST	RD_SCL_EN	WR_SCL_EN	SCLEN	SR	IIC_EN
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	-	-	0	1	0	0	0	0

Bit number	Bit symbol	Description
7~6	--	Reserved
5	IIC_RST	IIC module reset signal 1: IIC module reset operation 0: IIC module works properly

4	RD_SCL_EN	Host read pull low clock line control bit. 1: enable the host to read and pull the low clock line function; 0: disable the host to read and pull the low clock line function.
3	WR_SCL_EN	Host write pull low clock line control bit. 1: enable the host to write and pull the low clock line function; 0: disable the host to write and pull the low clock line function.
2	SCLEN	IIC clock enable bit 1= clock work properly 0= pull down the clock line.
1	SR	IIC conversion rate control bit 1: Conversion rate control is turned off to adapt to the standard speed mode (100K); 0: Conversion rate control is enabled to adapt to fast speed mode (400K)
0	IIC_EN	IIC work enable bit 1= IIC normal work; 0= IIC not work

IICSTAT (E8H) IIC status register

Bit number	7	6	5	4
Symbol	IIC_START	IIC_STOP	IIC_RW	IIC_AD
R/W	R	R	R	R
Reset value	0	1	0	0
Bit number	3	2	1	0
Symbol	IIC_BF	IIC_ACK	IIC_ACK	IIC_RECOV
R/W	R	R	R/W	R/W
Reset value	0	1	0	0

Bit number	Bit symbol	Description
7	IIC_START	Start signal flag 1: boot bit detected; 0: no boot bit detected
6	IIC_STOP	Stop signal flag 1: stop status detected; 0: no stop status detected
5	IIC_RW	Read and write flag. Record the read/write information obtained from the address byte after the last address match. 1: read; 0: write.
4	IIC_AD	Address data flag bit. 1: indicates that the most recently received or sent byte is data;

		0: indicates that the most recently received or sent byte is address.
3	IIC_BF	IICBUF full flag. Received in IIC bus mode: 1: received successfully, buffer is full; 0: received successfully, buffer is empty. Send in IIC bus mode 1: data transmission is in progress(does not include the acknowledge bit and the stop bit), buffer is full; 0: data transmission has been completed(does not include the acknowledge bit and the stop bit), buffer is empty.
2	IIC_ACK	Answer flag 1: invalid response signal; 0: effective response signal.
1	IIC_WCOL	Write conflict flag. 1: when the IIC is transmitting the current data, the new data is attempted to be written to the transmit buffer; new data cannot be written to the buffer. 0: no write conflict
0	IIC_RECOV	Receive overflow flag bit 1: When the previous data received by the IIC has not been taken, new data is received, the new data cannot be received by the buffer. 0: no receive overflow.

IICBUFFER (E9H) IIC transmit and receive data buffer register

Bit number	7	6	5	4	3	2	1	0
Symbol	IICBUFFER							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	IICBUFFER	IIC transmit receive data buffer register. RD_SCL_EN=0, when the host reads the data, the data in the IICBUFFER is send to the slave transmit buffer after 2 clocks after the interrupt is generated, the data sent as a salve. Therefore, the previously prepared IICBUFFER interrupt data is generated before the interruption.

PERIPH_IO_SEL (F2H) IIC/UART0/INT function control register

Bit number	7	6	5	4	3
Symbol	-	IIC_AFIL_SEL	IIC_DFIL_SEL	UART0_IO_SEL	
R/W	-	R/W	R/W	R/W	R/W
Reset value	-	1	0	0	0

Bit number	2	1	0	/
Symbol	INT2_IO_SEL	INT1_IO_SEL	INT0_IO_SEL	/
R/W	R/W	R/W	R/W	
Reset value	0	0	0	

Bit number	Bit symbol	Description
6	IIC_AFIL_SEL	IIC port analog filter selection enable 1: select analog filter function; 0: do not select analog filter function.
5	IIC_DFIL_SEL	IIC port digital filter selection enable. 1: select digital filter function; 0: do not select digital filter function.

IIC_START: Start signal status bit, IIC_START is set when the start signal is detected, Indicating that the bus is busy.

IIC_STOP: Stop signal status bit, IIC_START is set when the start signal is detected, indicating that the bus is idle. When the start signal is detected, the hardware is cleared, indicating that communication begins.

IIC_AD: Address data flag. It indicates whether the byte currently received or sent is an address or data. IIC_AD =0, flag is currently received or sent byte is the address; IIC_AD = 1 flag is currently received or sent byte is the data; Start signal, stop signal, non-response signal have no effect on this status bit. This status bit change occurs on the falling edge of the eighth clock.

IIC_RW: Read and write flag. The flag bit is recorded the read and write information bits obtained from the address is matched. IIC_RW = 1 means the host reads the slave. RW = 0 means the host writes the slave. Start signal, stop signal, non-answer signal (NACK) is cleared IIC_RW. This status bit change occurs on the falling edge of the ninth clock.

IIC_BF: BUFFER full flag. It indicates that the transceiver buffer is currently full or empty. IIC_BF=0 indicates that the buffer does not receive data and the buffer is empty; IIC_BF=1 indicates that the buffer receive data and the buffer is full. This status bit can only be set and cleared indirectly, not directly.

Address matching and IIC_RW=0, IIC_BF will be set after the falling edge of the eighth clock, indicating that the IICBUF has received the data. The IICBUF should be read during the execution of the interrupt routine, and the read IICBUF will indirectly clear the BF flag. If the host does not read IICBUF and the host continues to send data, a receive overflow will occur. Although the slave still receives the host to send data and is ballasted to the IICBUF.

IIC_RW=1 indicates the operation of the master to read the slave, the slave operation needs to write data to the IICBUF, and the slave writes IICBUF operation to set the IICBUF. The software then sets SCLEN to release the clock line; The host The host sends the synchronous clock. After the 8th clock is passed, the IICBUF is cleared by hardware after the data in the IICBUF is sent out.

IIC_ACK: Answer flag. Regardless of whether the host is a read or write operation, the slave samples the data line from the rising edge of the ninth clock and records the response information.

The acknowledge bits are divided into a valid acknowledgment ACK and a non-valid acknowledgement bit NACK. That is to say, the rising edge of the ninth clock samples the data to 0, indicating that the ACK is valid, and the IIC_ACK is cleared. If data 1 is sampled, NACK is set, indicating non-response. After the non-acknowledgment signal, the host will send a stop signal to announce the end of the communication. The start signal will clear this status bit.

IIC_WCOL: Write conflict flag. IICBUF only when IIC_RW=1, RD_SCL_EN=1 and SCLEN=0 can be written by the CPU. Any other attempt to write to IICBUF is forbidden. If the above conditions are not met, the write IICBUF operation occurs. Then the data will not be written to IICBUF, and the conflict flag IIC_WCOL will be set. This flag needs to be cleared by software.

IIC_RECOV: Receive overflow flag.In the case of IICBUF full, that is, in the case of data in the IICBUF. If IIC received new data, it will receive overflow and IIC RECOV will set.At the same time, the data in the IICBUF will not be updated, and the newly received data will be lost. This status bit also requires software to clear, otherwise it will affect the subsequent communication. This kind of situation will only appear in IICRW=0. BF=1, and the CPU will appear when it does not read IICBUF.

9.4. IICCON Registers

IIC control register, used to control communication work.

Bit	Name	R/W	Reset value	Description
7: 6	--	R	0	Reserved
5	IIC_RST	RW	0	IIC module reset control enable bit. 0: enable IIC module reset function ; 1: disable IIC module reset function .
4	RD_SCL_EN	RW	1	Host read pull low clock line control bit. 0: enable host read low line clock line function; 1: disable host read low line clock line function.
3	WR_SCL_EN	RW	0	Host write pull low clock line control bit. 0: enable host write low line clock line function; 1: disable host write low line clock line function.
2	SCLEN	RW	0	IIC clock enable bit. 1 = the clock is working properly; 0 = pull low clock line(IIC_EN=1 is valid).
1	SR	RW	0	IIC conversion rate control bit. 1 = conversion rate control is turned off to accommodate standard speed mode(100K) ;

				0 = conversion rate control is turned off to accommodate fast speed mode(400K) ;
0	IIC_EN	R	0	IIC module enable signal. 1 = IIC module work;

The role is describle in detail below:

IIC_EN is module enable signal, when IIC_EN=1, the circuit works.

SR is the conversion rate control bit, SR=1 conversion ratecontrol off, port adapted to 100Kbps communication.

SCLEN is clock enable control bit, although the slave cannot generate the communication clock, the slave can extend the low time of the clock according to the protocol. SCLEN=0, clock line is locked at low level; SCLEN=1, release clock line. The premise of extending the low level of the clock is IIC_EN=1, otherwise the internal circuit will not have any effect on the IIC bus. SCLEN is often used to extend low time and make the host enter the wait state, so that the slave has enough time to process the data.

WR_SCL_EN is write low line control bit. When it is 1 to enable the interrupt to pull down the clock line, when it is 0, it does not enable the interrupt to pull down the clock line.

IIC_RW=0, according to the communication rate of the host and the time of processing the interrupt, it is determinded whether to lower the clock line, that is, configure the WR_SCL_EN bit.

When the CPU can process the interrupt and exit the interrupt within 8 IIC clocks. WR_SCL_EN=0 disable pull down the clock clock line function. At this time, the hardware will not automatically pull down the clock line when the interrupt arrives. When the CPU cannot process the interrupt and exit in the 8 IIC clocks, WR_SCL_EN=1 enables the clock line to be pulled down. At this point, the hardware automatically pulls down the clock line when the interrupt arrives, forcing the host to enter the wait state. When the data written to the IIC is read by the CPU, the software sets SCLEN.

RD_SCL_EN is read low line control bit. When it is 1 to enable the interrupt to pull down the clock line, when it is 0, it does not enable the interrupt to pull down the clock line.

RD_SCL_EN=1, when the slave receives the address byte or sends one byte and the host sends, SCLEN wll be automatically pulled low by hardware, forcing the host to the enter the wait state. The release the IIC clock from the slave, the following two operations arerequired: first write the data to be sent to the IIC, set the software in IICBUF in SCLEN. The purpose of this design is to ensure that the data to be sent has been written in the IICBUF before the SCL is pulled high.

RD_SCL_EN=0, when the slave receives the address byte or sends one byte and the host sends an ACK, the slave immediately polls the data prepared in the IICBUFFER register to the transmit buffer register and then to the data line. Therefore, in order to ensure that data transmitted each time is correct, IICBUFFER prepares the next data to be sent in the interrupt service routine. The data received by the host is the last interrupted data, and the first time the data is received is ready for initialization.

Note: When you need to pull down the clock line, that is, WR_SCL_EN/RD_SCL_EN=1.

Software should turn off the clock line until the last Byte data is sent and received. That is, WR_SCL_EN/RD_SCL_EN=0, the software should turn on the write low pull clock line before sending and receiving the last Byte data. This kind of operation can be self-regulated according to whether the host is software or hardware.

IIC_RST is IIC module control enable bit, enable the IIC module reset function for 1 and disable the IIC module reset function when 0. Pay attention to configuration 1 reset IIC module all DFF triggers. The reset terminal of IIC_RST is global reset, and the other reset terminal are iic_rst_n. All iic_rst writes 0 first, then operate other register configurations.

9.5. IICBUF Register

IIC read and write buffer register, used to control communication work.

Bit	Name	R/W	Reset value	Description
7: 0	IICBUF	RW	0	IIC data receiving and transmitting buffer

The specific application process is as follows:

In the send state, after the data is ballasted into the IICBUF, under the synchronous clock of the host. The data is sequentially shifted and sent out, the high position is in front. After 8 clocks, one byte is sent.

In the receive state, after the host's 8 clocks have passed, the data is written to the BUF. After the 9th clock, an interrupt is generated, telling the CPU to read the data in the IICBUF.

Writing data to IICBUF is conditional, when RD_SCL_EN=1, only IIC_RW=1, and SCLEN=0 can write data into IICBUF; Otherwise, the operation of writing IICBUF is prohibited. That is to say, if the condition is not satisfied, the operation of writing IICBUF cannot be successful, and the data cannot be written. IICBUF data will not change, but will also cause write confilicts.

For example: IICBUF already has been 55h. In case the condition of writing IICBUF is not satisfied, we want to write data 00h into IICBUF. The result is that the data in IICBUF is still 55h, and the write conflict flag IIC_WCOL is set to tell the user that the operation is abnormal.

When RD_SCL_EN=0, the data to be the slave is the value of the ballast IICBUFFER register when the interrupt signal is generated.

9.6. IICBUFFER Register

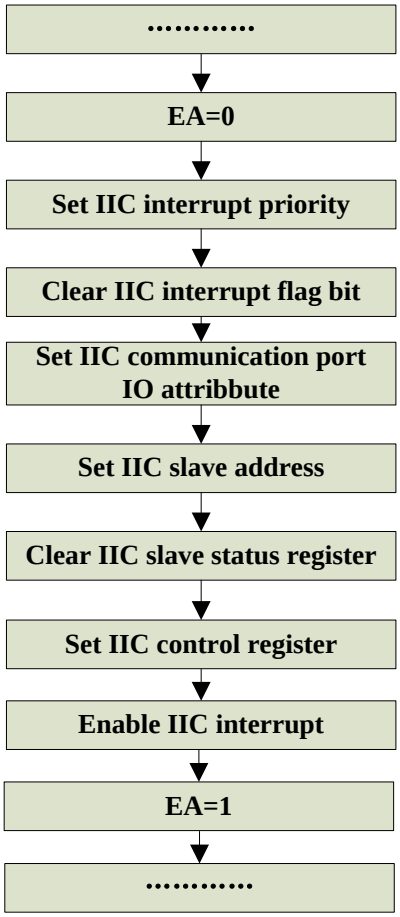
IIC data transmission buffer

Bit	Name	R/W	Reset value	Description
7: 0	IICBUFFER	RW	0	IIC data transmission buffer

The specific application process is as follows:

When RD_SCL_EN=0, and the host reads the data, the data in the IICBUFFER is sent to the slave transmit buffer register after the two clks after the interrupt is generated, and the data is sent as slave. Therefore, the data in the IICBUFFER should be prepared before the interrupt is generated. Generally, it is ready in the service routine. Device address generation interrupts send data to prepare for initialization.

9.7. IIC Configure Process



IIC configure process

Notes: IIC bus pull-up resistor 4.7K~10K, ground filter capacitor 10pF~ 100pF close to the lead chip.

10. UART

There are 2 UART modules in the BF7512CMXX-SJLX series, UART0 interface feature:

- Support full-duplex, half-duplex serial
- Independent dual buffer receiver and single buffer transmitter
- Programmed baud rate (10bit analog-to digital divider)
- Interrupt-driven or polling operation:
  - send completed
  - receiving full
  - receive overflow, parity error, frame error
- Supports hardware parity production and check
- Programmable 8bit or 9bit character length
- STOP bit 1 or 2 can be selected
- Supports multiprocessor mode
- UART0 supports 3 IO port mapping.

10.1. UART0 Function Description

10.1.1. Baud Rate Generation

Baud rate generation modules: Baud_Mod= {UART0_BDH[1:0], UART0_BDL}.

Baud rate calculation formula: Baud_Mod=0, does not generate baud rate clock. When Baud_Mod=1~1023, UART0 baud rate = BUSCLK/ (16x Baud_Mod).

BUSCLK uses the divided clock of the system clock source, fixed to 24M. Each time the baud rate register is configured, the internal counter is cleared and the baud rate signal is regenerated. Communication requires the transmitter and receiver to use the same baud rate. Baud rate deviation range allowed by communication: 8/(11*16)=4.5%.

10.1.2. Transmitter function

Sending data flow: Start sending by writing the value of UART0_BUF, set the sending interrupt after sending the stop bit, and clear the interrupt flag by software, waiting for the next write. Transmitter output pin (TXD) idle state, default is logic high state. The entire transmission process must be carried out when the module is enabled. By writing data to the data register (UART0_BUF), the data will be directly saved to the transmit data buffer and the transmission process will be started. In the subsequent complete transmission process, it is not allowed to write the data registers UART0_BUF and T8 until the stop bit is transmitted. , the transmission interrupt flag is set, and then the UART0_BUF can be written again to restart a new transmission.

The central element of the serial transmitter is a transmit shift register of length 10/11/12 bits (depending on the setting in the data_mode control bits). Assuming data_mode=0, select the normal 8-bit data mode. In the 8-bit data mode, there are 1 start bit, 8 data bits and 1/2 stop bits in the shift register, and the transmission and reception are all small endian mode (LSB first).

10.1.3. Receiver Function

The receiver is enabled by setting the receive enable bit in UART0_CON1. The entire receiving process must be performed when the module is enabled.

Receiving data flow: receive data at any time with the reception enable enabled. After receiving the stop bit, set the middle segment and the software clears the interrupt flag.

Currently acceptly data will detect wit, detect receive overflow, frame error, parity error three errors. Software clearance mark required. It is recommended to read the status flag and read the data buf after receiving the receive interrupt. Finally, the received data status flags are cleared （UART_STATE[3:0]).

Data character is started by logic 0, 8 or 9 data bit (LSB send first) and stop bits (1bit) of logic 1. After receiving the stop bit to the shifter, if the receive data shift register is not full (rx_full_if=0), data characters are transferred to the receive data register, setting the receive data register full(rx_full_if=1)status flag. If the rx_full_if of the receive data register is already set at this time,

set the overflow (rx_overflow_if) status flag, the new data will be lost. Because the receiver is double buffered, after setting rx_full_if, program has a full character time for reading before reading the data of the receive data buffer to avoid receiver overflow.

When the program detects that the receive data register is full (rx_full_if=1), it acquires data from the receive data register by reading UART0_BUF.

10.1.4. Receiver sampling method

The receiver uses a 16x baud clock for sampling. The receiver searches for falling edges on the RXD serial data input pin by extracting logic level samples at 16 times the baud rate. A falling edge is defined as a logic 0 sample after 3 consecutive logic 1 samples. The 16x baud clock is used to divide the bit time into 16 segments, labeled RT1 to RT16.

The receiver then samples each bit time at RT8, RT9 and RT10, including the start and stop bits, to determine the logic level of that bit. The logic level is the logic level of the vast majority of samples taken during the bit time. When the falling edge is located, make sure that this is the real start bit, not noise, by going to a logic level of 0, if at least two of the three samples are 0, the receiver assumes it is in sync with the receiver character and starts Shift to receive the following data, if the above conditions are not met, exit the state machine and return to the state of waiting for the falling edge. The falling edge detection logic constantly looks for a falling edge, and if an edge is detected, the sample clock resynchronizes the bit time.

10.1.5. Multiprocessor Mode

Multiprocessor mode, only works in 9-bit mode, when the received R8 bit=1, the receive interrupt is set, otherwise it is not set. The role of this mechanism is to eliminate the software overhead of handing unimportant information for diffrernt receivers.

In this application system, all receivers estimate the address character (ninth bit=1) of each message. Once it is determined that the information is intended for different receivers, subsequent data characters (ninth bit=0) are not received.

Configuration process: configuring receive enable, configuring multiprocessor mode, received address data (ninth bit=1), receive and generate an interrupt. The application confirms that the addresses match, and the match configures to turn off the multiprocessor mode. All subsequent data (ninth bit=0) can be received and interrupted until the next time the address data is received, the address does not match, then the multiprocessor mode is turned on. Then all subsequent data is not received until the next address data, and then cyclically applied.

10.2. UART0 Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Function description**

0xBD	UART0_BDL	RW	0x00	UART0 baudrate control register
0xBE	UART0_CON1	RW	0x00	UART0 mode control register 1
0xBF	UART0_CON2	RW	0x0C	UART0 mode control register 2
0xC0	UART0_STATE	RW	0x00	UART0 status flag register
0xC1	UART0_BUF	RW	0xFF	UART0 data register
0xF2	PERIPH_IO_SEL	RW	0x40	IIC/UART0/INT function control register

UART0 SFR register list

10.3. UART0 Register Details

UART0_BDL (BDH) UART0 Baudrate control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Baud rate control register. Baud rate modules divisor register lower 8 bits, Baud_Mod ={UART0_BDH[1:0], UART0_BDL}, Baud_Mod =0, does not generate baud rate clock. Baud_Mod =1~1023, bandrate = BUSCLK/(16x Baud_Mod).

UART0_CON1 (BEH) UART0 control register 1

Bit number	7	6	5	4
Symbol	-	uart0_enable	receive_enable	multi_mode
R/W	-	R/W	R/W	R/W
Reset value	-	0	0	0
Bit number	3	2	1	0
Symbol	stop_mode	data_mode	parity_en	parity_sel
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
6	uart0_enable	Module enable. 1: module enable; 0: module off.
5	receive_enable	Receiver enable. 1: receiver open; 0: receiver off.
4	multi_mode	Multiprocessor communication mode. 1: mode enable; 0: mode disable.
3	stop_mode	stop bit width selection.

		1: 2 bit; 0: 1 bit.
2	data_mode	Data mode select. 1: 9bit mode; 0: 8bit mode.
1	parity_en	Parity enable. 1: parity enable; 0: parity disable.
0	parity_sel	Parity select. 1: odd parity; 0: even parity.

UART0_CON2 (BFH) UART0 control register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	tx_empty_ie	rx_full_ie	UART0_BDH	
R/W	-	-	-	-	R/W		R/W	
Reset value	-	-	-	-	1	1	0	0

Bit number	Bit symbol	Description
3	tx_empty_ie	Send interrupt enable. 1: interrupt enable; 0: interrupt disable(used in polling mode)
2	rx_full_ie	Received interrupt enable 1: interrupt enable; 0: interrupt disable(used in polling mode)
1~0	UART0_BDH	Baud rate modulus divisor register high 2bit.

UART0_STATE (C0H) UART0 status flag register

Bit number	7	6	5	4
Symbol	-	r8	t8	tx_empty_if
R/W	-	R	R	R/W
Reset value	-	0	0	0
Bit number	3	2	1	0
Symbol	frx_full_i	rx_overflow_if	frame_err_if	parity_err_if
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
6	r8	Receiver’s ninth data, read only.
5	t8	Transmitter’s ninth data, read only when parity is enabled.
4	tx_empty_if	Send interrupt flag. 1: send buffer is empty; 0: send buffer is full, software write 0 clear 0, write 1 invalid.
3	frx_full_i	Receive interrupt flag,.

		1: receive buffer is full; 0: receive buffer is empty, software write 0 clear 0, write 1 invalid.
2	rx_overflow_if	Receive overflow flag; 1: receive overflow(lost new data); 0: no overflow, software write 0 clear 0, write 1 invalid.
1	frame_err_if	Framing error flag. 1: framing error flag; 0: no framing error flag, software write 0 clear 0, write 1 invalid.
0	parity_err_if	Parity error flag. 1: receiver parity error; 0: parity is correct, software write 0 clear 0, write 1 invalid.

UART0_BUF (C1H) UART0 data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	1	1	1	1	1	1	1	1

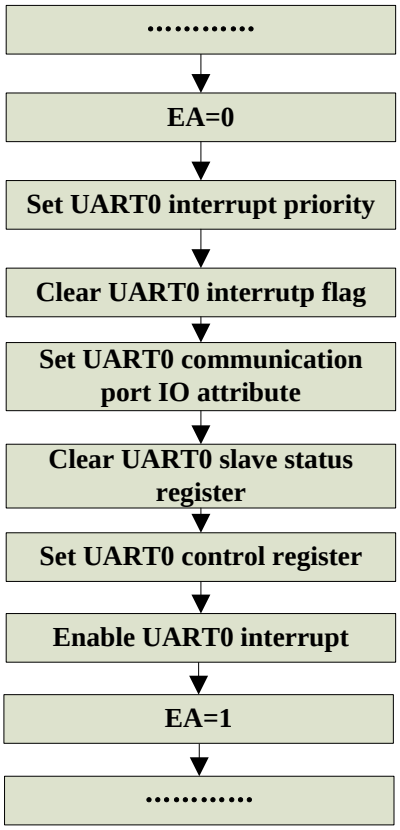
Bit number	Bit symbol	Description
7~0	--	Data register Read returns read-only receive data buffer contents, write into write-only send data buffer.

PERIPH_IO_SEL (F2H) IIC/UART0/INT function control register

Bit number	7	6	5	4	3
Symbol	-	IIC_AFIL_SEL	IIC_DFIL_SEL	UART0_IO_SEL	
R/W	-	R/W	R/W	R/W	R/W
Reset value	-	1	0	0	0
Bit number	2	1	0	/	
Symbol	INT2_IO_SEL	INT1_IO_SEL	INT0_IO_SEL	/	
R/W	R/W	R/W	R/W		
Reset value	0	0	0		

Bit number	Bit symbol	Description
4~3	UART0_IO_SEL	UART0 select enable. 00: select UART0(RXD0_A/TXD0_A) function 01: select UART0(RXD0_B/TXD0_B) function 1x: select UART0(RXD0_C/TXD0_C) function

10.4. UART0 Configuration Process



UART0 initial configuration process

Suggested application process:

1. Configuration module enable, receive enable, mode select: UART0_CON1;
2. Configure baudrate, open interrupt enable: UART0_BDL, UART0_CON2;
3. Write UART0_BF starts to send data, after detecting the transmission interrupt, clear the interrupt flag tx_empty_if;
4. Receive interrupt detected, first read status UART0_STATE. Then read R8 and UART0_BUF, finally clear the receive status flag (UART0_STAT[3:0] = B0000). One receiving process is completed, waiting for the next receiving interrupt;
5. If the configuration interrupt is not enabled, the program executes the UART0 function. Also read the status flag first, then read R8 and UART0_BUF, and finally clear the status flag.
6. Interrupt flag clear operation. In full-duplex operation, clear flag bit operation requires a vaild interrupt bit to be written 0, and other interrupt bits to be written as 1 (write 1 as invalid operation), otherwise it is easy to operate incorrectly. For example: when the send interrupt is vaild, you need tp write UART0_STATE = 0x0F; (configuration UART0_STATE[0:3] = 0x0F, R8 write is invalid, t8 needs to configure vaild transmit data when it is in 9 bit mode and does not have parity).

7. 8 bit mode: Parity enable is valid.

9 bit mode: When the parity bit is enabled, when the parity bit calculated by the ninth bit is not enable, the ninth bit is the T8 written in. Only send interrupts and receive interrupts.The error flag only marks the error detection of the current data, and only the corresponding bit writes 0 clear, do not jump out of error interrupt. The transmit interrupt is set after the stop bit is sent, and the software clears it. The receive interrupt is set after the stop bit is sent, and the software clears it.

Multiprocessor mode: Only works in 9 bit mode, received R8 = 1, receive interrupt is set, otherwise it is not set. When using multiprocessor mode, configuring receive enable and multiprocessor mode. Receive address data (the ninth bit=1) and generate an interrupt, confirm that the address matches. Matching configures the multiprocessor mode to be turned off, and all subsequent data(the ninth bit = 0)can be interrupted by the received interrupt, until the next time data is received. If the address do not match, the multiprocessor mode is turned on, and all subsequent data is not received until the next address data.

Hardware response: Send data is opened by the value written to UART0_BUF. The interrupt flag is sent after the stop bit is sent. The software clears the interrupt flag and waits for the next write. The receive data receives data at any time when the receiving enable is effective. Set receive interrupt after receiving stop bit, software clear interrupt flag. The currently received data has a detection mechanism that can detect three errors of receive overflow, frame error, and parity error. Both require a software clear flag. It is recommended to read the status flag after the receive interrupt and clear the receive status flag UART0_STATE[0:3].

Note: The mapping synchronization output function is not supported.

10.5. UART1 Function Description

10.5.1. Baud Rate Generation

Baud rate generation modules: Baud_Mod = {SCI_BDH [4:0], SCI_BDL}.

Baud rate calculation formula: Baud_Mod =0, does not generate baud rate clock. When Baud_Mod =1~8191, SCI baud rate = BUSCLK/(16x Baud_Mod). BUSCLK is the sci work clock, fixed 24MHz clock used in this project. Each time the baud rate register is configured, the internal counter is cleared and the baud rate signal is regenerated. Communication requires the transmitter and receiver to use the same baud rate. Baud rate deviation range allowed by communication: 8/(11*16)=4.5％.

Support automatic baud rate matching. In the LIN protocol, the sync segment character is 0x55. When the baud rate is detected, the measurement starts from the falling edge of the received START bit until the falling edge of the 8th data bit stops. A total of 8 bits will automatically update the Baud_Mod after the communication is completed, and can be read out through the register SCI_BDH/SCI_BDL. Note here that the receiving sync segment automatically matches the baud rate, and the receiving function is performed at the same time. After receiving the character, the receiving interrupt will occur. The maximum deviation before the baud rate match is not allowed to exceed 40%, otherwise the calibration fails.

10.5.2. Transmitter Function

The emitter output pin TXD idle state defaults to a logic high state(txd_inv =0 after reset). If txd_inv =1, the transmitter output is reversed.

The transmitter can send three characters: lead idle character, abort character, data character. Three characters are queued for sending, SCI_TRANS_CTRL[4]: trans_enable bit writes 0 and then writes 1 to queue leading idle characters. SCI_BREAK_CTRL[0]: break_trans_start bit writes 1 and then writes 0 to queue a stop characters, write data register SCI_BUFFER will queue a data character.

The transmitter is enabled by setting the trans_enable bit in the SCI_TRANS_CTRL. This will queue the leading idle characters, the leading idle character is a complete character frame in the idle state, and sends 12-bit or 11-bit or 10-bit idle characters (logic high) according to the data_mode and stop_mode controls. In the normal application process, idle characters need to be sent, the program will wait for tx_empty to be valid and set, the last character of the displayed information has been moved to the transmit shifted, and then 0 and 1 are sequentially written to the tran bit.

Notes: when trans_enable=0, as long as the characters (including three characters) in the shifter are not complete, the SCI transmitter will not stop sending.

By writing data to the SCI data register (SCI_BUFFER), the program saves the data to the transmit data buffer, which queues a data character. The transmit component of the SCI transmitter has a center component length of 10 or 11 or 12 bits (depending on the setting in the data_mode and stop_mode control bit). If data_mode=0, select normal 8-bit data mode. In 8-bit data mode, the shift register has 1 start bit, 8 data bits and 1/2 stop bit. When the transmit shift register can be used for a new SCI character, the value waiting in the transmit data register empty (tx_empty) status flag is set, indicating that another character can be written to the transmit data buffer of the SCI_BUFFER.

By register SCI_BREAK_CTRL[0]: break_trans_start bit writes 1 and then writes 0 to queue a stop character. The abort character is a full-character time of logic 0 (10-bit time), including start and stop bits. The longer pause of 13-bit time can be enabled by setting break_trans_size=1. At the same time, data_mode and stop_mode can each choose to add one time. In general, the program waits for tx_empty to be valid and then sets it to display that the last character of the message has been moved to the transmit shifter, and then writes 0 and 1 to the break bit in turn. Then, once the shifter is available, the operation immediately queues the abort characters that will be sent. If the break is still 1 when the abort that has entered the queue enters shifter, the extra abort character will enter the queue.

If no new characters (including three characters) are waiting in the transmit data buffer after stopping the TxD pin, the transmitter sets the transmission completion flag and enters the idle mode. TxD is in a high state, waiting for more characters to be sent.

Notes: send data empty interrupt generation conditions include: configure the transmitter to enable 0 to 1 enable an empty interrupt, and send a fifo to the shift register to enable a empty interrupt. Turning off the transmitter enable during transmission stops sending after the current character has been sent, clearing the previous queued characters.

Send completion interrupt generation condition: the queued characters are sent once and the completion interrupt is started.

10.5.3. Receiver Function

By setting rxd_inv=1, receiver input is inverted, received input is inverted. By setting SCI_TRANS_CTRL in receive_enable bit, receiver is enabled.

There are three types of received characters: data character, abort character and idle character.

The data character consists of the start bit of logic 0, 8 (or 9) data bits (LSB first) and the stop bit of logic 1. After receiving the stop bit to the receive shifter, if the receive data register is not full (rx_full_if=0), the data character is transferred to the receive data register, setting the receive data register full (rx_full_if=1) status flag. If the rx_full_if of the receive data register is already set at this time, the overflow heart state flag is set and the new data is lost. Because the SCI is double-buffered, the program has a full data in the receive data buffer after setting rx_full_if to avoid receiver overflow.

When the program detects that the receive data register is full (rx_full_if=1), it acquires data from the receive data register by reading the SCI_BUFFER.

The abort character counts from the 0 character of start until the stop bit detects 0 character. The break_check_en bit selects whether the 11-bit abort character detection is enabled. When a rising edge on the pin is detected, the count is cleared. Detected enough 0 characters (11/12/13bit), set abort character detection tag (break_check_if).

The idle character starts from the stop/start bit after the idle character bit count according to the idle bit selection, and starts to the idle bit selection, and starts to detect after the receiver has been active for a period of time (rx_full is effectively set once). Once the 0 character is detected, the count is cleared, and 1 character (10/11/12 bit) is detected, and the idle character detection flag (idle_if) is set.

Notes: enables only the abort character after the abort character detection, regardless of data reception, for lin protocol flow control; close the stop character detection enable, only receive data, ignore the abort character detection.

10.5.4. Receiver sampling method

SCI receiver samples with 16x baud rate. The receiver searches for falling edge on the RxD serial data input pin by extracting logic level samples at 16x baud rate. The falling edge to the definition is a logical 0 sample after 3 consecutive logic 1 samples. The 16x baud rate clock is used to divide the bit time into 16 segments, labeled RT1 and RT16 respectively. When the falling edge is located, three samples are taken from RT3, RT5 and RT7 to ensure that this is the true starting point, not juist the noise. If at least two samples of the three samples that it is synchronized with the receiver character, starts shifting to receive the following data, and if it does not satisfy the above,exits the state machine and returns to the state of waiting for the falling edge state.

The receiver samples each bit time of RT8, RT9 and RT10, including the start bit and the stop bit to determine the logic level of the bit. The logic level is the logic level of most samples extracted during bit time. In the start bit, if at least 2 samples in the sample on RT3, RT5 and RT7 are 0, then the bit is assumed to be 0, even if one or samples extracted on RT8, RT9 and RT10 are 1. If any sample in any bit time of a character frame (8 samples of the start bit RT3~RT10, 3 samples of the other bit 3 RT8~RT10) cannot match the logic level of the bit, a noise error flag is set when the received character is transmitted to the receive data buffer.

The falling edge detection logic constantly looks for the falling edge. If an edge is detected, the bit name so that when the noise or mismatch baud rate occurs, the receiver reliability can be improved.

10.5.5. Receiver Sleep Wake Up

Receiver sleep wake up is a hardware mechanism that uses hardware detection to eliminate software overhead for handling unimportant information characters. Allow SCI receiver to ignore characters in information used for different SCI receivers.

In this application system, the receiver estimates the first character of each message. Once it is determined that the message is intended for different receivers, they immediately write a logic 1 to the receiver wake-up (RWU) control bit in the SCI_TRANS_CTRL. When setting the RWU bit, it is forbidden to set the status flag related to the receiver (when setting rwu_idlesel bit, IDLE bit is set and interrupt is generated).

In the receiver sleep state (software sets the RWU bit to sleep), the wake up mode can be selected by the wake_sel bit (that is, the hardware automatically clears the RWU bit), including idle character wake up and address mark wake up.

Idel character detection are described above. Once the receiver detects a complete idle character, RWU is automatically cleared. After wake-up, the receiver will set the corresponding status flag when the next character is received.

The address mark wake-up is when the receiver detected a logic 1 in the highest bit of the received character (8th bit in data_mode=0; 9th bit r8 in data_mode=1), RWU is automatically cleared. After the wake-up, the receiver related status flag and interrupt and the current character can be set.

10.5.6. Pin Connection Mode

When cycle_mode=1, single_txd bit select cycle mode(single_txd= 0)or signle line mode(single_txd= 1).

Cycle mode:

Cycle mode is independent of external system connections and is sometimes used to check software to help isolate system problems. In this mode, the transmitter output internally supports connection to the receiver input, and SCI does not use the RxD pin.

Signle line mode:

In signle line mode, the txd_direct bit controls the serial data direction on the TxD pin. When txd_direct= 0, the TxD pin is the input of the SCI receiver, connected to the receiver input; when txd_direct=1, the TxD pin is an emitter driven output.

10.6. UART1 Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description**
0xC2	SCI_BDH	RW	0x00	UART1 baudrate control register [4:0]
0xC3	SCI_BDL	RW	0x00	UART1 baudrate control register [7:0]
0xC4	SCI_C1	RW	0x0C	UART1 control register 1
0xC5	SCI_C2	RW	0x00	UART1 control register 2
0xC6	SCI_C3	RO/RW	0x00	UART1 control register 3
0xC7	SCI_S2	RW	0x00	UART1 synchronization interval control register
0xC8	SCI_S1	RO	0x00	UART1 interrupt ststus flag register
0xC9	SCI_D	RW	0xFF	UART1 data register
0xD8	SCI_INT_CLR	W	0x00	UART1 module interrupt clear register

UART SFR register list

10.7. UART1 Register Details

SCI_BDH (C2H) UART0 Baudrate control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7	break_check_ie	Interval detection interrupt enable. 1: interrupt enable; 0: interrupt disable.
6	rx_edge_ie	RxD pin active edge interrupt enable. 1: interrupt enable; 0: interrupt disable.
5	--	Reserved
4~0	--	Baud rate modules divisor register high 5 bits.

SCI_BDL (C3H) UART1 baudrate control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Baud rate control register. Baud rate modules divisor register lower 8 bits, Baud_Mod ={UART0_BDH[1:0], UART0_BDL}, Baud_Mod =0, does not generate baud rate clock. Baud_Mod =1~1023, SCI bandrate = BUSCLK/(16x Baud_Mod)

SCI_C1 (C4H) UART1 control register 1

Bit number	7	6	5	4
Symbol	cycle_mode	stop_mode	single_txd	data_mode
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	parity_en	parity_sel	-	sci_enable
R/W	R/W	R/W	-	R/W
Reset value	0	0	-	0

Bit number	Bit symbol	Description

7	cycle_mode	Cycle mode enable. 1: cycle mode or signal mode, txd connection rxd; 0: normal two-wire mode.
6	stop_mode	stop bit selection. 1: 2bits; 0: 1bit.
5	single_txd	Signal line mode enable. 1: cycle_mode=1, select line mode, txd pin is valid; 0: internal cycle mode, txd pin is invalid.
4	data_mode	Transmission data mode selection. 1: 9 bit mode (the ninth bit is parity bit); 0: 8 bit mode.
3	parity_en	Parity enable. 1: parity enable; 0: parity disable.
2	parity_sel	Parity select. 1: odd parity; 0: even parity
1	--	Reserved
0	sci_enable	Clock gating enable when the module is working, and writing 1 indicates that the enable is valid. Open the module working clock, write 0 will close the module working clock, and reset the function module.

SCI_C2 (C5H) UART1 control register 2

Bit number	7	6	5	4
Symbol	tx_empty_ie	tx_finish_ie	rx_full_ie	idle_ie
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	trans_enable	receive_enable	rwu	break_trans_start
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	tx_empty_ie	Send buffer empty interrupt enable. 1: interrupt enable; 0: interrupt disable.
6	tx_finish_ie	Send complete interrupt enable. 1: interrupt enable; 0: interrupt disable.
5	rx_full_ie	Accept full interrupt enable. 1: interrupt enable; 0: interrupt disable.
4	idle_ie	Idle line interrupt enable.

		1: interrupt enable; 0: interrupt disable
3	trans_enable	Transmitter enable. 1: transmitter open,; 0: transmitter close
2	receive_enable	Receiver enable. 1: receiver open; 0: receiver close.
1	rwu	Receiver wake-up control. 1: receiver is in standby and waiting for the wake condition. 0: receiver is running normally.
0	break_trans_start	Send interval, write 1 and 0 to this bit, that is, a gap is placed in the data stream.

SCI_C2(C6H) UART1 control register 3

Bit number	7	6	5	4
Symbol	r8	t8	txd_direct	txd_inv
R/W	R	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	rxd_inv	rwu_idlesel	idle_sel	wake_sel
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	r8	Receiver’s ninth data, read only.
6	t8	Transmitter’s ninth data.
5	txd_direct	txd pin direction selection in signal line mode. 1: txd pin is the output in signal line mode; 0: txd pin is the input in signal line mode.
4	txd_inv	txd data inversion. 1: send data is reversed; 0: send data is not reserved.
3	rxd_inv	rxd data inversion . 1: receive data is reversed; 0: receive data is not reserved.
2	rwu_idlesel	Receive wake idle detection. 1: during the receive standby state (RWU=1), the idle bit is set when an IDLE character is detected; 0: during the receive standby state (RWU=1), the idle bit is not set when an IDLE character is detected.
1	idle_sel	Idle line type selection.

		1: idle character bit count starts after stop bit; 0: idle character bit count starts after start bit, and the 10-bit time is counted (if data_mode=1 or stop_mode=1, then add one time separately).
0	wake_sel	Receiver wake-up mode selection. 1: address mark wake up; 0: idle line wake up.

SCI_S2(C7H) UART1 sync segment control register

Bit number	7	6	5	4
Symbol	break_check_if	rx_edge_if	rx_active_flag	-
R/W	R/W	R/W	R/W	-
Reset value	0	0	0	-
Bit number	3	2	1	0
Symbol	-	-	break_trans_size	break_check_en
R/W	-	-	R/W	R/W
Reset value	-	-	0	0

Bit number	Bit symbol	Description
7	break_check_if	Interval detection interrupt flag. 1: interval detected; 0: no interval detected, this bit writes 1 clear, write 0 is invalid.
6	rx_edge_if	RxD pin active edge interrupt flag. 1: active edge on the receive pin; 0: active edge does not appear on the receive pin; this bit writes 1 clear, write 0 is invalid.
5	rx_active_flag	Receiver activity tag, read only. 1: receiver activity; 0: receiver idle.
4~2	--	Reserved
1	break_trans_size	Interval generation bit length. 1: send by 13-bit time (if data_mode=1 or stop_mode=1, add 1 bit length respectively); 0: send by 10-bit time (if data_mode=1 or stop_mode=1, add 1 bit length respectively).
0	break_check_en	Interval detection enable. 1: detected over 11 bit lengths (if data_mode=1 or stop_mode=1, add 1 bit length respectively); 0: not detecting.

SCI_S1(C8H) UART1 interrupt status flag register

Bit number	7	6	5	4

Symbol	tx_empty_if	tx_finish_if	rx_full_if	idle_if
R/W	R	R	R	R
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	rx_overflow_if	noise_err_if	frame_err_if	Parity_err_if
R/W	R	R	R	R
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	tx_empty_if	Send buffer empty interrupt flag. 1: send buffer is empty; 0: send buffer is full, read only.
6	tx_finish_if	Send completion interrupt flag. 1: send completed, transmitter idle; 0: the transmitter is working, read only.
5	rx_full_if	Accept full interrupt flag. 1: receiver buffer is full; 0: receiver buffer is empty, read only.
4	idle_if	Idle line break flag. 1: idle line detected; 0: no idle line detected, read only.
3	rx_overflow_if	Receive overflow mark. 1: receive overflow (new data loss); \ 0: no overflow, read only.
2	noise_err_if	Noise marker. 1: noise detected; 0: no noise detected, read only.
1	frame_err_if	Frame error flag. 1: framing error detected; 0: no framing error detected, read only.
0	parity_err_if	Parity error flag. 1: receiver parity error; 0: parity is correct, read only.

SCI_D(C9H) UART1 data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	-	SCI data register. Read returns the contents of the read-only receive data buffer,

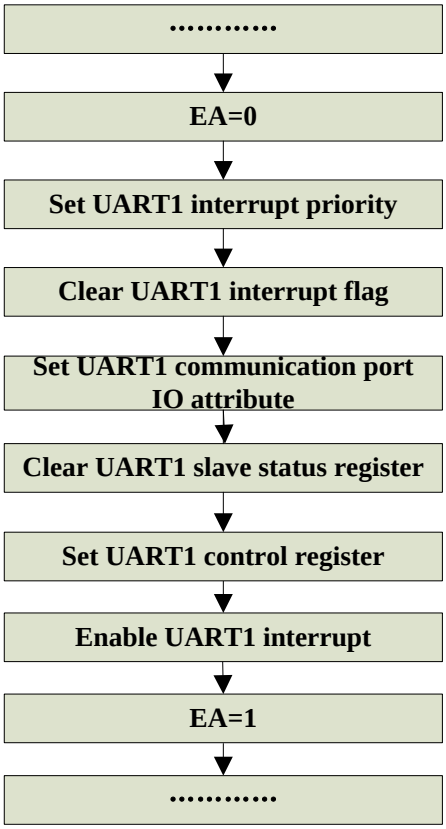
		writes to the write-only send data buffer.

SCI_INT_CLR (D8H) UART1 interrupt flag clear register

Bit number	7	6	5	4
Symbol	clr_tx_empty_if	clr_tx_finish_if	clr_rx_full_if	clr_idle_if
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	clr_rx_overflow_if	clr_noise_err_if	clr_frame_err_if	clr_parity_err_if
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	clr_tx_empty_if	Transmit buffer empty interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
6	clr_tx_finish_if	Transmit complete interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
5	clr_rx_full_if	Receive full interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
4	clr_idle_if	Idle line interrupt clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
3	clr_rx_overflow_if	Receive overflow flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
2	clr_noise_err_if	Noise flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
1	clr_frame_err_if	Frame flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.
0	clr_parity_err_if	Parity error flag clear bit, this bit writes 1 to clear the corresponding interrupt, write 0 is invalid.

10.8. UART1 Configuration Process



UART1 initial configuration process

11. PWM

11.1. PWM0 Function Description

PWM0 function description:

- Support 4 channels, each channel is individually enabled;
- 16bit counter;
- Counting cycle configurable, adjustable duty cycle per channel;
- Polar complementary output;
- PWM0_B/C/D(PWM0_CH1/2/3) duty cycle can be selected as PWM0_A(PWM0_CH0) configuration, can also choose to configure the duty cycle of your channel;
- Support common frequency: 38kHz (infrared application)

PWM0 pulse width modulation module can be configured by register for both cycle and pulse width, but the configuration of the register must be selected if the PWM0 output port is valid (highly effective), and each set of registers must be configured from low to high (include PWM0_MOD_L/H, PWM0_CHX_CNT_L/H), In order to ensure that the internal counter of the PWM0 modules is correctly counted, the error waveform is avoided. These configuration values update the register value when the counter changes from (PWM0_MOD) to (PWM0_MOD+1), is the update cycle and duty cycle after a full cycle.

PWM0 module support 4 channels, each channel can be individually controlled to enable. Share a 16-bit counter, the count clock is 24MHz and the system clock is synchronized. The PWM0 signal period is determined by the value of the period configuration register (PWM0_MOD), which is determined by the setting in channel register (PWM0_CHn_CNT). The polarity of the PWM0 signal is determined by the setting in the PWM0_CH_CTRL control bit. 0% and 100% duty cycle is possible.

Pulse width = (PWM0_CHn_CNT)

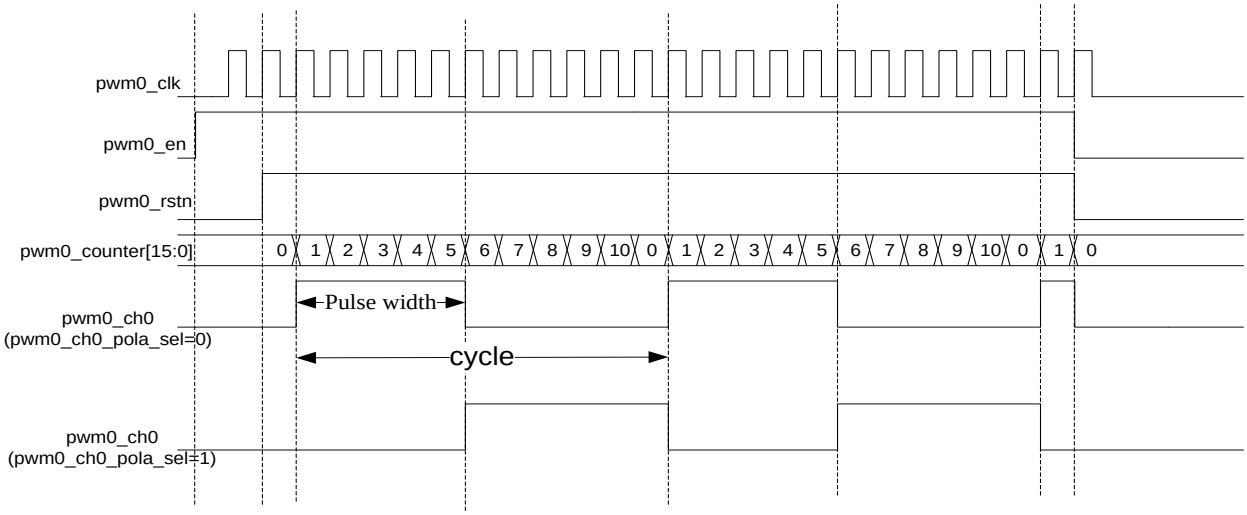
Cycle = (PWM0_MOD+1)

Duty cycle = pulse width / cycle

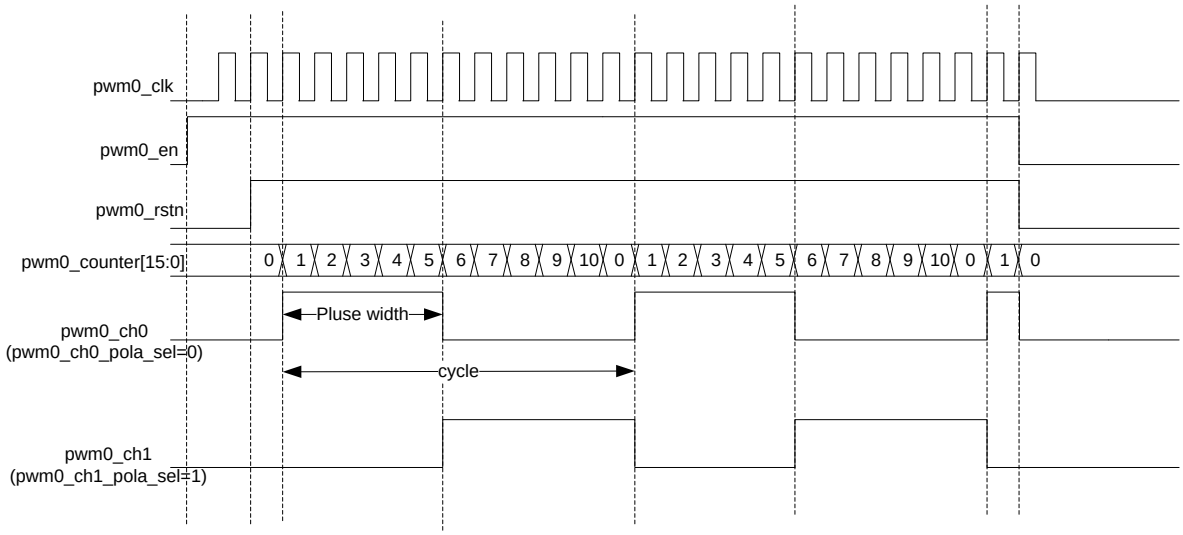
PWM0 counter counts up from 0x0000, when PWM0_CHn_CNT is counted, the output is inverted. This time is the pulse width. Countine counting until the count overflows at PWM0_MOD+1.

If PWM0_CH0_POLA_SEL=0, PWM0 signal enters high state when output is flipped. If PWM0_CH0_POLA_SEL=1, PWM0 signal enters high state when output is overflows.

When channel count register (PWM0_CHn_CNT) is set 0x0000, the duty cycle is 0. When channel count register (PWM0_CHn_CNT) is set to a value greater than the value set by the period configuration register (PWM0_MOD) to achieve a 100% duty cycle. The counter is automatically reloaded and will not stop by itself until the register PWM0 is enabled to stop and the counter is cleared.



(PWM0_CH0_CNT=5, PWM0_MOD=10, duty_cycle=5/11)



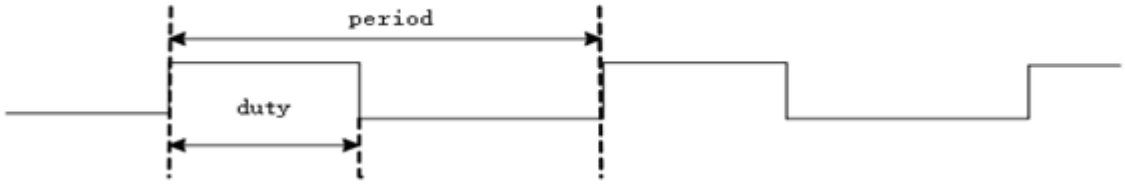
(PWM0_CH0, PWM0_CH1 complementary output)

11.2. PWM1/2 Function Description

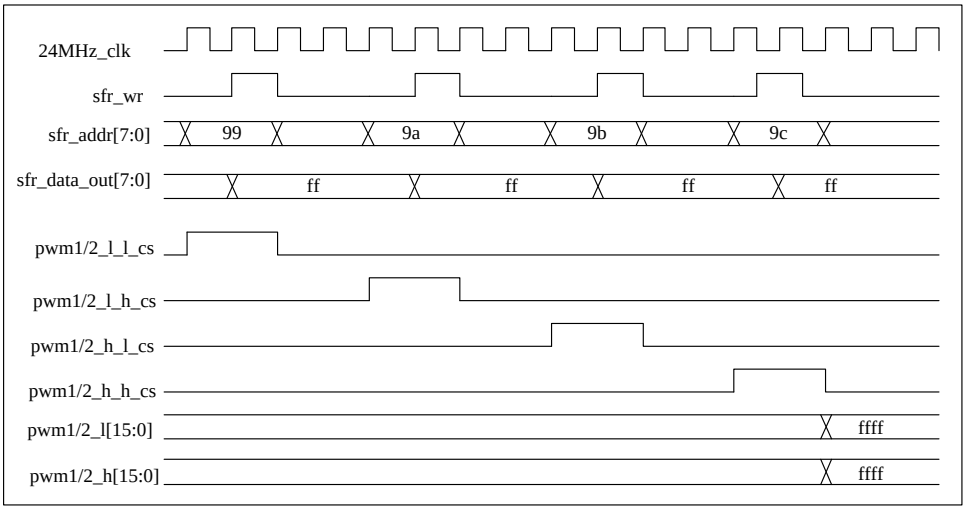
PWM1/2 features are as follows:

- The clock source is the clock CLK_24MHz;
- High level control register and low level control register : 16-bit register;
- Output cycle: TPWM1/2_data = (PWM1/2_H + PWM1/2_L)* $T_{\_CLK\_24MHz}$(us);
- Output duty cycle: DPWM1/2_data = PWM1/2_H/(PWM1/2_L + PWM1/2_H);
- Support common frequency: 38kHz (infrared application)

PWM1/2 waveform diagram



The PWM1/2 pulse width modulation module can be configured through registers in both high and low time, but the configuration of the register must be enabled again when PWM1/2 is enabled (high effective), the high level control register and the low level control register must be configured in descending order. In order to ensure that the internal counter of the PWM1/2 module is correctly counted, avoiding the generation of incorrect waveforms.



PWM1/2 timing diagram

11.3. PWM Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description**
0x99	PWM1_L_L	RW	0x00	PWM1 low level control register (lower 8 bits)
0x9A	PWM1_L_H	RW	0x00	PWM1 low level control register (high 8 bits)
0x9B	PWM1_H_L	RW	0x00	PWM1 high level control register (lower 8 bits)
0x9C	PWM1_H_H	RW	0x00	PWM1 high level control register (high 8 bits)
0x9D	PWM2_L_L	RW	0x00	PWM2 low level control register (low 8 bits)
0x9E	PWM2_L_H	RW	0x00	PWM2 low level control register (high 8 bits)
0x9F	PWM2_H_L	RW	0x00	PWM2 high level control register (lower 8 bits)
0xA1	PWM2_H_H	RW	0x00	PWM2 high level control register (high 8 bits)
0xA2	PWM_EN	RW	0x00	PWM Control Register
0xA3	PWM0_CH_CTRL	RW	0x00	PWM0 Control Register
0xA4	PWM0_CH0_CNT_L	RW	0x00	PWM0 channel 0 count value configuration register lower 8 bits
0xA5	PWM0_CH0_CNT_H	RW	0x00	PWM0 channel 0 count value configuration register upper 8 bits
0xA6	PWM0_CH1_CNT_L	RW	0x00	PWM0 channel 1 count value configuration register lower 8 bits
0xA7	PWM0_CH1_CNT_H	RW	0x00	PWM0 channel 1 count value configuration register upper 8 bits
0xA9	PWM0_CH2_CNT_L	RW	0x00	PWM0 channel 2 count value configuration register lower 8 bits
0xAA	PWM0_CH2_CNT_H	RW	0x00	PWM0 channel 2 count value configuration register upper 8 bits
0xAB	PWM0_CH3_CNT_L	RW	0x00	PWM0 channel 3 count value configuration register lower 8 bits
0xAC	PWM0_CH3_CNT_H	RW	0x00	PWM0 channel 3 count value configuration register upper 8 bits
0xAD	PWM0_MOD_L	RW	0x00	PWM0 period configuration register

				lower 8 bits
0xAE	PWM0_MOD_H	RW	0x00	PWM0 period configuration register upper 8 bits

11.3.1. PWM1 Low Level Control Registers

PWM1_L_L (99H) PWM1 low level control register (low 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM1_L_H (9AH) PWM1 low level control register (high 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

11.3.2. PWM1 High Level Control Registers

PWM1_H_L (9BH) PWM1 high level control register (low 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM1_H_H (9CH) PWM1 high level control register (high 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

11.3.3. PWM2 Low Level Control Registers

PWM2_L_L (9DH) PWM2 low level control register (low 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM2_L_H (9EH) PWM2 low level control register (high 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							

R/W	R/W
Reset value	0

11.3.4. PWM2 High Level Control Registers

PWM2_H_L (9FH) PWM2 high level control register (high 8bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

PWM2_H_H (A1H) PWM2 high level control register(high 8-bit)

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

11.3.5. PWM Control Registers

Channel 0: PWM0_A; Channel 1: PWM0_B; Channel 2: PWM0_C; Channel 3: PWM0_D.

PWM_EN (A2H) PWM control register

Bit number	7	6	5	4
Symbol	-	-	PWM0_CH3_CMOD	PWM0_CH2_CMOD
R/W	-	-	R/W	R/W
Reset value	-	-	0	0
Bit number	3	2	1	0
Symbol	PWM0_CH1_CMOD	PWM2_EN	PWM1_EN	PWM0_EN
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
5	PWM0_CH3_CMOD	PWM0 channel 3 duty cycle mode select register 1: select channel 0 duty cycle 0: select its own channel duty cycle
4	PWM0_CH2_CMOD	PWM0 channel 2 duty cycle mode select register 1: select channel 0 duty cycle 0: select its own channel duty cycle
3	PWM0_CH1_CMOD	PWM0 channel 1 duty cycle mode select registe 1: select channel 0 duty cycle 0: select its own channel duty cycle
2	PWM2_EN	PWM2 module enable register

		1: enable; 0: not enable
1	PWM1_EN	PWM1 module enable register 1: enable; 0: not enable
0	PWM0_EN	PWM0 module enable register 1: enable; 0: not enable

11.3.6. PWM0 Control Registers

Channel 0: PWM0_A; Channel 1: PWM0_B; Channel 2: PWM0_C; Channel 3: PWM0_D.

PWM0_CH_CTRL (A3H) PWM0 control register

Bit number	7	6	5	4
Symbol	PWM0_CH3_POLA_SEL	PWM0_CH2_POLA_SEL	PWM0_CH1_POLA_SEL	PWM0_CH0_POLA_SEL
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0
Bit number	3	2	1	0
Symbol	PWM0_CH3_EN	PWM0_CH2_EN	PWM0_CH1_EN	PWM0_CH0_EN
R/W	R/W	R/W	R/W	R/W
Reset value	0	0	0	0

Bit number	Bit symbol	Description
7	PWM0_CH3_POLA_SEL	Channel 3 polarity selection ch3_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
6	PWM0_CH2_POLA_SEL	Channel 2 polarity selection ch2_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
5	PWM0_CH1_POLA_SEL	Channel 1 polarity selection ch1_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
4	PWM0_CH0_POLA_SEL	Channel 0 polarity selection ch0_pola_sel 1: count value overflow makes the output low 0: count value overflow makes the output high
3	PWM0_CH3_EN	Channel 3 enable ch3_en 1: enable; 0: not enable
2	PWM0_CH2_EN	Channel 2 enable ch2_en 1: enable; 0: not enable
1	PWM0_CH1_EN	Channel 1 enable ch1_en 1: enable; 0: not enable
0	PWM0_CH0_EN	Channel 0 enable ch0_en

		1: enable; 0: not enable

11.3.7. PWM0 Channel Count Value Configuration Registers

The following registers: Configure the PWM0 output duty cycle.
Channel 0: PWM0_A; Channel 1: PWM0_B; Channel 2: PWM0_C; Channel 3: PWM0_D.
PWM0_CH0_CNT_L (A4H) PWM0 channel 0 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH0_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH0_CNT_L	Channel 0 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH0_CNT_H (A5H) PWM0 channel 0 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH0_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH0_CNT_H	Channel 0 count configuration register high 8 bits. Configure PWM output duty cycle.

PWM0_CH1_CNT_L (A6H) PWM0 channel 1 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH1_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH1_CNT_L	Channel 1 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH1_CNT_H (A7H) PWM0 channel 1 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH1_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description

7~0	PWM0_CH1_CNT_H	Channel 1 count configuration register high 8 bits. Configure PWM output duty cycle.

PWM0_CH2_CNT_L (A9H) PWM0 channel 2 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH2_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH2_CNT_L	Channel 2 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH2_CNT_H (AAH) PWM0 channel 2 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH2_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH2_CNT_H	Channel 2 count configuration register high 8 bits. Configure PWM output duty cycle.

PWM0_CH3_CNT_L (ABH) PWM0 channel 3 count value configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH3_CNT_L							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH3_CNT_L	Channel 3 count configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_CH3_CNT_H (ACH) PWM0 channel 3 count value configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_CH3_CNT_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_CH3_CNT_H	Channel 3 count configuration register low 8 bits. Configure PWM output duty cycle.

11.3.8. PWM0 Period Configuration Register

PWM0_MOD_L (ADH) PWM0 cycle configuration register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_MOD_L							
R/W	R/W							
Reset value	0							

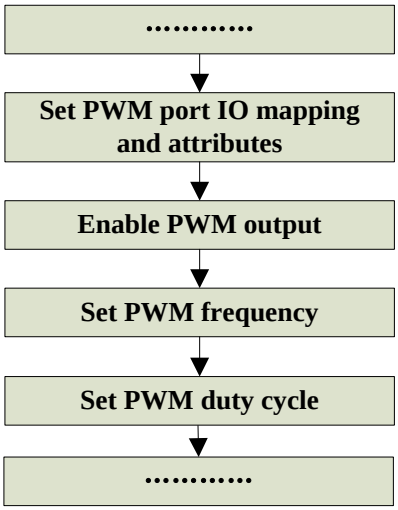
Bit number	Bit symbol	Description
7~0	PWM0_MOD_L	PWM0 count cycle configuration register low 8 bits. Configure PWM output duty cycle.

PWM0_MOD_H (AEH)PWM0 cycle configuration register high 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	PWM0_MOD_H							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	PWM0_MOD_H	PWM0 count cycle configuration register high 8 bits. Configure PWM output duty cycle.

11.4. PWM Configure Process



PWM configure process

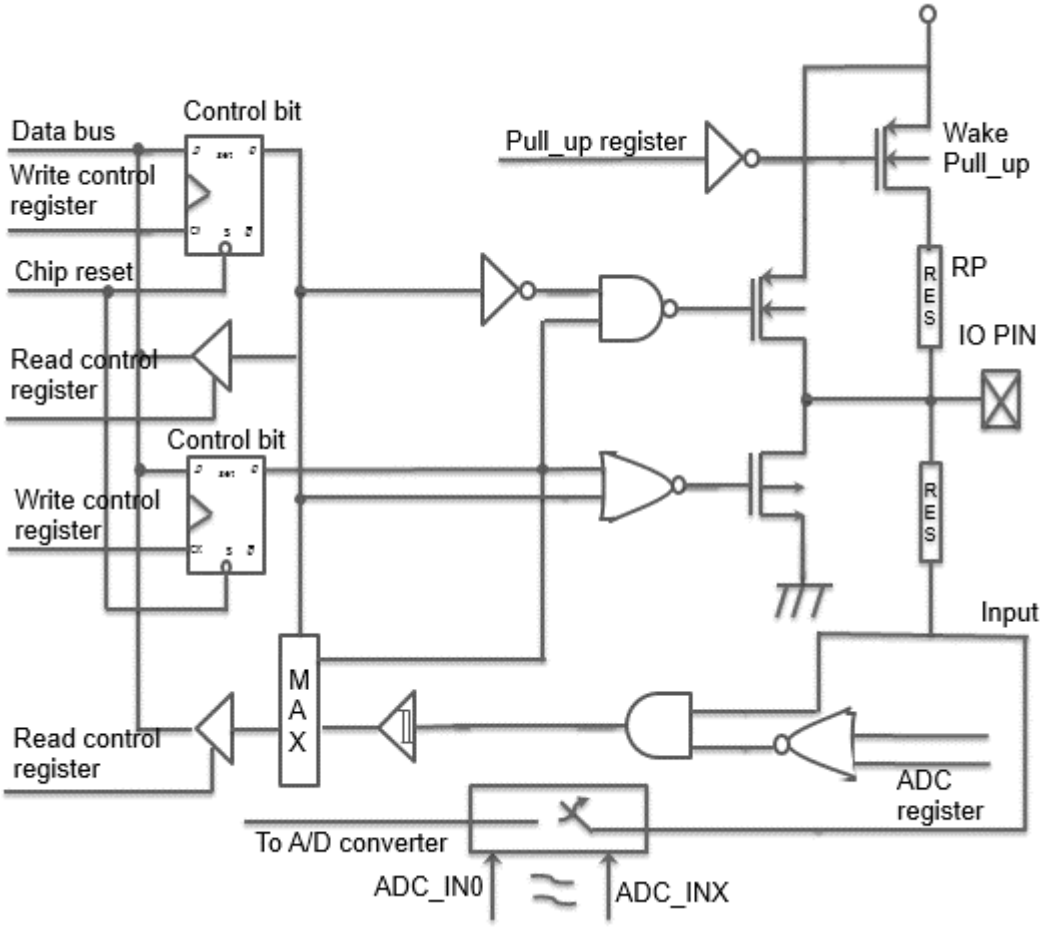
Note:

Frequency range: 370 Hz~369 kHz is recommended.

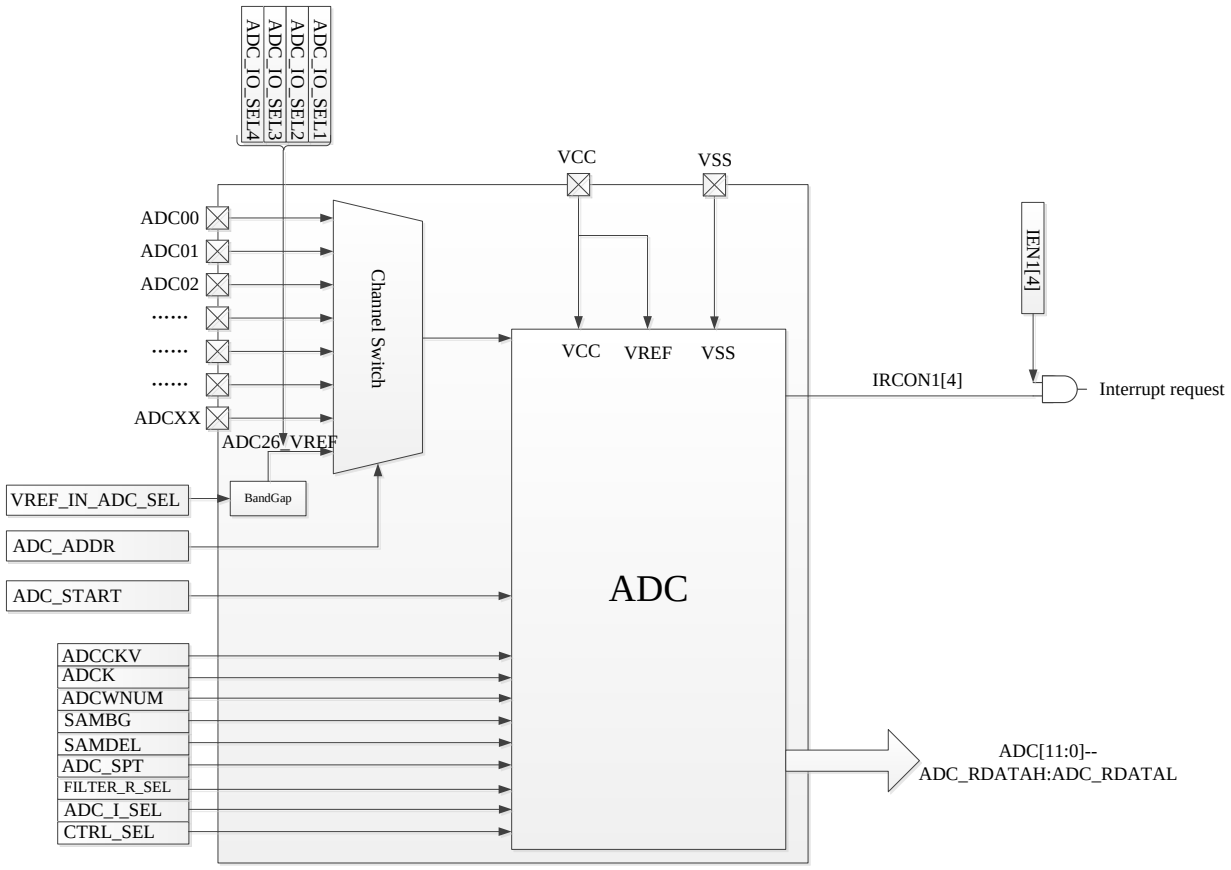
12. ADC

BF7512CMXX-SJLX chip include a signal-ended, 12-bit linear successive approximation analog -to-digital converter (ADC). The reference voltage of the ADC is connected to the VCC of the chip. ADC channels can input independent analog signals. ADC module converts one channel at a time, ADC_START=0→1(⎍) turn on conversion. Update the ADC result register and generate an interrupt after the conversion is complete.

- Liner successive approximation ADC with 12bit resolution and 10bit precision;
- Single conversion mode;
- Sampling time and conversion speed are configurable;



ADC module structure diagram



ADC block diagram

12.1. ADC Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0xB4	ADC_SPT	RW	0x00	ADC sampling time configure register
0xB5	ADC_SCAN_CFG	RW	0x00	ADC scan control register
0xB6	ADCCKC	RW	0x00	ADC clock control register
0xB9	ADC_RDATAH	R	0x00	ADC scan result register, high 4 bit.
0xBA	ADC_RDATAL	R	0x00	ADC scan result register, low 8 bit.
0xBB	ADC_CFG1	RW	0x00	ADC sampling time configure register 1
0xBC	ADC_CFG2	RW	0x02	ADC sampling time configure register 2
0xD9	ADC_IO_SEL1	RW	0x00	ADC function select register 1
0xDA	ADC_IO_SEL2	RW	0x00	ADC function select register 2
0xDB	ADC_IO_SEL3	RW	0x00	ADC function select register 3
0xDC	ADC_IO_SEL4	RW	0x00	ADC function select register 4
0xFE	PD_ANA	RW	0x1F	Module switch control register

ADC SFR register list

12.2. ADC Register Details

ADC_SPT (B4H) ADC sample time configure register

Bit number	7	6	5	4	3	2	1	0
Symbol	ADC_SPT							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	ADC_SPT	ADC sample time configure register sample time: sample_Timer = (ADC_SPT+1)*4Tadc_clk

ADC_SCAN_CFG (B5H) ADC scan control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	ADC_ADDR					ADC_START
R/W	-	-	R/W					R/W
Reset value	-	-	0					0

Bit number	Bit symbol	Description
5~1	ADC_ADDR	ADC channel address select register. 0~25: corresponding to ADC0~ADC25; 26: internal input channel of the chip
0	ADC_START	ADC scan open register ADC_START=0→1(↑) turn to conversion, ADC_START configuration is not allowed during scanning. ADC_START is set from 0 to 1, ADC start to scan, after scanning once, ADC_START hardware is automatically set to 0, corresponding to the interrupt flag set to 1, ADC interrupt flag bit needs to be cleared by software.

ADCCKC (B6H) ADC clock control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	ADCCKV		ADCK	
R/W	-	-	-	-	R/W		R/W	
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
3~2	ADCCKV	ADC comparator offset cancellation analog input clock. 0: 12MHz 1: 8MHz 2: 4MHz 3: 2MHz
1~0	ADCK	ADC_CLK frequency division selection. 0: 8MHz 1: 6MHz 2: 4MHz 3: 3MHz

ADC_RDATAH (B9H) ADC scan result register high 4 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	ADC_RAWDATA<11:8>			
R/W	-	-	-	-	R			
Reset value	-	-	-	-	0			

Bit number	Bit symbol	Description
3~0	ADC_RAWDATA<11:8>	ADC scan result register

ADC_RDATAL(BAH) ADC scan result register low 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	ADC_RAWDATA<7:0>							
R/W	R							
Reset value	0							

Bit number	Bit symbol	Description
7~0	ADC_RAWDATA<7:0>	ADC scan result register

ADC_CFG1(BBH) ADC sampling timing control register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	ADCWNUM					SAMBG	SAMDEL	
R/W	R/W					R/W	R/W	
Reset value	0					0	0	

Bit number	Bit symbol	Description
7~3	ADCWNUM	Distance conversion interval selection after sampling. (3+ADCWNUM)*ADC_CLK
2	SAMBG	Sampling timing and comparison timing interval selection. 0: interval 0; 1: interval 1*ADC_CLK.
1~0	SAMDEL	Sampling delay time selection. 0: 0*ADC_CLK; 1: 2*ADC_CLK; 2: 4*ADC_CLK; 3: 8*ADC_CLK.

ADC_CFG2 (BCH) ADC sampling timing control register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	-	FILTER_R_SEL	VREF_IN_ADC_SEL		ADC_I_SEL[1:0]		CTRL_SEL[1:0]	
R/W	-	R/W	R/W		R/W		R/W	
Reset value	-	0	0		0		10	

Bit number	Bit symbol	Description
6	FILTER_R_SEL	Input signal filtering selection, 0 means no RC filtering,

		1 means RC filtering.
5~4	VREF_IN_ADC_SEL	Input to ADC26 reference voltage selection 01: 2.253V; other: reserved; Need to read the calibration voltagevalue from the chip flash when using. VREF_IN_ADC_SEL voltage = { CBYTE[0x43C6], CBYTE[0x43C7]}mV.
3~2	ADC_I_SEL[1:0]	ADC bias current size selection register. ADC_I_SEL[0]: 0 is the comparator bias current is 4uA; 1 is the comparator bias current is 5uA; ADC_I_SEL[1]: 0 is the op amp bias current is 4uA; 1 is the op amp bias current is 5uA;
1~0	CTRL_SEL[1:0]	ADC comparator offset cancellation selection signal, the default is 10. CTRL_SEL[1:0]: 00/01: sampling first in offset cancellation; 10: all switches are disconnected together; 11: the switch is disconnected in turn.

ADC_IO_SEL1 (D9H) ADC function selection register 1

Bit number	7	6	5	4	3	2	1	0
Symbol	SEL_ADC[7:0]							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	SEL_ADC[7:0]	ADC function selection. 1: select ADC function; 0: do not select ADC function.

ADC_IO_SEL2(DAH) ADC function selection register 2

Bit number	7	6	5	4	3	2	1	0
Symbol	SEL_ADC[15:8]							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	SEL_ADC[15:8]	ADC function selection. 1: select ADC function; 0: do not select ADC function.

ADC_IO_SEL3(DBH) ADC function selection register 3

Bit number	7	6	5	4	3	2	1	0
Symbol	SEL_ADC[23:16]							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	SEL_ADC[23:16]	ADC function selection. 1: select ADC function; 0: do not select ADC function.

ADC_IO_SEL4(DCH) ADC function selection register 4

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	SEL_ADC[25:24]	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	

Bit number	Bit symbol	Description
1~0	SEL_ADC[25:24]	ADC function selection. 1: select ADC function; 0: do not select ADC function.

PD_ANA (FEH) Module switch control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	PD_LVDT	PD_BOR	PD_XTAL_32K	-	PD_ADC
R/W	-	-	-	R/W	R/W	R/W	-	R/W
Reset value	-	-	-	1	1	1	-	1

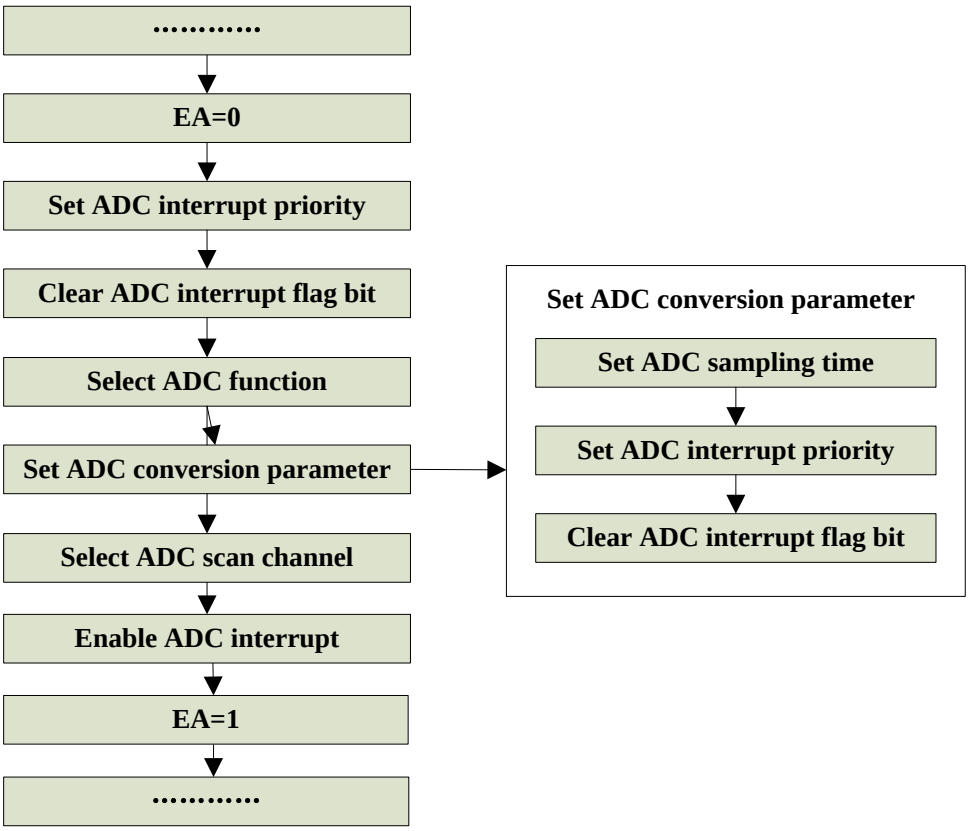
Bit number	Bit symbol	Description
0	PD_ADC	Analog ADC shutdown control register: PD_ADC=0 ADC module work properly; PD_ADC=1 ADC module don’t work.

Note: ADC detection time:

Formula	Description
$T_{AD}=T_{ADC\_SPT}+T_{W1}+T_{W2}$	ADC detection time
$T_{ADC\_SPT(\mu s)} = 4*(ADC\_SPT+1) *T_{adc\_clk}$	Sampling time
$T_{W1}=(ADCWNUM+3+ SAMDEL)*T_{adc\_clk}$	Distance conversion interval and delay time after sampling is completed
$T_{W2}=(2*1+12)*T_{adc\_clk}$	Fixed time
$F_{adc\_clk(MHz)}$	ADC division clock

1. Timing requirements: (3+ADCWNUM)/F_ADCK >4 /F_ADCCKV;
   F_ADCK: ADC frequency division clock;
   F_ADCCKV: ADC comparator offset cancellation analog input clock;
   ADC external signal plus RC filtered voltage setting time >= 2*(ADC sampling converts time);
2. When the power supply voltage fluctuates greatly or drops, the VCC voltage value can be inversely calculated by the formula ADCINNER_Data/VREF_IN_ADC_SEL = 4096/VCC, and the Vin voltage value can be inversely calculated by the formula Vin_Data/Vin=4096/VCC.
   ADCINNER_Data: ADC internal channel data;
   Vin_Data: ADC input channel data;
   Vin: input voltage;
   VREF_IN_ADC_SEL: Need to read the chip calibration value;
   Vin = (Vin_Data/ADCINNER_Data)*VREF_IN_ADC_SEL, VREF_IN_ADC_SEL needs to read the chip calibration value, first obtain the internal channel data, and then obtain the input voltage Vin_Data data, and the interval between the two acquisitions should be as short as possible;
3. ADC interrupt conditions: the configuration sequence is ADC_IO_SEL enable -> ADC interrupt enable -> ADC_ADDR (the address must correspond to ADC_IO_SEL) -> ADC_START. Pay attention to the initial configuration timing during application. If there is an application where the ADC and IO port functions are multiplexed, you need to pay attention to the switching timing. If the ADC_IO_SEL enable is turned off or the address does not correspond to ADC_IO_SEL, the ADC scan cannot be turned on. The configuration sequence must be followed: ADC_IO_SEL enable -> ADC interrupt enable -> ADC_ADDR (the address must correspond to ADC_IO_SEL) -> ADC_START sequence to start ADC scanning.
5. When a pin is configured as ADC function, the pin needs to be configured as IO input mode, and other multiplexing functions, such as pull-up resistors, are disabled.

12.3. ADC Configuration Process



ADC configuration process

13. LVDT

BF7512CMXX-SJLX series supports low pressure alarm function, effectively monitor voltage dynamics. Support four gear voltages are: 2.4V/3.0V/3.6V/4.2V (Preset point buck interrupt, hysteresis 0.1V to generate corresponding boost interrupt).

When the voltage monitoring configures the above threshold, the voltage drops to this threshold will trigger a low voltage interrupt. The system can be propely processed in low voltage interrupts according to the needs of the application.

13.1. LVDT Related Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Description**
0x86	INT_POBO_STAT	RW	0x00	LVDT boost/LVDT buck interrupt status register
0xFE	PD_ANA	RW	0x1F	Module switch control register
0xFF	SEL_LVDT_VTH	RW	0x00	LVDT threshold selection register

LVDT SFR register list

13.2. LVDT Register Details

INT_POBO_STAT (86H) LVDT boost/LVDT buck interrupt status register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	INT_PO_STAT	INT_BO_STAT
R/W	-	-	-	-	-	-	R/W	R/W
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1	INT_PO_STAT	Lvdt boost interrupt status 1: boost interrupt is valid 0: boost interrupt is invaild
0	INT_BO_STAT	Lvdt buck interrupt state 1: buck interrupt is valid 0: buck interrupt is invalid

PD_ANA (FEH) Module switch control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	PD_LVDT	PD_BOR	PD_XTAL_32K	-	PD_ADC
R/W	-	-	-	R/W	R/W	R/W	-	R/W
Reset value	-	-	-	1	1	1	-	1

Bit number	Bit symbol	Description
4	PD_LVDT	LVDT control register. 1: close; 0: open; default close.
3	PD_BOR	BOR control register. 1: close; 0: open; VBOR=2.1V , default close.
2	PD_XTAL_32K	RTC crystal oscillator circuit (32768Hz/4MHz) control register. 1: close; 0: open; default close.
1	--	Reserved
0	PD_ADC	Analog ADC shutdown control register: PD_ADC=0 ADC module work properly; PD_ADC=1 ADC module don’t work.

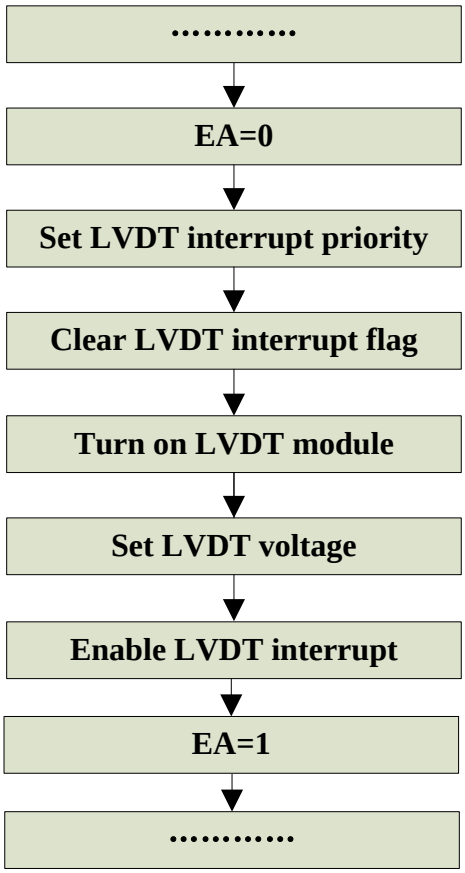
SEL_LVDT_VTH (FFH) LVDT threshold selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	
R/W	-	-	-	-	-	-	R/W	
Reset value	-	-	-	-	-	-	0	0

Bit number	Bit symbol	Description
1~0	--	LVDT threshold selection; 00=2.4V; 01=3.0V; 10=3.6V; 11=4.2V.

Note: It is recommended that the LVDT be configured with 3V. The low level of the LVDT voltage detection point has better suppression of power ripple. If the high-voltage detection voltage level is disturbed, the software needs to perform dry debounce processing to reduce the probability of misjudgment.

13.3. LVDT Configuration Process



LVDT Configuration Process

14. LED

LED dot matrix drive mode features:

- Support max 64 LED drive, configurable selection lattice 4x4, 5x5, 6x6, 6x7, 7x7, 7x8, 8x8(fixed pin after configuration);
- Dual lamp simulta neous conduction mode, the specific allocation is showen in the following lattice description;
- Signal lamp on time setting file: register 8 bits, configurable range is 16us-4.096ms, the step is 16us;
- Each lamp driver time can be selected separately;
- IO ports have multiple multiplexing relationships, each IO port needs to be configured through software to switch to LED port, and the LED function of LED0~LED8 corresponding to IO port will be automatically turned on according to the LED matrix mode selection;
- 64-light dot matrix address is unique. See the dot matrix description below for inputting switch light information.

14.1. Function Description

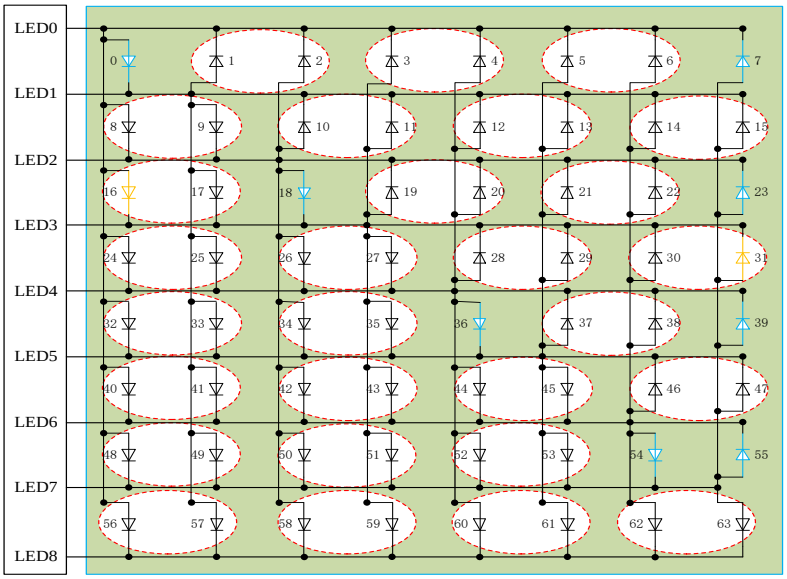
14.1.1. LED

LED dot matrix driver circuit consists of a controller, two counter, a comparator and a SRAM memory circuit.

LED dot matrix is a universal 8*8 matrix dual lamp mode scan, that is, two lights at a time (common cathode).

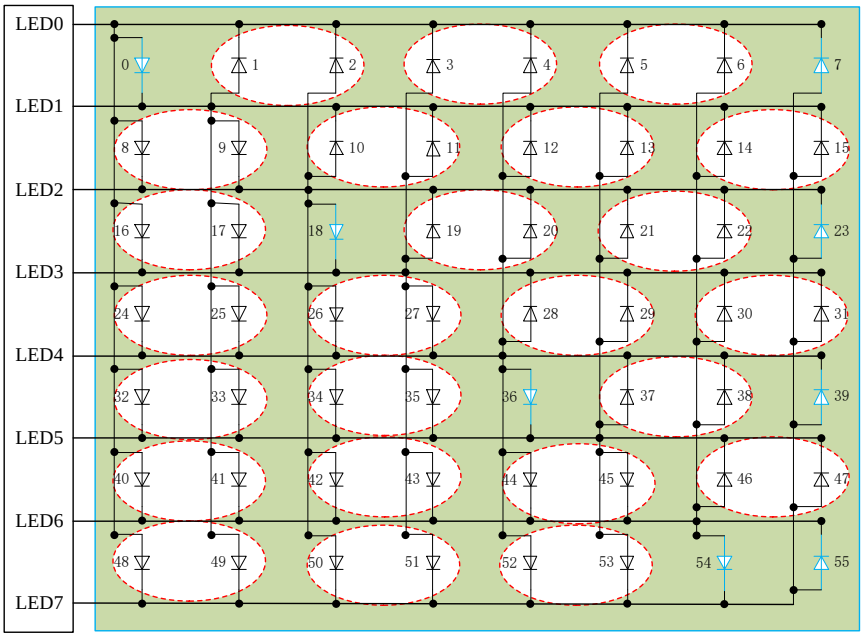
Corresponding to the LED0~LED8 port, up to 8x8=64 lights can be configured. The lights address of the corresponding position is marked in the 8*8 dot matrix below. The display configuration in the SRAM corresponding address lighting situation(1 means light, 0 means no light), The hardware code needs to resolve the lighting address and the current scanning address to automatically complete the corresponding IO port outport control. Configurable dot matrix 4x4, 5x5, 6x6, 6x7, 7x7, 7x8, 8x8, lamp addresses corresponding to different size lattices are unchanged.

8*8 lattice:



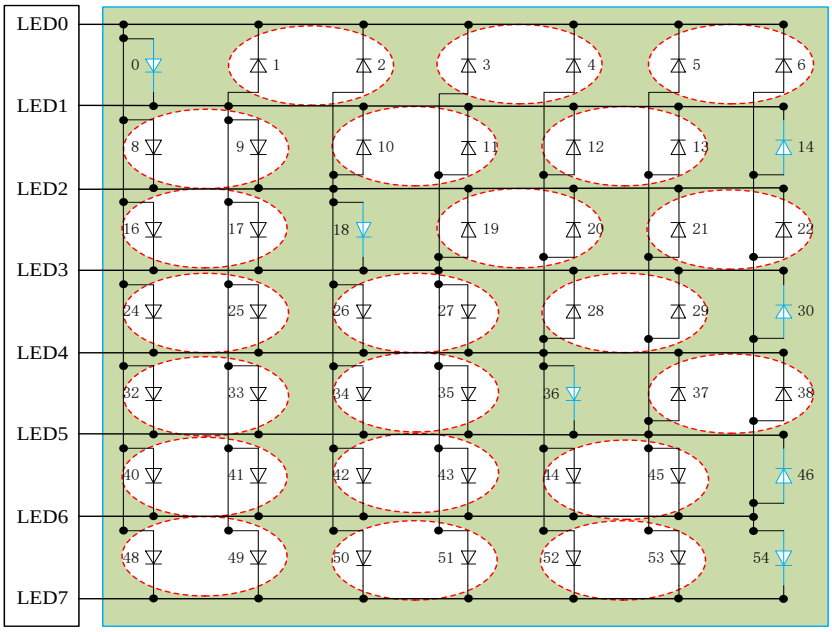
LED 8*8 lattice:

7*8 lattice:



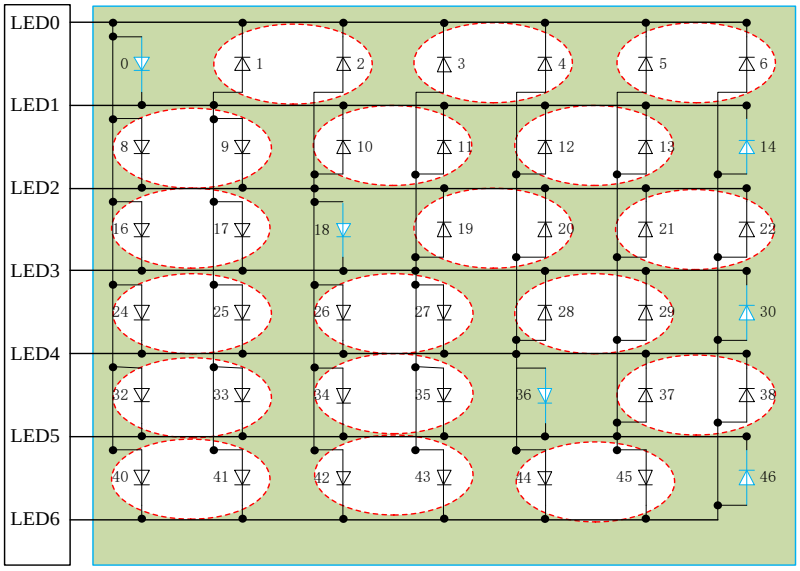
LED 7*8 lattice

7*7 lattice:



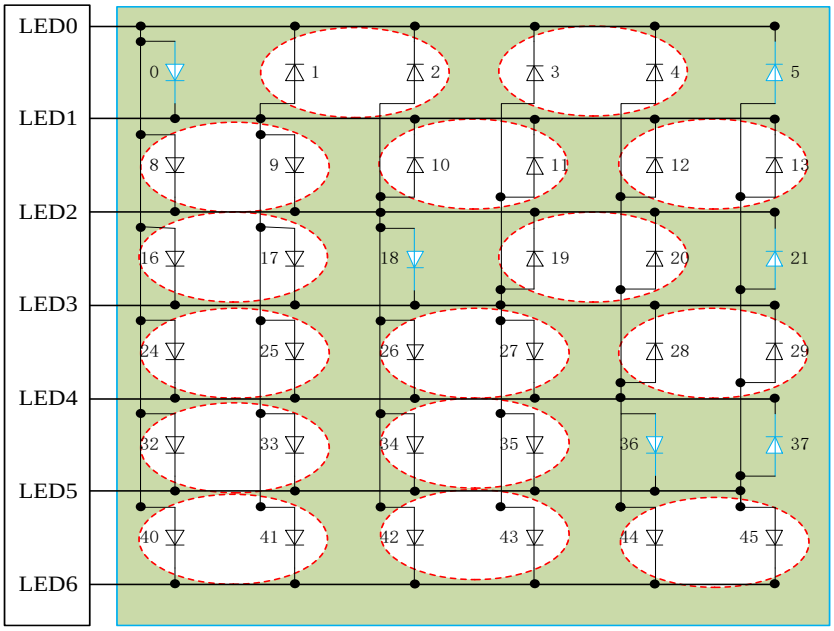
LED 7*7 lattice

6*7 lattice:



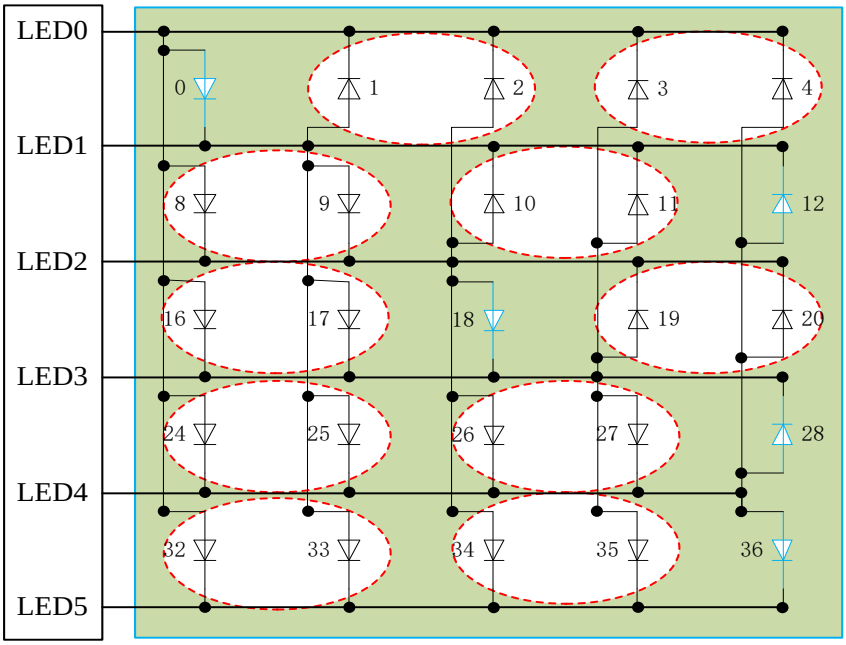
LED 6*7 lattice

6*6 lattice:



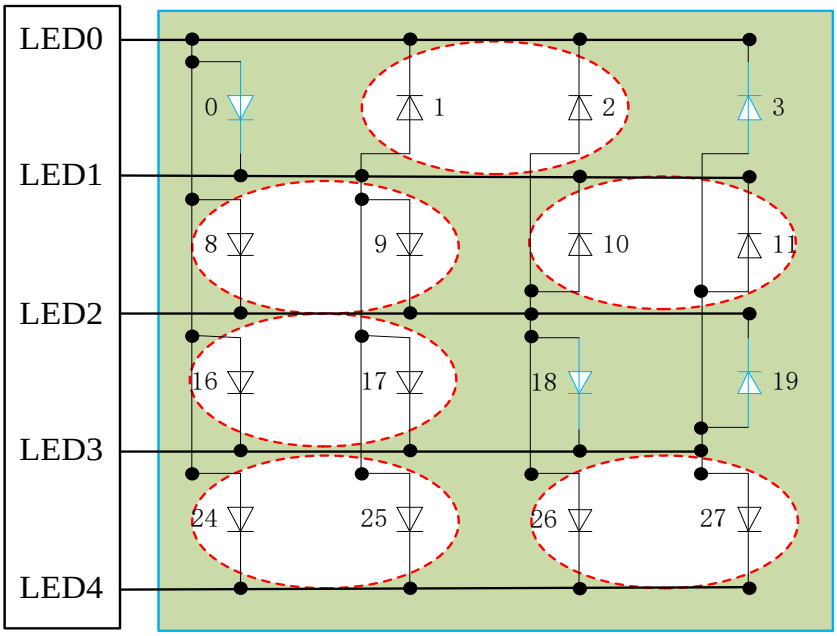
LED 6*6 lattice

5*5 lattice:



LED 5*5 lattice

4*4 lattice:

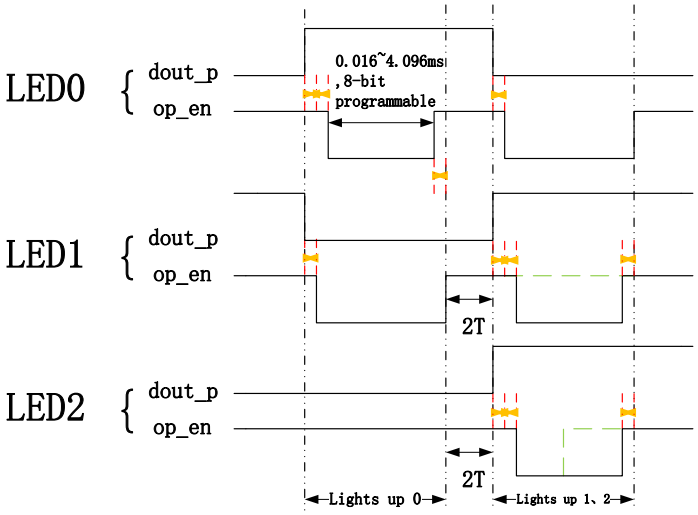


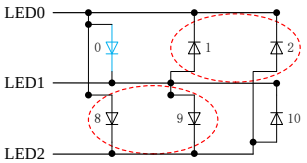
LED 4*4 lattice

Dot matrix scan timing example:

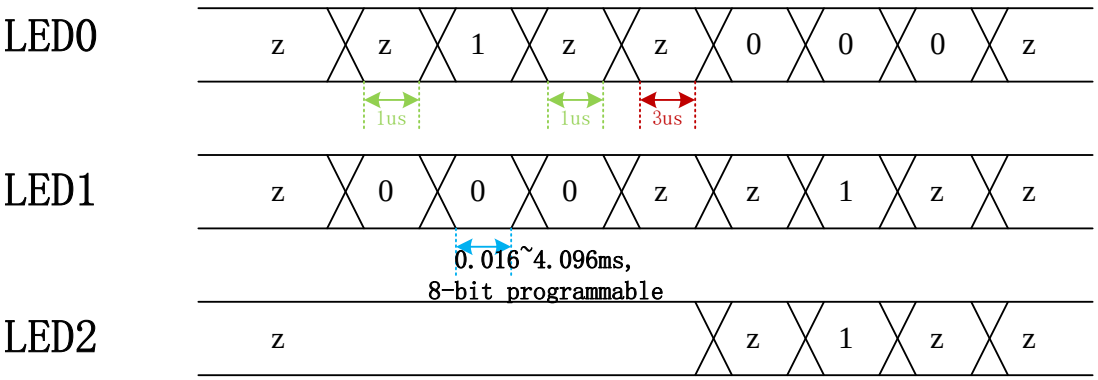
Take the lighting lamp 0, 1, 2 as an example, the timing is shown below:

- **One T，T is 1us**





The state of the IO port is as follows:



14.2. LED Related Register

SFR register				
Address	**Name**	**RW**	**Reset Value**	**Function description**
0xAF	SCAN_START	RW	0x00	LED scan open register
0XB0	DP_CON	RW	0x00	LED scan control register
0XB1	SCAN_WIDTH	RW	0x00	LED scan on time 1 control register
0xB2	LED2_WIDTH	RW	0x00	LED scan on time 2 control register
0xB3	LED_DRIVE	RW	0x00	LED drive capability configuration register

LED SFR register list

14.3. LED Register Details

SCAN_START(AFH) LED scan open register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	-
R/W	-	-	-	-	-	-	-	R/W
Reset value	-	-	-	-	-	-	-	0

DP_CON (B0H) LED scan control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	DUTY_SEL			SCAN_MODE	COM_MOD
R/W	-	-	-	R/W			R/W	R/W
Reset value	-	-	-	0	0	0	O	0

Bit number	Bit symbol	Description
4~2	DUTY_SEL	LED port drive mode matrix selection configuration register. 0: no matrix 1: 4x4 matrix (LED0~LED4) 2: 5x5 matrix (LED0~LED5) 3: 6x6 matrix (LED0~LED6) 4: 6x7 matrix (LED0~LED7) 5: 7x7 matrix (LED0~LED7) 6: 7x8 matrix (LED0~LED7) 7: 8x8 matrix (LED0~LED8)
1	SCAN_MODE	LED scan mode. 1: cycle scan mode 0: interrupt scan mode
0	COM_MOD	Large sink current ports drive enable. 1: COM port function lock, work as a large current IO port. 0: COM port function is not locked and can be configured as other functions. When the COM port locks the large sink current IO port, by configuring GPIO registers output drive timing, it is vaild when all of the following LED scan configurations are invalid.

SCAN_WIDTH (B1H) LED scan on time 1 control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	LED dot matrix drive mode, corresponding to a signal lamp lighting time configuration register——on time 1 configuration period=(scan_width+1)*16us, support configuration range 0.016~4.096ms.

LED2_WIDTH (B2H) LED scan on time 2 control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	LED dot matrix drive mode, corresponding to a signal lamp lighting time configuration register——on time 2 configuration period=(led2_width+1)*16us, support configuration range 0.016~4.096ms.

LED2_DRIVE (B3H) LED drive capability configuration register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-			
R/W	-	-	-	-	R/W			
Reset value	-	-	-	-	0	0	0	0

Bit number	Bit symbol	Description
7~0	--	LED port drive capability configuration register 0~15—3.77mA~69.14mA, please refer to LED drive ammeter for details.

14.4. LED Serial Dot Matrix Drive lamp voltage drop Description

(Ta = 27℃, VCC = 5V, LED lamp voltage drop 1.8V~2.3V)

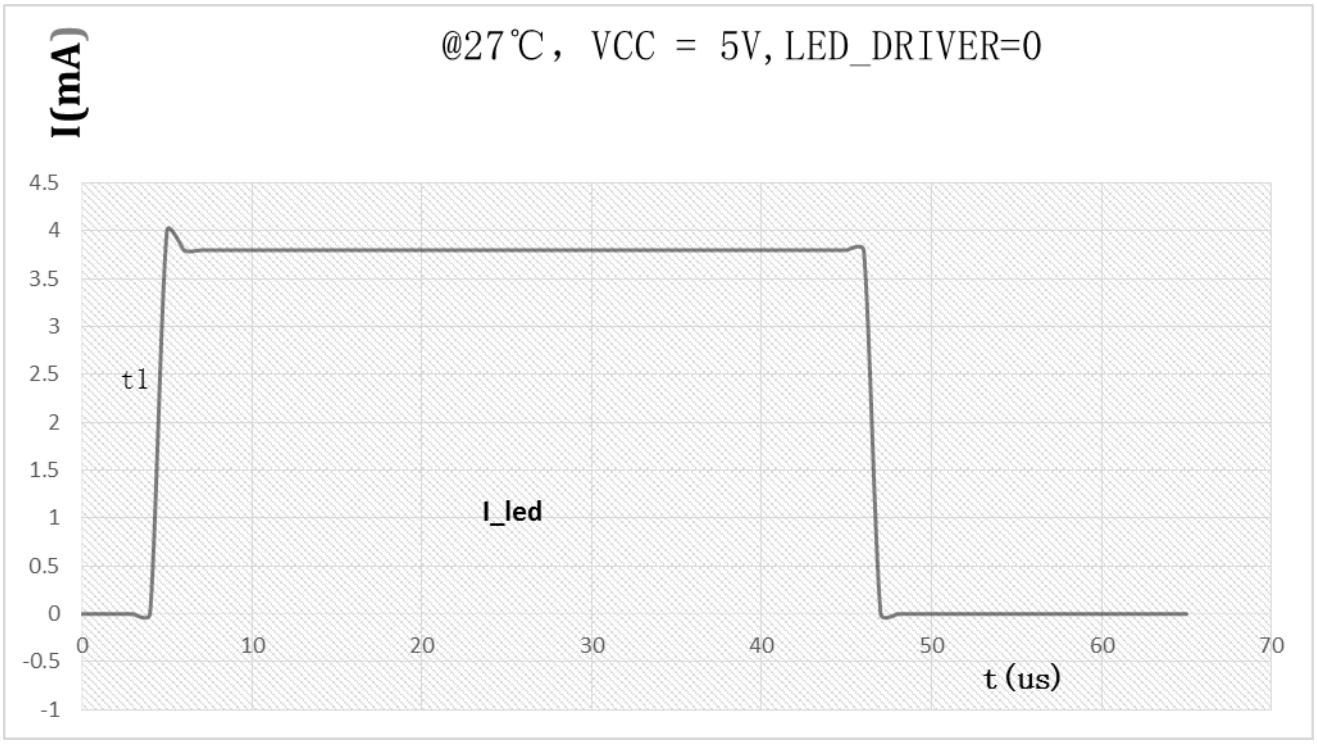
LED_DRIVE	I_led current (mA)
0	3.8
1	8.5
2	13.1
3	17.7
4	22.2
5	26.7
6	31.0
7	35.4
8	39.8
9	44.0
10	48.3
11	52.6
12	56.8
13	61.0
14	65.0
15	69.1

LED drive current register list

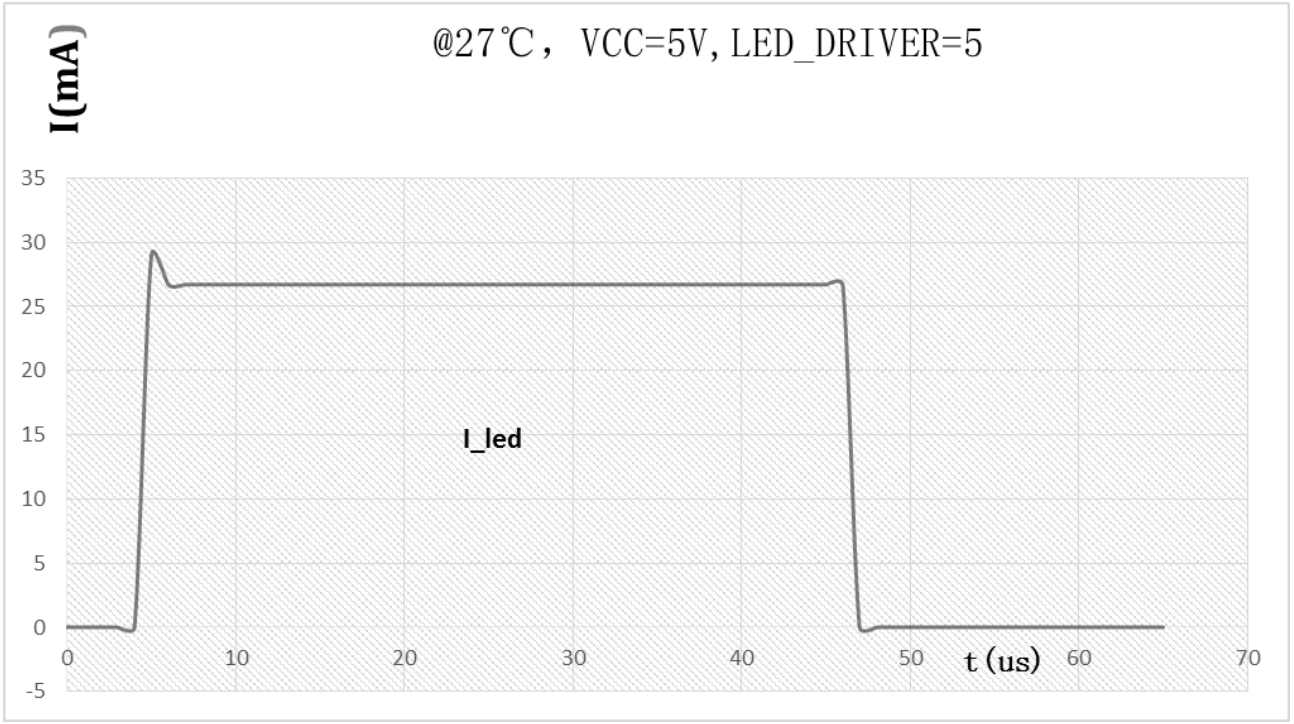
Notes:

1. **LED drive current deviation range(±8%)@VCC=5V, Ta=(-40℃~105℃), The setting of the LED_DRIVE is recommended to be smaller than the nominal Ifp of the LED lamp. The LED lamp to be driven should select the LED lamp with the same forward voltage V_F.**
2. **LED_DRIVE：LED drive capability configuration register;**
   I_led: LED lamp conducts steady state current.

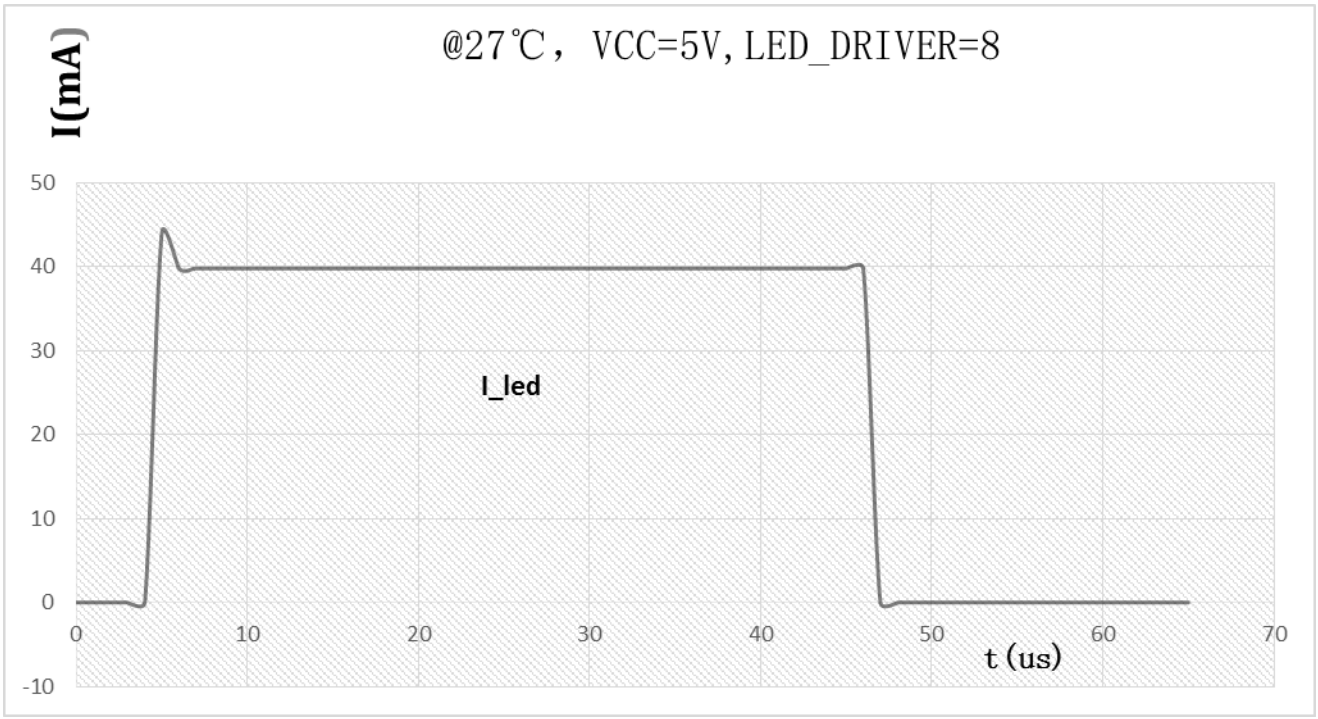
LED serial dot matrix drive current-time diagram under several common configurations:



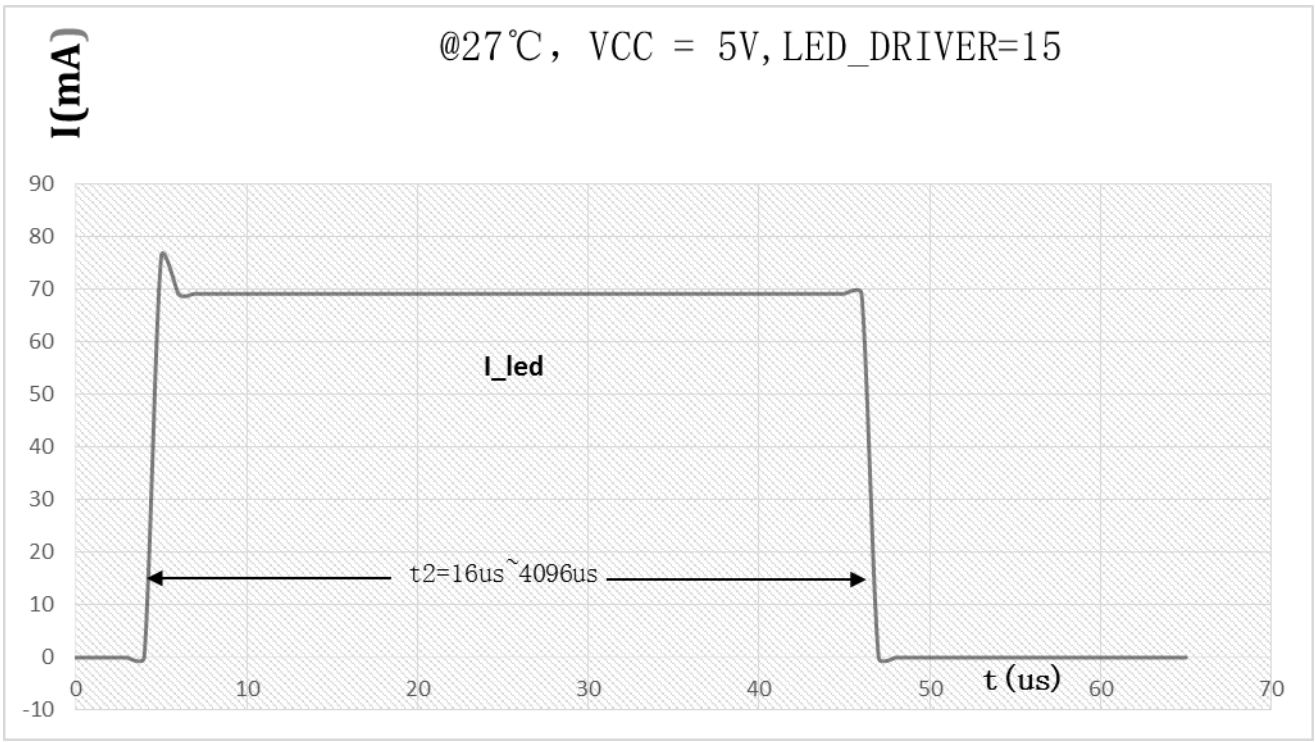
LED_DRIVER VS Time Figure1



LED_DRIVER VS Time Figure2



LED_DRIVER VS Time Figure3



LED_DRIVER VS Time Figure4

14.5. Display Configuration Address

LED dot matrix drive mode corresponding to display configuration:

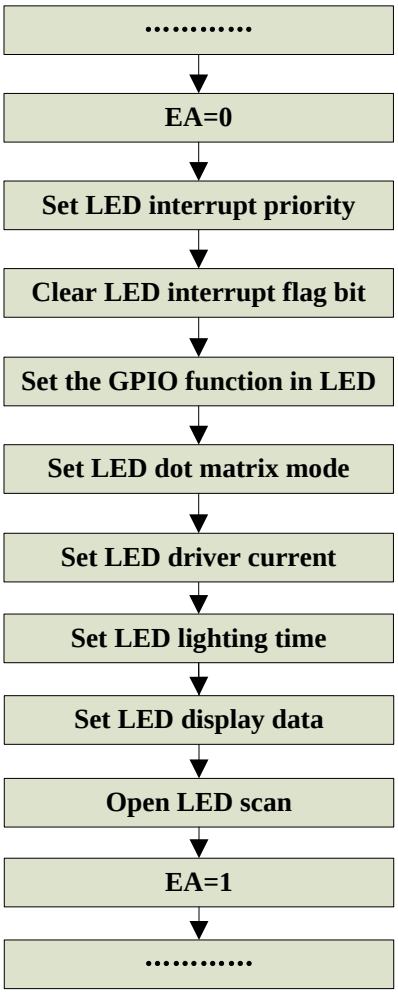
Dx indicates whether the light is selected or not, 0: not bright, 1: bright;

Dx_SEL indicates that the light is selected for the lighting cycle, 0: select the first segment of the light cycle, 1: select the second segment of the light cycle.

Address	7	6	5	4	3	2	1	0
200H	D7	D6	D5	D4	D3	D2	D1	D0
201H	D15	D14	D13	D12	D11	D10	D9	D8
202H	D23	D22	D21	D20	D19	D18	D17	D16
203H	D31	D30	D29	D28	D27	D26	D25	D24
204H	D39	D38	D37	D36	D35	D34	D33	D32
205H	D47	D46	D45	D44	D43	D42	D41	D40
206H	D55	D54	D53	D52	D51	D50	D49	D48
207H	D63	D62	D61	D60	D59	D58	D57	D56
208H	D7_SEL	D6_SEL	D5_SEL	D4_SEL	D3_SEL	D2_SEL	D1_SEL	D0_SEL
209H	D15_SEL	D14_SEL	D13_SEL	D12_SEL	D11_SEL	D10_SEL	D9_SEL	D8_SEL
20AH	D23_SEL	D22_SEL	D21_SEL	D20_SEL	D19_SEL	D18_SEL	D17_SEL	D16_SEL
20BH	D31_SEL	D30_SEL	D29_SEL	D28_SEL	D27_SEL	D26_SEL	D25_SEL	D24_SEL
20CH	D39_SEL	D38_SEL	D37_SEL	D36_SEL	D35_SEL	D34_SEL	D33_SEL	D32_SEL
20DH	D47_SEL	D46_SEL	D45_SEL	D44_SEL	D43_SEL	D42_SEL	D41_SEL	D40_SEL
20EH	D55_SEL	D54_SEL	D53_SEL	D52_SEL	D51_SEL	D50_SEL	D49_SEL	D48_SEL
20FH	D63_SEL	D62_SEL	D61_SEL	D60_SEL	D59_SEL	D58_SEL	D57_SEL	D56_SEL

LED dot matrix drive mode table

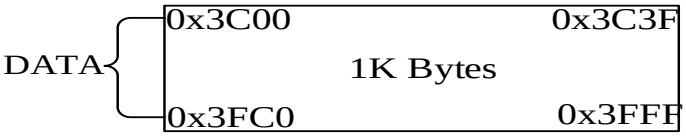
14.6. LED Configure Process



LED configure process

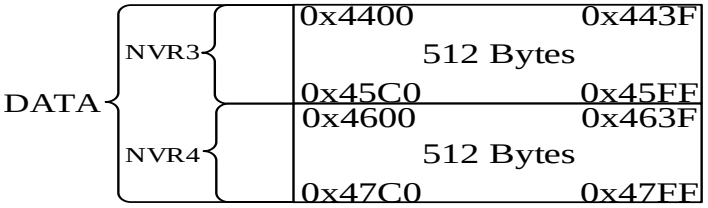
15. DATA

When the secondary bus register EEP_SELECT = 0: the size of the DATA area is 1024 Bytes for one page, and the address is (0x3C00~0x3FFF). When using it, it needs to be page erased, and then the byte write operation can only be written after erasing. Enter once.



{SPROG_ADDR_H[1:0],SPROG_ADDR_L[7:0]} The logical address (0~1023) corresponds to the physical address (0x3C00~0x3FFF).

When the secondary bus register EEP_SELECT = 1: select NVR3/4, 512 Bytes is one page.



When NVR3:SPROG_ADDR_H[0] = 0,

{SPROG_ADDR_H[0], SPROG_ADDR_L[7:0]} The logical address (0~511) corresponds to the physical address (0x4400~0x45FF).

When NVR4:SPROG_ADDR_H[0] = 1,

{SPROG_ADDR_H[0], SPROG_ADDR_L[7:0]} The logical address (0~511) corresponds to the physical address (0x4600~0x47FF).

15.1. Registers

SFR register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0xF9	SPROG_ADDR_H	RW	0x00	Address control register
0xFA	SPROG_ADDR_L	RW	0x00	Address control register
0xFB	SPROG_DATA	RW	0x00	Data register
0xFC	SPROG_CMD	RW	0x00	Command register
0xFD	SPROG_TIM	RW	0x5A	Erase time control register

Secondary bus register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0x20	EEP_SELECT	RW	0x00	DATA area selection register

15.2. Register Detailed Description

SPROG_ADDR_H (F9H) Address Control Register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-		
R/W	-	-	-	-	-	R/W		
Reset value	-	-	-	-	-	0	0	0

Bit number	Bit symbol	Description
2~0	--	When EEP_SELECT = 0, Bit[2]: DATA area selection enable, 0: Select 0x3C00~0x3FFF; 1: Reserved. {SPROG_ADDR_H[1:0], SPROG_ADDR_L[7:0]} means 0x3C00~0x3FFF address When EEP_SELECT = 1, Bit[2] = 0, select NVR3 (512Bytes); Bit[2] = 1, select NVR4 (512Bytes) {SPROG_ADDR_H[0], SPROG_ADDR_L[7:0]} represents the byte address within the page Bit[1]: reserved;

SPROG_ADDR_L(FAH)Address register,lower 8 bits

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Lower 8 bits of address

SPROG_DATA(FBH) Data register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Data to be written

SPROG_CMD(FCH) Command register

Bit number	7	6	5	4	3	2	1	0
Symbol	-							
R/W	R/W							
Reset value	0							

Bit number	Bit symbol	Description
7~0	--	Write 0x96: page erase; Write 0x69: byte burning.

SPROG_TIM(FDH) Erase time control register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset value	0	1	0	1	1	0	1	0

Bit number	Bit symbol	Description
7~5	--	Byte write time is fixed at 23.5us
4~0	SPROG_TIM[4:0]	When EEP_SELECT=0, bit[4:0]: 0~9 corresponds to the erasing time (1~10ms) + 0.13ms (step 1ms), >9 is 10.13ms. When EEP_SELECT=1, bit[4:0]: 0~9 corresponds to erasing time (0.5~5ms) + 0.065ms (step 0.5ms), and when >9, it is 5.065ms.

EEP_SELECT(20H)DATA area selection register

Bit number	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	-	-
R/W	-	-	-	-	-	-	-	R/W
Reset value	-	-	-	-	-	-	-	0

Bit number	Bit symbol	Description
7~1	--	Reserved
0	--	1: select NVR3/4 as DATA area. NVR3, 1 page, 512 Bytes; NVR4, 1 page, 512 Bytes 0: select DATA area (0x3C00~0x3FFF), 1 page, 1024 Bytes

15.3. Page Erase Step

EEP_SELECT = 0, select operation (0x3C00~0x3FFF), 1*1024 bytes.

EEP_SELECT = 1, select NVR3/NVR4, 2*512 bytes.

1. SPROG_TIM[4:0] = 0~9(suggest 5ms), byte write time is fixed at 23.5us, The main() program function is only configured once.
2. Close interrupt;
3. Configuration SPROG_ADDR_L = 0x00;
4. Configuration SPROG_ADDR_H = 0x00; select to erase the page;
5. Configuration SPROG_CMD = 0x96;
6. Write 4 NOP instructions;
7. Start erasing, the CPU turns off the Fsys_clk clock, and then turns on the Fsys_clk clock after completion;
8. Need to continue to erase data, jump to step 3;
9. Configuration SPROG_ADDR_L=0x00, SPROG_ADDR_H=0x00, restore interrupt settings.

15.4. Byte Write Step

EEP_SELECT = 0, select operation (0x3C00~0x3FFF), 1*1024 bytes.

EEP_SELECT = 1, select operation NVR3/NVR4, 2*512 bytes.

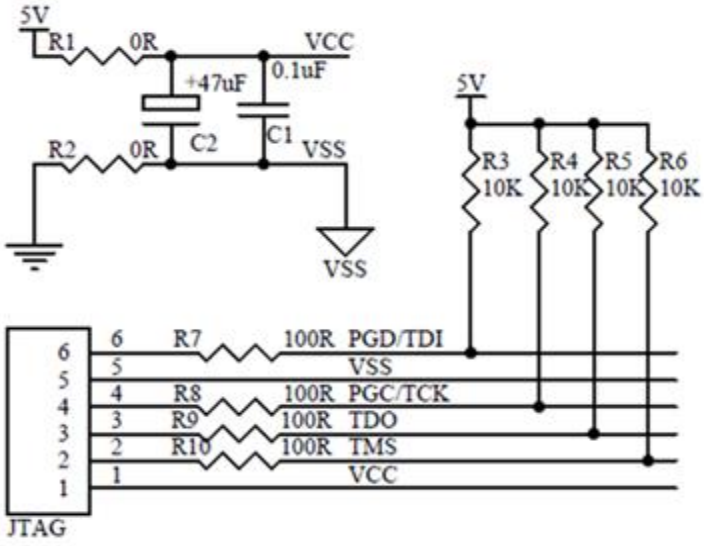
1. SPROG_TIM[4:0] = 0~9(suggest 5ms), byte write time is fixed at 23.5us, The main() program function is only configured once.
2. Close interrupt;
3. Configuration SPROG_ADDR_H, SPROG_ADDR_L, byte write address;
4. Configuration SPROG_DATA ;
5. Configuration SPROG_CMD = 0x69;
6. Write 4 NOP instructions;
7. Start writing, the CPU turns off the Fsys_clk clock, and then turns on the Fsys_clk clock after completion;
8. Need to continue to write data, jump to step 3;
9. Configuration SPROG_ADDR_L=0x00, SPROG_ADDR_H=0x00, restore interrupt settings.

Note: It is strongly not recommended that the DATA area address (0x3C00~0x3FFF) be stored as the user CODE.

16. Burning And Debugging

16.1. JTAG Circuit Connection

When debugging, you need to connect the TDI(PGD), TCK(PGC), TMS, TDO, VCC, VSS. In JTAG debug mode, the function of the JTAG port is blocked. It is not recommended to operate other functions that configure the JTAG debug I/O port to avoid affecting the JTAG debug function. Only four lines of TDI(PGD), TCK(PGC), VCC and VSS are connected during programming.



JTAG circuit connection

17. CPU Instruction System

17.1. Instruction Code

BF7512CMXX-SJLX instructions are divided into signal-byte instructions, double-byte instructions and three-byte instructions.

Signal-byte instructions: A signal-byte instruction consists of 8 bit binary code. There are only instruction opcodes in the instruction, no instruction operand or instruction operand is implied in the instruction opcode. There are 49 such instructions.

Double-byte instructions: Consists of two bytes, one for opcode and the other for the operand (or operand address), stored in order in program memory. There are 46 such instructions.

Three-byte instructions: Consists of one byte of instruction opcode and two bytes of operands (or operand address). There are 16 such instructions.

17.2. Instruction Set

In order to describe the instructions conveniently, some symbols are used in the instructions. The meanings of these symbols are as follows:

addr 11	Lower 11 bits of address
addr 16	16-bit address
direct	Direct addressing, 8-bit internal data and address (including special function registers)
bit	Bit address
#data	8-bit immediate
#data16	16-bit immediate
rel	Signed 8-bit relative displacement
n	Numbers 0~7
Rn	R0~R7 working registers of the current register group
i	Numbers 0, 1
Ri	Working registers R0, R1
@	Register indirect addressing
←	Data transfer direction
∧	Logical "and"
∨	Logical "or"
⊕	Logical "XOR"
√	Affects the flag bit
×	No effect on flags

CPU instruction symbol meaning table

The assembly instructions used, the function of each instruction, the number of bytes occupied, the instruction execution cycle and the impact on the corresponding flag bits are shown in the following table:

8 bit data transfer instructiion								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
MOV A	Rn	A←(Rn)	√	×	×	×	1	1
	direct	A←(direct)	√	×	×	×	2	1
	@Ri	A←((Ri))	√	×	×	×	1	1
	#data	A←data	√	×	×	×	2	1
MOV Rn	A	Rn←(A)	×	×	×	×	1	1
	direct	Rn←(direct)	×	×	×	×	2	2
	#data	Rn←data	×	×	×	×	2	1
MOV direct1	A	direct1←(A)	×	×	×	×	2	1
	Rn	direct1←(Rn)	×	×	×	×	2	1
	direct2	direct1←(direct2)	×	×	×	×	3	2

MOV direct,	@Ri	direct←((Ri))	×	×	×	×	2	2
	#data	direct←data	×	×	×	×	3	1
MOV @Ri	A	(Ri)←(A)	×	×	×	×	1	1
	direct	(Ri)←(direct)	×	×	×	×	2	2
	#data	(Ri)←data	×	×	×	×	2	1
16 bit data transfer instruction								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
MOV DPTR,#data16		DPTR←data16	×	×	×	×	3	1
External data transfer and table lookup instructions								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
MOVX @DPTR,A		(DPTR)←(A)	×	×	×	×	1	1
MOVC A,	@A+DPTR	A←((A)+(DPTR))	√	×	×	×	1	1
	@A+PC	A←((A)+(PC))	√	×	×	×	1	1
MOVX A,	@DPTR	A←(DPTR)	√	×	×	×	1	1
Notes: The number of cycles and the number of bytes of the MOVX instruction can be configured through registers CKCON<2:0>.								
Exchange class instruction								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
XCH A,	Rn	(Rn)←(A)	√	×	×	×	1	1
	direct	(A)←(direct)	√	×	×	×	2	2
	@Ri	(A)←((Ri))	×	×	×	×	1	2
XCHD A,@Ri		(A)3~0~((Ri))3~0	√	×	×	×	1	2
SWAP A		(A)7-4~(A)3-0	√	×	×	×	1	1
Arithmetic operation instruction								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
ADD A	Rn	A←(A)+(Rn)	√	√	√	√	1	1
	direct	A←(A)+(direct)	√	√	√	√	2	2
	@Ri	A←(A)+((Ri))	√	√	√	√	1	2
	#data	A←(A)+data	√	√	√	√	2	1
ADDC A	Rn	A←(A)+(Rn)+(C)	√	√	√	√	1	1
	direct	A←(A)+(direct)+(C)	√	√	√	√	2	2
	@Ri	A←(A)+((Ri))	√	√	√	√	1	2

		+(C)						
	#data	A←(A)+data+(C)	√	√	√	√	2	1
INC	A	A←(A)+1	√	×	×	×	1	1
	Rn	Rn←(Rn)+1	×	×	×	×	1	1
	direct	direct←(direct)+1	×	×	×	×	2	2
	@Ri	(Ri)←((Ri))+1	×	×	×	×	1	2
	DPTR	DPTR←((DPTR))+1	×	×	×	×	1	1
DA A		BCD code adjustment	√	×	√	√	1	1
SUBB A	Rn	A←(A)-(Rn)-(C)	√	×	×	×	1	1
	direct	A←(A)-(direct)-(C)	√	√	√	√	2	2
	@Ri	(A)←(A)-((Ri))-(C)	√	√	√	√	1	2
	#data	A←(A)-data-(C)	√	√	√	√	2	1
DEC	A	A←(A)-1	√	×	×	×	1	1
	Rn	Rn←(Rn)-1	×	×	×	×	1	1
	direct	direct←(direct)-1	×	×	×	×	2	2
	@Ri	(Ri)←((Ri))-1	×	×	×	×	1	2
MUL AB		BA←(A)*(B), after performing the multiplication operation, the lower byte is stored in A and the high byte is stored in B.	√	√	×	0	1	1
DIV AB		A←(A)/(B) B←remainder	√	√	×	0	1	1
Notes: When the DA instruction is used, the adjustment rules are as follows: if the low 4 bits of accumulator A are greater than 9 or AC=1, then A←A+06H; if the high 4 bits of accumulator A are greater than 9 or CY=1, then A←A+60H.								
Logical operation instruction								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
CLR A		A←00H	√	×	×	×	1	1
CPL A		A←($\overline{A}$)	√	×	×	×	1	1
ANL A,	Rn	A←(A)∧(Rn)	√	×	×	×	1	1
	direct	A←(A)∧(direct)	√	×	×	×	2	2

	@Ri	A←(A)∧((Ri))	√	×	×	×	1	2
	#data	A←(A)∧data	√	×	×	×	2	1
ANL direct,	A	direct←(A)∧(direct)	×	×	×	×	2	2
	#data	direct←(direct)∧data	×	×	×	×	3	2
ORL A,	Rn	A←(A)∨(Rn)	√	×	×	×	1	1
	direct	A←(A)∨(direct)	√	×	×	×	2	2
	@Ri	A←(A)∨((Ri))	√	×	×	×	1	2
	#data	A←(A)∨data	√	×	×	×	2	1
ORL direct,	A	direct←(direct)∨(A)	×	×	×	×	2	2
	#data	direct←(direct)∨data	×	×	×	×	3	2
XRL A,	Rn	A←(A)⊕(Rn)	√	×	×	×	1	1
	direct	A←(A)⊕(direct)	√	×	×	×	2	2
	@Ri	A←(A)⊕((Ri))	√	×	×	×	1	2
	#data	A←(A)⊕data	√	×	×	×	2	1
XRL direct,	A	direct←(direct)⊕(A)	×	×	×	×	2	2
	#data	direct←(direct)⊕data	×	×	×	×	3	2

Loop, shift class instruction

Mnemonic	Function	Impact on the flag				Number of bytes	Number of cycles
		P	OV	AC	CY		
RL A	The content in A is rotated left by one bit.	×	×	×	×	1	1
RLC A	A content with carry left shift one bit.	√	×	×	√	1	1
RR A	The content in A is rotated right by one bit.	×	×	×	×	1	1
RRC A	A content with carry right shift one bit.	√	×	×	√	1	1

Call, return class instruction

Mnemonic	Function	Impact on the flag				Number of bytes	Number of cycles
		P	OV	AC	CY		

LCALL addr16	(PC)←(PC)+3,(SP)←(PC), (PC)←addr16	×	×	×	×	3	2
ACALL addr11	(PC)←(PC)+2,(SP)←(PC), (PC10~0)←addr11	×	×	×	×	2	2
RET	(PC)←((SP))	×	×	×	×	1	2
RETI	(PC)←((SP)) return from interrupt	×	×	×	×	1	2
Transfer class instruction							
Mnemonic	Function	Impact on the flag				Number of bytes	Number of cycles
		P	OV	AC	CY		
LJMP addr16	PC←addr15~0	×	×	×	×	3	1
AJMP addr11	PC10~0←addr10~0	×	×	×	×	2	1
SJMP rel	PC←(PC)+rel	×	×	×	×	2	1
JMP @A+DPTR	PC←(A)+(DPTR)	×	×	×	×	1	1
JZ rel	PC←(PC)+2, If (A)=0, PC←(PC)+rel	×	×	×	×	2	2
JNZ rel	PC←(PC)+2, If (A)≠0, PC←(PC)+rel	×	×	×	×	2	2
JC rel	PC←(PC)+2, If (CY)=1, PC←(PC)+rel	×	×	×	×	2	2
JNC rel	PC←(PC)+2, If (CY)=0, PC←(PC)+rel	×	×	×	×	2	2
JB bit,rel	PC←(PC)+3, If (bit)=1, PC←(PC)+rel	×	×	×	×	3	2
JNB bit,rel	PC←(PC)+3, If (bit)=0, PC←(PC)+rel	×	×	×	×	3	2
JBC bit, rel	PC←(PC)+3, If (bit)=1,bit←0,	×	×	×	×	3	2

		PC←(PC)+rel						

CJNE	A, direct, rel	PC←(PC)+3,
		If (A) ≠direct,
		PC(PC)+rel
		If (A)<(direct),
		CY←1
			×	×	×	×	3	2

	A,#data,rel	PC←(PC)+3,
		If (A) ≠data,
		PC(PC)+rel
		If (A)<(data),
		CY←1
			×	×	×	×	3	2

	Rn,#data,rel	PC←(PC)+3,
		If (Rn) ≠data,
		PC←(PC)+rel
		If (Rn)<(data),
		CY←1
			×	×	×	×	3	1

	@Ri,#data,rel	PC←(PC)+3, if ((Ri)) ≠data, PC←(PC)+rel If ((Ri))<(data), CY←1	×	×	×	×	3	2
DJNZ	Rn,rel	PC←(PC)+2,Rn←(Rn)-1, If (Rn) ≠0, PC←(PC)+rel	×	×	×	×	2	1
	direct,rel	PC←(PC)+3, (direct)←(direct)-1, if (direct) ≠0, PC←(PC)+rel	×	×	×	×	3	2
Stack, empty operation class instruction								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		
PUSH	direct	SP←(SP)+1, (SP)←(direct)	×	×	×	×	2	2
POP	direct	direct←(SP), SP←(SP)-1	×	×	×	×	2	2
NOP		empty operation	×	×	×	×	1	1
Bit manipulation instruction								
Mnemonic		Function	Impact on the flag				Number of bytes	Number of cycles
			P	OV	AC	CY		

MOV	C, bit	CY←bit	×	×	×	√	2	2
	bit, C	bit←CY	×	×	×	×	2	2
CLR	C	CY←0	×	×	×	√	1	1
	bit	bit←0	×	×	×	×	2	2
SETB	C	CY←1	×	×	×	√	1	1
	bit	bit←1	×	×	×	×	2	2
CPL	C	CY←$(\overline{CY})$	×	×	×	√	1	1
	bit	bit←$(\overline{bit})$	×	×	×	×	2	2
ANL	C, bit	C←(C)∧(bit)	×	×	×	√	2	2
	C, /bit	C←(C)∧$(\overline{bit})$	×	×	×	√	2	2
ORL	C, bit	C←(C)∨(bit)	×	×	×	√	2	2
	C, /bit	C←(C)∨$(\overline{bit})$	×	×	×	√	2	2

Pseudo-instruction		
Mnemonic	Instruction format	Function description
ORG	【tab:】ORG addr16	Define the first address of tab
EQU	tab EQU data/tab	Assign values to labels
DB	【tab:】 DB item or item tabel	Define a-byte or multi-byte
DW	【tab:】 DW item or item tabel	16 bit word content used to define two or more cells in memory
DS	【tab:】DS expression	Specifies to leave several memory cells starting with the label
BIT	tab BIT address	Assign a bit address to a label
END	END is placed at the end of the assembly language program to tell the assembler that the source program ends here.	

CPU instruction symbol table

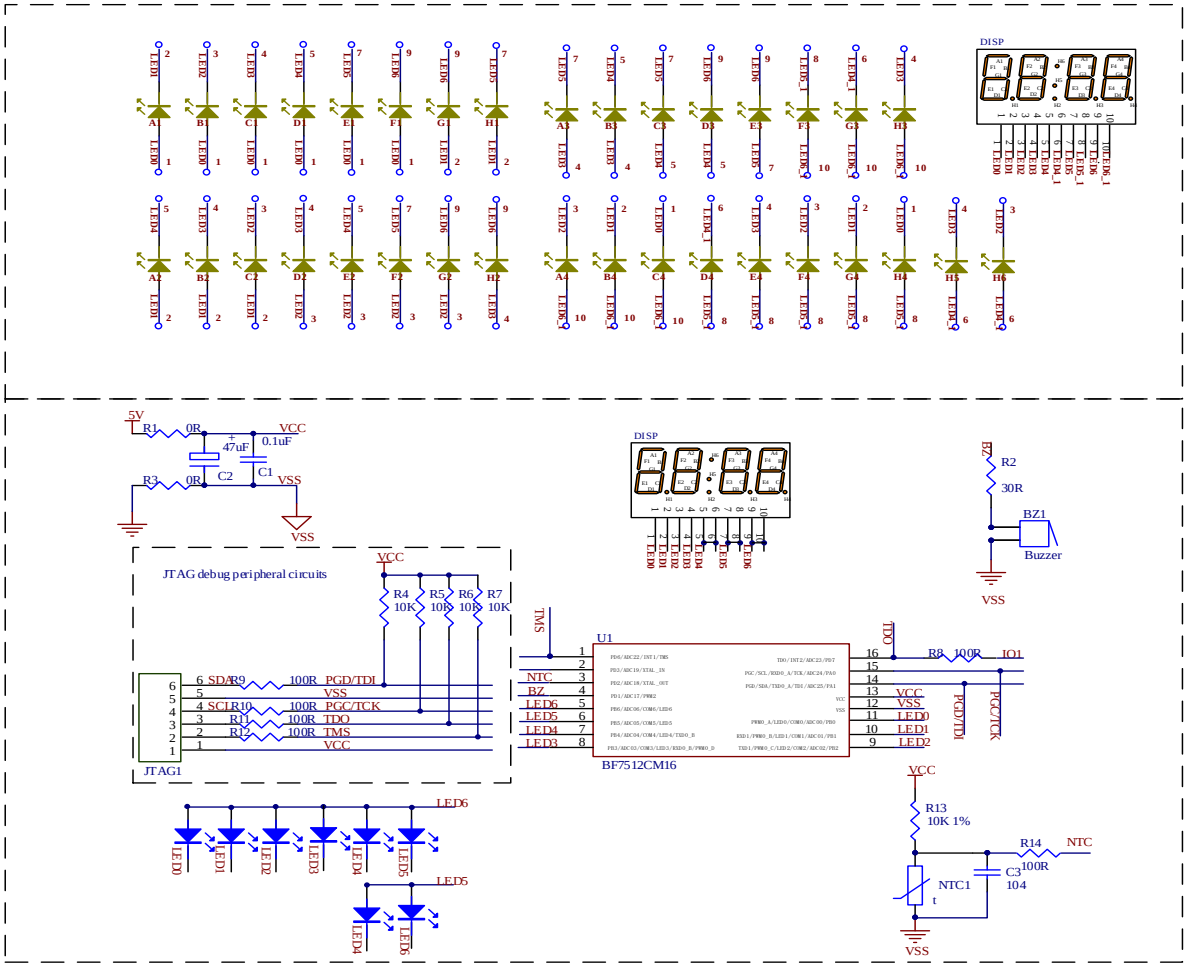
CPU related register

SFR register				
Address	**Name**	**RW**	**Reset value**	**Function description**
0x81	SP	RW	0x07	stack pointer register
0x82	DPL	RW	0x00	data pointer register 0 low 8 bit
0x83	DPH	RW	0x00	data pointer register 0 high 8 bit
0x87	PCON	RW	0x00	low power mode select register
0xE0	ACC	RW	0x00	accumulator
0xF0	B	RW	0x00	B register

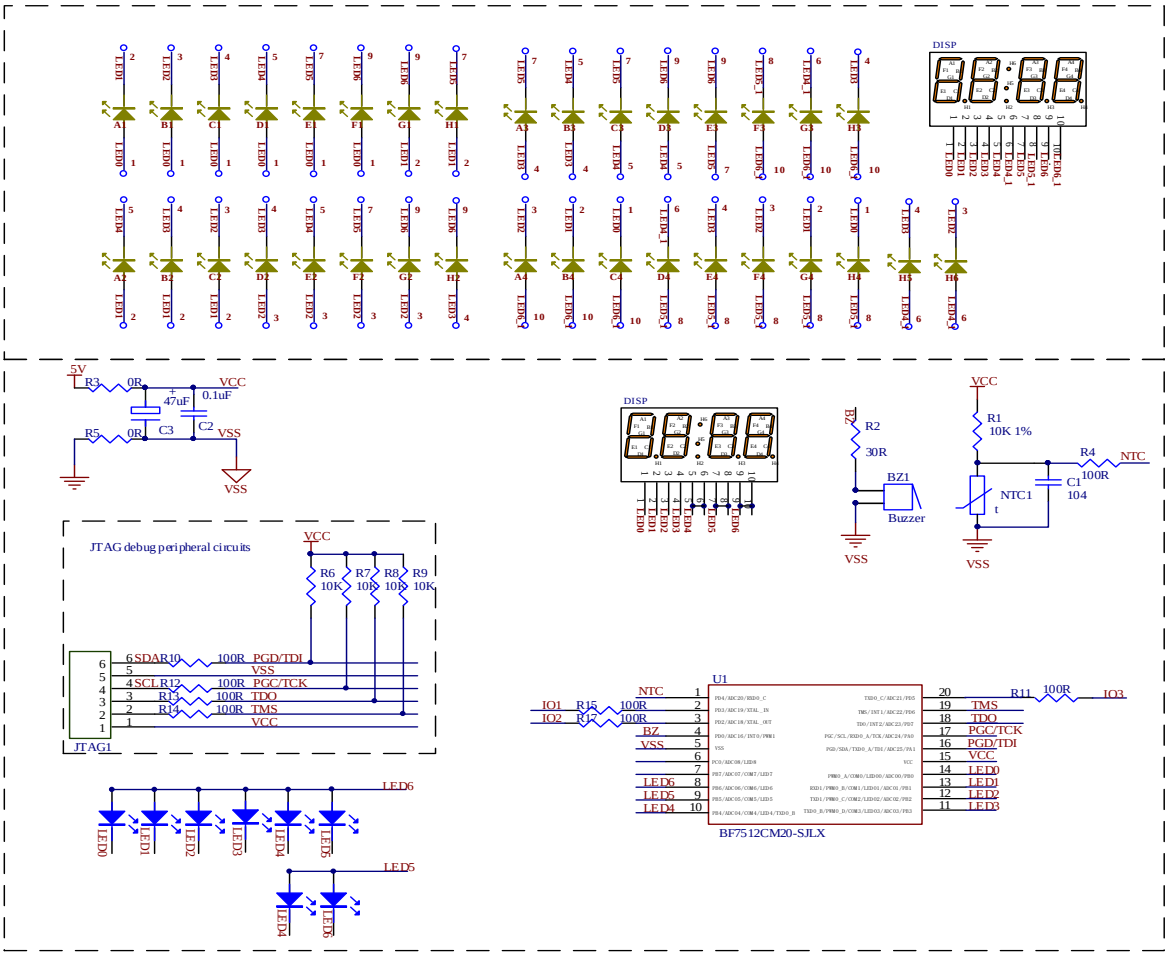
CPU SFR register list

18. Reference Application Circuits

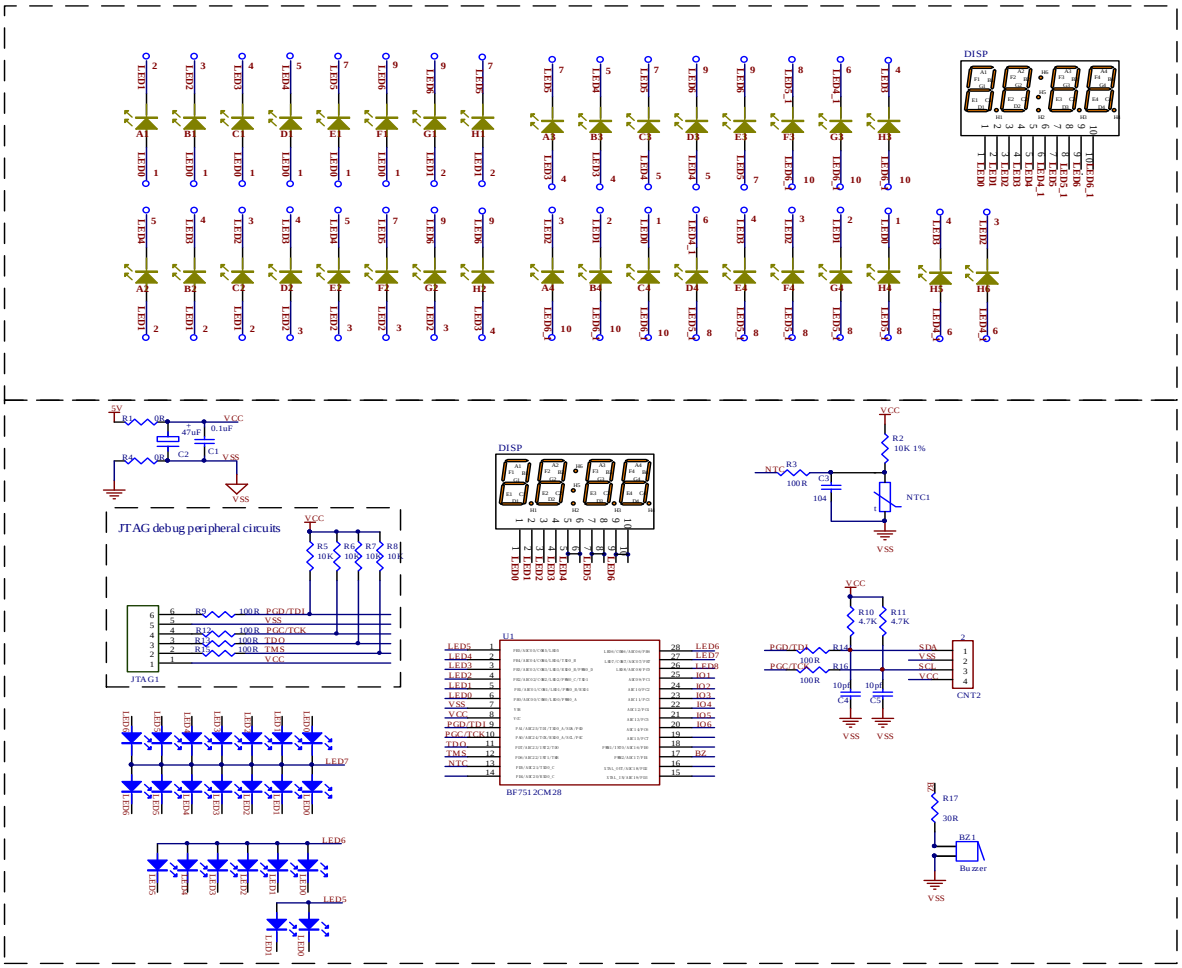
18.1. BF7512CM16-XXXX Reference Circuit



18.2. BF7512CM20-XXXX Reference Circuit



18.3. BF7512CM28-XXXX Reference Circuit

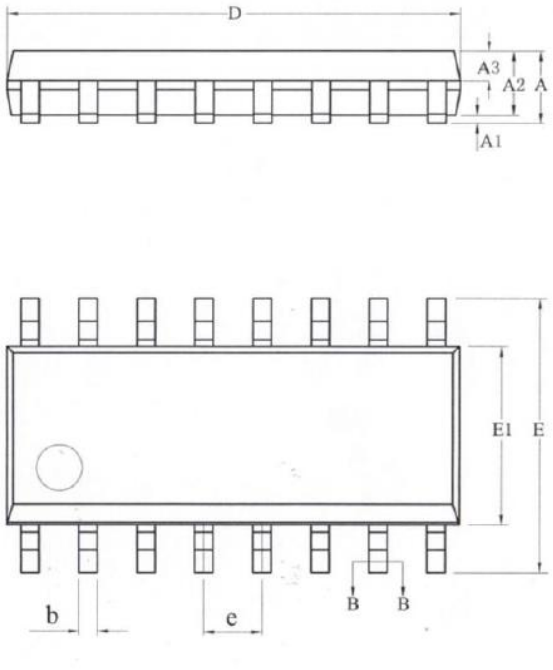


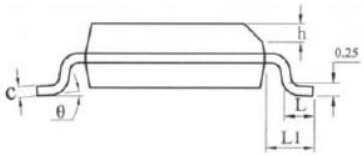
Note:

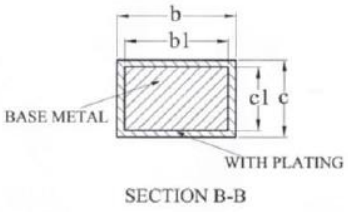
1. **The above schematic diagram is for reference only.**
2. **The JTAG debugging peripheral circuit is only used for JTAG debugging. If there is a pull-up resistor on the emulator or adapter board, there is no need to connect the JTAG pull-up resistor.**
3. **Replace the 0Ω resistors with parallel power and ground with magnetic beads. The EMI test item (RE) can increase the test margin. The recommended parameter is 600Ω@100MHz.**

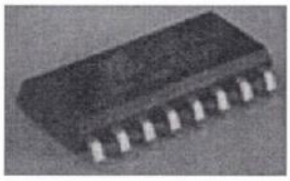
19. Packages

19.1. SOP16



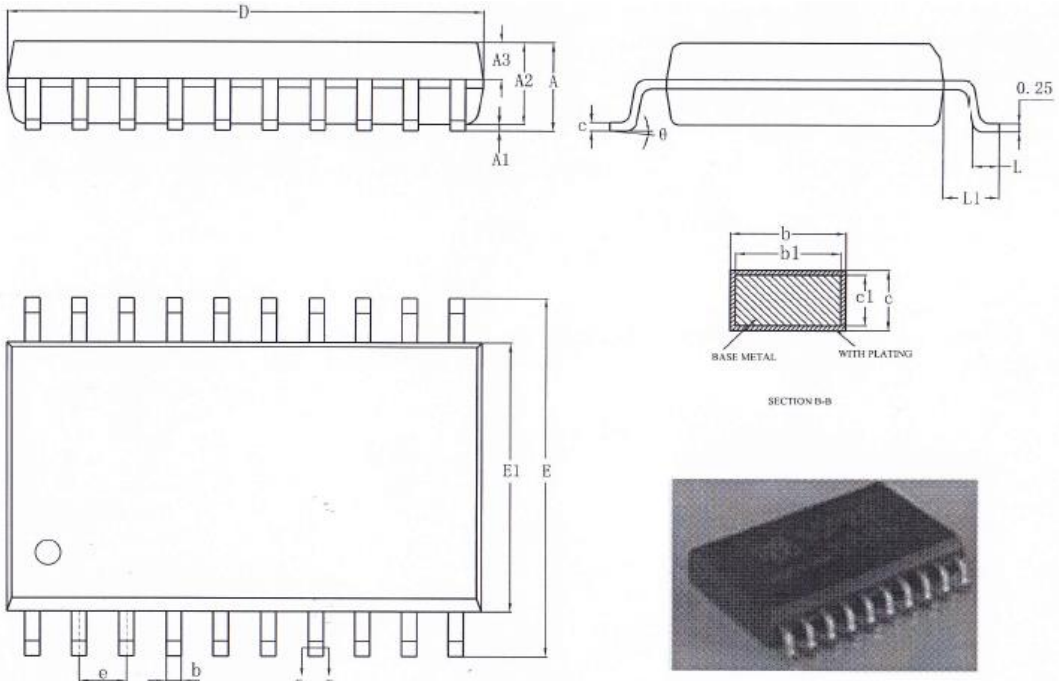






SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	—	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	—	0.48
b1	0.38	0.41	0.44
c	0.20	—	0.25
c1	0.19	0.20	0.21
D	9.80	9.90	10.00
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	1.27BSC		
h	0.25	—	0.50
L	0.50	—	0.80
L1	1.05REF		
θ	0	—	8°

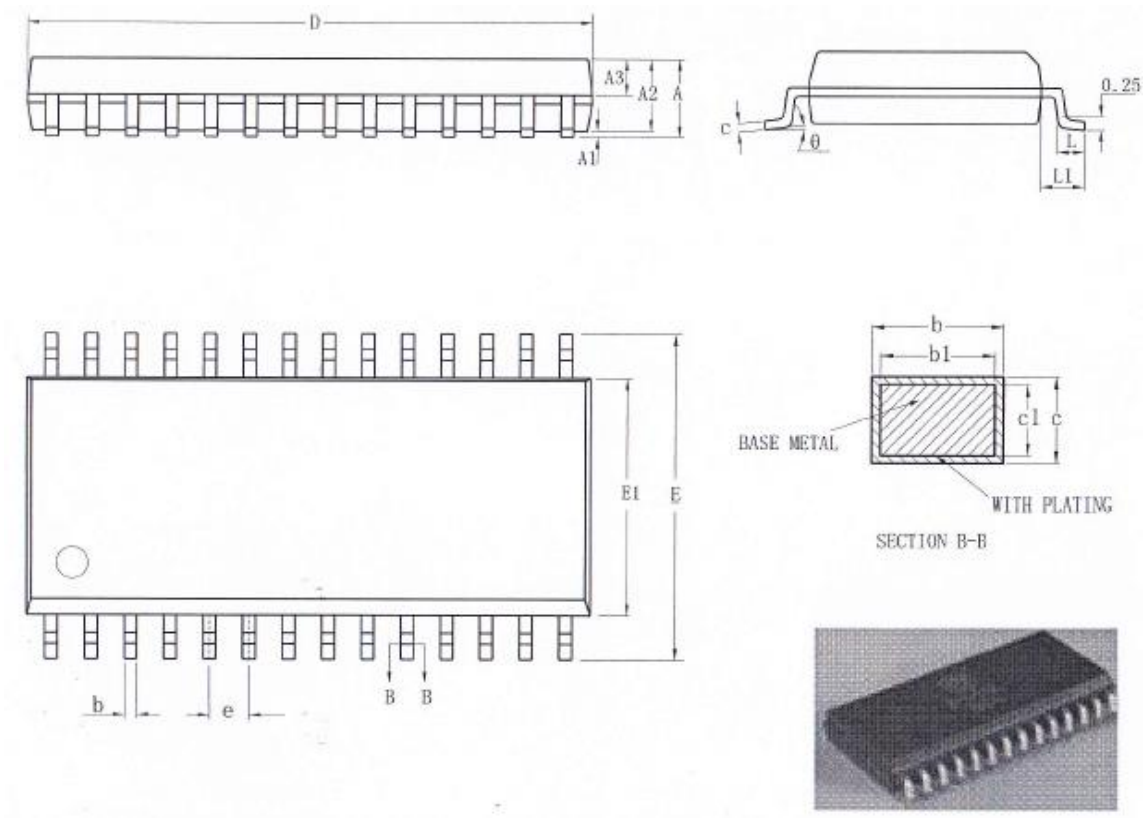
19.2. SOP20



SOP20 package

DIM	SOP020 MILLIMETERS		
	MIN	NOM	MAX
A	-	-	2.650
A1	0.100	0.200	0.300
A2	2.250	2.300	2.350
b	0.350	-	0.440
c	0.250	-	0.310
D	12.600	12.800	13.000
E1	7.300	7.500	7.700
E	10.100	10.300	10.500
e	1.270(BSC)		
L	0.7	-	1
θ	0 °	-	8 °
End face waste rubber	-	-	0.2
Total length of plastic body	12.800	13.000	13.300

19.3. SOP28

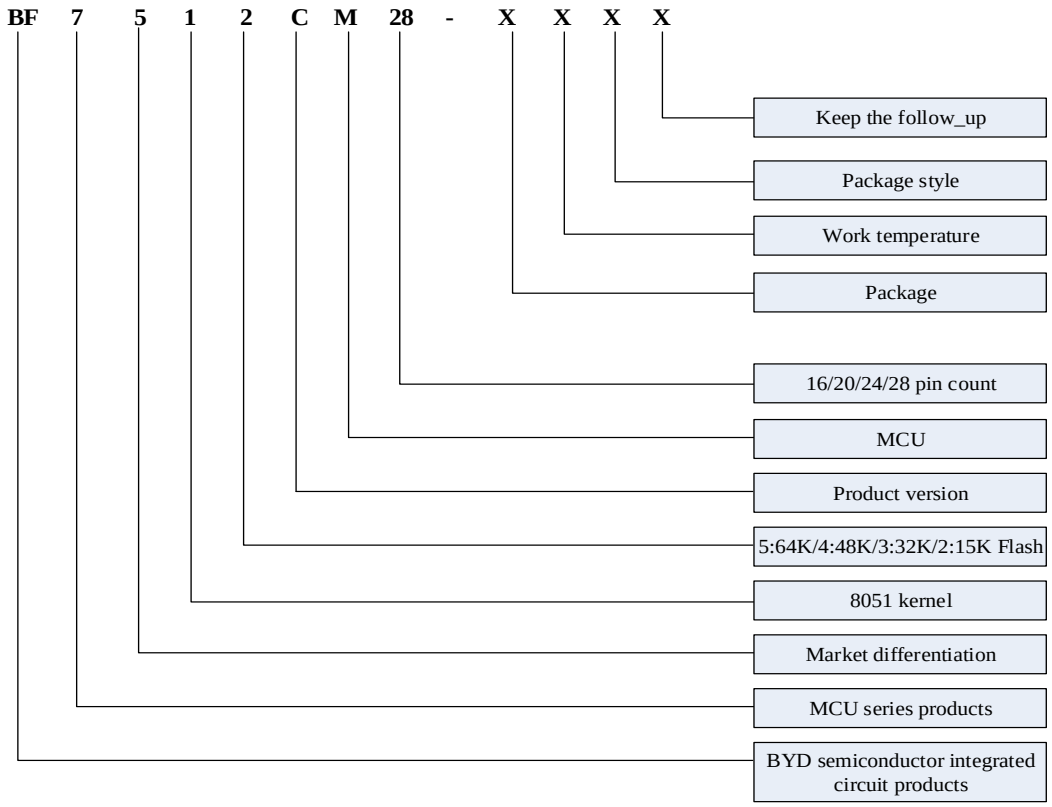


SOP28 package

DIM	SOP028 MILLIMETERS		
	MIN	NOM	MAX
A	2.250	2.400	2.650
A1	0.100	0.200	0.300
A2	2.250	2.300	2.350
b	0.300	0.425	0.480
c	0.250	0.285	0.310
D	17.800	18.000	18.200
E1	7.300	7.500	7.700
E	10.100	10.300	10.500
e	1.270(BSC)		
L	0.7	-	1
θ	0 °	-	8 °
End face waste rubber	-	-	0.2
Total length of plastic body	18.000	18.300	18.500

Ordering information

Package	Work temperature		Package style	Keep the follow-up
S: SOP	Car grade	A: -40℃~+150℃	B: tap	-
T: TSSOP		B: -40℃~+125℃	L: feed tube	-
M: MSSOP		C: -40℃~+105℃	T: tray	-
L: LQFP		D: -40℃~+85℃	-	-
Q: QFN	Industrial grade	K: -40℃~+85℃	-	-
B: BGA		J: -40℃~+105℃	-	-
D: DIP		L: -40℃~+125℃	-	-
-	Consumer grade	P: -25℃~+70℃	-	-
-		Q: 0℃~+70℃	-	-



Revision History

Revised date	Revised content	Reviser	Remarks
2020-11-19	V1.0	JX	V1.0
2021-11-25	1. Update chapter 5.1 Clock Block Diagram 2. Update Chapter 5.4 Working Mode 3. Update the FLASH memory chapter 4. Add secondary bus register 5. Add reset sequence 6. Update ADC, PWM chapter 7. Update the EEPROM-like page erasing and byte writing steps 8. Update feature introduction 9. Add BOR description 10. Update D2H description	YNN	V1.1
2022-03-09	1. Introduction to update features 2. Update the selection list 3. Update memory description 4. The name of "EEPROM-like" is updated to "DATA area" 5. Update the clock block diagram 6. Update the instruction set 7. Update the BYD logo	YNN	V1.2

Disclaimer

1. The information contained herein is subject to change without notice, see the revision record for details. Please contact FAE or the agent for the latest version.
2. BYD Semiconductor Co., Ltd. will do its utmost to ensure the high quality and high stability of the company's products. Nonetheless, due to the inherent characteristics of general semiconductor devices such as electrical sensitivity and vulnerability to external physical damage, our products may malfunction or fail under these circumstances. When using our company's products, users are responsible for designing a safe and stable system environment in compliance with safety rules. Users can avoid possible accidents, fires and public injuries by removing redundant devices, failure prevention and fire prevention measures. When the user uses the product, please follow the operating steps specified in the company's latest manual to use the product.
3. The products of BYD Semiconductor Co., Ltd. in this document are designed for general electrical applications (computers, personal tools, company tools, measuring tools, industrial mechanical devices, household appliances, etc.). The company's products can not be used in some special equipment that requires extremely high stability and quality, so as to avoid accidents such as casualties. The range of products that cannot be applied includes atomic energy control equipment, aircraft and aviation devices, transportation equipment, traffic signal equipment, combustion control equipment, medical equipment, and all safety equipment. The company shall not be responsible for any loss or injury caused by users who use it within the non-product application ranges listed above.