1 BF7807AMXX MCU general description

1.1. Features

- **Core: ARM Cortex-M0+**
  - Operating frequency: 48MHz, 32MHz, 24MHz, 12MHz
  - Clock error: ±1% @-20℃~65℃, 5V
    ±3% @-40℃~105℃, 5V
- **Memory (FLASH)**
  - FLASH: 128K Bytes, support erase and write protection and read protection functions
  - DATA: 512 Bytes
  - SRAM: 16K Bytes
- **Clock source, reset and power management**
  - Internal low-speed RC oscillator: LIRC 32kHz, clock error is ±10% @25℃, 5V, ±25% @-40℃ ~105℃, 5V
  - Internal high-speed RC oscillator: RC 1MHz
  - External crystal oscillator: XTAL 32768Hz/4MHz/8MHz
  - 8 kinds of resets, including power-down reset voltage (BOR): 2.8V/3.3V/3.7V/4.2V
  - Low voltage detection: 2.7V/3.0V/3.3V/3.6V/3.8V/4.0V/4.2V/4.4V
- **IO**
  - Both support built-in pull-up resistor 33k
  - High current sink port (PA0~PA7)
  - Support IO function remapping
  - Both support external interrupt function (rising edge, falling edge, double edge)
- **Communication module**
  - 5xUART communication module, support IO mapping
  - 1xIIC master-slave communication, both master and slave support 100kHz/400kHz/1MHz
  - 2xSPI master-slave communication, the master supports up to 8MHz, the slave supports up to 4MHz
- **16-Bit PWM**
  - PWM0/1 both support 5 channels, sharing period, duty cycle and polarity are configurable
  - PWM2/3 both support 1 channel and support mapping
  - PWM4 supports 1 channel
  - Support timing mode
- **Operating voltage: 2.7V ~5.5V**
- **Operating temperature: -40℃ ~105℃**
- **High precision 12-bit ADC**
  - Up to 59 analog input channels
  - Reference voltage: VCC/2V/4V
  - Single conversion mode
- **Interrupt**
  - 26 interrupt sources
- 4-level interrupt priority can be configured
- **Timer**
  - 16-bit Timer0/1/2/3
  - Timer2 clock source: LIRC 32k, XTAL 32768Hz/4MHz/8MHz
  - Watchdog timer, overflow time from 18ms to 2.304s
  - SysTick timer
- **LED Driver**
  - Support up to 8COM x 16SEG (1/8~8/8 duty cycle)
- **LCD Driver**
  - 4 COM x 28 SEG (1/4 duty cycle, 1/3 bias)
  - 5 COM x 27 SEG (1/5 duty cycle, 1/3 bias)
  - 6 COM x 26 SEG (1/6 duty cycle, 1/3 bias or 1/4 bias)
  - 8 COM x 24 SEG (1/8 duty cycle, 1/4 bias)
- **Touch key**
  - The sensitivity of each key can be set independently
  - Capacitive keys can be reused as GPIO
- **Low power management**
  - Idle mode 0, power consumption 1.7mA@5V typical
  - Idle mode 1, power consumption 8.0μA@5V typical
- **Cyclic redundancy check unit**
  - CRC8/16/32
- **96-bit/128-bit chip unique identification code**
- **Serial two-wire debugging interface SWD, PGC/PGD programming**
- **Package**
  - LQFP44/LQFP64

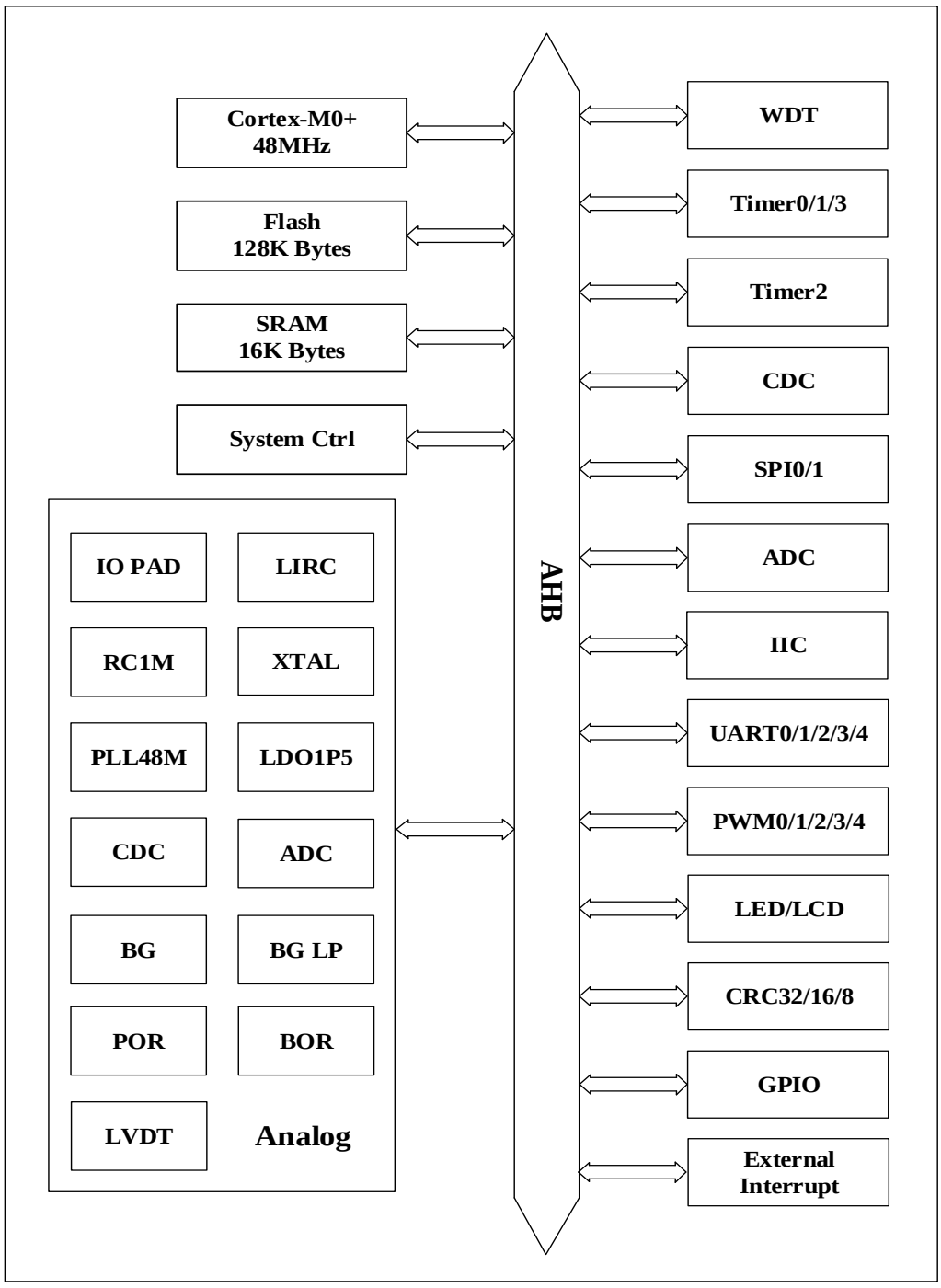
1.2. Overview

The BF7807AMXX adopts ARM Cortex-M0+ core, 32-bit high-performance microcontroller. The Cortex-M0+ core is based on the ARMv6-M architecture and supports the Thumb instruction set.

The BF7807AMXX includes peripheral watchdog, LED row and column matrix driver, LCD driver, capacitive touch key detection, IIC master-slave, SPI master-slave, multi-channel UART, PWM, Timer0, Timer1, Timer2, Timer3, 12bit successive approximation ADC, low-voltage detection, power-down reset, low-power management and other modules.

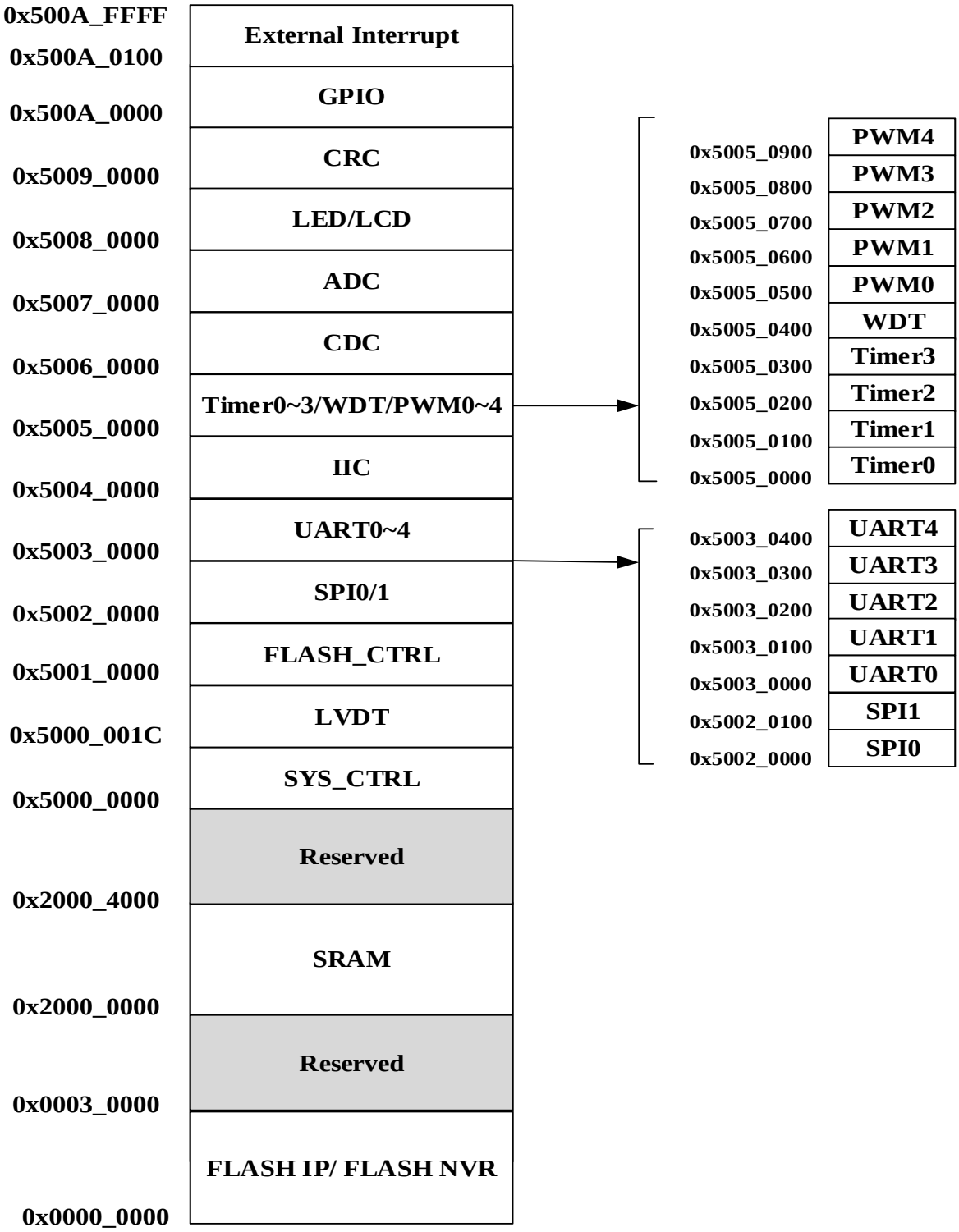
The BF7807AMXX integrates multiple capacitive detection channels, which can be used to detect proximity sensing or touch, and realize various applications such as keys, scroll wheels, and sliders. The BF7807AMXX can configure the corresponding function register to adjust the sensitivity of the capacitance detection channel.

1.3. System architecture



System architecture

1.4. Memory map

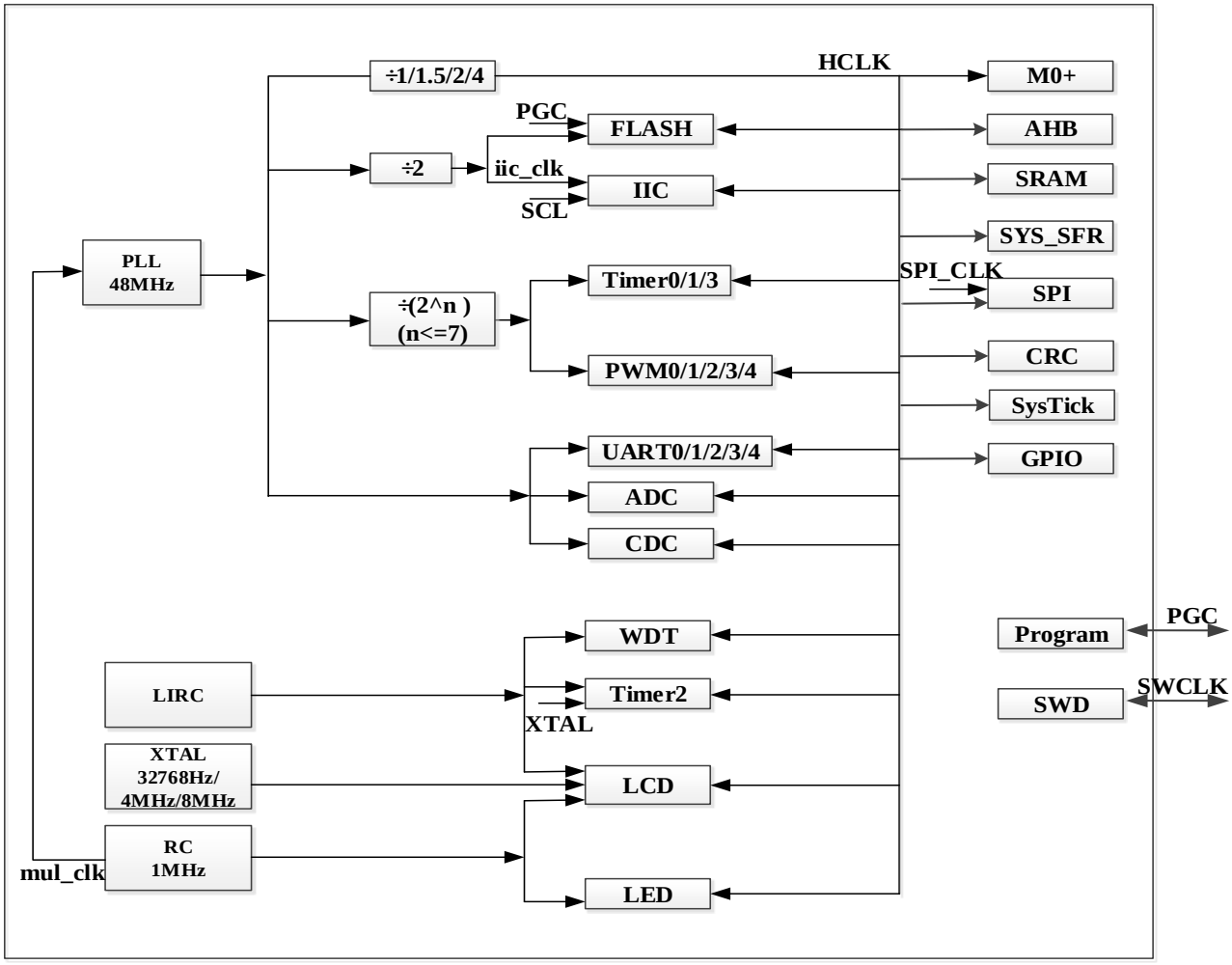


Address division map

Address range	Size(Bytes)	Module
0x0000_0000 ~ 0x0002_FFFF	192K	FLASH IP/FLASH NVR
0x2000_0000 ~ 0x2000_3FFF	16K	SRAM
0x5000_0000 ~ 0x5000_001B	64K	SYS_CTRL
0x5000_001C~ 0x5000_FFFF		LVDT
0x5001_0000 ~ 0x5001_FFFF	64K	FLASH_CTRL
0x5002_0000 ~ 0x5002_00FF	256	SPI0
0x5002_0100 ~ 0x5002_01FF	256	SPI1
0x5002_0200 ~ 0x5002_FFFF	-	Reserved
0x5003_0000 ~ 0x5003_00FF	256	UART0
0x5003_0100 ~ 0x5003_01FF	256	UART1
0x5003_0200 ~ 0x5003_02FF	256	UART2
0x5003_0300 ~ 0x5003_03FF	256	UART3
0x5003_0400 ~ 0x5003_04FF	256	UART4
0x5003_0500 ~ 0x5003_FFFF	-	Reserved
0x5004_0000 ~ 0x5004_FFFF	64K	IIC
0x5005_0000 ~ 0x5005_00FF	256	Timer0
0x5005_0100 ~ 0x5005_01FF	256	Timer1
0x5005_0200 ~ 0x5005_02FF	256	Timer2
0x5005_0300 ~ 0x5005_03FF	256	Timer3
0x5005_0400 ~ 0x5005_04FF	256	WDT
0x5005_0500 ~ 0x5005_05FF	256	PWM0
0x5005_0600 ~ 0x5005_06FF	256	PWM1
0x5005_0700 ~ 0x5005_07FF	256	PWM2
0x5005_0800 ~ 0x5005_08FF	256	PWM3
0x5005_0900 ~ 0x5005_09FF	256	PWM4
0x5005_0A00 ~ 0x5005_FFFF	-	Reserved
0x5006_0000 ~ 0x5006_FFFF	64K	CDC
0x5007_0000 ~ 0x5007_FFFF	64K	ADC
0x5008_0000 ~ 0x5008_FFFF	64K	LED/LCD
0x5009_0000 ~ 0x5009_FFFF	64K	CRC
0x500A_0000 ~ 0x500A_00FF	64K	GPIO
0x500A_0100 ~ 0x500A_FFFF		External Interrupt

Table Address division

1.5. Clock block diagram



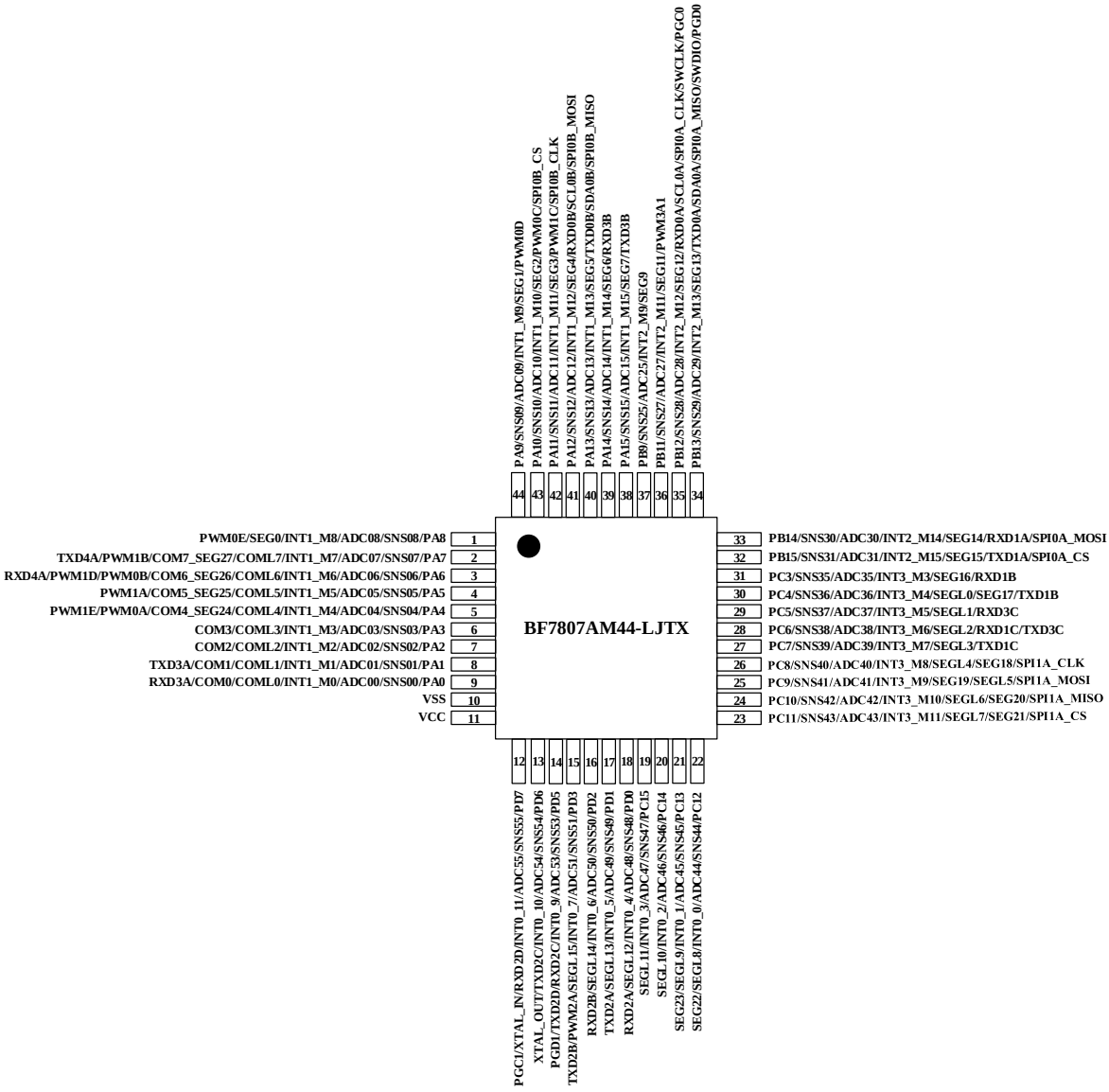
Clock block diagram

1.6. Selection list

Model		BF7807AM44-LJTX	BF7807AM64-LJTA	BF7807AM64-LJTX
Operating voltage (V)		2.7~5.5	2.7~5.5	2.7~5.5
Operating frequency (Hz)		48M	48M	48M
Core		ARM Cortex-M0+	ARM Cortex-M0+	ARM Cortex-M0+
Storage module (Bytes)	FLASH	128004B	128K	128K
	DATA	512	512	512
	SRAM	16K	16K	16K
Timer	WDT	1	1	1
	Timer0*16bit	1	1	1
	Timer1*16bit	1	1	1
	Timer2*16bit	1	1	1
	Timer3*16bit	1	1	1
Communication module	IIC	1	1	1
	UART	5	5	5
	SPI	2	2	2
GPIO		42	59	59
KEY		42	59	59
INT		42	59	59
COM		8	8	8
Analog module	ADC*12bit	42	59	59
Display module	LED ranks	8 COM x 16 SEG	8 COM x 16 SEG	8COM x 16SEG
	LCD	8 COM x 22 SEG	8 COM x 24 SEG	8COM x 24SEG
PWM module	PWM0*16bit	5	5	5
	PWM1*16bit	5	5	5
	PWM2*16bit	1	1	1
	PWM3*16bit	1	1	1
	PWM4*16bit	-	1	1
CRC		CRC8/16/32	CRC8/16/32	CRC8/16/32
Package		LQFP44 (10mm*10mm, e=0.8mm)	LQFP64 (10mm*10mm, e=0.5mm)	LQFP64 (14mm*14mm, e=0.8mm)

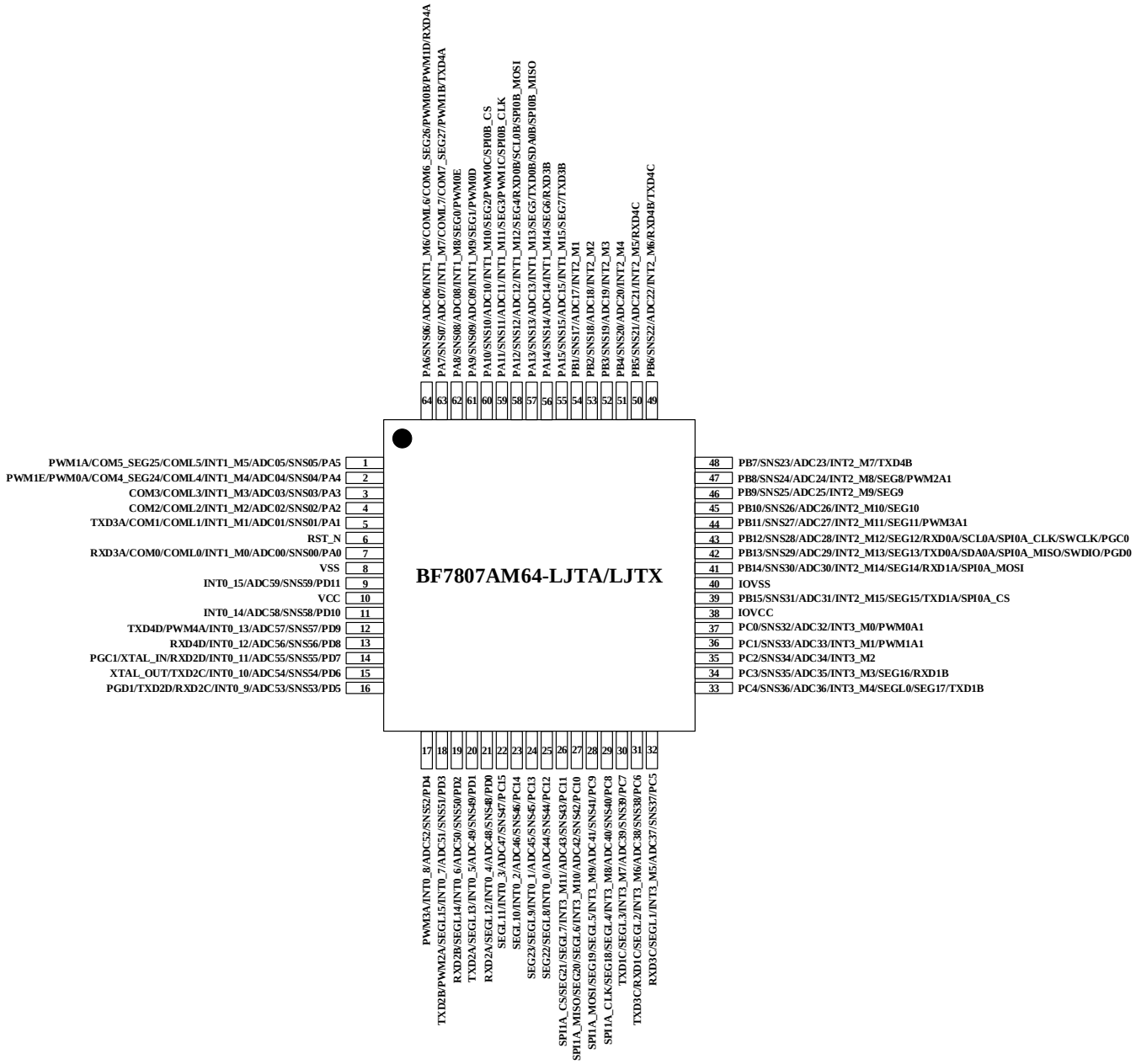
1.7. Pin configuration

1.7.1. BF7807AM44-LJTX



LQFP44 package pin diagram

1.7.2. BF7807AM64-LJTA/LJTX



LQFP64 package pin diagram

1.8. Pin description

BF7807AM64-LJTA/LJTX	BF7807AM44-LJTX	Function description
1	4	Default function: GPIO <PA5> Other functions: SNS05: Touch key channel ADC05: ADC channel INT1_M5: External interrupt COML5: COM of LED row column matrix; Large irrigation port COM5_SEG25: COM of LCD can be shared as SEG PWM1A: PWM output port
2	5	Default function: GPIO <PA4> Other functions: SNS04: Touch key channel ADC04: ADC channel INT1_M4: External interrupt COML4: COM of LED row column matrix; Large irrigation port COM4_SEG24: COM of LCD can be shared as SEG PWM0A: PWM output port PWM1E: PWM output port
3	6	Default function: GPIO <PA3> Other functions: SNS03: Touch key channel ADC03: ADC channel INT1_M3: External interrupt COML3: COM of LED row column matrix; Large irrigation port COM3: LCD COM
4	7	Default function: GPIO <PA2> Other functions: SNS02: Touch key channel ADC02: ADC channel INT1_M2: External interrupt COML2: COM of LED row column matrix; Large irrigation port COM2: LCD COM
5	8	Default function: GPIO <PA1> Other functions: SNS01: Touch key channel ADC01: ADC channel INT1_M1: External interrupt COML1: COM of LED row column matrix; Large irrigation port COM1: LCD COM TXD3A: Serial port transmission
6	-	RST_N: Reset pin

7	9	Default function: GPIO <PA0> Other functions: SNS00: Touch key channel ADC00: ADC channel INT1_M0: External interrupt COML0: COM of LED row column matrix; Large irrigation port COM0: LCD COM RXD3A: Serial port receiving
8	10	Default function: GND <VSS>
9	-	Default function: GPIO <PD11> Other functions: SNS59: Touch key channel ADC59: ADC channel INT0_15: External interrupt
10	11	Default function: Power supply <VCC>
11	-	Default function: GPIO <PD10> Other functions: SNS58: Touch key channel ADC58: ADC channel INT0_14: External interrupt
12	-	Default function: GPIO <PD9> Other functions: SNS57: Touch key channel ADC57: ADC channel INT0_13: External interrupt PWM4A: PWM output port TXD4D: Serial port transmission
13	-	Default function: GPIO <PD8> Other functions: SNS56: Touch key channel ADC56: ADC channel INT0_12: External interrupt RXD4D: Serial port receiving
14	12	Default function: GPIO <PD7> Other functions: SNS55: Touch key channel ADC55: ADC channel INT0_11: External interrupt RXD2D: Serial port receiving XTAL_IN: External crystal input PGC1: Programming port
15	13	Default function: GPIO <PD6> Other functions: SNS54: Touch key channel ADC54: ADC channel INT0_10: External interrupt TXD2D: Serial port transmission XTAL_OUT: External crystal oscillator output

16	14	Default function: GPIO <PD5> Other functions: SNS53: Touch key channel ADC53: ADC channel INT0_9: External interrupt TXD2D: Serial port transmission RXD2C: Serial port receiving PGD1: Programming port
17	-	Default function: GPIO <PD4> Other functions: SNS52: Touch key channel ADC52: ADC channel INT0_8: External interrupt PWM3A: PWM output port
18	15	Default function: GPIO <PD3> Other functions: SNS51: Touch key channel ADC51: ADC channel INT0_7: External interrupt SEGL15: SEG of LED row column matrix TXD2B: Serial port transmission PWM2A: PWM output port
19	16	Default function: GPIO <PD2> Other functions: SNS50: Touch key channel ADC50: ADC channel INT0_6: External interrupt SEGL14: SEG of LED row column matrix RXD2B: Serial port receiving
20	17	Default function: GPIO <PD1> Other functions: SNS49: Touch key channel ADC49: ADC channel INT0_5: External interrupt SEGL13: SEG of LED row column matrix TXD2B: Serial port transmission
21	18	Default function: GPIO <PD0> Other functions: SNS48: Touch key channel ADC48: ADC channel INT0_4: External interrupt SEGL12: SEG of LED row column matrix RXD2A: Serial port receiving
22	19	Default function: GPIO <PC15> Other functions: SNS47: Touch key channel ADC47: ADC channel INT0_3: External interrupt SEGL11: SEG of LED row column matrix

23	20	Default function: GPIO <PC14> Other functions: SNS46: Touch key channel ADC46: ADC channel INT0_2: External interrupt SEGL10: SEG of LED row column matrix
24	21	Default function: GPIO <PC13> Other functions: SNS45: Touch key channel ADC45: ADC channel INT0_1: External interrupt SEGL9: SEG of LED row column matrix SEG23: SEG of LCD
25	22	Default function: GPIO <PC12> Other functions: SNS44: Touch key channel ADC44: ADC channel INT0_0: External interrupt SEGL8: SEG of LED row column matrix SEG22: SEG of LCD
26	23	Default function: GPIO <PC11> Other functions: SNS43: Touch key channel ADC43: ADC channel INT3_M11: External interrupt SEGL7: SEG of LED row column matrix SEG21: SEG of LCD SPI1A_CS: SPI chip selection signal
27	24	Default function: GPIO <PC10> Other functions: SNS42: Touch key channel ADC42: ADC channel INT3_M10: External interrupt SEGL6: SEG of LED row column matrix SEG20: SEG of LCD SPI1A_MISO: SPI master data input
28	25	Default function: GPIO <PC9> Other functions: SNS41: Touch key channel ADC41: ADC channel INT3_M9: External interrupt SEGL5: SEG of LED row column matrix SEG19: SEG of LCD SPI1A_MOSI: SPI master data output
29	26	Default function: GPIO <PC8> Other functions: SNS40: Touch key channel ADC40: ADC channel INT3_M8: External interrupt

		SEGL4: SEG of LED row column matrix SEG18: SEG of LCD SPI1A_CLK: SPI clock
30	27	Default function: GPIO <PC7> Other functions: SNS39: Touch key channel ADC39: ADC channel INT3_M7: External interrupt SEGL3: SEG of LED row column matrix TXD1C: Serial port transmission
31	28	Default function: GPIO <PC6> Other functions: SNS38: Touch key channel ADC38: ADC channel INT3_M6: External interrupt SEGL2: SEG of LED row column matrix RXD1C: Serial port receiving TXD3C: Serial port transmission
32	29	Default function: GPIO <PC5> Other functions: SNS37: Touch key channel ADC37: ADC channel INT3_M5: External interrupt SEGL1: SEG of LED row column matrix RXD3C: Serial port receiving
33	30	Default function: GPIO <PC4> Other functions: SNS36: Touch key channel ADC36: ADC channel INT3_M4: External interrupt SEGL0: SEG of LED row column matrix SEG17: SEG of LCD TXD1B: Serial port transmission
34	31	Default function: GPIO <PC3> Other functions: SNS35: Touch key channel ADC35: ADC channel INT3_M3: External interrupt SEG16: SEG of LCD RXD1B: Serial port receiving
35	-	Default function: GPIO <PC2> Other functions: SNS34: Touch key channel ADC34: ADC channel INT3_M2: External interrupt
36	-	Default function: GPIO <PC1> Other functions: SNS33: Touch key channel ADC33: ADC channel

		INT3_M1: External interrupt PWM1A1: PWM output port
37	-	Default function: GPIO <PC0> Other functions: SNS32: Touch key channel ADC32: ADC channel INT3_M0: External interrupt PWM0A1: PWM output port
38	-	Default function: IOVCC
39	32	Default function: GPIO <PB15> Other functions: SNS31: Touch key channel ADC31: ADC channel INT2_M15: External interrupt SEG15: SEG of LCD SPI0A_CS: SPI chip selection signal TXD1A: Serial port transmission
40	-	Default function: IOVSS
41	33	Default function: GPIO <PB14> Other functions: SNS30: Touch key channel ADC30: ADC channel INT2_M14: External interrupt SEG14: SEG of LCD SPI0A_ MOSI: SPI master data output RXD1A: Serial port receiving
42	34	Default function: GPIO <PB13> Other functions: SNS29: Touch key channel ADC29: ADC channel INT2_M13: External interrupt SEG13: SEG of LCD SPI0A_ MISO: SPI master data input TXD0A: Serial port transmission SDA0A: Serial data line of IIC SWDIO: Data input /output PGD0: Programming port
43	35	Default function: GPIO <PB12> Other functions: SNS28: Touch key channel ADC28: ADC channel INT2_M12: External interrupt SEG12: SEG of LCD SPI0A_ CLK: SPI clock RXD0A: Serial port receiving SCL0A: Serial clock line of IIC SWCLK: Clock signal

		PGC0: Programming port
44	36	Default function: GPIO <PB11> Other functions: SNS27: Touch key channel ADC27: ADC channel INT2_M11: External interrupt SEG11: SEG of LCD PWM3A1: PWM output port
45	-	Default function: GPIO <PB10> Other functions: SNS26: Touch key channel ADC26: ADC channel INT2_M10: External interrupt SEG10: SEG of LCD
46	37	Default function: GPIO <PB9> Other functions: SNS25: Touch key channel ADC25: ADC channel INT2_M9: External interrupt SEG9: SEG of LCD
47	-	Default function: GPIO <PB8> Other functions: SNS24: Touch key channel ADC24: ADC channel INT2_M8: External interrupt SEG8: SEG of LCD PWM2A1: PWM output port
48	-	Default function: GPIO <PB7> Other functions: SNS23: Touch key channel ADC23: ADC channel INT2_M7: External interrupt TXD4B: Serial port transmission
49	-	Default function: GPIO <PB6> Other functions: SNS22: Touch key channel ADC22: ADC channel INT2_M6: External interrupt RXD4B: Serial port receiving TXD4C: Serial port transmission
50	-	Default function: GPIO <PB5> Other functions: SNS21: Touch key channel ADC21: ADC channel INT2_M5: External interrupt RXD4C: Serial port receiving
51	-	Default function: GPIO <PB4> Other functions: SNS20: Touch key channel ADC20: ADC channel

		INT2_M4: External interrupt
52	-	Default function: GPIO <PB3> Other functions: SNS19: Touch key channel ADC19: ADC channel INT2_M3: External interrupt
53	-	Default function: GPIO <PB2> Other functions: SNS18: Touch key channel ADC18: ADC channel INT2_M2: External interrupt
54	-	Default function: GPIO <PB1> Other functions: SNS17: Touch key channel ADC17: ADC channel INT2_M1: External interrupt
55	38	Default function: GPIO <PA15> Other functions: SNS15: Touch the key channel ADC15: ADC channel INT1_M15: External interrupt SEG7: SEG of LCD TXD3B: Serial port transmission
56	39	Default function: GPIO <PA14> Other functions: SNS14: Touch key channel ADC14: ADC channel INT1_M14: External interrupt SEG6: SEG of LCD RXD3B: Serial port receiving
57	40	Default function: GPIO <PA13> Other functions: SNS13: Touch key channel ADC13: ADC channel INT1_M13: External interrupt SEG5: SEG of LCD SPI0B_MISO: SPI master data input TXD0B: Serial port transmission SDA0B: Serial data line of IIC
58	41	Default function: GPIO <PA12> Other functions: SNS12: Touch key channel ADC12: ADC channel INT1_M12: External interrupt SEG4: SEG of LCD SPI0B_MOSI: SPI master data output RXD0B: Serial port receiving SCL0B: Serial clock line of IIC

59	42	Default function: GPIO <PA11> Other functions: SNS11: Touch key channel ADC11: ADC channel INT1_M11: External interrupt SEG3: SEG of LCD PWM1C: PWM output port SPI0B_CLK: SPI clock
60	43	Default function: GPIO <PA10> Other functions: SNS10: Touch key channel ADC10: ADC channel INT1_M10: External interrupt SEG2: SEG of LCD PWM0C: PWM output port SPI0B_CS: SPI chip selection signal
61	44	Default function: GPIO <PA9> Other functions: SNS09: Touch key channel ADC09: ADC channel INT1_M9: External interrupt SEG1: SEG of LCD PWM0D: PWM output port
62	1	Default function: GPIO <PA8> Other functions: SNS08: Touch key channel ADC08: ADC channel INT1_M8: External interrupt SEG0: SEG of LCD PWM0E: PWM output port
63	2	Default function: GPIO <PA7> Other functions: SNS07: Touch key channel ADC07: ADC channel INT1_M7: External interrupt COML7: COM of LED row column matrix; Large irrigation port COM7_SEG27: COM OF LCD can be shared as SEG TXD4A: Serial port transmission
64	3	Default function: GPIO <PA6> Other functions: SNS06: Touch key channel ADC06: ADC channel INT1_M6: External interrupt COML6: COM of LED row column matrix; Large irrigation port COM6_SEG26: COM OF LCD can be shared as SEG PWM0B: PWM output port PWM1D: PWM output port RXD4A: Serial port receiving

2 Electrical characteristics

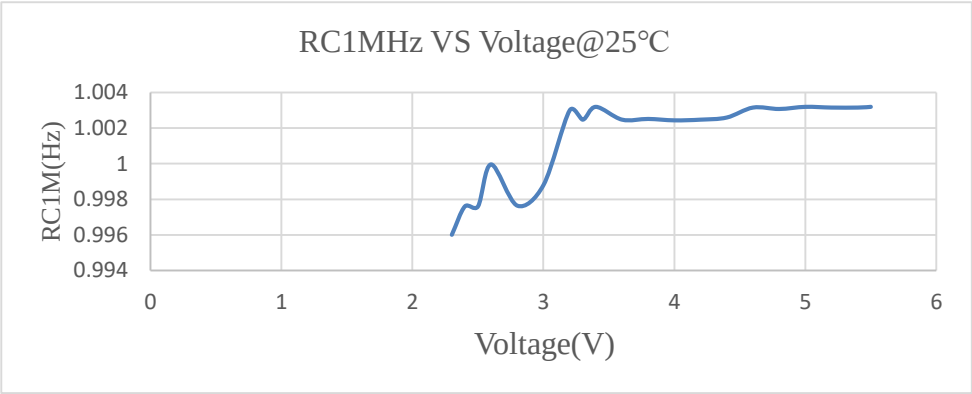
2.1. Limit parameters

Symbol	Parameter	Test Condition		Min	Typical	Max	Unit
		VCC	Condition				
VCC	Supply voltage when operating	-	-	VSS+2.7	-	VSS+5.5	V
T_{STG}	Storage temperature	-	-	-40	-	125	℃
Ta	Operating temperature	-	-	-40	-	105	℃
Vin	I/O input voltage	-	-	VSS-0.5	-	VCC+0.5	V
I_{OLA}	I_{OL} total current	-	-	130			mA
I_{OHA}	I_{OH} total current	-	-	-130			mA
ESD(HBM)	Port electrostatic discharge voltage	-	-	-8	-	8	kV

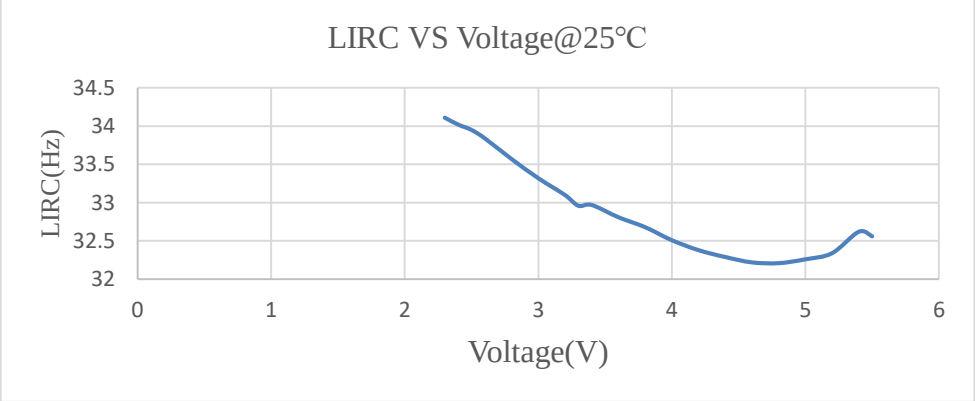
Note: Exceeding the range specified by the limit parameters will cause damage to the chip. It is impossible to predict the working state of the chip outside the above marked range, and if it works under the conditions outside the marked range for a long time, it may affect the reliability of the chip.

2.2. AC characteristics

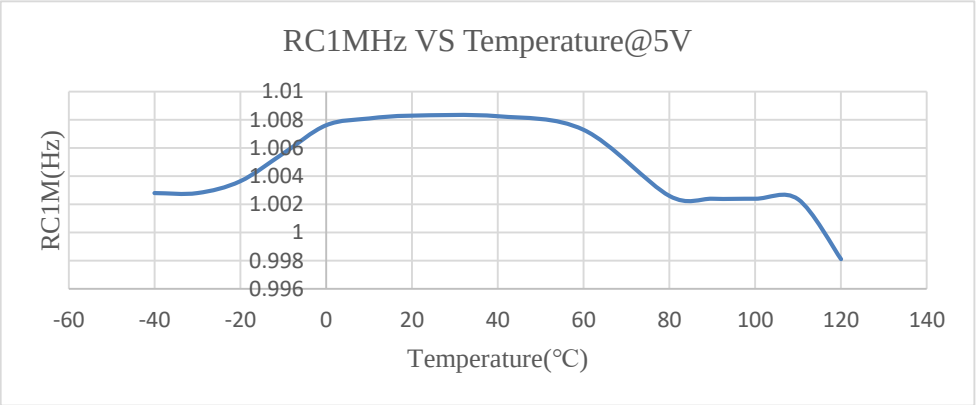
Symbol	Parameter	Test Conditions		Min	Typical	Max	Unit
		VCC	Temperature				
f_{RC1M}	Internal high-speed RC oscillator	5V	-20℃~65℃	-1%	1	+1%	MHz
			-40℃ ~105℃	-3%	1	+3%	
		2.7V~5.5V	25℃	-1%	1	+1%	
			-40℃ ~105℃	-3%	1	+3%	
f_{HCLK}	System clock	5V	-20℃~65℃	-1%	48/32/24/12	+1%	MHz
			-40℃ ~105℃	-3%	48/32/24/12	+3%	
		2.7V~5.5V	25℃	-1%	48/32/24/12	+1%	
			-40℃ ~105℃	-3%	48/32/24/12	+3%	
f_{LIRC}	Internal low speed RC oscillator	5V	25℃	-10%	32	+10%	kHz
			-40℃ ~105℃	-25%	32	+25%	
		2.7V~5.5V	25℃	-20%	32	+20%	



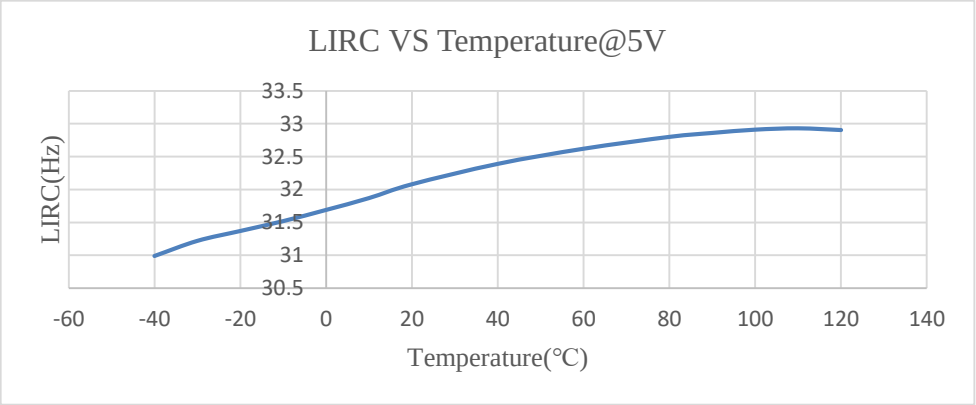
f_{RC1M} voltage graph



f_{LIRC} voltage graph



f_{RC1M} temperature curve



f_{LIRC} temperature curve

2.3. DC characteristics

Ta=25℃

Symbol	Parameter	Test Condition		Min	Typical	Max	Unit
		VCC	Condition				
VCC	Operating voltage	-	-	2.7	-	5.5	V
I_{OP1}	Active mode current	3.3V	f_{HCLK}=48MHz, all peripherals off	-	5.0	6.5	mA
		5V		-	5.2	6.7	
		3.3V	f_{HCLK}=32MHz, all peripherals off	-	4.4	5.7	
		5V		-	4.6	6.0	
		3.3V	f_{HCLK}=24MHz, all peripherals off	-	3.8	4.9	
		5V		-	4.0	5.2	
		3.3V	f_{HCLK}=12MHz, all peripherals off	-	2.8	3.6	
		5V		-	3.0	3.9	
I_{STB0}	Idle mode 0 current	3.3V	SCR = 0x0, all peripherals off	-	1.6	2.0	mA
		5V		-	1.7	2.2	
I_{STB1}	Idle mode 1 current	3.3V	SCR = 0x4, all peripherals off	-	7.8	10.1	μA
		5V		-	8.0	10.4	
V_{IL}	Input low voltage	2.7~5.5V	-	-	-	0.3*VCC	V
V_{IH}	Input high voltage	2.7~5.5V	-	0.7*VCC	-	-	V
V_{INTL}	INT input low voltage	2.7~5.5V	-	-	-	0.3*VCC	V
V_{INTH}	INT input high voltage	2.7~5.5V	-	0.7*VCC	-	-	V
V_{OL}	Output low voltage	5V	I_{OL}=75mA	-	-	0.1*VCC	V
V_{OH}	Output high voltage	5V	I_{OH}=16mA	0.9VCC	-	-	V
I_{OL}	IO sink current	5V	V_{OL}=0.1VCC	53	75	97	mA
I_{OH}	IO source current	5V	V_{OH}=0.9VCC	11	16	20	mA
I_{COM}	PA0-PA7 high current	5V	V_{OL}=0.1VCC	-	130	-	mA
I_{SPI_OL}	SPI high-speed mode sink current	5V	V_{SPI_OL}=0.1VCC	-	6.0	-	mA
I_{SPI_OH}	SPI high-speed mode source current	5V	V_{SPI_OH}=0.9VCC	-	5.0	-	mA
I_{Leak}	Input leakage current	5V	-	-	1	5	μA
R_{PH}	IO/RST_N internal pull-up resistor	5V	-	23	33	42	kΩ

Symbol	Parameter	Test Condition		Min	Typical	Max	Unit
		VCC	Condition				
I_{CDC}	CDC operating current	5V	f_{HCLK}=48MHz, enable CDC, open six channels	-	0.6	-	mA
I_{ADC}	ADC operating current	5V	f_{HCLK}=48MHz, enable ADC, open one channel	-	2.7	-	mA
I_{PWM}	PWM operating current	5V	f_{HCLK}=48MHz, enable PWM0, output 4kHz waveform	-	0.6	-	mA
I_{LVDT}	LVDT operating current	5V	f_{HCLK}=48MHz, enable LVDT, configure detection parameter 2.7V	-	0.6	-	mA

2.4. ADC characteristics

Ta=25℃

Symbol	Parameter	Test Condition		Min	Typical	Max	Unit
		VCC	Condition				
V_{ADC}	Supply voltage	-	-	2.7	-	5.5	V
N_R	Precision	-	-	-	9	10	Bit
V_{ADCI}	ADC input voltage	-	-	VSS		V_{REF}	V
R_{ADCI}	ADC input resistance	5V	No RC filtering	1.2	3.2	17.5	kΩ
			RC filtering	10	14.2	31.5	
I_{ADC}	ADC operating current	5V	f_{HCLK}=48MHz, enable ADC, open one channel	-	2.7		mA
I_{ADCI}	A/D input current	-	-	-	-	1	μA
DNL	Differential nonlinearity error	5V	-	-	±4	±6	LSB
INL	Integral nonlinearity error	5V	-	-	±4	±6	LSB
t_1	ADC sampling time	-	-	0.5	-	-	μs
t_{ADC}	ADC conversion time	-	-	2.875	-	-	μs
RESO	Resolution	-	-	12			Bit
N_{ADC}	Input channel	-	-	-	-	59	Channel

2.5. Memory characteristics

Ta=25℃

Symbol	Parameter	Test Condition		Min	Typical	Max	Unit
		VCC	Condition				
EC	Erase and write times	5V	-	20000	-	-	Cycle
t_{RET}	Data retention period	-	-	100	-	-	Year
t_{ERASE}	Page erase time	5V	f_{HCLK}=48MHz, in the while, the main memory block is page erased	-	4.5	-	ms
t_{PROG}	Programming time	5V	f_{HCLK}=48MHz, in the while, a word is written to the main memory block	-	750	-	ns
I_{ERASE}	Page erase current	5V	f_{HCLK}=48MHz, in the while, the main memory block is page erased	-	1.0	-	mA
I_{PROG}	Programming current	5V	f_{HCLK}=48MHz, in the while, a word is written to the main memory block	-	1.2	-	mA

3 System control (SYS_CTRL)

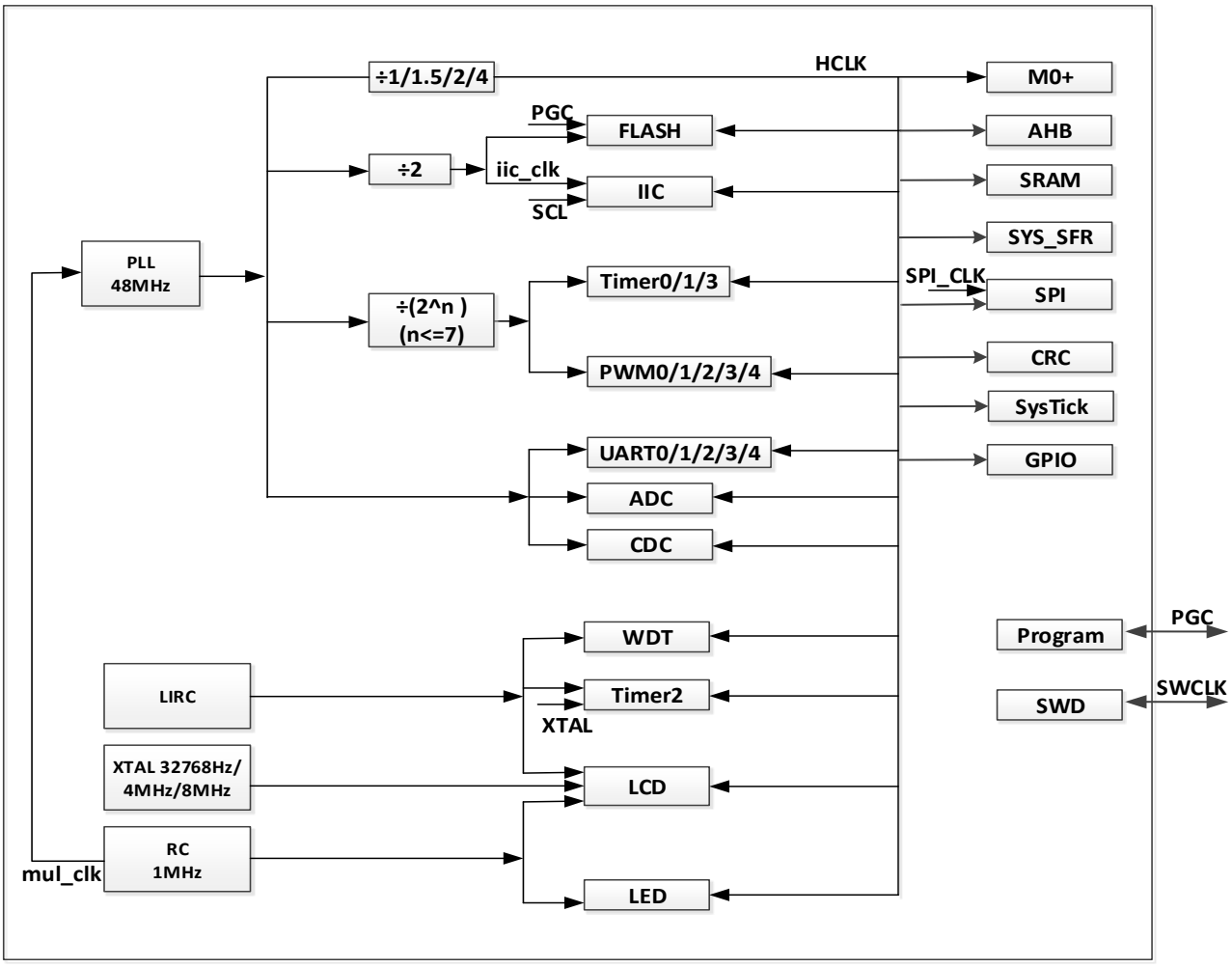
3.1. Clock description

The clock control module mainly controls the system clock and peripheral clock. It can configure different system clock frequency divisions, and can enable or disable peripheral clocks. Clock source:

- Internal high-speed RC oscillator: RC 1MHz
- Internal low-speed RC oscillator: LIRC 32kHz
- External crystal oscillator: 32768Hz/4MHz/8MHz
- RC1M multiplication to get PLL clock: PLL multiplied clock

The BF7807AMXX series clocks are defined as follows:

- **RC1M:** Internal high-speed RC oscillator, LED/LCD module working clock, frequency multiplier (mul_clk) to get PLL48MHz.
- **PLL48M:** System clock source, CDC/ADC/UART module working clock.
- **HCLK:** System clock, AHB bus peripheral clock, 48MHz/32MHz/24MHz/12MHz frequency optional.
- **PLL48M divide by 2:** FLASH programming clock, IIC digital filter clock and IIC master working clock(iic_clk).
- **PLL48M 1~128 frequency division:** Timer0/Timer1/Timer3/PWM module working clock.
- **LIRC 32kHz:** Internal low-speed RC oscillator, WDT/Timer2/LCD module clock source.
- **XTAL 32768Hz /4MHz /8MHz:** External crystal oscillator, Timer2/LCD module clock source.
- **PGC:** The clock within 5M is used as the FLASH programming clock source.
- **SCL:** The highest 400kHz clock is used as the IIC slave communication clock source.
- **SPI_CLK:** The highest 4MHz clock is used as the slave SPI communication clock source.
- **SWCLK:** Clock within 20M, as the clock source for DEBUG and FLASH programming.



Clock block diagram

3.2. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x00	CLK_CFG	Clock configuration register
0x04	RCU_EN	Peripheral module clock control register
0x0C	XTAL_HS_SEL	Hysteresis voltage selection of comparator in crystal oscillator register
0x10	ANA_CFG	Analog module switch register
0x18	WAKE_CFG	System wake-up configuration register

3.2.1. Clock registers

3.2.1.1. Clock configuration register (CLK_CFG)

Address offset: 0x00

Reset value: 0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:2	-	Reserved
1:0	HCLK_SEL	HCLK clock frequency division selection register 00: 48MHz 01: 32MHz 10: 24MHz 11: 12MHz

3.2.1.2. Peripheral module clock control register (RCU_EN)

This register is a register that allows or prohibits the provision of clocks to peripheral modules.

Address offset: 0x04

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16

LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8
TIMER1_CLKEN	TIMER0_CLKEN	PWM4_CLKEN	PWM3_CLKEN	PWM2_CLKEN	PWM1_CLKEN	PWM0_CLKEN	IIC_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

7	6	5	4	3	2	1	0
UART4_CLKEN	UART3_CLKEN	UART2_CLKEN	UART1_CLKEN	UART0_CLKEN	SPI1_CLKEN	SPI0_CLKEN	Reserved
RW	RW	RW	RW	RW	RW	RW	

31:24	-	Reserved
23	LED_LCD _CLKEN	LCD/LED module operation enable 1: Work 0: Off, the default is 0
22	GPIO_CLKEN	GPIO module operation enable 1: Work 0: Off, the default is 0
21	CRC_CLKEN	CRC module operation enable 1: Work 0: Off, the default is 0
20	ADC_CLKEN	ADC module operation enable 1: Work 0: Off, the default is 0
19	CDC_CLKEN	CDC module operation enable 1: Work 0: Off, the default is 0
18	WDT_CLKEN	WDT module operation enable 1: Work 0: Off, the default is 0
17	TIMER3_CLKEN	TIMER3 module operation enable 1: Work 0: Off, the default is 0
16	TIMER2_CLKEN	TIMER2 module operation enable 1: Work 0: Off, the default is 0
15	TIMER1_CLKEN	TIMER1 module operation enable

		1: Work 0: Off, the default is 0
14	TIMER0_CLKEN	TIMER0 module operation enable 1: Work 0: Off, the default is 0
13	PWM4_CLKEN	PWM4 module operation enable 1: Work 0: Off, the default is 0
12	PWM3_CLKEN	PWM3 module operation enable 1: Work 0: Off, the default is 0
11	PWM2_CLKEN	PWM2 module operation enable 1: Work 0: Off, the default is 0
10	PWM1_CLKEN	PWM1 module operation enable 1: Work 0: Off, the default is 0
9	PWM0_CLKEN	PWM0 module operation enable 1: Work 0: Off, the default is 0
8	IIC_CLKEN	IIC module operation enable 1: Work 0: Off, the default is 0
7	UART4_CLKEN	UART4 module operation enable 1: Work 0: Off, the default is 0
6	UART3_CLKEN	UART3 module operation enable 1: Work 0: Off, the default is 0
5	UART2_CLKEN	UART2 module operation enable 1: Work 0: Off, the default is 0
4	UART1_CLKEN	UART1 module operation enable 1: Work 0: Off, the default is 0
3	UART0_CLKEN	UART0 module operation enable 1: Work 0: Off, the default is 0
2	SPI1_CLKEN	SPI1 module operation enable 1: Work

		0: Off, the default is 0
1	SPI0_CLKEN	SPI0 module operation enable 1: Work 0: Off, the default is 0
0	-	Reserved

3.2.1.3. System wake-up configuration register (WAKE_CFG)

Address offset: 0x18

Reset value: 0x0000 0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:3	-	Reserved
2:0	PLL_WAKE_TIM	Wake up PLL timing time 000: 0.2ms 001: 0.3ms 010: 0.4ms 011: 0.5ms 100: 0.6ms 101: 0.7ms 110: 0.9ms 111: 1ms

3.2.2. Hysteresis voltage selection of comparator in crystal oscillator register (XTAL_HS_SEL)

Address offset: 0x0C

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:2	1	0
Reserved	HS_SEL[1:0]	
	RW	

31:2	-	Reserved

1:0	HS_SEL[1:0]	Hysteresis voltage selection of comparator in crystal oscillator 00: 300mV 01: 400mV 10: 500mV 11: 600mV

3.2.3. Analog module switch register (ANA_CFG)

Address offset: 0x10

Reset value: 0x0000 0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	XTAL_HFR_SEL	XTAL_SEL	PD_XTAL	PD_CDC	PD_ADC
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	XTAL_HFR_SEL	Analog high frequency crystal oscillator circuit selection 0: 4MHz 1: 8MHz
3	XTAL_SEL	Frequency selection of analog crystal oscillator circuit 0: 32768Hz 1: 4MHz/8MHz
2	PD_XTAL	Analog crystal oscillator circuit (32768Hz/4MHz/8MHz) control register 1: Off 0: On, off by default
1	PD_CDC	CDC work control register 0: CDC module works normally 1: CDC module does not work
0	PD_ADC	Analog ADC shutdown control register 0: ADC module works normally 1: ADC module does not work

3.3. SysTick timer

3.3.1. System timer description

SysTick timer, used to generate periodic interrupt requests.

Features:

- 24-bit cyclic count down
- Automatic loading count initial value
- Configurable interrupts
- The count clock is the system clock

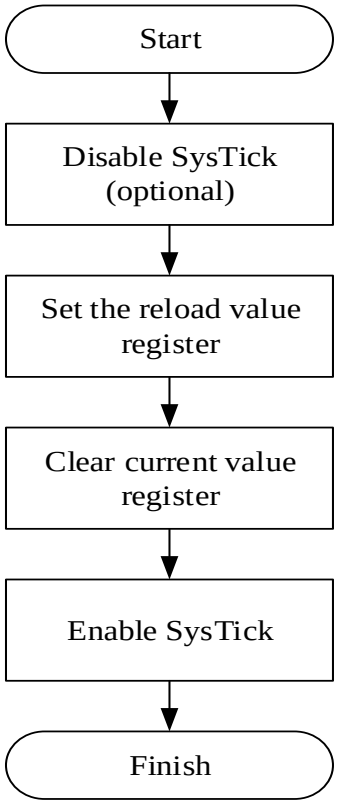
SysTick cannot wake up from standby mode.

SysTick is a 24-bit timer and counts down. After the timer count is reduced to 0, a programmable value will be reloaded and a SysTick exception will be generated at the same time. SysTick can be operated by polling or interruption. Programs that use polling can read the SysTick control and status registers and check COUNTFLAG. If the flag is set, it means that the SysTick count has been reduced to 0.

Directly operate register to control SysTick:

```
SysTick->CTRL = 0;          // Disable SysTick
SysTick->LOAD = 999;        // Count down from 999 to 0
SysTick->VAL = 0;           // Clear the current value to 0
SysTick->CTRL = 0x7;        // Enable SysTick
```

The process of setting SysTick is as follows:



3.3.2. Registers

Address	Register	CMSIS symbol	Description
0xE000E010	SYST_CSR	SysTick->CTRL	SysTick control and status register
0xE000E014	SYST_RVR	SysTick->LOAD	SysTick reload value register
0xE000E018	SYST_CVR	SysTick->VAL	SysTick current value register

3.3.2.1. SysTick control and status register (SYST_CSR)

Bit	Bit symbol	Description	RW	Reset value
31:17	Reserved	-	-	-
16	COUNTFLAG	SysTick timer overflow flag, read-only bit 1: Timer counts to 0 0: Timer did not count to 0 Reading register will be cleared	R	0
15:3	Reserved	-	-	-
2	CLKSOURCE	SysTick clock source 1: Use HCLK clock 0: Reserved	R/W	0
1	TICKINT	SysTick interrupt enable 1: Enable interrupt 0: Prohibit interrupt	R/W	0
0	ENABLE	SysTick timer enable 1: Enable SysTick 0: Prohibit SysTick	R/W	0

3.3.2.2. SysTick reload value register (SYST_RVR)

Bit	Bit symbol	Description	RW	Reset value
31:24	Reserved	-	-	-
23:0	RELOAD	Reload initial value of Systick timer	R/W	Undefined

3.3.2.3. SysTick current value register (SYST_CVR)

Bit	Bit symbol	Description	RW	Reset value
31:24	Reserved	-	-	-
23:0	CURRENT	The current value of the Systick timer, writing any value will clear the register, and COUNTFLAG will also be cleared (does not cause an exception to the Systick timer)	R/W	Undefined

4 FLASH

4.1. FLASH features

- Main storage block: 128K Bytes, 256 pages
- NVR1/2/3/4: 512 Bytes per NVR, 1 page
- Whole word programming
- Whole chip erase and page erase
- Support IAP upgrade
- Erase and program protection states to prevent accidental write operations
- Support read protection function and write configuration word protection function
- Support security protection status, which can prevent illegal read access to code or data
- Erasing/programming times: At least 20000 times@25℃
- Data retention period: 100 years@25℃

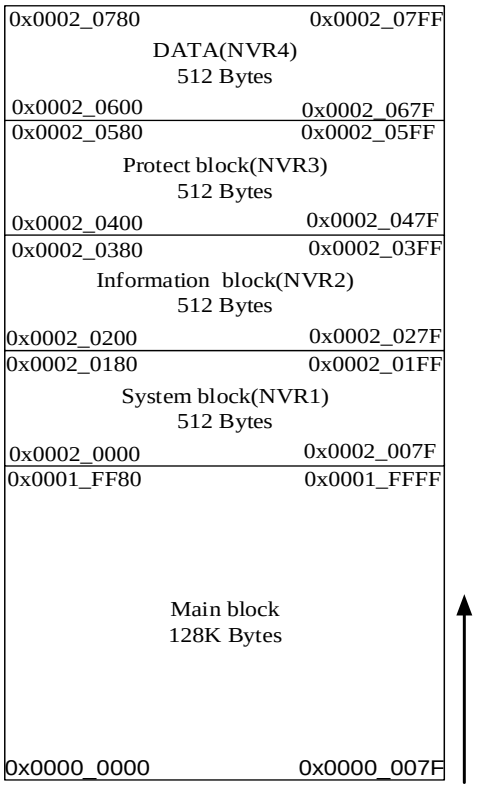
4.2. FLASH memory allocation

Main block: Used to store the chip running program.

NVR1 (system block) addresses 0x201F8 and NVR3 addresses 0x205FC~0x205E0 are option byte areas. Users can configure according to specific application requirements.

NVR2 (information block): Used to store information about the factory configuration, which cannot be changed by the user.

NVR4 (DATA): User data storage area.



Module	Address	Size(Bytes)	Page
Main block	0x0000_0000 ~ 0x0000_01FF	512	Page 0
	0x0000_0200 ~ 0x0000_03FF	512	Page 1
	0x0000_0300 ~ 0x0000_04FF	512	Page 2
	0x0000_0500 ~ 0x0000_06FF	512	Page 3
	…	…	…
	0x0001_FE00~ 0x0001_FFFF	512	Page 255
NVR1 (System block)	0x0002_0000 ~ 0x0002_01FF	512	1 page
NVR2 (Information block)	0x0002_0200 ~ 0x0002_03FF	512	1 page
NVR3(Protect block)	0x0002_0400 ~ 0x0002_05FF	512	1 page
NVR4	0x0002_0600 ~ 0x0002_07FF	512	1 page

Address allocation table

4.3. FLASH function overview

When the FLASH is operated and the erase and programming instructions are executed, the AHB bus is occupied. When the erase and programming instructions are executed, the bus is released and the program continues to execute. If an interrupt occurs, wait for the erase and programming instructions to complete, and then execute the interrupt service routine after the BUSY bit in the FMC_STATE register is cleared to 0.

4.3.1. Key value

In order to enhance security, when performing an operation, it is necessary to write specific values to a bit to verify whether it is a safe operation. These values are called key values. The FLASH of BF7807AMXX has three key values:
RDP = 0xA5, used to release read protection;
KEY1 = 0x45670123, to unlock the flash memory lock;
KEY2 = 0xCDEF89AB, to unlock the flash memory lock.

4.3.2. FLASH unlock

After reset, the FMC module is protected and the FMC control register (FMC_CTRL) does not allow write operations. The unlock sequence is as follows:

1. Write KEY1 = 0x45670123 to the FMC key register (FMC_KEY)
2. Write KEY2 = 0xCDEF89AB to the FMC key register (FMC_KEY)

Wrong sequence of operations will lock the FMC module and the FMC_CTRL register before the next reset, and writing the wrong key sequence will also generate a bus error. A bus error occurs when either of the following conditions occur:

- The first write is not KEY1
- The first write is KEY1 but the second write is not KEY2

The first 4 pages of FLASH and NVR are still protected after writing the key value to the FMC key register (FMC_KEY). For erasing and programming of the first 4 pages of FLASH, PER0KEY is also required to unlock. The unlocking process is two write operations:

1. Write 0x45670123 to the FLASH first 4 page key register (FMC_PER0KEY)
2. Write 0xCDEF89AB to the FLASH first 4 page key register (FMC_PER0KEY)

For the unlocking steps of NVR1/3/4, see "NVR1/3/4 unlocked", and for the unlocking steps of NVR2, see "NVR2 read".

The program can set the LOCK bit in the FMC control register (FMC_CTRL) to lock the FMC module and the FMC control register (FMC_CTRL).

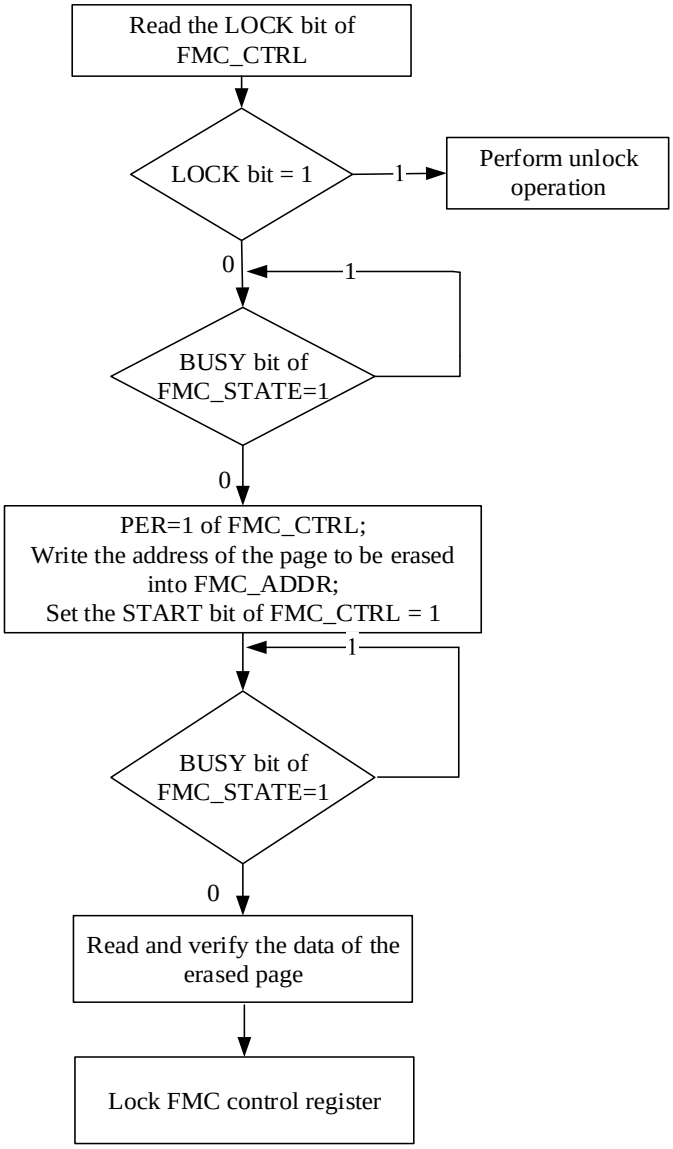
4.3.3. Erase

4.3.3.1. Main memory block page erase

The page erase function of FMC initializes the page content of the main storage flash memory to high level. Each page can be erased independently without affecting the contents of other pages. FMC erase page steps are as follows:

1. Make sure that the FMC_CTRL register is not in a locked state;
2. Check the BUSY bit of the FMC_STATE register to determine whether the flash memory is in the erase and write access state, if the BUSY bit is 1, you need to wait for the operation to end, and the BUSY bit becomes 0;
3. Position FMC_CTRL the PER bit of the register;
4. Writes the absolute address of the page to be erased to the FMC_ADDR register;
5. Send the page erase command to FMC by setting the START bit of the FMC_CTRL register to 1;
6. Wait for the erase command to be executed, and the BUSY bit of the FMC_STATE register is cleared to 0;
7. If necessary, the CPU can read to verify whether the page has been successfully erased;
8. After completing the relevant operations, re-lock the FMC control register to prevent FLASH from being misoperated.

When the page erase is successfully performed, the END bit of the FMC_STATE register is set. The user needs to make sure that the correct erase address is written. Otherwise, when the address of the page to be erased is used to fetch instructions or access data, the software will run away. In this case, the FMC does not provide any notification of the error. At the same time, erasing a write-protected page will have no effect. The figure below shows the page erase operation flow.

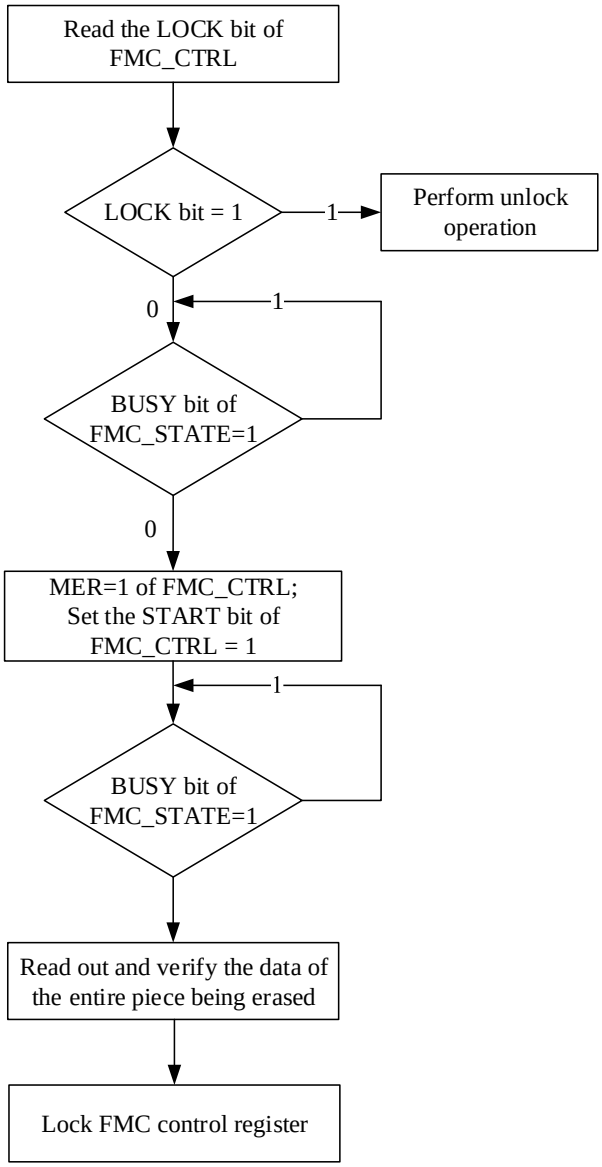

Read the LOCK bit of FMC_CTRL
LOCK bit = 1
1
Perform unlock operation
0
1
BUSY bit of FMC_STATE=1
0
PER=1 of FMC_CTRL;
Write the address of the page to be erased into FMC_ADDR;
Set the START bit of FMC_CTRL = 1
1
BUSY bit of FMC_STATE=1
0
Read and verify the data of the erased page
Lock FMC control register

4.3.3.2. Main memory block full chip erase

The chip erase function of the BF7807AMXX initializes the contents of the main memory block to a high level. Chip Erase does not affect the NVR. For chip erase operation, the specific steps of register setting are as follows:

1. Make sure that the FMC_CTRL register is not in a locked state;
2. Wait for the BUSY bit of the FMC_STATE register to become 0;
3. Set the MER bit in the FMC_CTRL register;
4. Send the entire chip erase command to FMC by setting the START bit of the FMC_CTRL register to 1;
5. Wait for the erase command to be executed and the BUSY bit of the FMC_STATE register is cleared to 0;
6. If necessary, you can verify whether the programming is successful by reading the CPU (little endian mode);
7. After completing the relevant operations, re-lock the FMC control register to prevent FLASH from being misoperated.

When the entire chip erase is successfully executed, the END bit of the FMC_STATE register is set. Since all the flash memory data will be reset to 0xFFFF_FFFF, the whole chip erase operation can be realized by directly accessing the FMC register by a program running in SRAM or using a debugging tool.


Read the LOCK bit of FMC_CTRL
LOCK bit = 1
1
Perform unlock operation
0
1
BUSY bit of FMC_STATE=1
0
MER=1 of FMC_CTRL; Set the START bit of FMC_CTRL = 1
1
BUSY bit of FMC_STATE=1
Read out and verify the data of the entire piece being erased
Lock FMC control register

4.3.4. Main memory block programming

FMC provides a 32-bit whole word programming function to modify the contents of the main memory block. The programming operation using each register flow is as follows:

1. Make sure that the FMC_CTRL register is not in a locked state;
2. Wait for the BUSY bit of the FMC_STATE register to become 0;
3. Set the PG bit of the FMC_CTRL register;
4. The CPU writes a 32-bit whole word to the destination absolute address (0x000X_XXXX);
5. Wait for the completion of the programming instruction, and the BUSY bit of the FMC_STATE register is cleared to 0;
6. Clear the PG bit of the FMC_CTRL register;
7. If necessary, it can be verified by CPU read whether the programming is successful (little endian mode);
8. After completing the relevant operations, re-lock the FMC control register to prevent FLASH from being misoperated.

When the main memory block programming is successfully executed, the END bit of the FMC_STATE register is set.

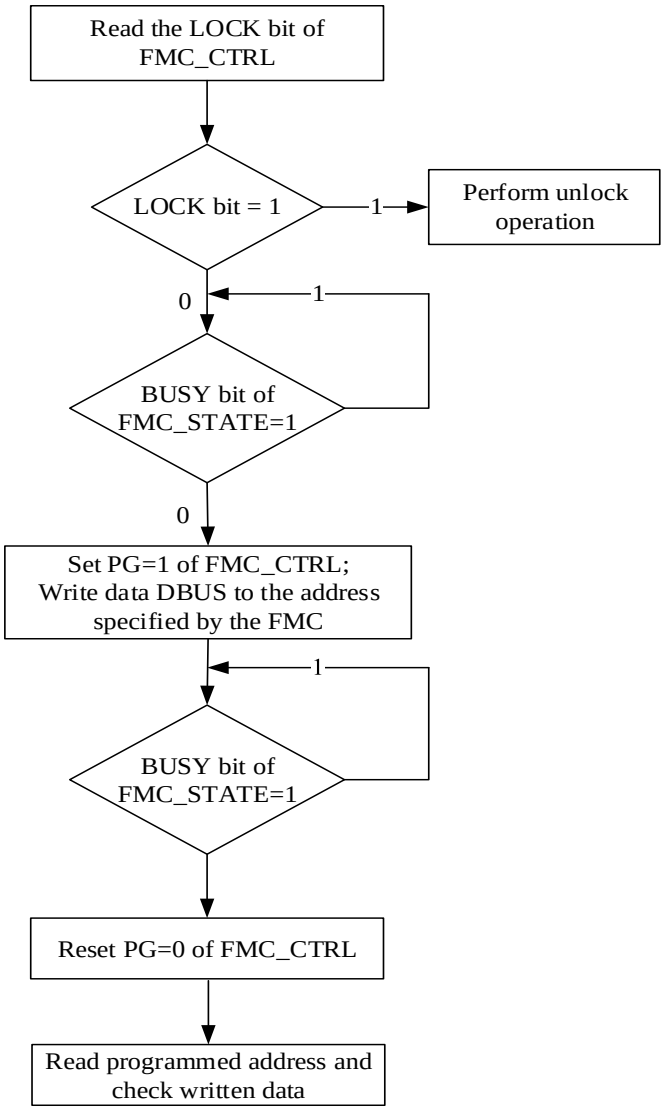
Programming errors: The following errors can be detected.

- PGERR: Programming error

When performing a whole word programming operation, the hardware checks whether the data at the destination address is 0xFFFF. When the data at the address is not 0xFFFF, a programming error occurs, PGERR is set to 1, and programming is aborted.

- WPERR: Erase/Program protection error

During an erase/program operation on a protected page, WPERR is set by hardware and the erase/program is aborted.


Read the LOCK bit of FMC_CTRL
LOCK bit = 1
1
Perform unlock operation
0
1
BUSY bit of FMC_STATE=1
0
Set PG=1 of FMC_CTRL; Write data DBUS to the address specified by the FMC
1
BUSY bit of FMC_STATE=1
Reset PG=0 of FMC_CTRL
Read programmed address and check written data

4.3.5. Protect

The user code area in the flash memory can prevent illegal reading. The pages of the flash memory area are protected to prevent them from being accidentally changed when the program runs away. The basic unit of write protection is: 2 pages.

4.3.5.1. Main memory read protection

Read protection can be activated by setting the RDP option byte and then a system reset to load the new RDP option byte.

Read protection turned on:

1. Only allow the read operation of the main flash memory from the user code.
2. All the functions of loading and executing code to the built-in SRAM through SWD are still valid, and it can also be started from the built-in SRAM through SWD. This function can be used to release the read protection.
3. When the read-protected option byte is changed to the unprotected value of the memory, the whole chip erase process will be executed.

Steps to remove read protection:

1. Erase the entire option byte area, the read protection code (RDP) will become 0xFF, at this time the read protection is still valid;
2. Write the correct nRDP (0x5A) and RDP (0xA5) to release the protection of the memory. This operation will firstly cause the entire chip erase operation of all user flash memories;
3. Perform a reset to reload the option byte (and the new RDP code), at this time the read protection is released.

4.3.5.2. Main memory write protection

Write protection is implemented in units of every 2 pages.

If you try to program or erase a protected page, a protection error flag will be returned in the FMC status flag register (FMC_STATE).

The configuration option byte WRP[7:0] will set the write protection, and the subsequent system reset will load the new WRPx option byte.

Pages 0~3 are automatically written-protected, and other parts of the memory can be programmed through the code executed in the main memory (to achieve IAP or data storage functions), but it is not allowed to be in the debug mode or in the slave internal SRAM Perform write or erase operations after booting (except for mass erase).

IAP (In Application Programming): It means that the Flash of the microcontroller can be reprogrammed during the running of the user program.

Remove write protection:

1. Erase the entire option byte area;
2. Reload option bytes (including new WRP[7:0] bytes);
3. Perform a system reset and the write protection is released.

4.4. NVR1/3/4

4.4.1. NVR1/3 option bytes

NVR1 (system block) addresses 0x201F8 and NVR3 addresses 0x205FC~0x205E0 of BF7807AMXX are option byte areas. The option bytes are configured by the user according to specific application requirements. Each 32-bit word in the option byte is divided into the following format:

Bit[31:16]	Bit[15:0]
Inversion of byte 0	Option byte 0

The selected byte block must be programmed in 32-bit word, Bit[31:16] is the inverse of Bit[15:0], otherwise the default value is restored.

NVR1 option byte description:

FLASH address	Bit	Bit symbol	Description	Defaults
0x201F8	[10]	PD_WDT_EN	0: Do not allow the program to turn off the watchdog function 1: Allow the program to turn off the watchdog function	1
	[9]	WDTRST_SD	0: When entering idle mode 1, a watchdog reset is generated 1: When entering idle mode 1, no watchdog reset is generated, wake-up standby, and WDT interrupt is generated	1
	[8]	WDTRST_SP	0: When entering idle mode 0, a watchdog reset is generated 1: When entering idle mode 0, no watchdog reset is generated, wake up standby, and generate WDT interrupt	1
	[7:0]	MAIN_READ_PROTECT	Code read protection option byte. Unprotected state (can be read): RDP=0xA5 Protected state (cannot be read): RDP!=0xA5	0xFF

NVR3 option byte description: The basic unit of write protection is: 2 pages.

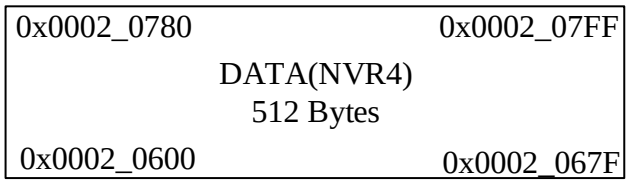
Each representative: 0: Write protection is effective; 1: Write protection is invalid

FLASH address	Bit	Bit symbol	Description	Defaults
0x205FC	[15:0]	WRP0	Write protection of pages 0~31 of the main memory block Bit[0]: Write protection of page 0 and page 1 Bit[1]: Write protection for pages 2 and 3 … Bit[14]: Write protection on pages 28 and 29 Bit[15]: Write protection on page 30 and page 31	0xFFFF
0x205F8	[15:0]	WRP1	Write protection of pages 32~63 of the main memory block Bit[0]: Write protection for pages 32 and 33 … Bit[15]: Write protection on pages 62 and 63	0xFFFF
0x205F4	[15:0]	WRP2	Write protection of pages 64~95 of the main memory block Bit[0]: Write protection of page 64 and page 65 … Bit[15]: Write protection on pages 94 and 95	0xFFFF
0x205F0	[15:0]	WRP3	Write protection of pages 96~127 of the main memory block Bit[0]: Write protection of page 96 and page 97 … Bit[15]: Write protection on pages 126 and 127	0xFFFF
0x205EC	[15:0]	WRP4	Write protection of pages 128~159 of the main memory block Bit[0]: Write protection on page 128 and page 129 … Bit[15]: Write protection on pages 158 and 159	0xFFFF
0x205E8	[15:0]	WRP5	Write protection of pages 160~191 of the main memory block Bit[0]: Write protection on pages 160 and 161 … Bit[15]: Write protection on pages 190 and 191	0xFFFF
0x205E4	[15:0]	WRP6	Write protection of pages 192~223 of the main memory block Bit[0]: Write protection on pages 192 and 193 … Bit[15]: Write protection on pages 222 and 223	0xFFFF

0x205E0	[15:0]	WRP7	Write protection of pages 224~255 of the main memory block Bit[0]: Write protection for pages 224 and 225 … Bit[15]: Write protection on pages 254 and 255	0xFFFF

4.4.2. NVR4 (DATA area)

The NVR of BF7807AMXX includes a DATA storage area for storing user data. The address range of the DATA area is 0x0002_0600 ~ 0x0002_07FF, one page, 512 bytes.



DATA area address map

4.4.3. NVR1/3/4 unlocked

By default, NVR1/3/4 are always readable and write-protected.

Write operation (program/erase) to NVR1/3/4 must first write the correct key sequence in the FMC option key register, then allow write operation to NVR1/3/4, OTPWRE bit of FMC control register (FMC_CTRL) Indicates that writing is allowed, clearing this bit will disable writing.

The unlock sequence is as follows:

1. Write KEY1 = 0x45670123 to the FMC key register (FMC_KEY);
2. Write KEY2 = 0xCDEF89AB to the FMC key register (FMC_KEY);
3. Write 0x45670123 to the FMC option key register (FMC_OTPKEY);
4. Write 0xCDEF89AB to the FMC Option Key Register (FMC_OTPKEY).

4.4.4. NVR1/3/4 erase

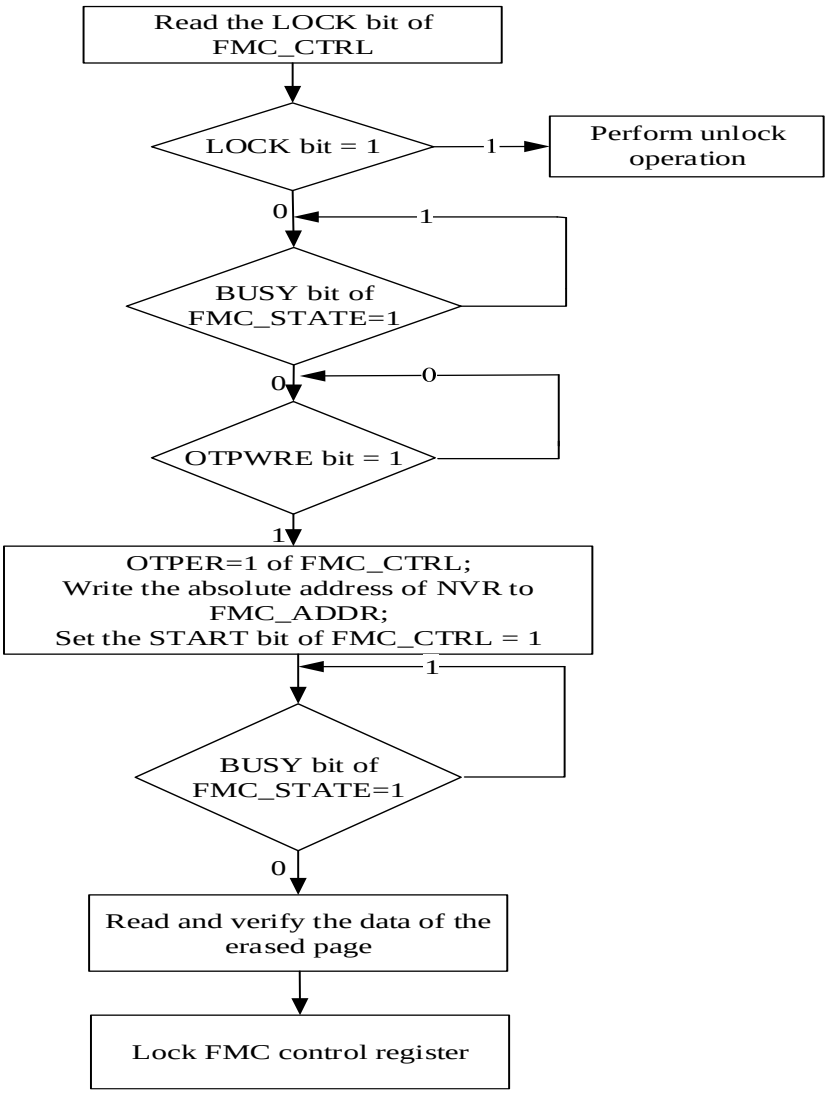
After the erase operation is completed, the page content of the NVR is initialized to a high level. The NVR1/3/4 erasing steps are as follows.

1. Unlock according to NVR1/3/4 unlocking sequence;
2. Make sure the FMC_CTRL register is not locked;
3. Wait for the BUSY bit of the FMC_STATE register to become 0;
4. Wait for the OTPWRE bit of the FMC_CTRL register to be 1;
5. Set the OTPER bit in the FMC_CTRL register;
6. Write the absolute address of the NVR to be erased to the FMC_ADDR register;
7. Send the optional byte block erase command to FMC by setting the START bit of the

FMC_CTRL register to 1;

8. Wait for the erase command to be executed and the BUSY bit of the FMC_STATE register is cleared to 0;
9. If necessary, the CPU can read to verify whether the erase is successful;
10. After completing the relevant operations, re-lock the FMC control register to prevent FLASH from being misoperated.

When the NVR1/3/4 erase is successfully executed, the END bit of the FMC_STATE register is set.



4.4.5. NVR1/3/4 programming

FMC provides a 32-bit whole word programming function, which can be used to modify the content of NVR1/3/4.

The NVR1/3/4 programming operation steps are as follows.

1. Unlock according to NVR1/3/4 unlocking sequence;
2. Make sure the FMC_CTRL register is not locked;
3. Wait for the BUSY bit of the FMC_STATE register to become 0;
4. Wait for the OTPWRE bit of the FMC_CTRL register to be 1;
5. Set the OTPPG bit in the FMC_CTRL register;
6. The CPU writes a 32-bit whole word to the destination address;

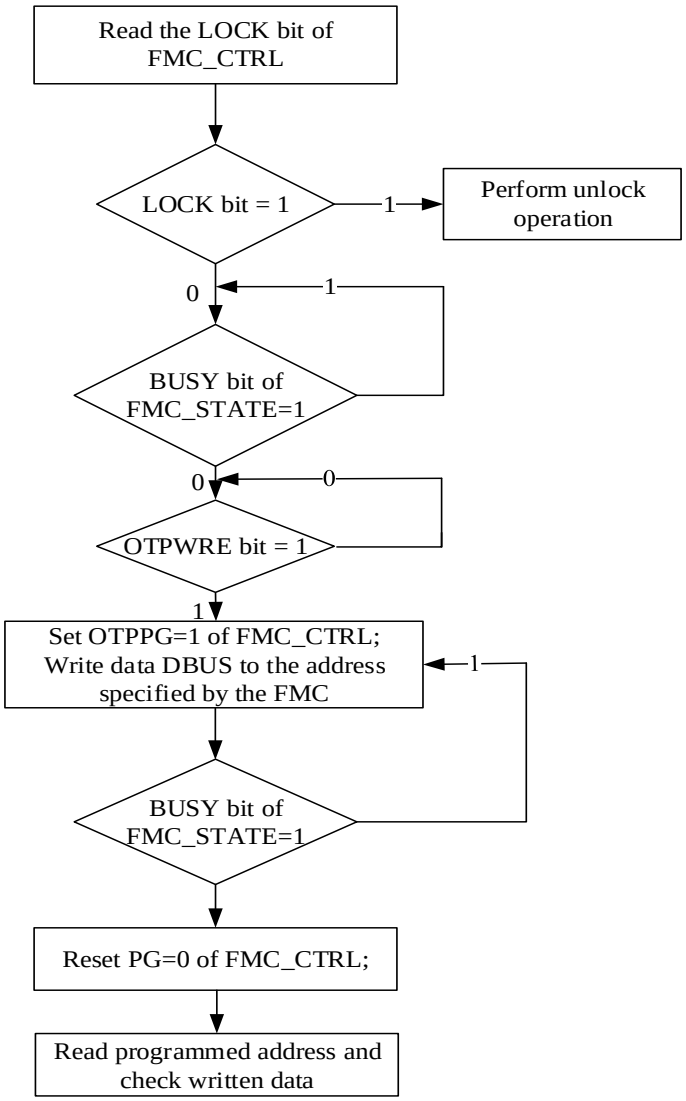
When programming the option byte area (NVR1/3), Bit[31:16] must be the one's complement of Bit[15:0], otherwise the default value will be restored.

7. Wait for the completion of the programming instruction, and the BUSY bit of the FMC_STATE register is cleared to 0;
8. Reset the OTPPG bit of the FMC_CTRL register;
9. If necessary, verify whether the programming is successful by reading the CPU (little endian mode);
10. After completing the relevant operations, re-lock the FMC control register to prevent FLASH from being misoperated.

When the NVR1/3/4 programming is successfully executed, the END bit of the FMC_STATE register is set.

Programming error

When performing a whole word programming operation, the hardware checks whether the data at the destination address is 0xFFFF. When the data at the address is not 0xFFFF, a programming error occurs, PGERR is set to 1, and programming is aborted.


Read the LOCK bit of FMC_CTRL
LOCK bit = 1
1
Perform unlock operation
0
1
BUSY bit of FMC_STATE=1
0
0
OTPWRE bit = 1
1
Set OTPPG=1 of FMC_CTRL; Write data DBUS to the address specified by the FMC
1
BUSY bit of FMC_STATE=1
Reset PG=0 of FMC_CTRL;
Read programmed address and check written data

4.5. NVR2 read

The NVR2 (information block) module stores the configuration word, RB80k calibration value, ADC calibration value and electronic signature of the device, etc. It needs to be unlocked to read, and the user cannot change it. The Information block key register (FMC_INFKEY) is used to unlock the read protection.

The reading steps are as follows:

1. Write KEY1 = 0x45670123 to the FMC key register (FMC_KEY);
2. Write KEY2 = 0xCDEF89AB to the FMC key register (FMC_KEY);
3. Write 0x896DBA23 to the Information block key register (FMC_INFKEY);
4. Write 0x7EA56C8F to the Information block key register (FMC_INFKEY);
5. Read the data;
6. Configure the FMC control register FMC_CTRL. LOCK = 1 to lock the NVR2 module.

4.6. Registers

Base address: 0x5001 0000

Address offset	Register	Description
0x00	FMC_KEY	FMC key register
0x04	FMC_OTPKEY	FMC option key register
0x08	FMC_STATE	FMC status flag register
0x0C	FMC_CTRL	FMC control register
0x10	FMC_ADDR	FMC address register
0x14	FMC_PER0KEY	The first 4 pages of FLASH key register
0x18	FMC_INFKEY	Information block key register
0x1C	FMC_CFG	CFG register
0x20	FMC_WRP0	Write protection register 0
0x24	FMC_WRP1	Write protection register 1
0x28	FMC_WRP2	Write protection register 2
0x2C	FMC_WRP3	Write protection register 3

The following registers are configuration word registers

Address offset	Register	RW	Description	Defaults
0x0100	CFG00_REG	R	Configuration word 00 register	0x0000 1FFF
0x0104	CFG01_REG	R	Configuration word 01 register	0x0000 0DA9
0x0108	CFG02_REG	R	Configuration word 02 register	0x0000 01A9
0x010C	CFG03_REG	R	Configuration word 03 register	0x0000 00FF
0x0110	CFG04_REG	R	Configuration word 04 register	0x0000 0018
0x0114	CFG05_REG	R	Configuration word 05 register	0x0000 77FF
0x0118	CFG06_REG	R	Configuration word 06 register	0x0000 FFFF
0x011C	CFG07_REG	R	Configuration word 07 register	0x0000 FFFF
0x0120	CFG08_REG	R	Configuration word 08 register	0x0000 FFFF
0x0124	CFG09_REG	R	Configuration word 09 register	0x0000 0FFF
0x0128	CFG10_REG	R	Configuration word 10 register	0x0000 007F
0x012C	CFG20_REG	R	Configuration word 20 register	0x0000 00FF
0x0130	CFG21_REG	R	Configuration word 21 register	0x0000 07FF
0x0134	CFG22_REG	R	Configuration word 22 register	0x0000 FFFF
0x0138	CFG23_REG	R	Configuration word 23 register	0x0000 FFFF
0x013C	CFG24_REG	R	Configuration word 24 register	0x0000 FFFF
0x0140	CFG25_REG	R	Configuration word 25 register	0x0000 FFFF
0x0144	CFG26_REG	R	Configuration word 26 register	0x0000 FFFF
0x0148	CFG27_REG	R	Configuration word 27 register	0x0000 FFFF
0x014C	CFG28_REG	R	Configuration word 28 register	0x0000 FFFF
0x0150	CFG29_REG	R	Configuration word 29 register	0x0000 FFFF

4.6.1. FMC key register (FMC_KEY)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[31:16]															
W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
W															

31:0	KEY[31:0]	Unlock for FLASH erasing and programming The register is write-only, the read back data is 0 Unlock method: Write KEY1 = 0x45670123, KEY2 = 0xCDEF89AB in sequence Note: Only after configuring this register successfully, other FMC registers can be configured

4.6.2. FMC option key register (FMC_OTPKEY)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OTPKEY [31:16]															
W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OTPKEY [15:0]															
W															

31:0	OTPKEY[31:0]	Used to unlock NVR1/3/4 erasing and programming The register is write-only, the read back data is 0 Unlock method: Write sequentially 0x45670123, 0xCDEF89AB

4.6.3. FMC status flag register (FMC_STATE)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										END	WPERR	Res.	PGERR	Res.	BUSY
										RC_W1	RC_W1		RC_W1		RC_W1

31:6	-	Reserved
5	END	Operation end flag After the operation is successfully executed, this bit is set by hardware Software write 1 to clear 0
4	WPERR	Erase/program protection error flag During an erase/program operation on a protected page, this bit is set by hardware Software write 1 to clear 0
3	-	Reserved
2	PGERR	Programming error flag When the state of the programmed area is not 0xFFFF, program the flash memory, this bit is set by hardware Software write 1 to clear 0
1	-	Reserved
0	BUSY	Busy, this bit indicates that the flash memory operation is in progress At the beginning of the flash memory operation, this bit is set to '1' At the end of the operation or an error occurs, this bit is cleared to '0'

4.6.4. FMC control register (FMC_CTRL)

Address offset: 0x0C

Reset value: 0x0000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	INFRDE	Res.	OTPWRE	PER0WRE	LOCK	START	OTPER	OPTPG	Res.	MER	PER	PG
	RC_W0		RC_W0	RC_W0	RS	RS	RW	RW		RW	RW	RW

31:12	-	Reserved
11	INFRDE	Allow reading information block When this bit is '1', the read operation of the information block is allowed When the correct key sequence is written in the FMC_INFKEY register, this bit is set to '1' Software can write 0 to clear this bit

10	-	Reserved
9	OTPWRE	Allow write NVR1/NVR3/NVR4 When this bit is '1', the NVR1/NVR3/NVR4 can be programmed When the correct key sequence is written in the FLASH_OPTKEY register, this bit is set to '1' Software can write 0 to clear this bit
8	PER0WRE	Allow to write the first 4 pages of flash When this bit is '1', the option byte can be programmed When the correct key sequence is written in the FLASH_PER0KEY register, this bit is set to '1' Software can write 0 to clear this bit
7	LOCK	Lock Only '1' can be written. When this bit is '1', it means the FLASH_CTRL register is locked This bit is cleared by hardware after a correct unlock sequence is detected If the unlock operation fails, this bit remains set until the next system reset
6	START	Send erase command bit to FLASH Set by software to send erase command to FLASH When the BUSY bit is cleared to 0, this bit is cleared to 0 by hardware When there is no valid erase command, after START writes 1, it can only be completed by writing FLASH command or cleared by error
5	OTPER	Erase NVR1/NVR3/NVR4 1 effective Set and cleared by software
4	OPTPG	Programming NVR1/NVR3/NVR4 1 effective Set and cleared by software
3	-	Reserved
2	MER	Option to erase all pages of main memory block 1 effective Set and cleared by software
1	PER	Select erase main memory block pages 1 effective Set and cleared by software
0	PG	Main memory block programming operation 1 effective Set and cleared by software

4.6.5. FMC address register (FMC_ADDR)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FMC_ADDR[31:16]															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FMC_ADDR[15:0]															
RW															

31:0	FMC_ADDR[31:0]	Flash erase address, this bit is set by software The ADDR bit is the address of the flash erase command

4.6.6. The first 4 pages of FLASH key register (FMC_PER0KEY)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PER0KEY [31:16]															
W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PER0KEY [15:0]															
W															

31:0	PER0KEY	Used to erase and program the first 4 pages of FLASH to unlock The register is write-only, the read back data is 0 Unlock method: Write sequentially 0x45670123, 0xCDEF89AB

4.6.7. Information block key register (FMC_INFKEY)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INFKEY [31:16]															
W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INFKEY [15:0]															
W															

31:0	INFKEY[31:0]	Used to unlock the read protection of the information block 0x0002_0200~0x0002_03FF The register is write-only, the read back data is 0 Unlock method: Write sequentially 0x896DBA23, 0x7EA56C8F

4.6.8. CFG register (FMC_CFG)

Address offset: 0x1C

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	WDT_EN	WDTRST_SD	WDTRST_SP	RDPRT	CFGERR
	R	R	R	R	R

31:5	-	Reserved
1	RDPRT	Read protection, read-only bit 1: Indicates that the flash memory is read protected 0: Indicates that the flash memory is not read-protected
0	CFGERR	Reset read configuration word error, read-only bit When this bit is 1, it means that the configuration word and its inverse code do not match

4.6.9. Write protection register 0 (FMC_WRP0)

Address offset: 0x20

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRP1															
R															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRP0															
R															

31:16	WRP1	Write protection of pages 0~63 of the main memory block Bit[0]: Write protection for pages 0 and 1 Bit[1]: Write protection for pages 2 and 3

15:0	WRP0	… Bit[31]: Write protection for pages 62 and 63 0: The corresponding two-page write protection takes effect 1: The corresponding two-page write protection is invalid This register contains the write protection option byte loaded by SYS

4.6.10. Write protection register 1 (FMC_WRP1)

Address offset: 0x24

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRP3															
R															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRP2															
R															

31:16	WRP3	Write protection of pages 64~127 of the main memory block Bit[0]: Write protection for pages 64 and 65 Bit[1]: Write protection for pages 66 and 67 …
15:0	WRP2	Bit[31]: Write protection for pages 126 and 127 0: The corresponding two-page write protection takes effect 1: The corresponding two-page write protection is invalid This register contains the write protection option byte loaded by SYS

4.6.11. Write protection register 2 (FMC_WRP2)

Address offset: 0x28

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRP5															
R															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRP4															
R															

31:16	WRP5	Write protection of pages 128~191 of the main memory block Bit[0]: Write protection for pages 128 and 129 Bit[1]: Write protection for pages 130 and 131 … Bit[31]: Write protection for pages 190 and 191 0: The corresponding two-page write protection takes effect 1: The corresponding two-page write protection is invalid This register contains the write protection option byte loaded by SYS
15:0	WRP4	

4.6.12. Write protection register 3 (FMC_WRP3)

Address offset: 0x2C

Reset value: 0xFFFF FFFF

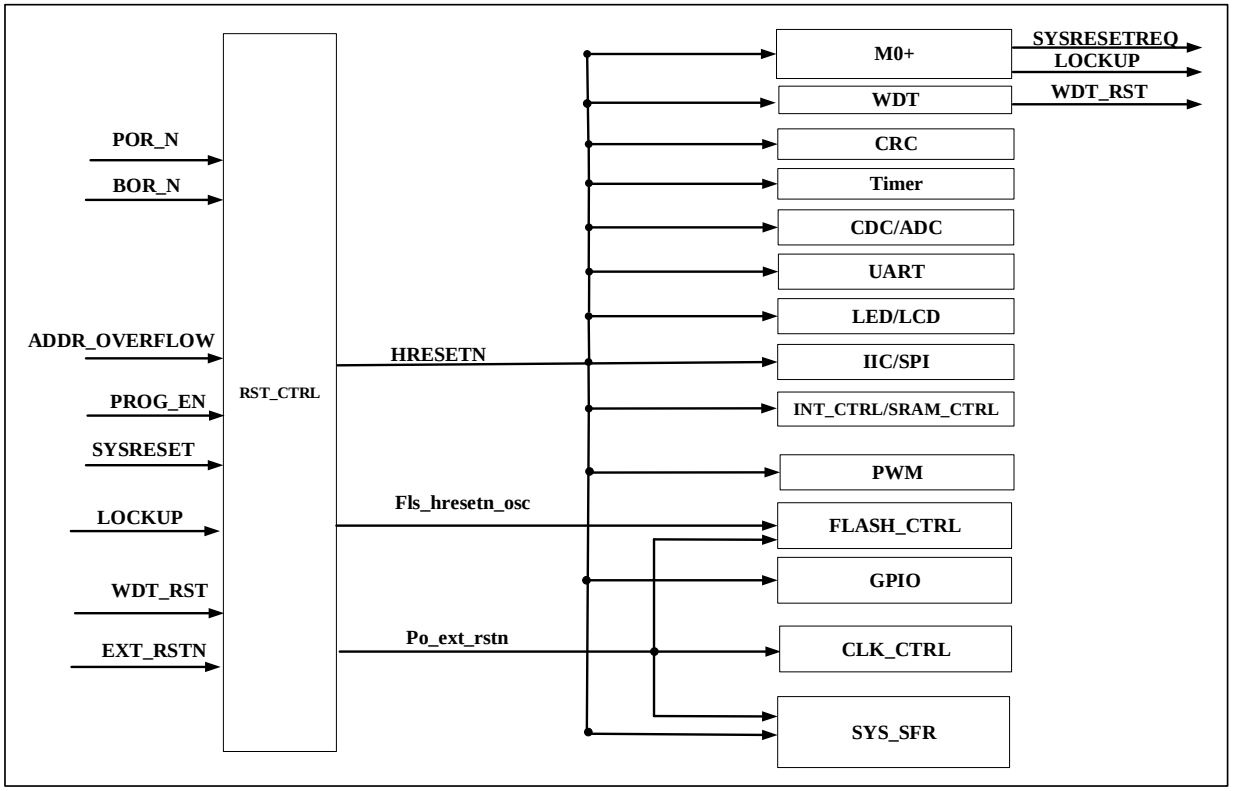
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WRP7															
R															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRP6															
R															

31:16	WRP7	Write protection of pages 192~255 of the main memory block Bit[0]: Write protection for pages 192 and 193 Bit[1]: Write protection for pages 194 and 195 … Bit[31]: Write protection for pages 254 and 255 0: The corresponding two-page write protection takes effect 1: The corresponding two-page write protection is invalid This register contains the write protection option byte loaded by SYS
15:0	WRP6	

5 Reset

There are 8 reset sources in BF7807AMXX: Power-on reset, power-down reset, watchdog reset, CPU software reset, CPU lock reset, external reset, PC pointer overflow reset, and FLASH programming reset. As long as any one of these resets occurs, the system's global reset signal resets the entire chip. The reset flag register (RST_STATE) can be used to determine what kind of reset the chip has performed. The reset flag bit needs to be cleared by software.



Reset block diagram

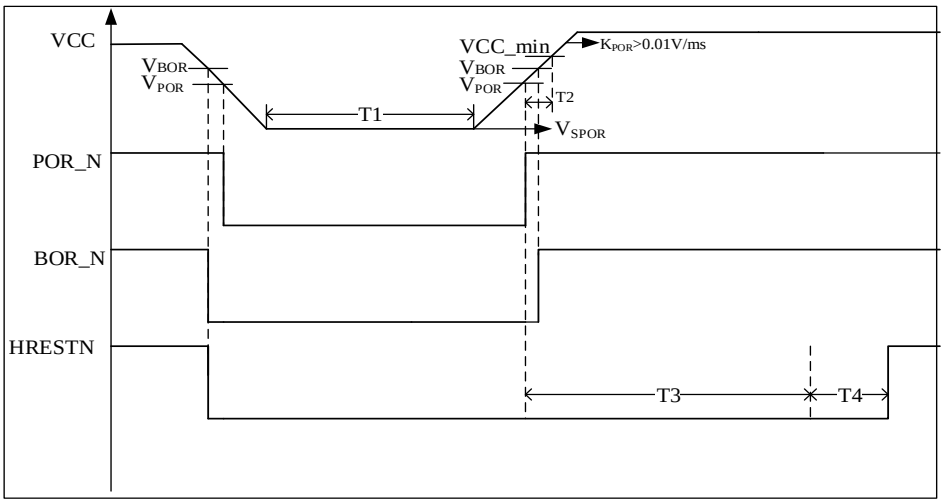
5.1. Reset introduction

5.1.1. Power-on/Power-down reset

POR_N: After the system is powered on, after the POR reset module releases the POR reset signal, the global reset signal lasts for 93ms, and after the global reset signal continues to be valid for 5ms, the system exits the reset mode.

BOR_N: After the system is powered off, the BOR reset module generates a BOR reset signal, the global reset signal is valid for 5ms.

Power-up/power-down sequence:



Power-on reset schematic

Power-on/power-down reset parameters:

Symbol	Parameter	Test Conditions		Min	Typical	Max	Unit
		VCC	Temperature				
V_{SPOR}	Power-on reset start voltage	-	25℃	-	-	300	mV
K_{POR}	Power-on reset voltage rate	-	25℃	0.01	-	-	V/ms
V_{POR}	Power-on reset voltage, 0.3V hysteresis	-	25℃	1.1	1.4	2.2	V
V_{BOR}	Power-down reset voltage (±10%), 0.2V hysteresis	-	25℃	-	V_{BOR}	-	V
VCC_min	Minimum working voltage	-	25℃	2.7	-	-	V
T1	VCC hold V_{SPOR} time	-	25℃	0.1	-	-	ms
T2	V_{POR} to VCC_min time	-	25℃	-	-	0.6*T3	ms
T3	Analog POR module delay time	-	25℃	75	93	112	ms
T4	Global reset valid time	-	25℃	-	5	-	ms

Power-on reset characteristic parameter table

Note: The V_{BOR} power-down reset voltage is selected by the VTH_SEL bits of the BOR controller BOR_CFG[4:2].

5.1.2. Watchdog timer overflow reset

WDT_RST: After the watchdog timer overflows, the global reset is performed for 5ms. After 5ms, the system exits the reset mode.

5.1.3. CPU software reset

SYSRESETREQ: The soft reset signal is valid by writing the CPU register, and the global reset signal is valid for 5ms. After 5ms, the system exits the reset mode (AIRCR. SYSRESETREQ, writing 1 will activate the external SYSRESETREQ signal).

5.1.4. CPU lockup reset

LOCKUP: If there is an error in program execution, the CPU is locked to make the reset signal valid, and the global reset signal is valid for 5ms. After 5ms, the system exits the reset mode.

5.1.5. External reset

RESET_N: When the external reset pin detects a low level, a system reset is generated. The reset pin has a built-in 33k pull-up resistor. The signal added to this pin should exceed 300us to ensure reliable reset of the chip.

The chip port is connected to an external reset signal. When reset_n is low, the entire chip is in a reset state. After it becomes high, the global reset signal continues to be valid for 5ms. After 5ms, the system exits the reset mode.

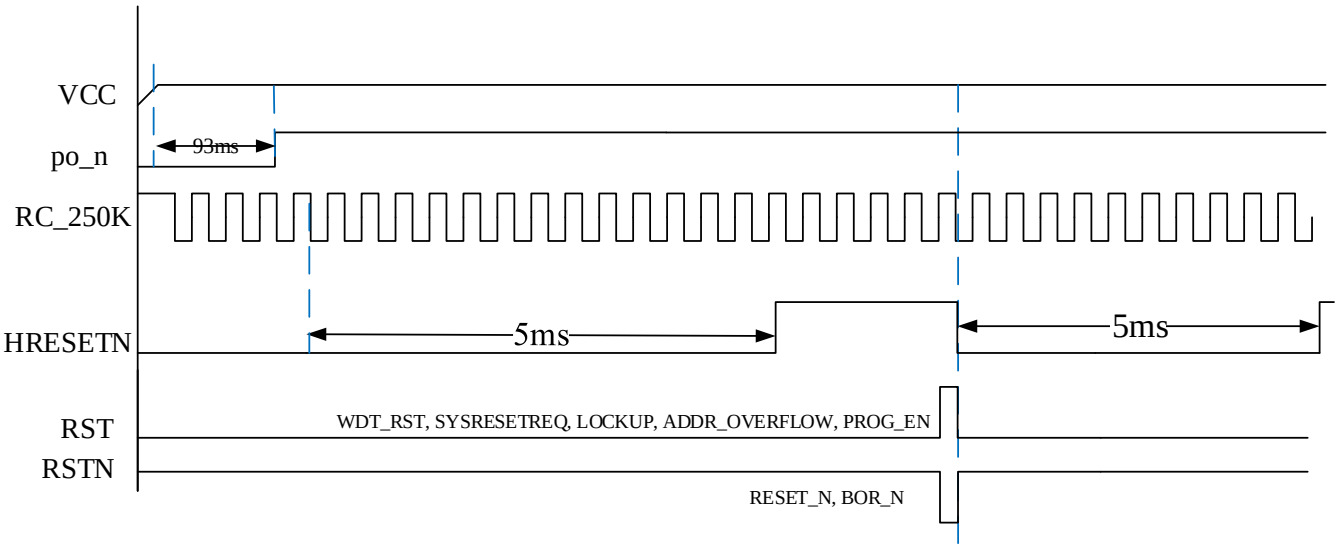
5.1.6. PC pointer overflow reset

ADDR_OVERFLOW: 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 rising edge of the sys_clk clock detects the high level of addr_overflow (it takes 1 clock cycle) to reset the global for 5ms, and the reset signal will reset the addr_overflow signal. The signal is cleared, and after 5ms, the system exits the reset mode.

5.1.7. FLASH programming reset

PROG_EN: When prog_en is high, it is the programming mode of FLASH. At this time, the global reset signal is valid. After it becomes low, the global reset signal continues to be valid for 5ms.

5.2. Reset timing diagram



Reset sequence description:

1. When the chip has a power-on reset, 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), after completing the programming, exit the programming mode. After a delay of 5ms, and HRESETN are pulled high. The chip enters normal operation.
3. In normal operation, when any one of watchdog reset, address overflow reset, soft reset, and LOCKUP occurs, HRESETN is pulled low, after a delay of 5ms, HRESETN is pulled high. The chip enters normal operation.
4. After normal work, you can no longer enter the programming mode.
5. In normal operation, after an external reset RESET_N occurs, after filtering 4 clocks with LIRC32k, HRESETN is pulled low, after RESET_N is pulled high, and after a delay of 5ms, HRESETN is pulled high. The chip enters normal operation.
6. In normal operation, after BO_N occurs, HRESETN is pulled low, after BO_N is pulled high, and after a delay of 5ms, HRESETMN is pulled high. The chip enters normal operation.

5.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x08	RST_STATE	Reset flag register
0x14	BOR_CFG	BOR configuration register

5.3.1. Reset flag register (RST_STATE)

Address offset: 0x08

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Reserved	PROG	ADDROF	EXTI_RSTN	LOCKUP	SYSRESET	WDTRST	BOR	POR
	RC_W0							

31:8	-	Reserved
7	PROG	FLASH programming reset flag, software can write 0 to clear the corresponding flag bit 1: FLASH programming reset occurs 0: No FLASH programming reset occurs
6	ADDROF	PC pointer overflow reset flag, software can write 0 to clear the corresponding flag bit 1: PC pointer overflow reset occurs 0: No PC pointer overflow reset occurs
5	EXTI_RSTN	External reset flag, software can write 0 to clear the corresponding flag bit 1: External reset occurred 0: No external reset occurred
4	LOCKUP	CPU lock reset flag, software can write 0 to clear the corresponding flag bit 1: CPU lock reset occurred 0: No CPU lock reset occurs
3	SYSRESET	CPU software reset flag, software can write 0 to clear the corresponding flag bit 1: CPU software reset occurred 0: No CPU software reset occurs
2	WDTRST	Watchdog reset flag, software can write 0 to clear the corresponding flag bit 1: Watchdog reset occurred 0: No watchdog reset occurs
1	BOR	Power-down reset flag, software can write 0 to clear the corresponding flag bit

		1: Power-down reset occurred 0: No power-down reset occurs
0	POR	Power-on reset flag, software can write 0 to clear the corresponding flag bit 1: Power-on reset occurred 0: No power-on reset occurred

5.3.2. BOR configuration register (BOR_CFG)

Address offset: 0x14

Reset value: 0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	VTH_SEL			DELAY_SEL	Res.
	RW			RW	

31:5	-	Reserved
4:2	VTH_SEL	BOR power-down threshold configuration register 000: Reserved 001: 2.8V 010: 3.3V 011: 3.7V 1xx: 4.2V After power-on reset, the BOR threshold defaults to 2.3V, and the program configures the above gears See table "Threshold and delay selection"
1	DELAY_SEL	BOR's Power-Down Delay Configuration Register 0: Delay selection 1 1: Delay option 2 See table "Threshold and delay selection"
0	-	Reserved

DELAY_SEL	VTH_SEL	Power down threshold (V)	Recovery threshold (V)	Hysteresis (mV)	Delay (μs)
0	001	2.8	2.6	140.3	84.0
	010	3.3	3.4	144.5	97.7
	011	3.7	3.8	121	107.3
	1XX	4.2	4.3	129.7	117.3
1	001	2.8	2.9	144.5	168
	010	3.3	3.4	149.5	195.9
	011	3.7	3.8	126.4	215.3
	1XX	4.2	4.3	135.6	235.6

Table BOR threshold and delay selection

6 System working mode

6.1. Working mode description

The working mode of BF7807AMXX: Active mode, debug mode, standby mode.

- **Active Mode**

The RC1M, PLL, HCLK, LIRC work, XTAL depends on software settings. The core is running, the peripherals keep working normally, and the functions of each peripheral are controlled by software configuration.

- **Debug Mode**

Debugging is performed by transmitting commands through the SWD port.

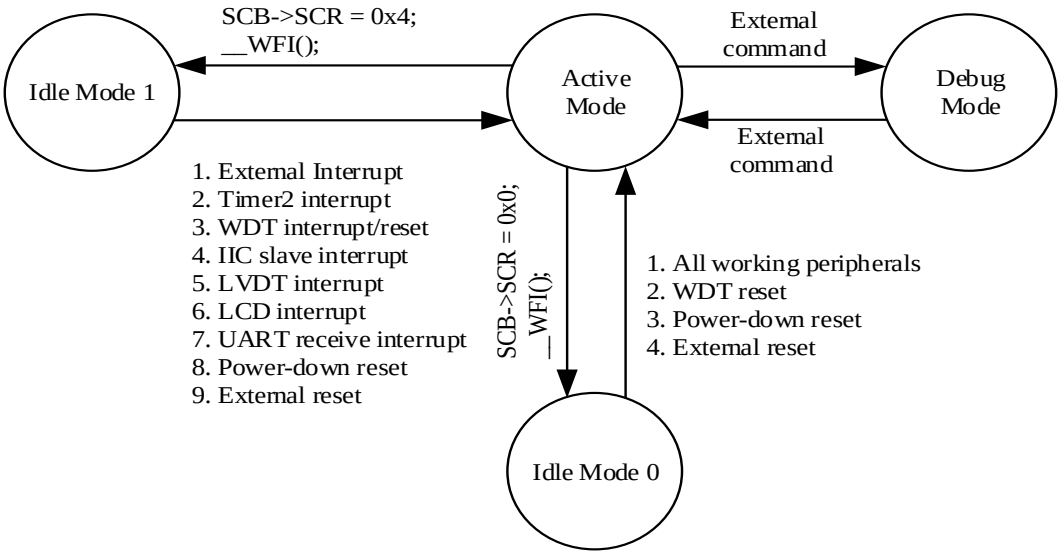
- **Standby mode is divided into idle mode 0 and idle mode 1**
  - Idle Mode 0

    Only the HCLK clock is turned off, RC1M, PLL and LIRC work, XTAL depends on software settings. The core stops running, and the rest of the peripherals that do not need HCLK as the working clock can work.
  - Idle Mode 1

    RC1M, PLL and HCLK are off, XTAL depends on software settings, LIRC works. The core is stopped and the peripherals work fine using the LIRC clock.

The working status of the clock source:

Work mode	Entry conditions	Effects on the clock	
Active Mode	Wake-up from power-on reset/standby mode	RC1M	Work
		PLL	Work
		HCLK	Work
		LIRC	Work
		XTAL	Depends on software configuration
Idle Mode 0	SCB->SCR = 0x0; __WFI();	RC1M	Work
		PLL	Work
		HCLK	Off
		LIRC	Work
		XTAL	Depends on software configuration
Idle Mode 1	SCB->SCR = 0x4; __WFI();	RC1M	Off
		PLL	Off
		HCLK	Off
		LIRC	Work
		XTAL	Depends on software configuration



Working mode conversion diagram

6.1.1. Active mode

Reduce system clock speed

In active mode, the system clock (HCLK) speed can be reduced by configuring the clock configuration register (CLK_CFG).

Peripheral clock gating

In active mode, peripherals can be stopped at any time to reduce power consumption.

To further reduce power consumption in idle mode, peripheral clocks can be disabled before executing a WFI or WFE instruction. Peripheral clock gating is controlled by the peripheral module clock control register (RCU_EN).

6.1.2. Debug mode

Debugging is performed by transmitting commands through the SWD port.

If the user application puts the MCU in standby mode, debugging will break because the core clock is stopped.

6.1.3. Low power management

The BF7807AMXX saves all CPU states before entering standby mode, SRAM and register contents are preserved, and GPIOs remain in run-time state.

Enter idle mode 0

SCB->SCR = 0x0; __WFI();

Enter idle mode 1

SCB->SCR = 0x4; __WFI();

Status of peripherals in standby mode:

Module	Clock	Standby mode	
		Idle mode 0	Idle mode 1
Cortex-M0+	HCLK	×	×
SRAM	HCLK	×	×
SYS_SFR	HCLK	×	×
SysTick	HCLK	×	×
FLASH	Divide by 2 of HCLK/PLL_48M	√	×
UART0/1/2/3/4	HCLK/ PLL_48M	√	×
PWM0/1/2/3/4	1~128 frequency division of HCLK/PLL_48M	√	×
Timer0/1/3	1~128 frequency division of HCLK/PLL_48M	√	×
WDT	HCLK/LIRC	√	√
Timer2	HCLK/LIRC/ XTAL	√	√
CDC	HCLK/PLL_48M	√	×
ADC	HCLK/PLL_48M	√	×
LED	HCLK/RC1M	√	×
LCD	HCLK/LIRC/RC1M/XTAL	√	√
SPI	HCLK/SPI_CLK	√	×
IIC	HCLK/PLL_48M divide by 2/SCL	√	×
CRC	HCLK	×	×
GPIO	--	√	√

Note: In the figure, √ means support, according to the program configuration. × means unavailable.

Exit idle mode 0

WDT interrupt, external interrupt 0~3 interrupt, IIC slave interrupt, UART0/1/2/3/4 interrupt, Timer0/1/2/3 interrupt, LED interrupt, LCD interrupt, ADC interrupt, CDC interrupt, PWM0/1 /2/3/4 interrupt, LVDT interrupt, any of which can wake up the chip, exit idle mode 0, and the CPU executes the interrupt service routine.

External reset, power-down reset, and WDT reset can also wake up the chip to exit idle mode 0.

Exit idle mode 1

WDT interrupt, external interrupt, IIC slave interrupt, Timer2 interrupt, LCD interrupt, LVDT interrupt, UART receive interrupt, any of which can wake up the chip, exit idle mode 1, and the CPU executes the interrupt service routine.

External reset, power-down reset, and WDT reset can also wake up the chip to exit idle mode 1.

6.2. Register

Enter idle mode 0 or idle mode 1, which is determined by the register SCR.SLEEPDEEP bit.

Address	Register	Description
0xE000ED10	SCR	System control register

Bit	Bit symbol	Description	RW	Reset value
31:3	Reserved	-	-	-
2	SLEEPDEEP	Select standby mode register 0: Enter idle mode 0 1: Enter idle mode 1	R/W	0
1	SLEEPONEXIT	0: Do not enter standby mode (WFI) when exiting exception handling and returning to the program thread 1: When exiting exception handling and returning to the program thread, the processor automatically enters standby mode (WFI)	R/W	0
0	Reserved	-	-	-

7 WDT

7.1. WDT features

- The WDT clock is provided by internal low-speed clock LIRC32kHz, which can work in standby mode
- Timing range: 18ms~2.304s
- In debug mode, it can be configured to automatically pause at the current value and continue counting after exiting

7.2. WDT function overview

Due to the particularity of the system application, the watchdog timer overflow signal is classified:

Watchdog time-out reset: Reset generated in active mode, reset generated by option byte selection in standby mode. If the watchdog timing overflow occurs, the overflow signal is the watchdog overflow reset signal at this time, and the watchdog overflow reset affects the global reset. At this time, the system implements the global reset action and reloads the configuration information.

Watchdog time-out interrupt: Option byte selection generates an interrupt in standby mode. If the watchdog timing overflow occurs, the overflow signal is the watchdog interrupt signal at this time. In this case, no reset occurs, and the interrupt wakes up the chip to exit the sleep mode and execute the watchdog interrupt service function.

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

Note: When the watchdog peripheral clock is turned on, the configuration register WDT_EN=0x55 can turn off the watchdog.

7.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: 0x5001 0000

Address offset	Register	Description
0x1C	FMC_CFG	CFG register

Base address: 0x5005 0400

Address offset	Register	Description
0x00	WDT_EN	Watchdog timer enable configuration register
0x04	WDT_TIME_SEL	Watchdog overflow timing configuration register
0x08	WDT_INT_CFG	Watchdog interrupt register

7.3.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

18	WDT_CLKEN	WDT module operation enable 1: Work 0: Off, the default is 0

7.3.2. CFG register (FMC_CFG)

Address offset: 0x1C

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	WDT_EN	WDTRST_SD	WDTRST_SP	RDPRT	CFGERR
	R	R	R	R	R

31:5	-	Reserved
4	WDT_EN	User option bytes loaded by NVR1

		0: The program is not allowed to turn off the watchdog function 1: Allow the program to turn off the watchdog function
3	WDTRST_SD	User option bytes loaded by NVR1 0: When entering idle mode 1, a watchdog reset is generated 1: When entering idle mode 1, no watchdog reset is generated, wake-up standby, and WDT interrupt is generated
2	WDTRST_SP	User option bytes loaded by NVR1 0: When entering idle mode 0, a watchdog reset is generated 1: When entering idle mode 0, no watchdog reset is generated, wake up standby, and generate WDT interrupt

7.3.3. Watchdog timing enable configuration register (WDT_EN)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								EN[7:0]							
								RW							

31:8	-	Reserved
7:0	EN[7:0]	Watchdog timer enable configuration register When the configuration value of this register is 0x55, the watchdog is turned off, and when it is configured to other values, it is turned on Configuring this register will clear the WDT counter Configuration word control enable is closed

7.3.4. Watchdog overflow timer configuration register (WDT_TIME_SEL)

Address offset: 0x04

Reset value: 0x0000 0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	TIME_SEL[2:0]		
	RW		

31:3	-	Reserved
2:0	TIME_SEL[2:0]	Watchdog overflow timing configuration register, the timing length is as follows: 000: 18ms 001: 36ms 010: 72ms 011: 144ms 100: 288ms 101: 576ms 110: 1152ms 111: 2304ms

7.3.5. Watchdog interrupt register (WDT_INT_CFG)

Address offset: 0x08
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:2	1	0
Reserved	INT_CLR	INT_FLAG
	W	R

31:2	-	Reserved
1	INT_CLR	Interrupt status flag clear register Write 1 to clear INT_FLAG, write only
0	INT_FLAG	Interrupt Status Flag Register 1: Counting complete 0: Count not completed, read only The option byte controls whether an interrupt is generated in standby mode

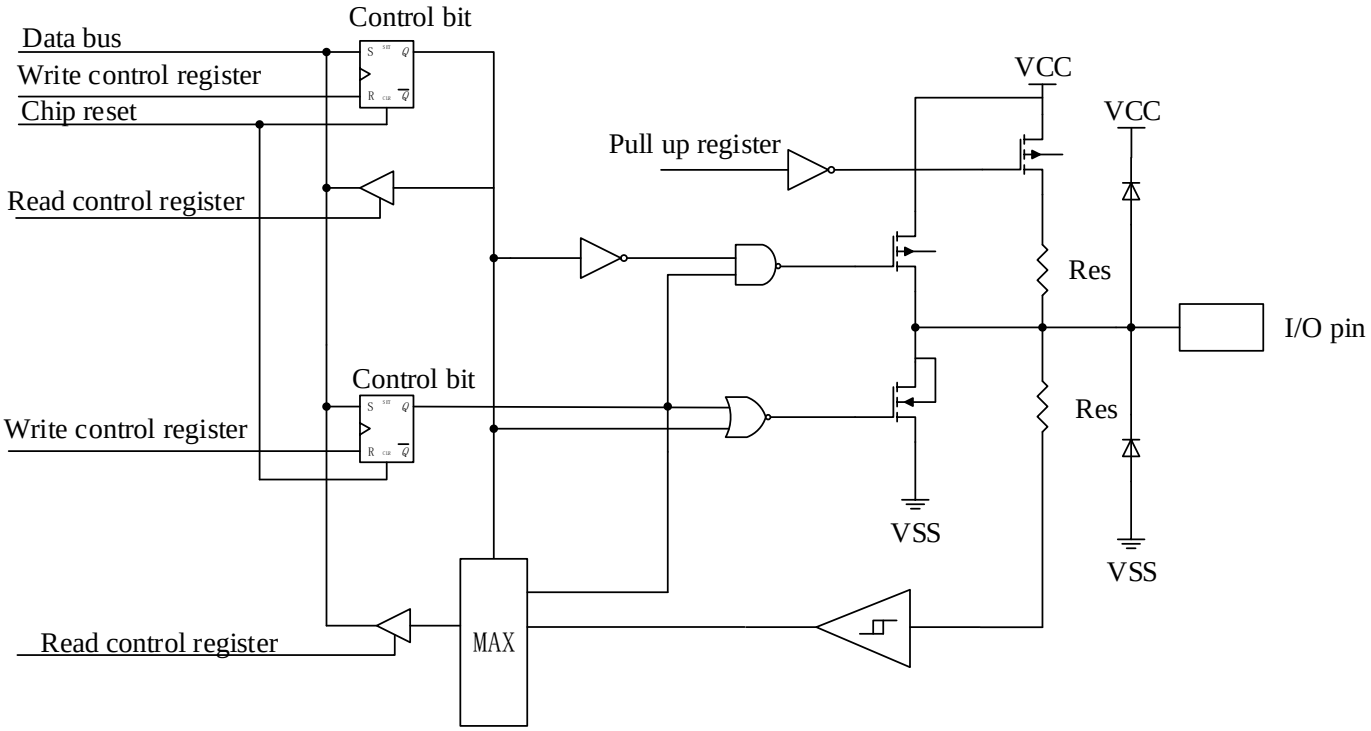
8 GPIO port

8.1. GPIO port description

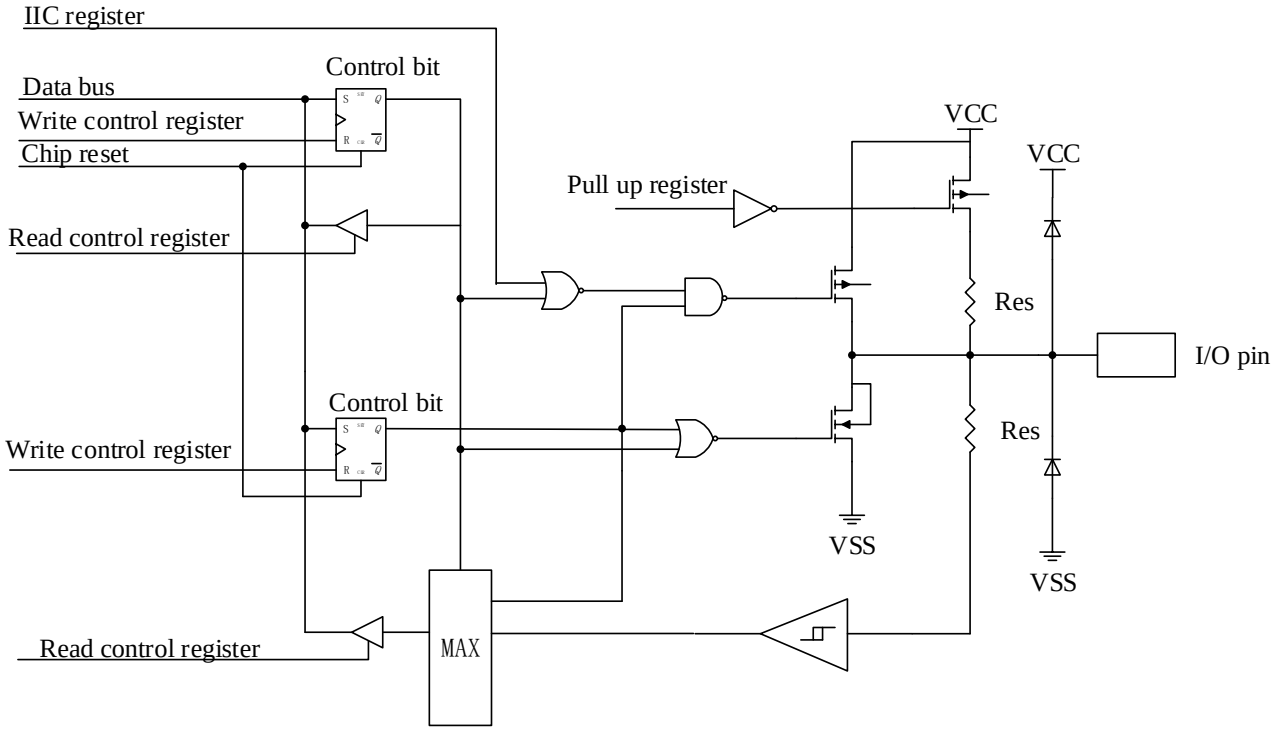
Some pins of the GPIO port are multiplexed with the peripheral functions of the device, and they cannot be configured for multiple functions at the same time, otherwise it will cause functional disorder.

Configure the corresponding registers to achieve the following features:

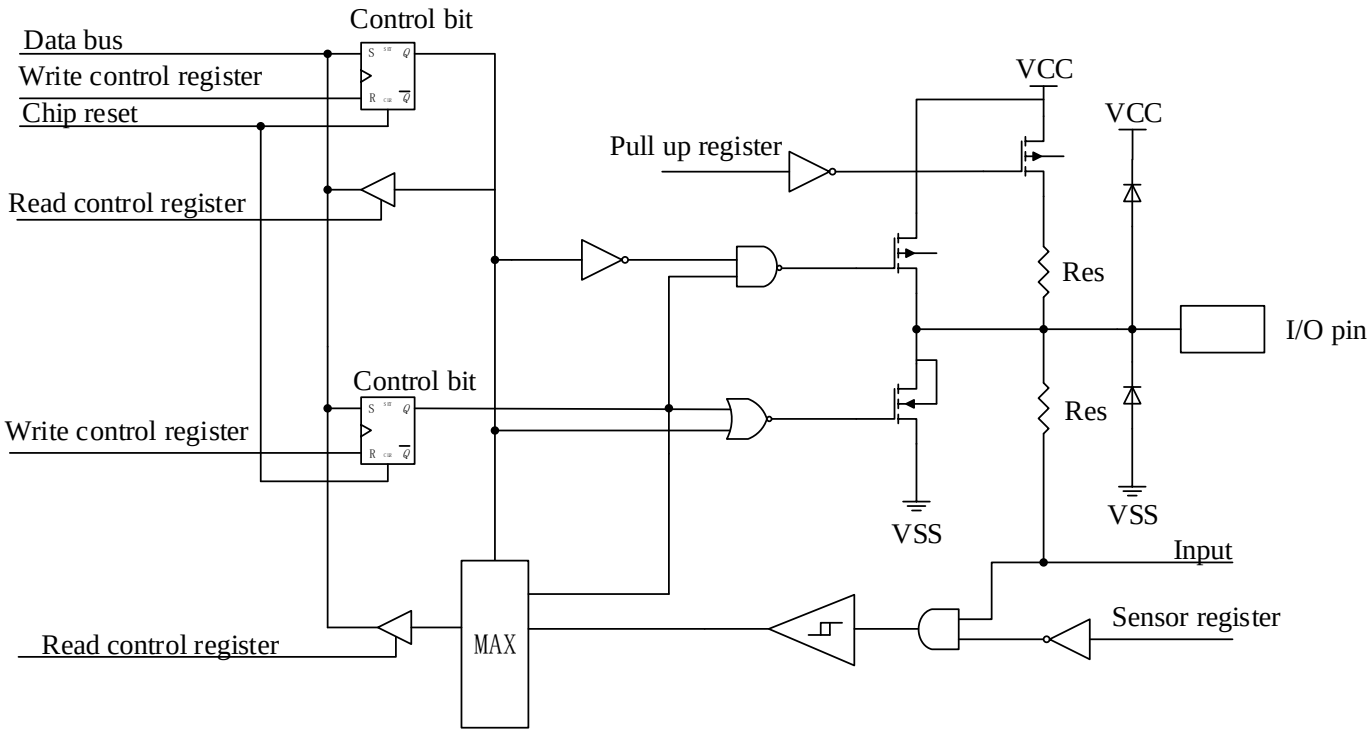
- Direction selection (Px_TRIS)
  Configuration 1 configures the corresponding pin as an input, and clears it to configure the corresponding pin as an output.
- Output high and low level selection (Px_DATA)
  Configuration 1 configures the corresponding pin to output high, and clearing it to 0 configures the corresponding pin to output low.
- Internal pull-up resistor (PU_Px)
  The corresponding pin pull-up resistor of configuration 1 is enabled, and the corresponding pin is cleared to not enable the pull-up resistor, and the pull-up resistor is 33k.
- Open drain output (ODRAIN_EN)
  The pin corresponding to configuration 1 enables open-drain output. Clearing to zero will disable the open-drain output function. After enabling the IIC function, the open-drain output is automatically turned on. IIC/UART recommends using an external pull-up resistor.
  The pin corresponding to configuration 1 enables the high-current drive function, and the pin corresponding to 0 is the IO port function.
- All ports can provide external interrupt, and each interrupt can be configured as rising edge trigger, falling edge trigger, double edge trigger, see chapter "External interrupt" for details.



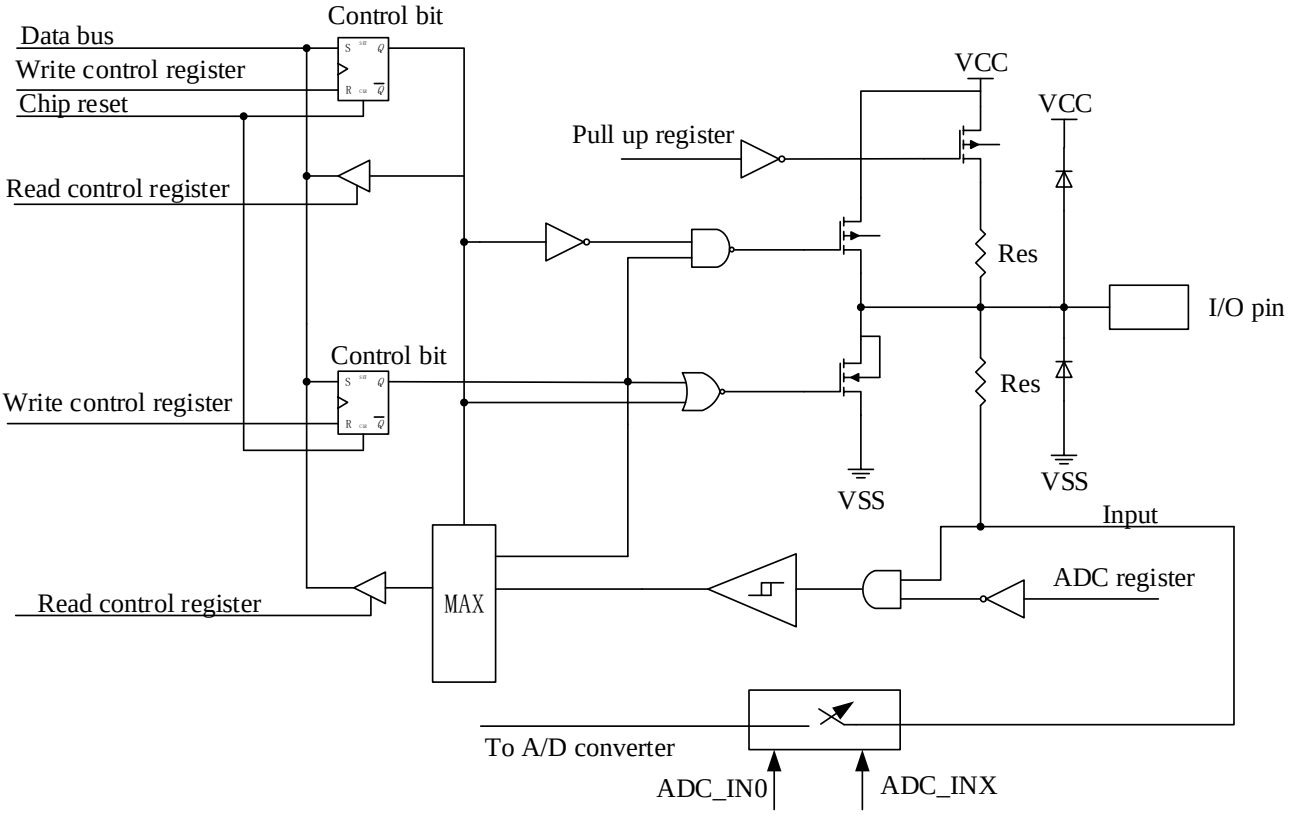
Common IO structure diagram



Open drain output IO structure diagram



SNS IO structure diagram



ADC IO structure diagram

8.2. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: 0x5008 0000

Address offset	Register	Description
0x24	COM_IO_SEL	COM port selection configuration register

Base address: 0x500A 0000

Address offset	Register	Description
0x00	PA_TRIS	PA direction register
0x04	PB_TRIS	PB direction register
0x08	PC_TRIS	PC direction register
0x0C	PD_TRIS	PD direction register
0x10	PA_DATA	PA data register
0x14	PB_DATA	PB data register
0x18	PC_DATA	PC data register
0x1C	PD_DATA	PD data register
0x20	PU_PA	Port PA pull-up resistor control register
0x24	PU_PB	Port PB pull-up resistor control register
0x28	PU_PC	Port PC pull-up resistor control register
0x2C	PU_PD	Port PD pull-up resistor control register
0x30	SW_IO_EN	SWD port selection enable register
0x34	ODRAIN_EN	Open drain output enable register
0x3C	PWM_OUT_EN	PWM output enable register
0x40	UARTx_IO_SEL	UART control register
0x44	IIC_IO_CTRL	IIC control register
0x48	SPI_CLK_CFG	SPI input clock configuration register
0x4C	HSPEED_EN	SPI high-speed mode enable register
0x50	SPI0_IO_CTRL	SPI select enable register

8.2.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

22	GPIO_CLKEN	GPIO module operation enable 1: Work 0: Off, the default is 0

8.2.2. Direction registers

8.2.2.1. PA direction register (PA_TRIS)

Address offset: 0x00

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:16	-	Reserved
15:0	TR[15:0]	Bit[15]~ Bit[0]: Direction of PA15~PA0 port pins 0: Output; 1: Input

8.2.2.2. PB direction register (PB_TRIS)

Address offset: 0x04

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	Res.
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

31:16	-	Reserved
15:1	TR[15:1]	Bit[15]~ Bit[1]: PB15~PB1 port pin direction 0: Output; 1: Input
0	-	Reserved

8.2.2.3. PC direction register (PC_TRIS)

Address offset: 0x08

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:16	-	Reserved
15:0	TR [15:0]	Bit[15]~ Bit[0]: PC15~PC0 port pin direction 0: Output; 1: Input

8.2.2.4. PD direction register (PD_TRIS)

Address offset: 0x0C

Reset value: 0x0000 0FFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
				RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:12	-	Reserved
11:0	TR[11:0]	Bit[11]~ Bit[0]: PD11~PD0 port pin direction 0: Output; 1: Input

8.2.3. Data registers

8.2.3.1. PA data register (PA_DATA)

Address offset: 0x10

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:16	-	Reserved
15:0	D[15:0]	Bit[15]~ Bit[0]: PA15~PA0 data register Configurable output level when IO port of PA group is used as GPIO port The read value is the current level state of the IO port (input) or the configured output value (output)

8.2.3.2. PB data register (PB_DATA)

Address offset: 0x14

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	Res.
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

31:16	-	Reserved
15:1	D[15:1]	Bit[15]~ Bit[1]: PB15~PB1 data register Configurable output level when PB group IO port is used as GPIO port The read value is the current level state of the IO port (input) or the configured output value (output)
0	-	Reserved

8.2.3.3. PC data register (PC_DATA)

Address offset: 0x18

Reset value: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:16	-	Reserved
15:0	D[15:0]	Bit[15]~ Bit[0]: PC15~PC0 data register

		Configurable output level when IO port of PC group is used as GPIO port The read value is the current level state of the IO port (input) or the configured output value (output)

8.2.3.4. PD data register (PD_DATA)

Address offset: 0x1C

Reset value: 0x0000 0FFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
				RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:12	-	Reserved
11:0	D[11:0]	Bit[11]~ Bit[0]: PD11~PD0 data register Configurable output level when PD group IO port is used as GPIO port The read value is the current level state of the IO port (input) or the configured output value (output)

8.2.4. Pull-up resistor control registers

8.2.4.1. PA pull-up resistor control register (PU_PA)

Address offset: 0x20

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:16	-	Reserved
15:0	PU[15:0]	Bit[15]~ Bit[0]: PA15~PA0 pull-up resistor control register 1: PAx port pull-up resistor is enabled 0: PAx port pull-up resistor is not enabled

8.2.4.2. PB pull-up resistor control register (PU_PB)

Address offset: 0x24

Reset value: 0x0000 3000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	Res.
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

31:16	-	Reserved
15:1	PU[15:1]	Bit[15]~ Bit[1]: PB15~PB1 pull-up resistor control register 1: PBx pull-up resistor is enabled 0: PBx pull-up resistor is not enabled
0	-	Reserved

8.2.4.3. PC pull-up resistor control register (PU_PC)

Address offset: 0x28

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:16	-	Reserved
15:0	PU[15:0]	Bit[15]~ Bit[0]: PC15~PC0 pull-up resistor control register 1: PCx pull-up resistor is enabled 0: PCx pull-up resistor is not enabled

8.2.4.4. PD pull-up resistor control register (PU_PD)

Address offset: 0x2C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

Reserved	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:12	-	Reserved
11:0	PU[11:0]	Bit[11]~ Bit[0]: PD11~PD0 pull-up resistor control register 1: PDx pull-up resistor is enabled 0: PDx pull-up resistor is not enabled

8.2.5. Open drain output enable register (ODRAIN_EN)

Address offset: 0x34

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												OD3	OD2	OD1	OD0
												RW	RW	RW	RW

31:4	-	Reserved
3	OD3	PB13 port open drain output enable 1: Open drain output 0: CMOS output Note: When IIC is enabled, the open-drain output enable of the corresponding port is automatically turned on, 2-way IIC mapping.
2	OD2	PB12 port open drain output enable 1: Open drain output 0: CMOS output
1	OD1	PA13 port open drain output enable 1: Open drain output 0: CMOS output
0	OD0	PA12 port open drain output enable 1: Open drain output 0: CMOS output

8.2.6. COM port selection configuration register (COM_IO_SEL)

Address offset: 0x24

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Reserved	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	RW	RW	RW	RW	RW	RW	RW	RW

31:8	-	Reserved
7:0	COM[7:0]	COM port selection configuration register, bit[0]~bit[7] correspond to PA0~PA7 ports 1: Select COM port mode 0: Select IO port mode Note: This register is valid when selecting LED row-column matrix mode, valid when selecting high-current IO port driver enable, and invalid in other cases

8.2.7. IO multiplex registers

8.2.7.1. SWD port selection enable register (SW_IO_EN)

Address offset: 0x30

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:1	0
Reserved	SW_SEL
	RW

31:1	-	Reserved
0	SW_SEL	SWD port selection enable 1: PB12/PB13 are SWD functions 0: PB12/PB13 are GPIO functions

8.2.7.2. PWM output enable register (PWM_OUT_EN)

Address offset: 0x3C

Reset value: 0x000 0000

The output of different PWM modules on the same port has priority: PWM0 prior to PWM1

31:17	16
Reserved	PWM4A
	RW

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

PWM 3A1	PW M3A	PWM 2A1	PW M2A	PWM 1A1	PW M1E	PW M1D	PW M1C	PW M1B	PW M1A	PWM 0A1	PW M0E	PW M0D	PW M0C	PW M0B	PW M0A
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

16	PWM4	PWM4A port output enable 1: Output, 0: No output
15:14	PWM3x[1:0]	Bit[0]: PWM3A port output enable Bit[1]: PWM3A1 port output enable The corresponding bits of PWM3 are: 1: Output, 0: No output
13:12	PWM2x[1:0]	Bit[0]: PWM2A port output enable Bit[1]: PWM2A1 port output enable The corresponding bits of PWM2 are: 1: Output, 0: No output
11:6	PWM1x[5:0]	Bit[0]: PWM1A port output enable Bit[1]: PWM1B port output enable Bit[2]: PWM1C port output enable Bit[3]: PWM1D port output enable Bit[4]: PWM1E port output enable Bit[5]: PWM1A1 port output enable The corresponding bits of PWM1 are: 1: Output, 0: No output
5:0	PWM0x[5:0]	Bit[0]: PWM0A port output enable Bit[1]: PWM0B port output enable Bit[2]: PWM0C port output enable Bit[3]: PWM0D port output enable Bit[4]: PWM0E port output enable Bit[5]: PWM0A1 port output enable The corresponding bits of PWM0 are: 1: Output, 0: No output

8.2.7.3. UART port control register (UARTx_IO_SEL)

Address offset: 0x40

Reset value: 0x000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:14	13:12	11:10	9:8	7:6	5	4	3	2	1	0
Reserved	UART 4_SEL	UART 3 _SEL	UART 2_SEL	UART 1_SEL	UART 0_SEL	UART 4_ EC	UART 3_ EC	UART 2_ EC	UART 1_ EC	UART 0_ EC

	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:14	-	Reserved
13:12	UART4 _SEL	UART4 port selection enable 00: PA6/PA7 port select UART4 function 01: PB6/PB7 port select UART4 function 10: PB5/PB6 port select UART4 function 11: PD8/PD9 port select UART4 function
11:10	UART3 _SEL	UART3 port selection enable 00: PA0/PA1 port select UART3 function 01: PA14/PA15 port select UART3 function 10: PC5/PC6 port select UART3 function
9:8	UART2_SEL	UART2 port selection enable 00: PD0/PD1 port select UART2 function 01: PD2/PD3 port select UART2 function 10: PD5/PD6 port select UART2 function 11: PD7/PD5 port select UART2 function
7:6	UART1_SEL	UART1 port selection enable 00: PB14/PB15 port select UART1 function 01: PC3/PC4 port select UART1 function 10: PC6/PC7 port select UART1 function
5	UART0_SEL	UART0 port selection enable 0: PB12/PB13 port select UART0 function 1: PA12/PA13 port select UART0 function
4	UART4_ EC	UART4 port TXD/RXD pin interchange 1: Pin interchange 0: Pins are not interchangeable
3	UART3_ EC	UART3 port TXD/RXD pin interchange 1: Pin interchange 0: Pins are not interchangeable
2	UART2_ EC	UART2 port TXD/RXD pin interchange 1: Pin interchange 0: Pins are not interchangeable
1	UART1_ EC	UART1 port TXD/RXD pin interchange 1: Pin interchange 0: Pins are not interchangeable
0	UART0_EC	UART0 port TXD/RXD pin swap 1: Pin interchange 0: Pins are not interchangeable

8.2.7.4. IIC control register (IIC_IO_CTRL)

Address offset: 0x44

Reset value: 0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	DFIL_SEL	AFIL_SEL	IIC_SEL
	RW	RW	RW

31:3	-	Reserved
2	DFIL_SEL	IIC function digital filter enable 1: Enable 0: Disable
1	AFIL_SEL	IIC function analog filter enable 1: Enable 0: Disable
0	IIC_SEL	IIC port selection enable 0: PB12/PB13 port select IIC function 1: PA12/PA13 port select IIC function

8.2.7.5. SPI input clock configuration register (SPI_CLK_CFG)

Address offset: 0x48

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:2	1	0
Reserved	SPI1_CLK_CFG	SPI0_CLK_CFG
	RW	RW

31:2	-	Reserved
1	SPI1_CLK_CFG	SPI1 input clock selection configuration 1: Use digital internally generated clock (existing in master mode) 0: Use analog input signal
0	SPI0_CLK_CFG	SPI0 input clock selection configuration 1: Use digital internally generated clock (existing in master mode) 0: Use analog input signal

8.2.7.6. SPI high-speed mode enable register (HSPEED_EN)

Address offset: 0x4C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:9	8	7	6	5	4	3	2	1	0
Reserved	HSSEL8	HSSEL7	HSSEL6	HSSEL5	HSSEL4	HSSEL3	HSSEL2	HSSEL1	HSSEL0
	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:9	-	Reserved
8:0	HSSEL[8:0]	Bit0: SPI communication port PA11 configuration high-speed mode enable Bit1: SPI communication port PA12 configuration high-speed mode enable Bit2: SPI communication port PA13 configuration high-speed mode enable Bit3: SPI communication port PB12 configuration high-speed mode enable Bit4: SPI communication port PB13 configuration high-speed mode enable Bit5: SPI communication port PB14 configuration high-speed mode enable Bit6: SPI communication port PC8 configuration high-speed mode enable Bit7: SPI communication port PC9 configuration high-speed mode enable Bit8: SPI communication port PC10 configuration high-speed mode enable The corresponding bits of HSSEL[8:0] are: 1: High-speed mode; 0: Normal mode

8.2.7.7. SPI0 select enable register (SPI0_IO_CTRL)

Address offset: 0x50

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

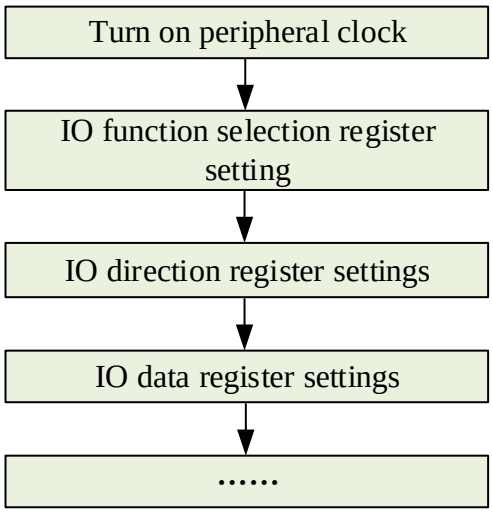
15:1	0
Reserved	SPI0_SEL
	RW

31:1	-	Reserved
0	SPI0_SEL	SPI0 port selection enable 0: PB12/PB13/PB14/PB15 port select SPI0 function 1: PA10/PA11/PA12/PA13 port select SPI0 function

8.3. GPIO configuration process

PB12 and PB13 are SW download and debugging function ports by default. Configuration register SW_IO_EN = 0x00, PB12 and PB13 are set as common IO ports.

To set the port as GPIO, the following 3 sets of registers need to be set accordingly.



9 Interrupt controller (NVIC)

9.1. Interrupt source and entry address

There are 26 interrupt sources in BF7807AMXX. Some of these interrupts will multiplex an interrupt request flag. This flag can be set by software and hardware. If multiple IO are used as External interrupt to multiplex the same interrupt entry, you should refer to the interrupt source settings. INT1/INT2/INT3 can only have one IO as an external interrupt at a time, and INT0 is not restricted.

The built-in simple 24-bit down-counting timer (SysTick) is used as the tick timer of the real-time operating system (RTOS), or as a simple counter. SysTick counts down from a preset value, and when it reaches zero, a system interrupt is generated. For details, see the "SysTick timer" chapter.

Interrupt priority	Interrupt number	CPU interrupt list	Interrupt vector address	Description
-1	-13	HardFault	0x0C	Hardware error
Programmable	-5	SVCall	0x2C	Request management call realized by SVC
Programmable	-2	PendSV	0x38	Suspendable system service request
Programmable	-1	SysTick	0x3C	System tick timer
Programmable	0	IRQ0	0x40	WDT (only exists after wake-up from standby mode)
Programmable	2	IRQ2	0x48	INT_EXTI0
Programmable	3	IRQ3	0x4C	INT_EXTI1
Programmable	4	IRQ4	0x50	INT_EXTI2
Programmable	5	IRQ5	0x54	INT_EXTI3
Programmable	10	IRQ10	0x68	INT_IIC
Programmable	11	IRQ11	0x6C	INT_UART0
Programmable	12	IRQ12	0x70	INT_UART1
Programmable	13	IRQ13	0x74	INT_UART2
Programmable	14	IRQ14	0x78	INT_UART3
Programmable	15	IRQ15	0x7C	INT_UART4
Programmable	16	IRQ16	0x80	INT_SPI0
Programmable	17	IRQ17	0x84	INT_SPI1
Programmable	18	IRQ18	0x88	INT_TIMER0
Programmable	19	IRQ19	0x8C	INT_TIMER1
Programmable	20	IRQ20	0x90	INT_TIMER2
Programmable	21	IRQ21	0x94	INT_TIMER3

Programmable	22	IRQ22	0x98	INT_CDC
Programmable	23	IRQ23	0x9C	INT_ADC
Programmable	24	IRQ24	0xA0	INT_LED_LCD
Programmable	25	IRQ25	0xA4	INT_PWM0
Programmable	26	IRQ26	0xA8	INT_PWM1
Programmable	27	IRQ27	0xAC	INT_PWM2
Programmable	28	IRQ28	0xB0	INT_PWM3
Programmable	29	IRQ29	0xB4	INT_PWM4
Programmable	30	IRQ30	0xB8	INT_LVDT

9.2. Interrupt function

9.2.1. Features

- Flexible interrupt management
- Support nested interrupt
- Programmable priority

9.2.2. Interrupt enable and interrupt clear

The interrupt control register is programmable and used to control the enable and disable of interrupt requests.

For example: Enable or clear interrupt 2:

```
*( (volatile unsigned long* ) (0xE000E100 ) ) = 0x4;      // Enable interrupt 2
*( (volatile unsigned long* ) (0xE000E180 ) ) = 0x4;      // Disable interrupt 2
```

Address	Register	Description
0xE000E100	SETENA	Interrupt set enable register
0xE000E180	CLRENA	Interrupt clear enable register

9.2.2.1. Interrupt set-enable register (SETENA)

Bit	Description	RW	Reset value
31:0	Set enable interrupt 0 to 31, write 1 to set the bit to 1, writing 0 has no effect Bit[0]: Used for interrupt 0 Bit[1]: Used for interrupt 1 ... Bit[30]: Used to interrupt 30 Bit[31]: Reserved	R/W	0x00000000

9.2.2.2. Interrupt clear enable register (CLRENA)

Bit	Description	RW	Reset value
31:0	Clear enable interrupt 0 to 31, write 1 to set the bit to 1, writing 0 has no effect Bit[0]: Used for interrupt 0 Bit[1]: Used for interrupt 1 ... Bit[30]: Used to interrupt 30 Bit[31]: Reserved	R/W	0x00000000

9.2.3. Interrupt suspension and clear suspension

If an interrupt occurs but cannot be processed immediately, the interrupt request will be suspended. Access or modify the interrupt pending status by operating the interrupt to set the pending register and the interrupt clear-pending register.

The interrupt pending status register allows software to trigger interrupts. If the interrupt has been enabled and has not been shielded, and there is no higher priority interrupt currently running, the interrupt service routine will be executed immediately.

For example: Trigger interrupt 2:

```
* ( (volatile unsigned long* ) (0xE000E100 ) ) = 0x4;      // Enable interrupt 2
* ( (volatile unsigned long* ) (0xE000E200 ) ) = 0x4;      // Pending interrupt 2
```

Clear the suspended state of interrupt 2:

```
* ( (volatile unsigned long* ) (0xE000E280 ) ) = 0x4;      // Clear the suspended state of
                                                              interrupt 2
```

Address	Register	Description
0xE000E200	SETPEND	Interrupt set pending register
0xE000E280	CLRPEND	Interrupt clear pending register

9.2.3.1. Interrupt set-pending register (SETPEND)

Bit	Description	RW	Reset value
31:0	Set the suspended state of interrupts 0 to 31, write 1 to set the bit to 1, writing 0 has no effect Write, 1: Change interrupt pending state; 0: No effect Read, 1: Interrupt pending; 0: Interrupt not pending Bit[0]: Used for interrupt 0 Bit[1]: Used for interrupt 1 ... Bit[30]: Used for interrupt 30	R/W	0x00000000

	Bit[31]: Reserved The read value indicates the current suspended state		

9.2.3.2. Interrupt clear-pending register (CLRPEND)

Bit	Description	RW	Reset value
31:0	Clear the suspended state of interrupts 0 to 31, write 1 to set the bit to 1, writing 0 has no effect Bit[0]: Used for interrupt 0 Bit[1]: Used for interrupt 1 ... Bit[30]: Used for interrupt 30 Bit[31]: Reserved The read value indicates the current suspended state	R/W	0x00000000

9.2.4. Interrupt priority

The CPU contains 8 32-bit register configurations. Each interrupt entry corresponds to an 8-bit priority register, of which only the highest two bits [7:6] are valid, and the priority levels that can be used are 0x00 (highest), 0x40, 0x80 and 0xC0 (lowest). The default priority is interrupt 0 to 31.

If two interrupts occur at the same time and their priority is the same, the interrupt with the smaller interrupt number will be executed first. The ongoing interrupt service routine can only be interrupted by high-priority interrupt requests.

Address	[31:30]	[29:24]	[23:22]	[21:16]	[15:14]	[13:8]	[7:6]	[5:0]
0xE000E41C	Reserved		IRQ30		IRQ29		IRQ28	
0xE000E418	IRQ27		IRQ26		IRQ25		IRQ24	
0xE000E414	IRQ23		IRQ22		IRQ21		IRQ20	
0xE000E410	IRQ19		IRQ18		IRQ17		IRQ16	
0xE000E40C	IRQ15		IRQ14		IRQ13		IRQ12	
0xE000E408	IRQ11		IRQ10		Reserved		Reserved	
0xE000E404	Reserved		Reserved		IRQ5		IRQ4	
0xE000E400	IRQ3		IRQ2		Reserved		IRQ0	

Interrupt priority register

Note: The gray is an unused bit, the write operation is invalid, and the readout is 0.

Each access to the priority register is equivalent to accessing the priority of 4 interrupts. If you change one of them, you need to read the entire word, modify one byte, and then write the entire word back.

For example: Set the priority of interrupt #2 to 0xC0:

```
unsigned long temp;                                    // A temporary variable
temp = * ( (volatile unsigned long * ) ( 0xE000E400 ) );   // Get IRP0
temp = temp & ( 0xFF00FFFF ) | ( 0xC0<<16);            // Modify priority
* ( (volatile unsigned long* ) (0xE000E400 ) ) = temp;     // Set up IRP0
```

9.3. External interruption

9.3.1. Features

- All IO ports support external interrupt function (rising edge, falling edge, double edge)
- Trigger source type: Rising edge, falling edge or both edges
- Standby mode: High and low level trigger mode can wake up the chip
- Single interrupt enable, rising edge trigger enable, falling edge trigger enable, trigger request flag bit

INT1/INT2/INT3 can only have one IO as an external interrupt at a time, which is selected by the EXTIx interrupt source selection register (EXTI_SEL). INT0 is not restricted.

When the selected edge event occurs on the external interrupt line, the corresponding bit of the EXTIx trigger request flag register is set to 1. It can be cleared by writing 1 to this bit.

9.4. External interrupt registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: 0x500A 0100

Address offset	Register	Description
0x00	EXTI_EN	EXTIx enable register
0x04	EXTI_RTEN	EXTIx rising edge trigger enable register
0x08	EXTI_FTEN	EXTIx falling edge trigger enable register
0x0C	EXTI_PR	EXTIx trigger request flag register
0x10	EXTI_SEL	EXTIx interrupt source selection register

9.4.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN

RW	RW	RW	RW	RW	RW	RW	RW

22	GPIO_CLKEN	GPIO module work enable 1: Work 0: Off, the default is 0

9.4.2. EXTIx enable register (EXTI_EN)

Address offset: 0x00
Reset value: 0x0000 0000

31:19	18	17	16
Reserved	EXTI3	EXTI2	EXTI1
	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI 0_15	EXTI 0_14	EXTI 0_13	EXTI 0_12	EXTI 0_11	EXTI 0_10	EXT I0_9	EXT I0_8	EXT I0_7	EXT I0_6	EXT I0_5	EXT I0_4	EXT I0_3	EXT I0_2	EXT I0_1	EXT I0_0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:19	-	Reserved
18	EXTI3	External interrupt 3 interrupt enable 1: Enable; 0: Disable
17	EXTI2	External interrupt 2 interrupt enable 1: Enable; 0: Disable
16	EXTI1	External interrupt 1 interrupt enable 1: Enable; 0: Disable
15:0	EXTI0_x[15:0]	Bit[15]~ Bit[0]: External interrupt 0, INT0_15~INT0_0 interrupt enable Bit[0]: INT0_0 Bit[1]: INT0_1 … Bit[15]: INT0_15 The corresponding bits are: 1: Enable; 0: Disable

9.4.3. EXTIx rising edge trigger enable register (EXTI_RTEN)

Address offset: 0x04
Reset value: 0x0000 0000
The trigger mode is rising edge, and the high level wakes up the standby; in the case of

EXTI_INTEN=0, the configuration is modified.

31:19	18	17	16
Reserved	RT3	RT2	RT1
	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT0_15	RT0_14	RT0_13	RT0_12	RT0_11	RT0_10	RT0_9	RT0_8	RT0_7	RT0_6	RT0_5	RT0_4	RT0_3	RT0_2	RT0_1	RT0_0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:19	-	Reserved
18	RT3	External interrupt 3 rising edge trigger enable 1: The rising edge trigger is valid; 0: The rising edge trigger is invalid
17	RT2	External interrupt 2 rising edge trigger enable 1: The rising edge trigger is valid; 0: The rising edge trigger is invalid
16	RT1	External interrupt 1 rising edge trigger enable 1: The rising edge trigger is valid; 0: The rising edge trigger is invalid
15:0	RT0_x[15:0]	Bit[15]~ Bit[0]: External interrupt 0, INT0_15~INT0_0 rising edge trigger enable Bit[0]: INT0_0 rising edge trigger enable Bit[1]: INT0_1 rising edge trigger enable … Bit[15]: INT0_15 rising edge trigger enable The corresponding bits are: 1: The rising edge trigger is valid; 0: The rising edge trigger is invalid

9.4.4. EXTIx falling edge trigger enable register (EXTI_FTEN)

Address offset: 0x08

Reset value: 0x0007 FFFF

The trigger mode is falling edge/double edge, low-level wake-up standby; in the case of EXTI_INTEN=0, the configuration is modified.

31:19	18	17	16
Reserved	FT3	FT2	FT1
	RW	RW	RW

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

FT0_15	FT0_14	FT0_13	FT0_12	FT0_11	FT0_10	FT0_9	FT0_8	FT0_7	FT0_6	FT0_5	FT0_4	FT0_3	FT0_2	FT0_1	FT0_0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:19	-	Reserved
18	FT3	External interrupt 3 falling edge trigger enable 1: Falling edge trigger is valid; 0: The falling edge trigger is invalid
17	FT2	External interrupt EXTI2 falling edge trigger enable 1: Falling edge trigger is valid; 0: The falling edge trigger is invalid
16	FT1	External interrupt EXTI1 falling edge trigger enable 1: Falling edge trigger is valid; 0: The falling edge trigger is invalid
15:0	FT0_x[15:0]	Bit[15]~ Bit[0]: External interrupt 0, INT0_15~INT0_0 falling edge trigger enable Bit[0]: INT0_0 Bit[1]: INT0_1 … Bit[15]: INT0_15 The corresponding bits are: 1: Falling edge trigger is valid; 0: Falling edge trigger is invalid

9.4.5. EXTIx trigger request flag register (EXTI_PR)

Address offset: 0x0C

Reset value: 0x0000 0000

This bit is set to 1 when the selected edge event occurs on the external interrupt line. Writing a 1 to this bit clears it.

31:19	18	17	16
Reserved	PR3	PR2	PR1
	RC-W1	RC-W1	RC-W1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PR0_15	PR0_14	PR0_13	PR0_12	PR0_11	PR0_10	PR0_9	PR0_8	PR0_7	PR0_6	PR0_5	PR0_4	PR0_3	PR0_2	PR0_1	PR0_0
RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1	RC-W1

31:19	-	Reserved

18	PR3	External interrupt EXTI3 trigger request 1: The selected trigger request occurred 0: No trigger request occurred
17	PR2	External interrupt EXTI2 trigger request 1: The selected trigger request occurred 0: No trigger request occurred
16	PR1	External interrupt EXTI1 trigger request 1: The selected trigger request occurred 0: No trigger request occurred
15:0	PR0_x [15:0]	Bit[15]~ Bit[0]: External interrupt 0, INT0_15~INT0_0 trigger request Bit[0]: INT0_0 Bit[1]: INT0_1 … Bit[15]: INT0_15 The corresponding bits are: 1: The selected trigger request has occurred; 0: The trigger request has not occurred

9.4.6. EXTIx interrupt source selection register (EXTI_SEL)

Address offset: 0x10

Reset value: 0x0000 0000

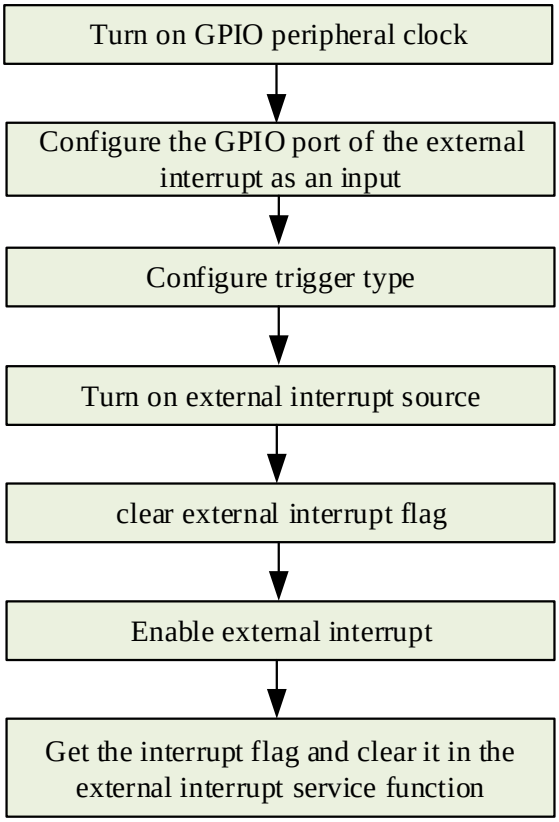
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				EXTI3_S				EXTI2_S				EXTI1_S			
				RW				RW				RW			

31:12	-	Reserved
11:8	EXTI3_S	EXTI3 interrupt source selection 0000: PC0 0001: PC1 ... 1010: PC10 1011: PC11 Other values are invalid
7:4	EXTI2_S	EXTI2 interrupt source selection 0000: Reserved 0001: PB1 …… 1110: PB14

		1111: PB15
3:0	EXTI1_S	EXTI1 interrupt source selection 0000: PA0 0001: PA1 …… 1110: PA14 1111: PA15

9.5. External interrupt configuration flow



10 Timer

10.1. Features

- 16-bit up counting Timer0/1/2/3
- Timer0/Timer1/Timer3 is connected to PLL48M, the internal frequency of the counting clock is 1/2/4/8/16/32/64/128
- Timer2 can select internal LIRC 32kHz and external crystal oscillator clock XTAL, frequency 32768Hz/4MHz/8MHz
- 16-bit automatic reload timing and manual reload timing
- Timer0/1/2/3 can wake up idle mode 0
- When Timer2 count clock selects XTAL/LIRC, it supports interrupt wake-up idle mode 1

10.2. Function description

10.2.1. Timer0/1/3

Timer0, Timer1 and Timer3 have the same functions.

Two working modes are supported: Single timing mode and automatic reload mode, and no matter which mode, the timing is complete can be configured to generate an interrupt.

Single timing mode: After a timing is completed, the hardware will automatically pull down TIMERx_CFG[0] to stop timing.

Automatic reload mode: the hardware will automatically reload the setting value, and the register TIMERx_CFG[0] will continue to maintain 1, and the next time will be restarted; the software will stop counting by writing 0 to the register TIMERx_CFG[0], or modify the timing mode midway.

Timing duration formula:

$$T_{TIMER} = T_{CLK}*(TIMERx\_SET+1)$$

Note: It is strictly forbidden to change the relevant configuration during the timing process. If TIMERx_SET and TIMERx_CFG are configured, the counter will be cleared. If the current count enable is valid, it will count from zero again.

10.2.2. Timer2

Two working modes are supported: Single timing mode and automatic reload mode, and no matter which mode, the timing is complete can be configured to generate an interrupt.

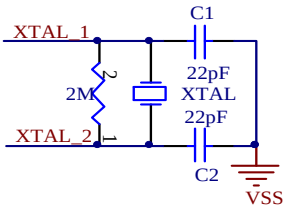
Single timing mode: After a timing is completed, the hardware will automatically pull down TIMER2_CFG[0] to stop timing.

Auto-reload mode: The hardware will automatically reload the setting value, and the register TIMER2_CFG[0] will continue to maintain 1, and the next time will be restarted; the software will stop counting by writing 0 to the register TIMER2_CFG[0], or modify the timing mode midway.

Timing duration formula:

$$T_{TIMER2} = T_{CLK}*(TIMER2\_SET+1)$$

Note: Any configuration of TIMER2_SET and TIMER2_CFG in the timing process can clear the counter. If the current counting enable is valid, it will count from zero again.



External crystal oscillator circuit reference

Note:

1. The external crystal oscillator circuit is for reference only, and the actual parameters refer to the crystal oscillator specifications;
2. XTAL 32768Hz excitation power is recommended to be greater than 1μW;
3. XTAL 32768Hz recommends parallel resistance of 2MΩ;
4. XTAL 4M/8M recommends parallel resistance of 1MΩ.

10.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register
0x0C	XTAL_HS_SEL	Comparator hysteresis voltage selection register in crystal oscillator
0x10	ANA_CFG	Analog module switch register

Base address: Timer0: 0x5005_0000; Timer1: 0x5005_0100; Timer2: 0x5005_0200;
Timer3: 0x5005_0300

Timer0, Timer1 and Timer3 registers have the same function: x=0/1/3.

Address offset	Register	Description
0x00	TIMERx_CFG	TIMER0/1/3 configuration register
0x04	TIMERx_SET	TIMER0/1/3 counting cycle configuration register
0x08	TIMERx_INT_CFG	TIMER0/1/3 interrupt configuration register

Timer2:

Address offset	Register	Description
0x00	TIMER2_CFG	TIMER2 configuration register
0x04	TIMER2_SET	TIMER2 counting cycle configuration register
0x08	TIMER2_INT_CFG	TIMER2 interrupt configuration register

10.3.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8
TIMER1_CLKEN	TIMER0_CLKEN	PWM4_CLKEN	PWM3_CLKEN	PWM2_CLKEN	PWM1_CLKEN	PWM0_CLKEN	IIC_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

17	TIMER3_CLKEN	TIMER3 module operation enable 1: Work 0: Off, the default is 0
16	TIMER2_CLKEN	TIMER2 module operation enable 1: Work

		0: Off, the default is 0
15	TIMER1_CLKEN	TIMER1 module operation enable 1: Work 0: Off, the default is 0
14	TIMER0_CLKEN	TIMER0 module operation enable 1: Work 0: Off, the default is 0

10.3.2. Timer0/1/3 registers

10.3.2.1. Timer0/1/3 configuration register (TIMERx_CFG)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	CLK_SEL			RLD	EN
	RW			RW	RW

31:5	-	Reserved
4:2	CLK_SEL	Timer0/1/3 module clock selection 000: Select PLL48M 001: Select PLL24M 010: Select PLL12M 011: Select PLL6M 100: Select PLL3M 101: Select PLL1.5M 110: Select PLL0.75M 111: Select PLL0.375M Regardless of the mode, configuring this register during the counting process will clear the counter, and if the current counting enable is valid, it will count from zero again
1	RLD	Counting mode selection 1: Auto reload mode 0: Manual reload mode Regardless of the mode, configuring this register during the counting process will clear the counter, and if the current counting enable is valid, it will count from zero again
0	EN	Count enable

		1: Enable counting 0: Stop counting In the manual reload mode, the counting will stop after the counting is completed. The software needs to write 0 to the register during interrupt processing. To re-count requires the software to write 0 to the register and then write 1; in the automatic reload mode, it will be after the counting is completed. Automatically restart counting from zero. Regardless of the mode, configuring this register during the counting process will clear the counter, and if the current counting enable is valid, it will count from zero again

10.3.2.2. Timer0/1/3 counting cycle configuration register (TIMERx_SET)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMERx_SET[15:0]															
RW															

31:16	-	Reserved
15:0	SET[15:0]	Counting cycle configuration register, configuring this register during counting will clear the counter, if the current counting enable is valid, it will count from zero again Note: When the configuration selects PLL_48M, the counting period should be greater than 4

10.3.2.3. Timer0/1/3 interrupt configuration register (TIMERx_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

31:3	-	Reserved

2	INT_CLR	Interrupt status flag clear Write 1 to clear INT_FLAG, write only
1	INT_FLAG	Interrupt status flag register 1: Counting is complete 0: Count incomplete, read only
0	INT_EN	Interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

10.3.3. Timer2 registers

10.3.3.1. Comparator hysteresis voltage selection register in crystal oscillator (XTAL_HS_SEL)

Address offset: 0x0C

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:2	1	0
Reserved	HS_SEL[1:0]	
	RW	

31:2	-	Reserved
1:0	HS_SEL[1:0]	The hysteresis voltage selection of the comparator in the crystal oscillator 00: 300mV 01: 400mV 10: 500mV 11: 600mV

10.3.3.2. Analog module switch register (ANA_CFG)

Address offset: 0x10

Reset value: 0x0000 0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	XTAL_HFR_SEL	XTAL_SEL	PD_XTAL	PD_CDC	PD_ADC
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	XTAL_HFR_SEL	Analog high frequency crystal oscillator circuit selection 0: 4MHz 1: 8MHz
3	XTAL_SEL	Analog crystal oscillator circuit frequency selection 0: 32768Hz 1: 4MHz/8MHz
2	PD_XTAL	Analog crystal oscillator circuit (32768Hz/4MHz/8MHz) control register 1: Off 0: On, off by default

10.3.3.3. Timer2 configuration register (TIMER2_CFG)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	CLK_SEL	RLD	EN
	W	R	RW

31:3	-	Reserved
2	CLK_SEL	Count clock selection register 1: Select the clock XTAL 0: Select clock LIRC No matter which mode, configuring this register during counting will clear the counter.
1	RLD	Counting mode selection register 1: Auto reload mode 0: Manual reload mode No matter which mode, configuring this register during counting will clear the counter.
0	EN	Count enable register 1: Enable counting 0: Stop counting In manual reload mode, it will stop counting after the count is completed. The software needs to write 0 to the register during interrupt processing. To re-count requires software to write 0 to this register and then write 1; in automatic reload mode, it will be after the count is completed. Automatically restart counting from zero.

		Regardless of the mode, configuring this register during the counting process will clear the counter, and if the current counting enable is valid, it will count from zero again.

10.3.3.4. Timer2 counting cycle configuration register (TIMER2_SET)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMER2_SET[15:0]															
RW															

31:16	-	Reserved
15:0	SET[15:0]	Counting cycle configuration register, configuring this register during counting will clear the counter, if the current counting enable is valid, it will count from zero again.

10.3.3.5. Timer2 interrupt configuration register (TIMER2_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

31:3	-	Reserved
2	INT_CLR	Interrupt status flag clear register Write 1 to clear INT_FLAG, write only
1	INT_FLAG	Interrupt status flag register 1: Counting is complete 0: Count incomplete, read only The way to clear the interrupt flag: System reset; write 1 to INT_CLR to clear; configure register CFG/TIMER2_SET.
0	INT_EN	Interrupt enable register 1: Interrupt enable

		0: Interrupt disabled (used in polling mode) The wake-up function is only available when the interrupt is enabled.

10.4. Configuration process

1. Turn on the peripheral clock;
2. Configure the clock selection register TIMERx_CFG[2] and the counting cycle configuration register TIMERx_SET;
3. Configure the counting mode selection register TIMERx_CFG[1];
4. Configure the counting enable register TIMERx_CFG[0], turn on the timing TIMERx_CFG[0] = 0x1;
5. Stop timing TIMERx_CFG[0] = 0x0.

No matter which mode, after the count is completed, TIMERx_INT_CFG[1] is set. If interruption is required, the interrupt enable can be configured.

Interrupt processing flow:

One-shot timing mode: Configure TIMERx_INT_CFG[2]=1 in the interrupt handler to clear the interrupt flag. If you want to turn it on again, you need to configure TIMERx_CFG[0]= 0x0, and then configure TIMERx_CFG[0]= 0x1.

Auto-reload mode: Configure TIMERx_INT_CFG[2] = 1 in the interrupt handler to clear the interrupt flag, and do not allow the TIMERx_SET and TIMERx_CFG registers to be configured. To stop counting, configure TIMERx_CFG[0] = 0x0 directly.

Note:

1. **The TIMERx_CFG[0]=0x1 operation should be placed at the end of all configurations.**
2. **During the timing of TIMER, it is strictly forbidden to change the relevant configuration. If you want to modify it, you need to stop the timing first.**
3. **For precise timing, configuration of the TIMERx_SET and TIMERx_CFG registers is not allowed in interrupt handling in auto-reload mode.**

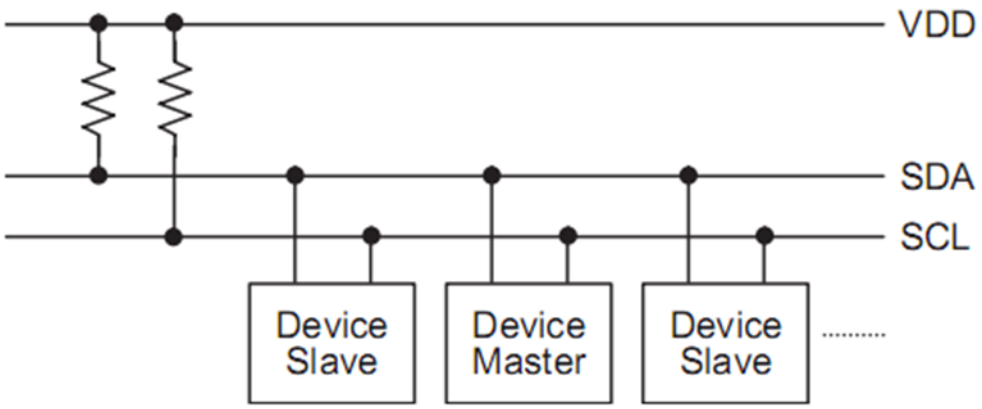
11 IIC bus

11.1. IIC slave mode

11.1.1. Features

- Two serial interfaces: Serial data line SDA and serial clock line SCL
- Compliant with the standard communication protocol of philips
- Support standard mode 100kHz, fast mode 400kHz, super fast mode 1MHz①
- Support 7-bit address addressing
- With the function of extending the low level of the clock
- The core can be woken up by IIC interrupt in standby mode
- Detect write conflict and buffer BUF overflow abnormal situation

Note①: When the communication rate of the slave device reaches 1MHz or the slave device needs to process other transactions, the slave device can lengthen the low level time of the clock line by pulling down the clock line, thereby reducing the communication frequency.



IIC master-slave connection diagram

The master and slave are connected by SCL (serial clock) line and SDA (serial data) line. SCL and SDA must be connected with pull-up resistors (4.7k~10k recommended).

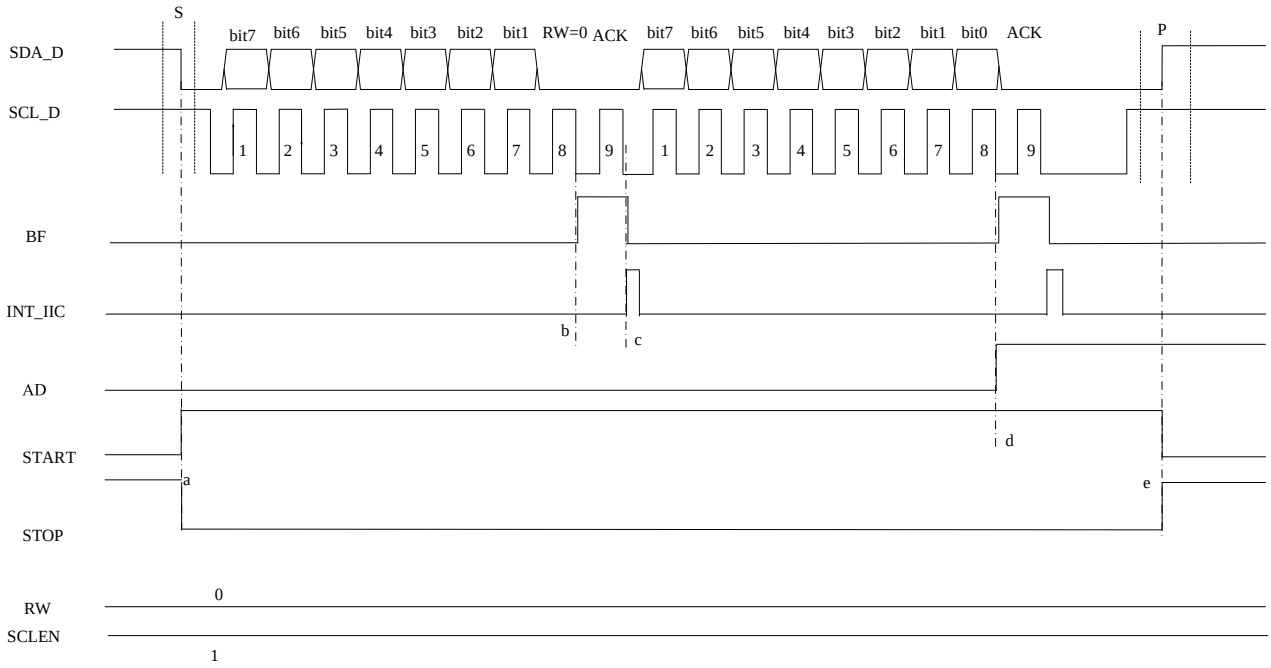
11.1.2. Port configuration

The BF7807AMXX provides register IIC_IO_CTRL[0] to select IIC function.
Write 0 to the register IIC_IO_CTRL[0], then configure PB12 and PB13 as IIC functions:
SCL0A, PB12 are IIC serial clock lines;
SDA0A, PB13 are IIC serial data lines.
Write 1 to the register IIC_IO_CTRL[0] to configure PA12 and PA13 as IIC functions:
SCL0B, PA12 are IIC serial clock lines;
SDA0B, PA13 are IIC serial data lines.

11.2. IIC slave communication timing

The BF7807AMXX adopts hardware slave. When the master reads/writes 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 eighth clock when the master writes data, and the interrupt signal will not be generated when the master sends a stop signal. The following is a simple sequence diagram of IIC communication:

IIC master write timing description



IIC write not pull down clock line diagram

As shown in the figure above, it is a schematic diagram of the master not pulling down the clock line during write operation, from which you can see the changes of the IIC bus and the changes of some internal signals.

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

Then, the master sends the address byte and the read and write flag bits, and the slave hardware automatically matches its own address after receiving the address byte. Compare and set BF after the falling edge of the 8th clock if there is a match, as shown by the dotted line b.

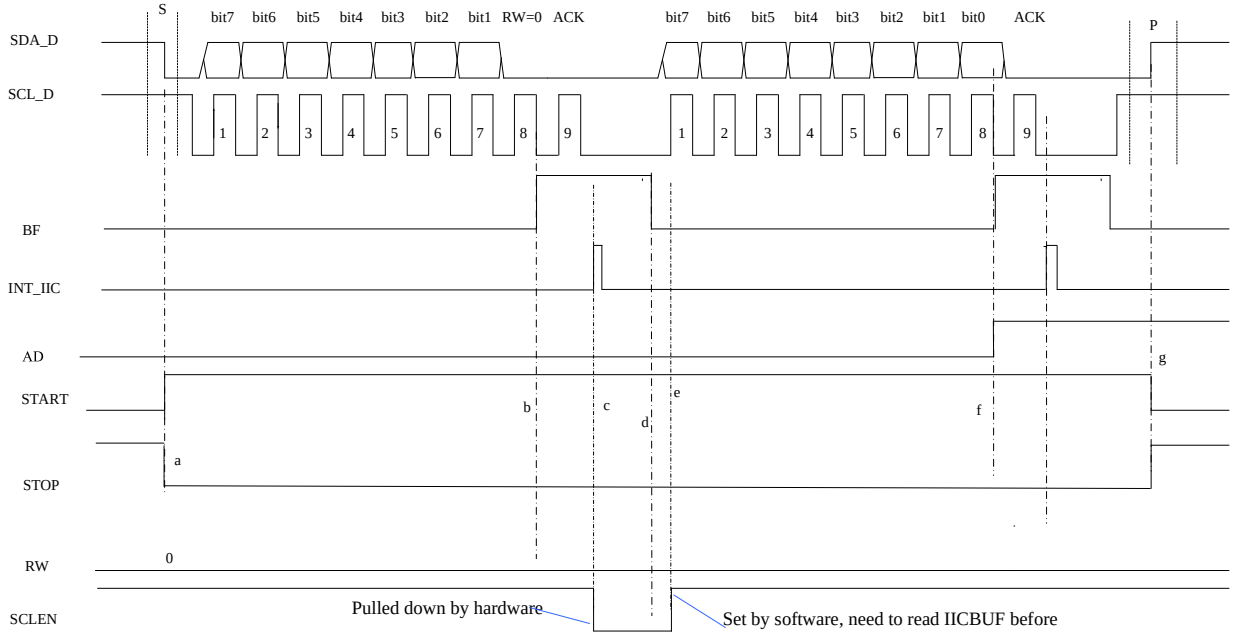
The interrupt signal INT_IIC will be generated after the falling edge of the ninth clock. As shown by the dotted line c, the MCU needs to read IICBUF during the execution of the interrupt subroutine. Even if this data is useless, the operation of reading IICBUF will indirectly clear BF.

The master continues to send data, BF is also set after the falling edge of the 8th clock of the 2nd byte, and the AD flag is also set, indicating that the currently received byte is data, as shown by the dotted line d , the stop signal has no effect on the AD flag, that is, when the stop signal STOP is

detected, the AD flag will not be cleared; an interrupt will also be generated after the falling edge of the ninth clock, and the interrupt subroutine needs to do the same operation.

If the master wants to send multiple bytes, it can continue to send. The above figure only shows the situation where the host sends one data. Finally, the host sends a stop signal STOP after sending all the data, which marks the end of the communication, releases the IIC bus, and the bus enters the idle state.

IIC master write pull low timing description



IIC master write pull low clock line diagram

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

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

Then, the master sends the address byte and the read and write flag bits, and the slave hardware automatically matches its own address after receiving the address byte. Compare and set BF after the falling edge of the 8th clock if there is a match, as shown by the dotted line b. The interrupt signal INT_IIC is generated after the falling edge of the ninth clock, as shown by the dotted line c.

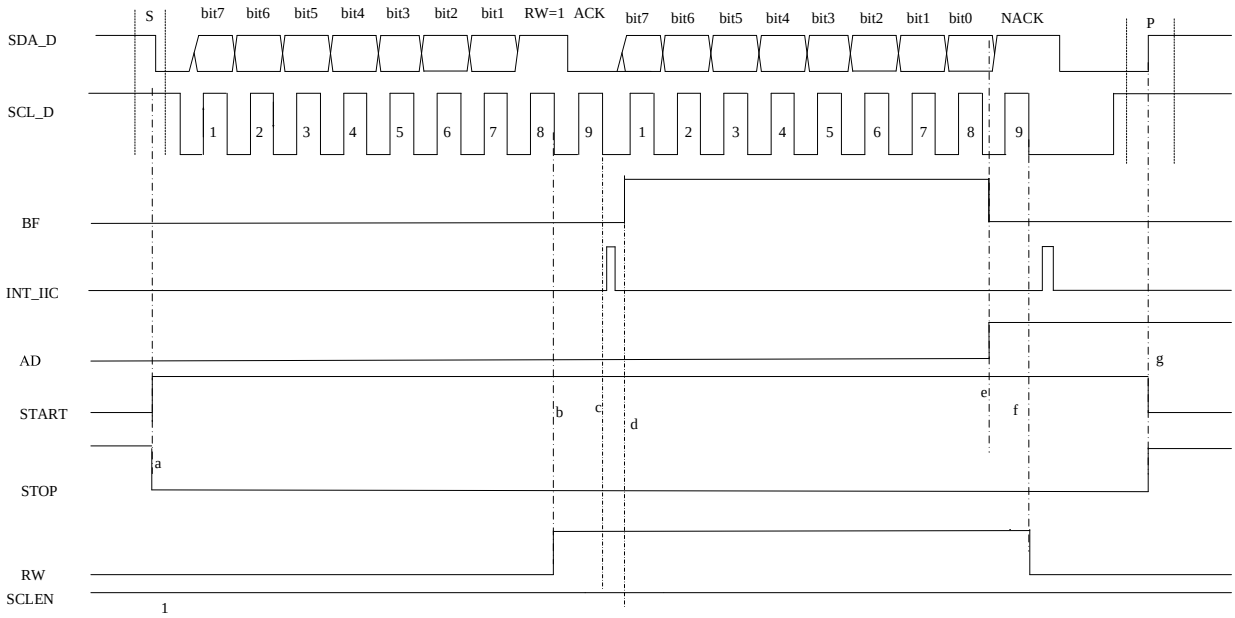
After the falling edge of the ninth clock, SCLEN will be automatically cleared by hardware. During this period, it is used for slave processing or reading data. Even if the data is useless, the operation of reading IICBUF will indirectly clear BF, such as the dotted line d. shown. Then software sets SCLEN to release the clock line, as shown by the dotted line e.

After the master detects that the slave releases SCL, it will continue to send the synchronous clock. After the falling edge of the 8th clock of the 2nd byte, BF will also be set, and the AD flag will also be set, indicating that the current received Byte is data, as shown by the dotted line f, the stop signal has no effect on the AD flag bit, that is, if the stop signal STOP is detected, the AD flag bit will not be cleared; an interrupt will also be generated after the falling edge of the ninth clock.

If the master wants to send multiple bytes, it can continue to send, as shown in the figure above, only the host sends one data. It should be noted that when the host sends the last data, the function of pulling down the clock line is not enabled.

Finally, the host sends a stop signal STOP after sending all the data, which marks the end of the communication, releases the IIC bus, and the bus enters the idle state.

IIC master read timing description



IIC master reading does not pull down the clock line diagram

As shown in the figure above, it is the timing diagram of the master reading the slave clock line low. From the figure, we can know the changes of the bus and the changes of the internal signals of some circuits.

First, the master sends a START signal to mark the beginning of communication. As shown by the dotted line a, the internal circuit sets the status flag bit START after detecting the timing of the START signal.

Then, after the START signal, the master sends the address byte, and RW=1, which means that the master reads the slave. In the case of address matching, after the falling edge of the eighth clock, the status bit RW is set, as shown by the dotted line b. If the address does not match, RW will not be set.

After the falling edge of the ninth clock, an interrupt signal is generated, as shown by the

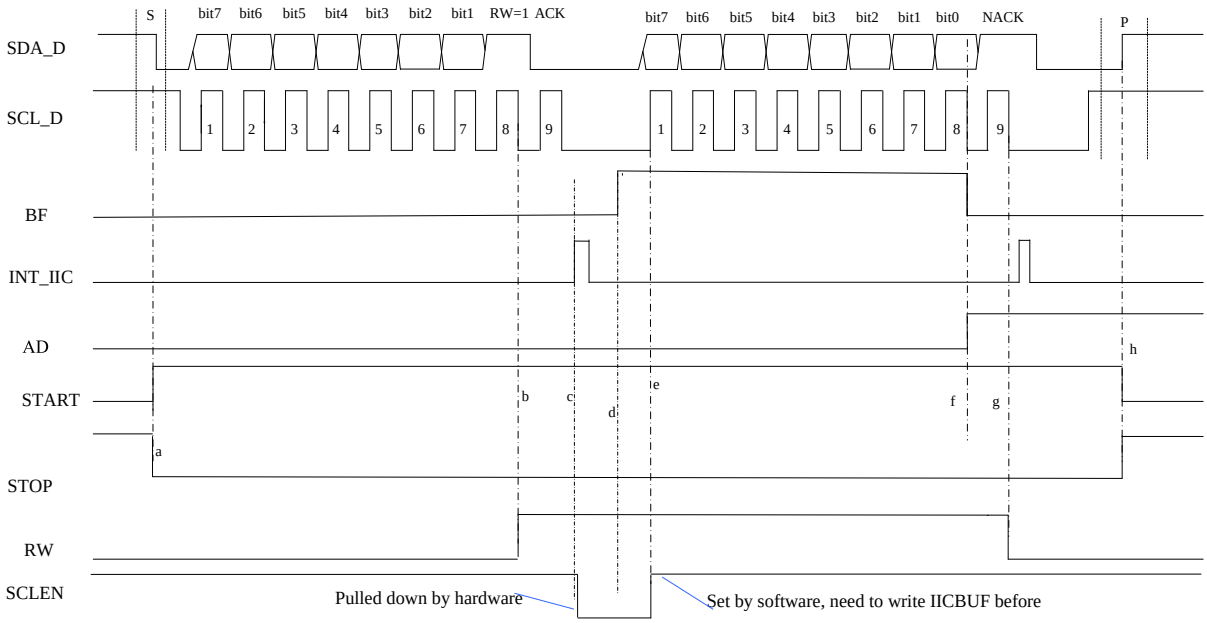
dashed line c. And ballast the data in IICBUFFER to IICBUF, BF is set, as shown by the dotted line d, and the highest bit is sent to the bus. After 8 clocks, one byte of data is sent, and the BF flag is cleared; at the same time, the address data flag is also set, indicating the byte data currently being sent. As shown by the dotted line e.

After the falling edge of the ninth clock, an interrupt will be generated. If the master needs to continue to read the slave, the master will reply with a valid acknowledge bit ACK and continue communication; if the data required by the master has been read, the master will reply with an invalid reply NACK, and then Send the stop signal STOP to terminate the communication. In the schematic diagram, the master only reads one piece of data, and replies with NACK, and then sends a STOP signal to terminate the communication. When NACK is detected, the read and write flag bit RW is cleared by hardware, as shown by the dotted line f.

If the master sends a NACK, the slave SCLEN will not be automatically pulled down, this should be paid attention to in the application.

Finally, the master sends a stop signal STOP after reading all the data, which marks the end of communication. When the STOP signal is detected, the status bit STOP is set, START is cleared, and the IIC bus is released. As shown by the dotted line g, the bus enters idle state.

IIC master read pull low timing description



IIC master reads and pulls down the clock line diagram

As shown in the figure above, it is the timing diagram of the master reading the slave clock line low. From the figure, we can know the changes of the bus and the changes of the internal signals of some circuits.

First, the master sends a START signal to mark the beginning of communication. As shown by

the dotted line a, the internal circuit sets the status flag bit START after detecting the timing of the START signal.

Then, after the START signal, the master sends the address byte, and RW=1, which means that the master reads the slave. In the case of address matching, after the falling edge of the eighth clock, the status bit RW is set, as shown by the dotted line b. If the address does not match, RW will not be set.

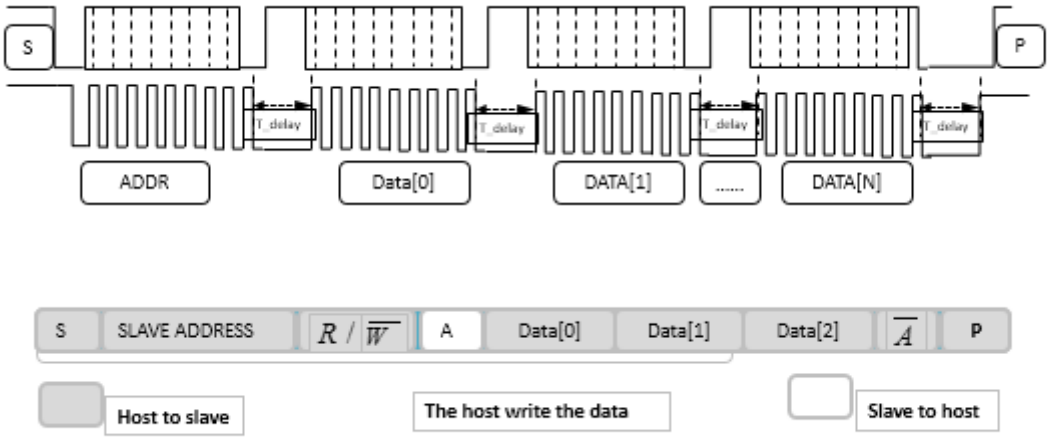
After the falling edge of the ninth clock, an interrupt signal is generated, as shown by the dashed line c. After the falling edge of the ninth clock, SCLEN will also be automatically pulled low by the hardware. During this period, it is used by the slave to process or prepare data, and then write the prepared data into IICBUF, and then the software sets SCLEN to release the clock line. As shown by the dashed line d, after writing data into IICBUF, BF will be set to indicate that IICBUF is full. As shown by the dotted line e, the software sets SCLEN and releases the clock line.

After the master detects that the slave releases the SCL, it will continue to send the synchronous clock and read the data from the slave. After the falling edge of the eighth clock, one byte of data is sent, and the BF flag bit is cleared; at the same time, the address data flag The bit will also be set to indicate the byte data currently being sent. As shown by the dashed line f.

After the falling edge of the 9th clock, an interrupt will be generated. If the master needs to continue to read the slave, it will reply with the valid response bit ACK and continue communication; if the data required by the master has been read, it will reply with an invalid response NACK, and then stop sending The signal STOP terminates the communication. In the schematic diagram, the master only reads one piece of data, and replies with NACK, and then sends a STOP signal to terminate the communication. When NACK is detected, the read and write flag bit RW is cleared by hardware, as shown by the dotted line g.

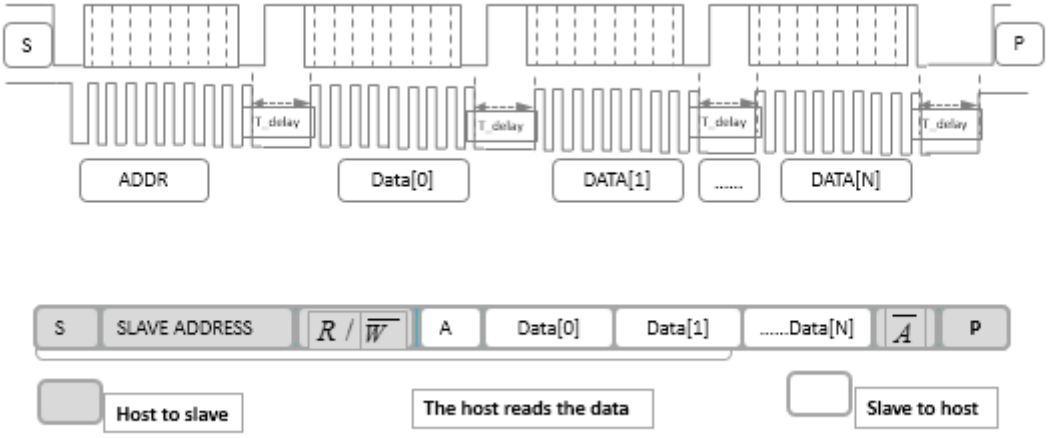
Finally, the master sends a stop signal STOP after reading all the data, marking the end of communication. When the STOP signal is detected, the status bit STOP is set, START is cleared, and the IIC bus is released. As shown by the dotted line h, the bus enters idle state.

IIC master 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 .

IIC master 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.

The slave gives an ACK signal on the falling edge of the eighth clock, and an IIC interrupt is generated after the falling edge of the ninth clock. It is recommended that the master delay 60μs~300μs to reserve the slave IIC when the ninth clock falling edge is sent. Interrupt the service data preparation time, and then send the clock signal.

11.3. IIC slave registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: Slave: 0x5004_0000

Address offset	Register	Description
0x00	IICADD	IIC address register
0x04	IICBUF	IIC send and receive data register
0x08	IICCON	IIC configuration register
0x0C	IICSTAT	IIC status register
0x10	IICBUFFER	IIC transmit and receive data buffer register

Base address: 0x500A 0000

Address offset	Register	Description
0x44	IIC_IO_CTRL	IIC control register

11.3.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8
TIMER1_CLKEN	TIMER0_CLKEN	PWM4_CLKEN	PWM3_CLKEN	PWM2_CLKEN	PWM1_CLKEN	PWM0_CLKEN	IIC_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

22	GPIO_CLKEN	GPIO module operation enable 1: Work 0: Off, the default is 0
8	IIC_CLKEN	IIC module operation enable 1: Work 0: Off, the default is 0

11.3.2. IIC address register (IICADD)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								IICADD[7:0]							
								RW							

31:8	-	Reserved
7:0	IICADD[7:0]	Address register

11.3.3. IIC send and receive data register (IICBUF)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								IICBUF [7:0]							
								RW							

31:8	-	Reserved
7:0	IICBUF [7:0]	IIC transmit and receive data buffer

The specific application process is as follows:

In the sending state, after the data is ballasted into IICBUF, the data is shifted and sent out in turn under the synchronization clock of the master, with the high bit first. After 8 clocks, one byte is sent.

In the receiving state, after 8 clocks from the master, the data is written into the BUF. After the 9th clock, an interrupt is generated to tell the CPU to read the data in IICBUF.

The operation of writing data into IICBUF is conditional. When RD_SCL_EN=1, only RW=1 and SCLEN=0 can the data be written into IICBUF; otherwise, the operation of writing IICBUF is prohibited. In other words, if the conditions are not met, the operation of writing IICBUF cannot be successful, the data cannot be written in, and the data of IICBUF will not change, and it will also cause write conflicts.

For example: IICBUF already has data 55h, if the conditions for writing IICBUF are not met, I want to write data 00h into IICBUF. The result is that the data in IICBUF is still 55h, and the write conflict flag WCOL is set to tell the user that the operation is abnormal.

When RD_SCL_EN=0, the data to be sent by the slave is obtained by ballasting the IICBUFFER register value when the interrupt signal is generated.

11.3.4. IIC configuration register (IICCON)

Address offset: 0x08

Reset value: 0x0000 0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:6	5	4	3	2	1	0
Reserved	IIC_RST	RD_SCL_EN	WR_SCL_EN	SCLEN	SR	IICEN
	RW	RW	RW	RW	RW	RW

31:6	-	Reserved
5	IIC_RST	IIC module reset signal 1: IIC module reset operation 0: IIC module works normally
4	RD_SCL_EN	The master reads and pulls the clock line control bit low 1: Enable the master to read and pull down the clock line function 0: Disable the function of pulling down the clock line when the master reads
3	WR_SCL_EN	The master writes and pulls the clock line low control bit 1: Enable the function of writing and pulling down the clock line 0: Disable the function of writing and pulling down the clock line
2	SCLEN	IIC clock enable bit 1: The clock works normally 0: Pull down the clock line
1	SR	IIC conversion rate control bit 1: The 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	IICEN	IIC work enable bit 1: IIC works normally 0: IIC does not work

The IICCON register is used to control the communication operation.

IIC_EN is the enable signal of the IIC module, and the circuit works only when IIC_EN=1.

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

SCLEN is the clock enable control bit. Although the slave cannot generate the communication clock, the slave can extend the low-level time of the clock according to the protocol. The clock line with SCLEN=0 is locked at low level, and the clock line with SCLEN=1 is released. The prerequisite for extending the clock low level is IIC_EN=1, otherwise the internal circuit will not have any impact on the IIC bus. SCLEN is often used to extend the low-level time and make the master enter the waiting state, so that the slave has enough time to process the data.

WR_SCL_EN is the control bit of the write pull-low line. When it is 1, it enables the interrupt pull-down function of the clock line. When it is 0, it does not enable the interrupt pull-down function of the clock line.

In the case of IIC_RW=0, you can decide whether to pull down the clock line according to the communication rate of the master and the time to process the interrupt, 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 disables the function of pulling down the clock line. At this time, the hardware will not automatically pull down the clock line when the interrupt comes. When the CPU cannot finish processing the interrupt and exit within 8 IIC clocks, WR_SCL_EN=1 enables the function of pulling down the clock line. At this time, the hardware automatically pulls down the clock line when the interrupt comes, forcing the master to enter the waiting state. After the data in IICBUF is read by the CPU, the software sets SCLEN.

RD_SCL_EN is the read pull-low control bit. When it is 1, it enables the interrupt pull-down function of the clock line. When it is 0, it does not enable the interrupt pull-down function of the clock line.

When RD_SCL_EN=1, when the slave receives an address byte or sends a byte and the master sends an ACK, SCLEN will be automatically pulled low by the hardware, forcing the master to enter the waiting state. To release the IIC clock from the slave, the following two operations are required: first write the data to be sent into the IICBUF, and then the software sets SCLEN. The purpose of this design is to ensure that the data to be sent has been written in IICBUF before SCL is pulled high.

When RD_SCL_EN=0, when the slave receives an address byte or sends a byte and the master sends an ACK, the slave will immediately load the data prepared in the IICBUFFER register to the sending buffer register, and then send it to the data line superior. Therefore, in order to ensure that the data transmitted each time is correct, IICBUFFER prepares the next data to be sent in the interrupt service routine. The data received by the master is the data processed by the previous interrupt, and the first received data is prepared during initialization.

Note: When the clock line needs to be pulled down, that is, WR_SCL_EN/RD_SCL_EN=1, before sending and receiving the last Byte data, the software should turn off the function of pulling down the clock line, that is, WR_SCL_EN/RD_SCL_EN=0, after finishing sending and receiving the last Byte data after that, software should turn on writing to pull down the clock line. This kind of operation can be adjusted by itself according to the host computer is software and hardware, and the interrupt processing time.

IIC_RST is the IIC module control enable bit. When it is 1, it enables the reset function of the

IIC module; when it is 0, it does not enable the reset function of the IIC module. Pay attention to configuration 1 to reset all DFF flip-flops of the IIC module during application. The reset terminal of IIC_RST is a global reset, and the other reset terminals are iic_rst_n. All the iic_rst bits are written to 0 before operating other register configurations.

11.3.5. IIC status register (IICSTAT)

Address offset: 0x0C

Reset value: 0x0000 0044

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Reserved	START	STOP	RW	AD	BF	ACK	WCOL	RECOV
	R	R	R	R	R	R	RW	RW

31:8	-	Reserved
7	START	Start signal flag 1: Indicates that the start bit is detected 0: Indicates that the start bit is not detected
6	STOP	Stop signal flag 1: Indicates that it is in a stopped state 0: Indicates that the stop bit is not detected
5	RW	Read and write flags Record the read/write information obtained from the address byte after the last address match 1: Indicates read operation 0: Indicates write operation
4	AD	Address data flag 1: Indicates that the most recently received or sent byte is data 0: Indicates that the most recently received or sent byte is an address
3	BF	IICBUF full flag When receiving in IIC bus mode: 1: Indicates that the reception is successful and the buffer is full 0: Indicates that the reception is not completed and the buffer is still empty When sending in IIC bus mode: 1: Indicates that data transmission is in progress (not including the response bit and stop bit), and the buffer is still full; 0: Indicates that the data transmission has been completed (not including the response bit and stop bit), and the buffer is empty

2	ACK	Response flag 1: Indicates an invalid response signal 0: Indicates a valid response signal
1	WCOL	Write conflict flag 1: Indicates that when the IIC is sending the current data, new data is trying to be written into the sending buffer; the new data cannot be written into the buffer 0: No write conflict occurred
0	RECOV	Receive overflow flag 1: Indicates that new data is received when the previous data received by IIC has not been taken away, and the new data cannot be received by the buffer 0: Indicates that there is no receiving overflow

The IIC status register is used to reflect the status in the communication process and can be inquired by the user.

START: Start signal status bit. When the start signal is detected, START will be set, indicating that the bus is busy.

STOP: Stop signal status bit. STOP will be set when the chip is in the stop state, indicating that the bus is in an idle state, and cleared by hardware when the start signal is detected, indicating that the communication has started.

AD: Address data flag bit. It indicates that the byte currently received or sent is an address or data. AD=0 indicates that the byte received or sent is an address; AD=1 indicates that the byte received or sent is data. The start signal, stop signal, and non-acknowledgment signal have no effect on this status bit. The change of this status bit occurs on the falling edge of the eighth clock.

RW: Read and write flag bit. This flag bit records the read and write information bits obtained from the address byte after the address matches. RW=1 means that the master reads from the slave, and RW=0 means that the master writes to the slave. The start signal, stop signal, and non-acknowledgment signal (NACK) will all clear RW. The change of this status bit occurs on the falling edge of the ninth clock.

BF: Buffer full flag bit. It indicates that the transceiver buffer is currently full or empty. BF=0 means that the buffer is not receiving data and the buffer is empty; BF=1 means that the buffer has received data and the buffer is full. This status bit can only be set and cleared indirectly, not directly.

When the address matches and RW=0, BF will be set after the falling edge of the 8th clock, which indicates that IICBUF has received data. IICBUF should be read during the execution of the interrupt program, and the operation of reading IICBUF will indirectly clear the BF flag bit. If IICBUF is not read and the master continues to send data, a reception overflow will occur. Although the slave still receives the data sent by the master and ballasts it to IICBUF, it will still send a NACK signal and give an invalid response.

When the address matches and RW=1, the BF flag will not be set after the slave receives the address byte; RW=1 means the master reads the operation of the slave, and the slave needs to write data into IICBUF, and the slave writes The operation of IICBUF will set BF, and then the software will set SCLEN to release the clock line; the master will send the synchronous clock. After the

eighth clock, after the data in IICBUF is sent out, BF is cleared by hardware.

ACK: Acknowledgement status bit. Regardless of whether the master is a read operation or a write operation, the slave will sample the data line on the rising edge of the 9th clock and record the response information. The response bit is divided into a valid response bit ACK and an ineffective response bit NACK. That is to say, the rising edge of the ninth clock samples the data as "0", which means that the ACK is effectively acknowledged, and at the same time ACK is cleared. If the data is sampled as "1", ACK is set, which means non-acknowledgment. After the non-response signal, the master will send a stop signal to announce the end of the communication. The start signal will clear this status bit.

WCOL: Write conflict flag bit. IICBUF can be written by CPU only when RW=1, RD_SCL_EN=1, SCLEN=0. Attempting to write IICBUF under any other circumstances is prohibited. If the above conditions are not met and an operation to write IICBUF occurs, the data will not be written to IICBUF, and the write conflict flag WCOL is set, indicating that a write conflict has occurred , This flag bit needs to be cleared by software.

RECOV: Receive overflow flag bit. When IICBUF is full, that is, when there is data in IICBUF, when IIC receives new data, a receive overflow will occur, RECOV will be set, and the data in IICBUF will not be updated. Newly received data Will be lost. This status bit also needs to be cleared by software, otherwise it will affect the subsequent communication process. This situation will only occur when RW=0, BF=1, and the CPU does not read IICBUF.

11.3.6. IIC data transmission buffer register (IICBUFFER)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								IICBUFFER [7:0]							
								RW							

31:8	-	Reserved
7:0	IICBUFFER [7:0]	IIC data transmission buffer register

The specific application process is as follows:

When RD_SCL_EN is 0, when the master reads data, it sends the data in IICBUFFER to the slave send buffer register 2 clocks after the interrupt, as the data sent by the slave. So prepare IICBUFFER interrupt data before interrupt generation.

11.3.7. IIC control register (IIC_IO_CTRL)

Address offset: 0x44

Reset value: 0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	DFIL_SEL	AFIL_SEL	IIC_SEL
	0	1	0

31:3	-	Reserved
2	DFIL_SEL	IIC function digital filter enable 1: Enable 0: Disable
1	AFIL_SEL	IIC function analog filter enable 1: Enable 0: Disable
0	IIC_SEL	IIC port selection enable 1: PA12/PA13 port select IIC function 0: PB12/PB13 port select IIC function

11.4. IIC master mode

11.4.1. Features

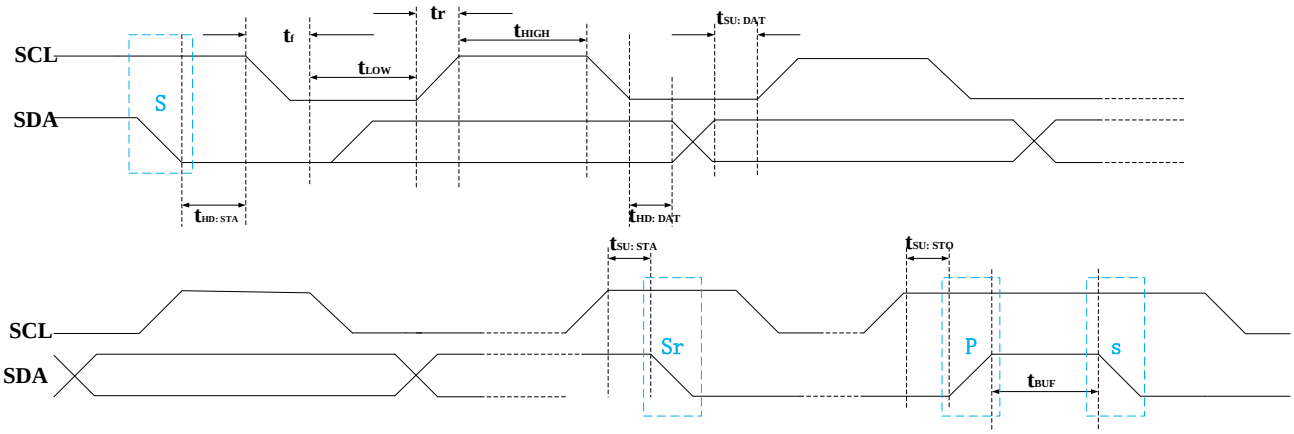
- Multi-master arbitration mode function is optional
- Support standard mode 100kHz, fast mode 400kHz, super fast mode 1MHz
- 7-bit addressing mode
- All 7-bit address response mode
- The timing and time parameters are configurable
- Easy to use interrupt event management
- Support detection of slave clock low-level extension function
- The master extends the clock low level when it is not ready for data and when it is not receiving data
- Support register close enable reset internal state machine and related timing
- Independent clock: iic_clk (24MHz) to make communication speed unaffected by system clock changes

11.4.2. Port configuration

The BF7807AMXX provides register IIC_IO_CTRL[0] to select IIC function.
Write 0 to the register IIC_IO_CTRL[0], then configure PB12 and PB13 as IIC functions:
SCL0A, PB12 are IIC serial clock lines;
SDA0A, PB13 are IIC serial data lines.
Write 1 to the register IIC_IO_CTRL[0] to configure PA12 and PA13 as IIC functions:
SCL0B, PA12 are IIC serial clock lines;
SDA0B, PA13 are IIC serial data lines.

11.4.3. IIC master communication time

The BF7807AMXX meets IIC communication requirements. The following time parameters can be configured. For details, see the description of register IIC_TIMINGR.



S: Start signal, SCL and SDA lines are high at the same time, and a signal from high to low appears on the SDA line.

Sr: Repeated start signal, when there is no stop signal between two start signals, a repeated start signal is generated.

P: Stop signal, when the SCL line is high, a low to high signal appears on the SDA line.

t_{HIGH}: The high level of the SCL clock, which is configured by SCLH;

t_{LOW}: SCL clock low level, configured by SCLL;

$t_{HD:\ DAT}$: The holding time of SDA data is configured by SDADEL;

$t_{HD:\ STA}$: Hold time of the initial condition, configured by SCLH;

$t_{SU:\ STA}$: The establishment time of the repeated start condition, which is configured by SCLL;

$t_{SU:\ STO}$: The establishment time of the stop condition, which is configured by SCLH;

t_{BUF}: Bus idle time between stop and start conditions, configured by SCLL.

- **Without extending the clock low level:**
  The response signal holding time is fixed as: 3*hclk + 4*iic_clk
  The setup time of the highest bit of data is: t_{LOW} – (3*hclk + 4*iic_clk)
  Limiting situation: When the IIC communication frequency is 1M and the system clock is 12M, 3*hclk + 4*iic_clk=416.3ns, the SCLL configuration can be enlarged appropriately, and the SCLH configuration can be reduced.
- **In the case of extending the clock low level:**
  t_{LOW} must be greater than 3*hclk + 2*iic_clk. In the mode of restarting after the transmission is stopped, the START bit needs to be written after the STOPF flag is detected, otherwise the stop bit may not be transmitted successfully. Therefore, the minimum value of t_{LOW} is 3*hclk + 4*iic_clk + $t_{SU:\ SDA}$.
- t_{HIGH} is at least greater than 3*iic_clk.
- $t_{HD:\ DAT}$: SDA data retention time (SDADEL+1)*(PRESC+1)* iic_clk.

Limit case: Both SDADEL and PRESC are configured to 0, and the minimum data retention time is 1 iic_clk.

11.4.4. Interrupt event management

To ensure proper communication, the IIC master mode has several interrupt management events. Open the corresponding enable configuration is valid. For specific configuration, see the IIC_ISR register and the IIC timing description.

- The sending data register is empty INT_TXIS: The data to be sent is not yet ready at the first clock. It occurs when the transmit data register is empty (TXE=1) after the 9th SCL falling edge. This interrupt will not be generated after the last data transmission in non-reload mode is completed.
- Transmission completion interrupt INT_TC: Transmission completion (TC) or transmission completion waiting for reload (TCR). Generates the 9th SCL falling edge when the last data in non-reload mode is completed or 255 data in reload mode is completed.
- The receive data register is not empty INT_RXNE: The last received data has not been read until the eighth clock falling edge. Generated on the 8th SCL falling edge, when the receive data register is full (RXNE = 1).
- Received negative acknowledgement interrupt INT_NACK: Received negative acknowledgement. Generated on the 9th SCL falling edge.
- Stop bit detection interrupt INT_STOP: The master has issued a stop bit.
- Error interrupt INT_ERRIR: Arbitration loss (ARLO) or bus error detection (BERR). This interrupt is generated when arbitration loss or bus error occurs during transmission.

Parameter	After the 8th SCL falling edge	After the 9th SCL falling edge	Number of CPU interrupts during 1-9
Send			
Sending	-	INT_TXIS	1
Send the reloaded 255th data + do not extend the clock low level	-	INT_TXIS + INT_TC	1
Send the reloaded 255th data + extend the clock low level	-	INT_TXIS + INT_TC	2
Send the last non-reload data + software end	-	INT_TC	1
Send the last data of non-reload + auto end	-	-	0
Receive			
In the process of receiving	INT_RXNE	-	1
Receive reloaded 255th data	INT_RXNE	INT_TC	2
Receive the last non-reloaded data +	-	INT_TC	1

software end			
Receive the last data of non-reload + auto end	-	-	0
Other cases			
NACK received, NACKEND=1	-	INT_NACK	1
Received NACK+NACKEND=0 during sending	-	INT_NACK + INT_TXIS	1
Send the reloaded 255th data and receive NACK+Do not extend the clock low level	-	INT_NACK + INT_TXIS + INT_TC	1
Send the reloaded 255th data and receive NACK + extended clock low level	-	INT_NACK + INT_TXIS + INT_TC	2
Send the last data and receive NACK++NACKEND=0+software end	-	INT_NACK + INT_TXIS + INT_TC	1
Send the last data and receive NACK+NACKEND=0+automatically end	-	INT_NACK	1

11.4.5. Software reset

A software reset can be performed by clearing the PE bit in the I2C_CR1 register. In this case, the I2C lines SCL and SDA are released. The internal state machine is reset, and the communication control bits and status bits are restored to their reset values. The configuration registers are not affected.

The affected register bits are listed below:

IIC_CR2 register: START and STOP

IIC_ISR register: TXE, TXIS, RXNE, NACKF, STOPF, TC, TCR, BERR, ARLO, BUSY

When arbitration loss and bus error occur, the internal state machine can be reset by setting the ERR_RST_EN bit in the IIC_CR2 register to 1, release SCL and SDA, and the register will not be affected.

System reset will reset the entire IIC module.

11.5. IIC master registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: Master: 0x5004_0100

Address offset	Register	Description
0x00	IIC_CR1	IIC control register 1
0x04	IIC _CR2	IIC control register 2
0x08	IIC _TIMINGR	IIC timing register
0x0C	IIC_ISR	IIC interrupt and status register
0x10	IIC_ICR	IIC interrupt clear register
0x14	IIC _RXDR	IIC receive data register
0x18	IIC _TXDR	IIC send data register

Base address: 0x500A 0000

Address offset	Register	Description
0x44	IIC_IO_CTRL	IIC control register

11.5.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8
TIMER1_CLKEN	TIMER0_CLKEN	PWM4_CLKEN	PWM3_CLKEN	PWM2_CLKEN	PWM1_CLKEN	PWM0_CLKEN	IIC_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

22	GPIO_CLKEN	GPIO module operation enable 1: Work 0: Off, the default is 0
8	IIC_CLKEN	IIC module operation enable 1: Work 0: Off, the default is 0

11.5.2. IIC control register 1 (IIC_CR1)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:14	13	12:8	7	6	5	4	3	2	1	0
Reserved	IIC_SR	Reserved	ERRIE	TCIE	STOPIE	NACKIE	Res.	RXIE	TXIE	PE
	RW		RW	RW	RW	RW		RW	RW	RW

31:14	-	Reserved
13	IIC_SR	Conversion rate control bit 1: The conversion rate control is turned on to adapt to the standard speed mode (100K) 0: The conversion rate control is turned off to adapt to the fast speed mode (400K)
12:8	-	Reserved
7	ERRIE	Error interrupt enable 0: Disable error detection interrupt 1: Enable error detection interrupt Any of the following errors will generate an interrupt: Arbitration Loss (ARLO), Bus Error Detection (BERR)
6	TCIE	Transfer complete interrupt enable 0: Disable transmission completion interrupt 1: Enable transfer completion interrupt Any of the following errors will generate an interrupt: transfer completion (TC), transfer completion waiting for reload (TCR)
5	STOPIE	Stop bit detection interrupt enable 0: Disable stop bit detection (STOPF) interrupt 1: Enable stop bit detection (STOPF) interrupt
4	NACKIE	Receive negative acknowledge interrupt enable 0: It is forbidden to receive a negative acknowledgement (NACKF) interrupt 1: Enable the reception of a negative acknowledgement (NACKF) interrupt
3	-	Reserved
2	RXIE	RX interrupt enable 0: Disable receiving (RXNE) interrupt 1: Enable receive (RXNE) interrupt
1	TXIE	TX interrupt enable 0: Disable sending (TXIS) interrupt

		1: Enable transmit (TXIS) interrupt
0	PE	Peripheral enable 0: Disable peripherals 1: Enable peripherals

11.5.3. IIC control register 2 (IIC_CR2)

Address offset: 0x04

Reset value: 0x1000 0000

31:29	28	27	26	25	24	23:16
Reserved	NACKEND	ERR_RST_EN	Res.	AUTOEND	RELOAD	NBYTES[7:0]
	RW	RW		RW	RW	RW

15	14	13	12 11	10	9 8	7 6 5 4 3 2 1	0
Res.	STOP	START	Reserved	RD_WRN	Reserved	SADD[7:1]	Res.
	RW	RW		RW		RW	

31:29	-	Reserved
28	NACKEND	Receive NACK automatic end mode, set and cleared by software 0: Not affected after receiving NACK, continue transmission 1: Stop bit will be sent automatically after receiving NACK
27	ERR_RST_EN	Error reset-enable, set and cleared by software 0: Lost arbitration, continue transmission after a bus error occurs 1: Loss of arbitration, reset sequence after bus occurrence. Including releasing the SCL line and SDA line of IIC and resetting the internal state machine
26	-	Reserved
25	AUTOEND	Auto end mode (main mode), set and cleared by software 0: Software end mode: When the NBYTES data transmission is completed, the TC flag will be set to 1, and the low level time of SCL will be extended until the end of the corresponding software operation. (Set the START or STOP position to 1) 1: Auto end mode: When the NBYTES data transmission is completed, the stop bit will be sent automatically Note: When RELOAD is set to 1, this bit has no effect
24	RELOAD	NBYTES reload mode Set and cleared by software 0: After transmitting NBYTES data (followed by stop bit or repeated start bit), the transmission is completed 1: After transmitting NBYTES data, the transmission is not completed (NBYTES will be reloaded). When the NBYTES data transmission is completed, the TCR

		flag will be set to 1, and the low time of SCL will be extended until the end of the corresponding software operation. (NBYTES writes a non-zero value)
23:16	NBYTES[7:0]	Number of bytes Set the number of bytes to be sent/received here Note: When the START position is 1, it is not allowed to change these bits
15	-	Reserved
14	STOP	Stop bit generation (main mode) Set by software, and can be cleared by hardware when the stop bit is detected or when PE=0 In main mode: 0: Do not generate stop bits 1: Generate a stop bit after the current byte transfer is completed Note: Writing "0" to this bit has no effect
13	START	Start bit generation Set by software, and can be cleared by hardware after sending the start bit (followed by the address sequence), when arbitration is lost, or when PE=0. It can also be cleared by software by writing "1" to the ADDRCF bit in the IIC_ICR1 register 0: Do not generate start bit 1: Generate repeated start/start bit If AUTOEND=0, then this bit will generate a repeated start bit after the NBYTES transmission ends and RELOAD=0 Note: Writing "0" to this bit has no effect Set the START bit to 1 even if the bus is busy When RELOAD is set to 1, this bit has no effect
12:11	-	Reserved
10	RD_WRN	Transmission direction (main mode) 0: The master device requests a write transfer 1: The master device requests a read transfer Note: When the START position is 1, it is not allowed to change this bit
9:8	-	Reserved
7:1	SADD[7:1]	Slave address bits[7:1] (master mode) In 7-bit addressing mode: These bits should be written to the 7-bit slave address to be transmitted Note: This bit is not allowed to be changed when the START bit is set
0	-	Reserved

11.5.4. IIC timing register (IIC_TIMINGR)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC				Reserved								SDADEL			
RW												RW			

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

31:28	PRESC	Timing prescaler factor This field is used to prescale IIC_CLK (24MHz) to generate clock period t_{PRESC} for data setup and hold counters and SCL high and low counters t_{PRESC} = (PRESC+1) *t_{IIC_CLK}
27:20	--	Reserved
19:16	SDADEL	Data retention time This field is used to generate the delay t_{SDADEL} between the falling edge of SCL and the edge of SDA. The low time of the SCL line will be extended during t_{SDADEL} t_{SDADEL}= (SDADEL+1) *t_{PRESC} Note: SDADEL is used to generate $t_{HD:\ DAT}$ timing
15:8	SCLH	SCL high period This field is used to generate SCL high level period t_{SCLH}= (SCLH+1) *t_{PRESC} Note: SCLH is also used to generate $t_{SU:\ STO}$ and $t_{HD:\ STA}$ timing
7:0	SCLL	SCL low period, not configured as 0 This field is used to generate SCL low period t_{SCLL}= (SCLL+1) *t_{PRESC} Note: SCLL is also used to generate t_{BUF} and $t_{SU:\ STA}$ timing

11.5.5. IIC interrupt and status register (IIC_ISR)

Address offset: 0x0C

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14:10	9	8	7	6	5	4	3	2	1	0
BUSY	Reserved	ARLO	BERR	TCR	TC	STOPF	NACKF	Res.	RXNE	TXIS	TXE
R		R	R	R	R	R	R		R	R	RS

31:16	-	Reserved

15	BUSY	Bus is busy This flag is used to indicate that there is communication on the bus. When the start bit is detected, the bit is set by hardware; when the stop bit is detected or PE=0, this bit is cleared by hardware
14:10	-	Reserved
9	ARLO	Arbitration lost When arbitration is lost, this flag is set by hardware This flag is cleared by software by setting ARLOCF to 1 Note: When PE=0, this bit is cleared by hardware
8	BERR	Bus error When a misplaced start bit or stop bit is detected, and the peripheral is also involved in the transmission, the flag is set by hardware. This flag is cleared by software by setting BERRCF to 1 Note: When PE=0, this bit is cleared by hardware
7	TCR	Transmission is complete and waiting for reload When RELOAD=1 and NBYTES data transmission is completed, this flag is set by hardware When NBYTES writes a non-zero value, the flag is cleared by software Note: When PE=0, this bit is cleared by hardware
6	TC	Transfer complete (main mode) When RELOAD=0, AUTOEND=0 and NBYTES data transmission is completed, this flag is set by hardware. When the START bit or STOP bit is set to 1, the flag is cleared by software Note: When PE=0, this bit is cleared by hardware
5	STOPF	Stop bit detection flag When the stop bit is detected on the bus and the peripheral is also participating in this transfer, the flag is set by hardware. The premise of this bit is that the peripheral has issued a stop bit This flag is cleared by software by setting STOPCF to 1 Note: When PE=0, this bit is cleared by hardware
4	NACKF	Negative response sign received When a NACK is received after the byte is transmitted, the flag is set by hardware. This flag is cleared by software by setting NACKF to 1. Note: When PE=0, this bit is cleared by hardware
3	-	Reserved
2	RXNE	Receive data register is not empty (receiver) When the received data has been copied to the IIC_RXDR register and is ready for reading, this flag is set by hardware. When reading IIC_RXDR, this bit will be cleared Note: When PE=0, this bit is cleared by hardware

1	TXIS	Transmit interrupt status (transmitter) When the IIC_TXDR register is empty, this bit is set by hardware. The data to be sent must be written into the IIC_TXDR register. When the next data to be sent is written to the IIC_TXDR register, this bit is cleared Write "1" to this bit, invaild Note: When PE=0, this bit is cleared by hardware
0	TXE	Send data register is empty (transmitter) When the IIC_TXDR register is empty, this flag is set by hardware. When the next data to be sent is written to the IIC_TXDR register, this bit is cleared This bit can be written "1" by software to refresh the transmit data register IIC_TXDR After the last data is sent, the hardware is set to 1 Note: When PE=0, this bit is set by hardware

11.5.6. IIC interrupt clear register (IIC_ICR)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:10	9	8	7	6	5	4	3:0
Reserved	ARLOCF	BERRCF	Res.		STOPCF	NACKCF	Reserved
	W	W			W	W	

31:10	-	Reserved
9	ARLOCF	Arbitration lost flag cleared Write 1 to this bit, the ARLO flag in the IIC_ISR register will be cleared
8	BERRCF	Bus error flag is cleared Write 1 to this bit, the BERR flag in the IIC_ISR register will be cleared
7:6	-	Reserved
5	STOPCF	Clear the stop bit detection flag Write 1 to this bit, the STOPF flag in the IIC_ISR register will be cleared
4	NACKCF	Negative response flag is cleared Write 1 to this bit, the NACKF flag in the IIC_ISR register will be cleared
3:0	-	Reserved

11.5.7. IIC receive data register (IIC_RXDR)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								RXDATA[7:0]							
								RW							

31:8	-	Reserved
7:0	RXDATA[7:0]	8-bit receive data Data byte received from IIC bus

11.5.8. IIC transmit data register (IIC_TXDR)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TXDATA[7:0]							
								RW							

31:8	-	Reserved
7:0	TXDATA[7:0]	8-bit send data Data bytes to be sent to the IIC bus Note: These bits can only be written when TXE=1

11.5.9. IIC control register (IIC_IO_CTRL)

Address offset: 0x44

Reset value: 0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	DFIL_SEL	AFIL_SEL	IIO_SEL
	RW	RW	RW

31:3	-	Reserved
2	DFIL_SEL	IIC function digital filter enable

		1: Enable 0: Disable
1	AFIL_SEL	IIC function analog filter enable 1: Enable 0: Disable
0	IIC_SEL	IIC port selection enable 1: PA12/PA13 port select IIC function 0: PB12/PB13 port select IIC function

11.6. IIC master configuration process

The master mode configuration process is performed according to the following steps:

1. Clear the PE (peripheral enable) in IIC_CR1.
2. Configure digital filter IIC_DFIL_SEL and analog filter IIC_AFIL_SEL.
3. Configure SDADEL[3:0], SCLH[7:0] and SCLL[7:0] in IIC_TIMINGR.
4. Set the PE bit (peripheral enable) in IIC_CR1 to 1.
5. Slave address to be sent: SADD[7:1]; transmission direction: RD_WRN; number of bytes to be transmitted: NBYTES[7:0]; and other enable control bits.

a) The maximum value of NBYTES[7:0] is 255, so if the data to be transmitted is greater than 255, a combination of reload mode and non-reload mode must be used. When the data to be transmitted is less than or equal to 255, just use the non-reload mode.

b) When the number of bytes to be transmitted is less than or equal to 255, RELOAD=0 must be configured. Configure AUTOEND=1 to automatically generate a stop bit after the end of the transmission; configure AUTOEND=0 to detect the TC flag after the end of the transmission, so that the configuration can restart (START) or stop (STOP). (The above configuration can be reconfigured before the configuration restarts). When the number of bytes to be transmitted is equal to 255, NBYTES[7:0] must be filled with 0xFF during initialization.

c) When the number of bytes to be transmitted is greater than 255, RELOAD=1 must be configured, and NBYTES[7:0] must be filled with 0xFF during initialization. After the 255 data is transmitted, TCR is detected. If the remaining number to be transmitted is still greater than 255, NBYTES[7:0] must be filled with 0xFF again. Until the remaining number to be transmitted is less than or equal to 255, when TCR is detected, configure RELOAD=0, and configure AUTOEND and NBYTES[7:0].

6. Detect that the BUSY flag is 0, set the START bit in the IIC_CR2 register to 1. When the START bit is 1, it is not allowed to change all the bits in step 5. TXDATA[7:0] can be configured in step 5 or after the START bit is set.
7. When it detects that the bus is free, it will automatically send the start bit after a delay of t_{BUF}, and then send out the slave device address.

Note:

1. **Before opening the peripheral enable, the filter and timing registers should be configured. After the enable is turned on, the filtering and timing configuration are not allowed to be changed.**
2. **If NBYTES[7:0] is filled with 0, the stop bit will be sent automatically after sending the address.**

12 Serial peripheral interface (SPI)

12.1. SPI function description

The SPI of BF7807AMXX is a serial, synchronous, full/half duplex communication bus, which supports intermittent communication and high-speed continuous communication. The BF7807AMXX contains two modules, SPI0 and SPI1.

The maximum communication frequency of the master is 8M. The slave receives the highest communication frequency of 4M, and sends the highest communication frequency of 4M.

SPI normal mode: MCU writes into SPI transmit buffer SPID through interrupt (when SPI enable is turned on, immediately generates send empty interrupt) or polling, data is automatically loaded into shift register, and sent to SPI_MOSI synchronously via SCLK; at the same time from SPI_MISO receives data and loads it into the SPI receive buffer. When a receive full interrupt is generated, the received data can be read from SPID.

SPI high-speed mode: MCU advances to SRAM to write and send data (up to 4K can be written), during communication, SPI directly reads the data to be sent from SRAM without interrupting or polling; at the same time, each time a piece of data is received (8Bits), write the corresponding address of SRAM immediately. When the communication is completed, the SPI simultaneously generates the empty sending flag and the receiving full flag, and sends an interrupt. Only one SPI high-speed communication is allowed at the same time.

12.1.1. Features

- Standard SPI, two-wire SPI mode
- SPI master mode, SPI slave mode
- Multi-slave function
- In the slave mode, the master can pull up the chip select to reset the SPI counter, and re-communication in the high-speed mode
- Four communication modes are optional
- There is a first level cache for sending and receiving
- High-speed mode supports communication completion interrupt
- SPI baud rate baudrate=F_{HCLK} / [(SPR+1) *2], SPI clock duty cycle is 50%
- The effective bit width of the SPI data buffer is configurable: 8/9/10/11/12/16/24/32 bits, the high-speed mode can only operate SRAM by word according to the selection
- High-speed mode supports SRAM address overflow interrupt. In the high-speed mode, after the access to the SRAM address reaches the maximum value, the SPI stops working and generates an interrupt when the SRAM address reaches the maximum value to prevent the system from entering hardfault

Hardfault: Read the corresponding overflow flag, turn off the high-speed mode, reconfigure the corresponding register, and restart the high-speed mode.

12.1.2. Port configuration

To use the SPI0 function, you need to configure the relevant port as an SPI channel, and select the corresponding port input through SPI0_IO_CTRL[0] register. As shown in the

SPI0_IO_CTRL[0]= 1	SPI0_IO_CTRL[0]= 0
SPI0B_CS: PA10, SPI chip selection signal SPI0B_CLK: PA11, SPI clock SPI0B_MOSI: PA12, SPI master data output SPI0B_MISO: PA13, SPI master data input	SPI0B_CS: PB12, SPI chip selection signal SPI0B_CLK: PB13, SPI clock SPI0B_MOSI: PB14, SPI master data output SPI0B_MISO: PB15, SPI master data input

SPI high-speed mode: Only one channel is allowed at the same time, through the SRAM_SPI_SEL[0] register configuration:

SRAM_SPI_SEL[0]= 1	SRAM_SPI_SEL[0]= 0
SPI1	SPI0

12.1.3. SPI multi-slave function

1. In SPI master mode, the SPI_CS chip select port is a normal IO port; when SPI_EN=1, the SPI_MISO port can be used as a normal IO port;
2. In SPI slave mode, the SPI_CS chip select port is a dedicated chip select input port for SPI;
3. In SPI multi-slave mode (MULTI_SLAVE_EN=1), when SPI_CS=1, MISO is always an input port, and SPI_CS=0 becomes an output port only after SPI_CS=0 is selected;
4. In SPI single slave mode (MULTI_SLAVE_EN=0), MISO is always the output port.

12.1.4. Four modes

Four modes of SFR configuration:

CPOL: Select the clock idle state level:

0: The idle state of the clock is low;

1: The idle state of the clock is high.

CPHA: Select the data moment of each cycle:

0: Data sampling is performed on the first transition edge (rising or falling edge) of the clock;

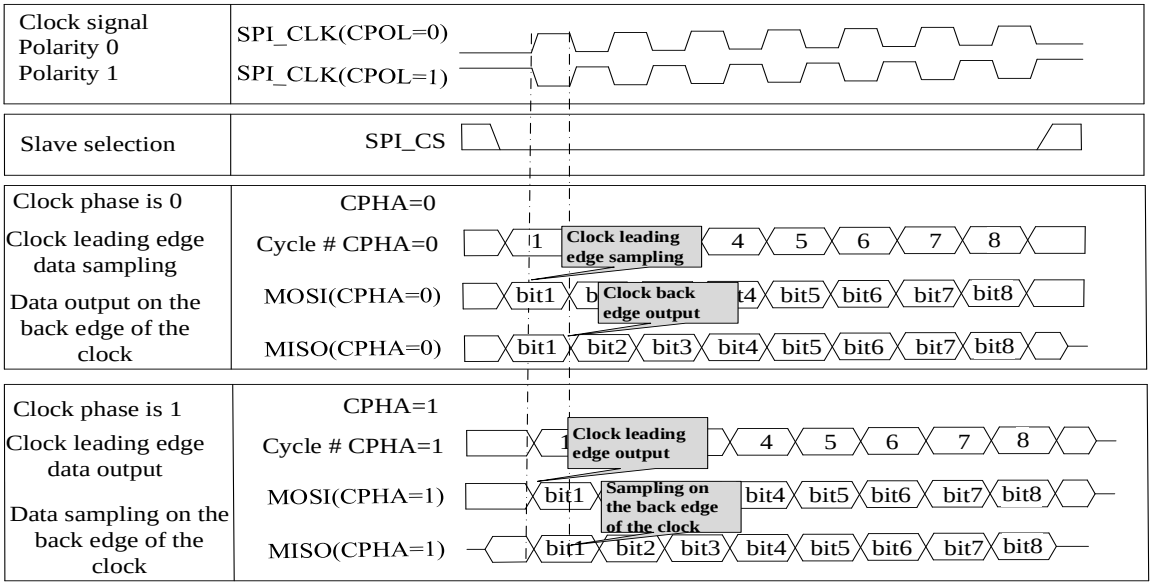
1: Data sampling is performed on the second transition edge (rising or falling edge) of the clock.

Mode 0 (CPOL=0, CPHA=0): The idle level of the clock is low, and the master and slave are sampling on the rising edge.

Mode 1 (CPOL=0, CPHA=1): The idle level of the clock is low, and the master and slave are sampling on the falling edge.

Mode 2 (CPOL=1, CPHA=0): The idle level of the clock is high, and the master and slave are sampling on the falling edge.

Mode 3 (CPOL=1, CPHA=1): The idle level of the clock is high, and the master and slave are sampling on the rising edge.



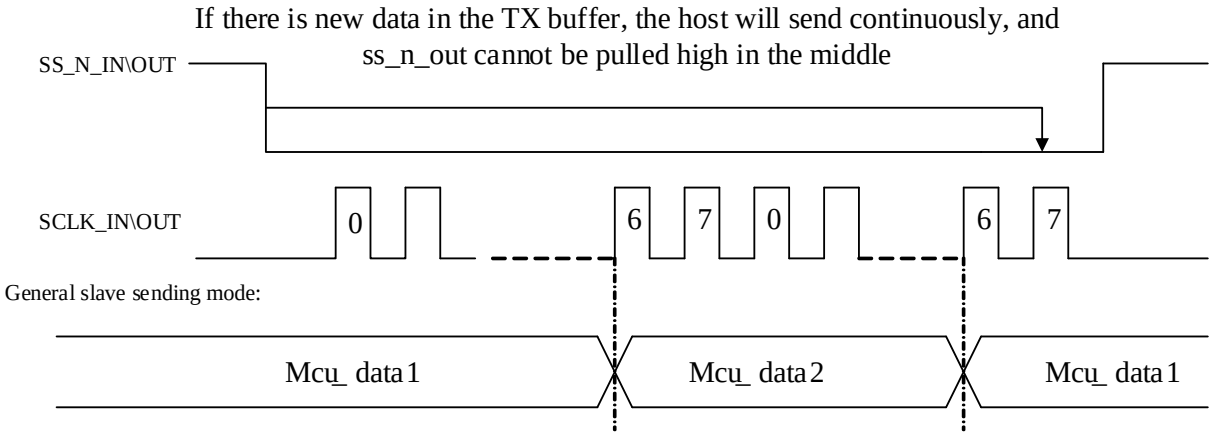
SPI working mode timing diagram

Description: SI: Slave sampling data; SO: Slave sending data; MI: Master sampling data; MO: Master sending data; SPI_CS high level minimum time requirement is 1 SPI clock cycle.

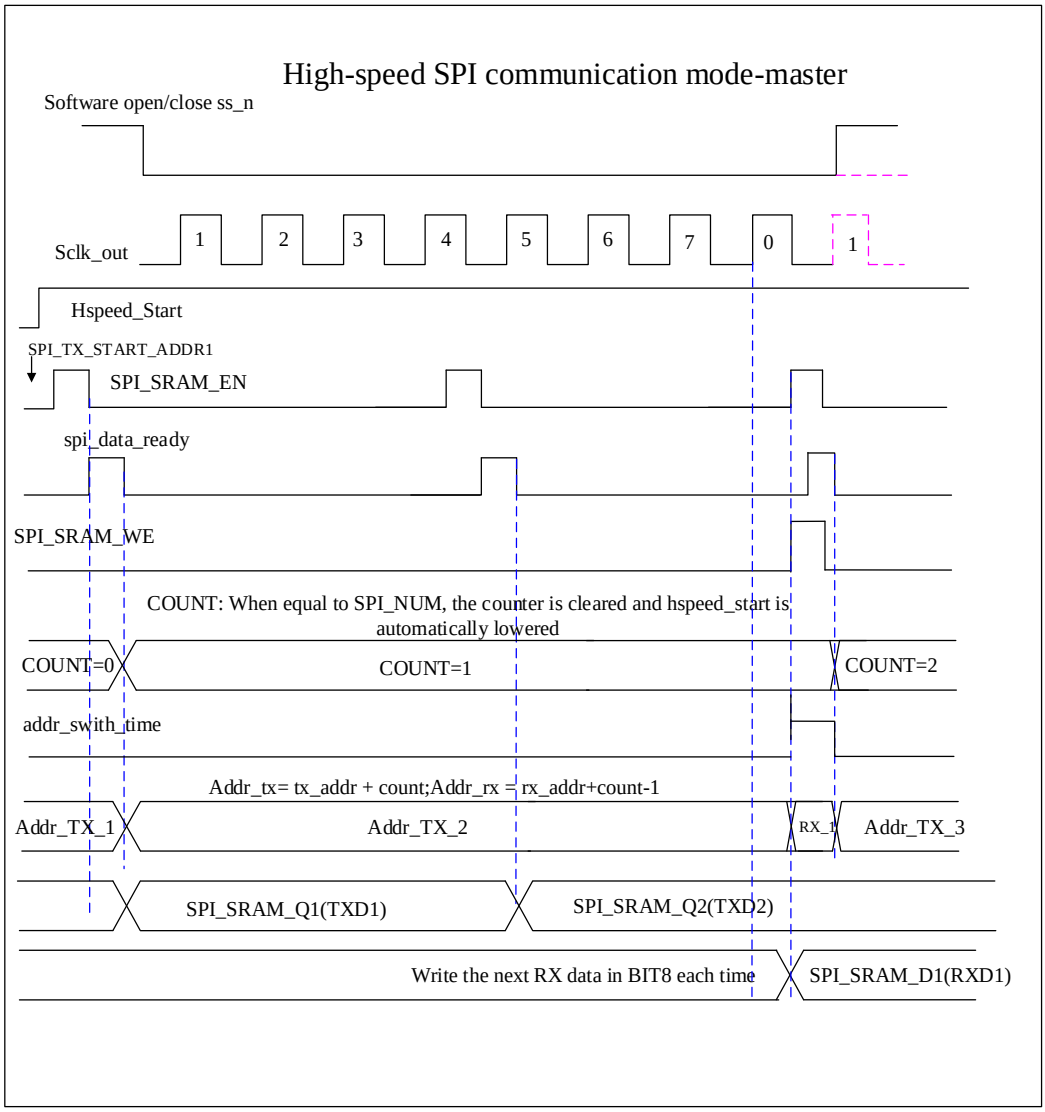
12.1.5. Communication timing

There are three flag bits, two interrupt mask bits and an interrupt vector related to the SPI system. The SPI receive interrupt enable bit (RX_IE) allows interrupts from the SPI receiver full flag (SPRF) to occur. The SPI transmit interrupt enable bit (TX_IE) allows interrupts from the SPI transmit buffer empty flag (SPTIEF) to occur. When a flag bit is set and the related interrupt enable bit is set, the hardware interrupt request is sent to the CPU. If the interrupt enable bit is cleared, the software can poll the relevant flag bit without interruption. The SPI interrupt service routine (ISR) should check the flag bit to determine the event that caused the interrupt. Before returning from the ISR (usually near the starting point of the ISR), the service program should also clear the flag bit.

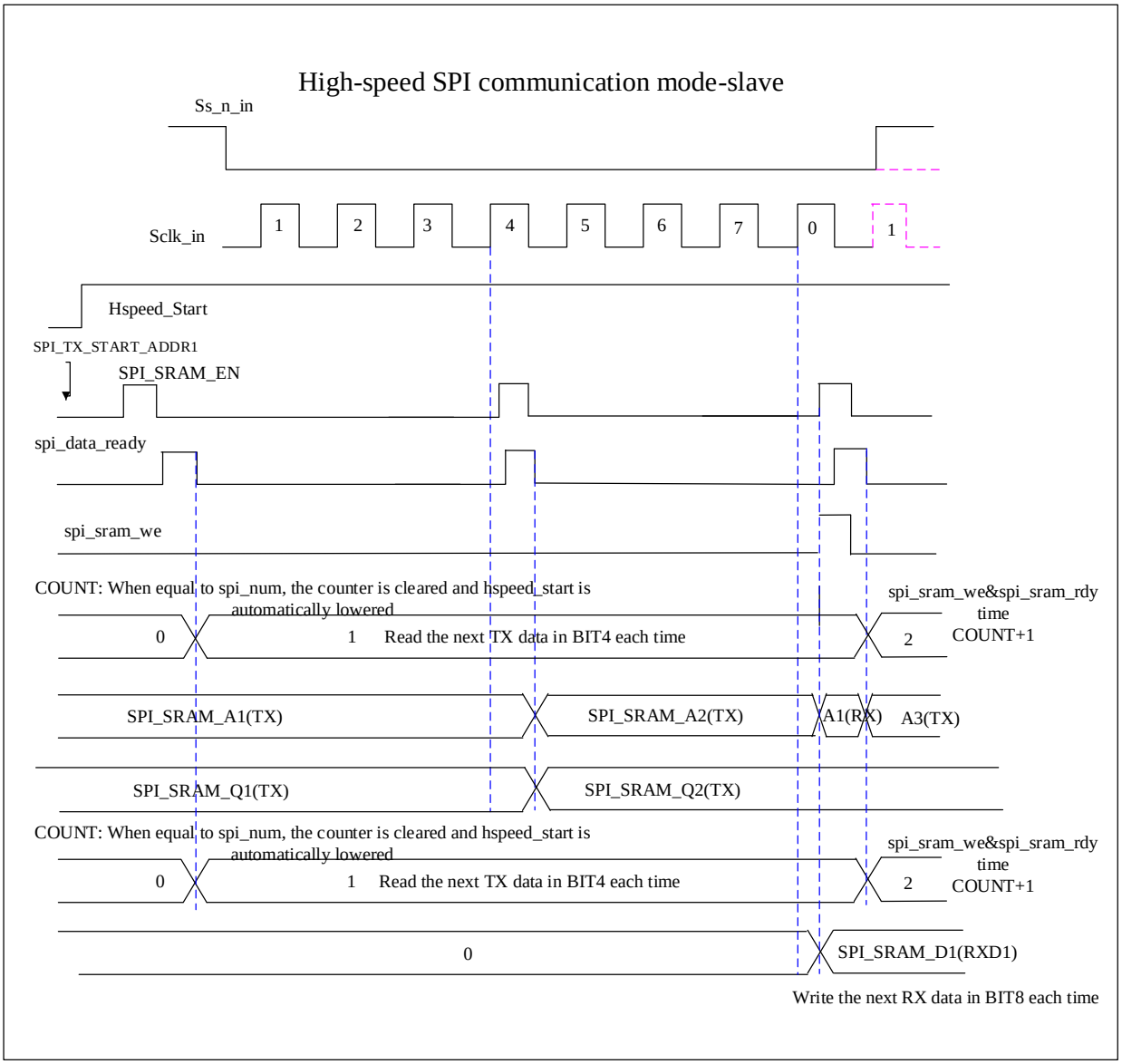
Schematic diagram of SPI continuous working in normal communication mode:



Schematic diagram of SPI continuous working in high-speed communication mode (master):



Schematic diagram of SPI continuous working in high-speed communication mode (slave):



12.2. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register
0x34	SRAM_SPI_SEL	SPI configuration register

Base address: SPI0: 0x5002_0000; SPI1: 0x5002_0100; x=0/1

Address offset	Register	Description
0x00	SPIx_CFG1	SPI0/1 configuration register 1
0x04	SPIx_CFG2	SPI0/1 configuration register 2
0x08	SPIx_STATE	SPI0/1 status register
0x0C	SPIx_SPID	SPI0/1 cache operation register
0x10	SPIx_TX_START_ADDR	SPI0/1 high-speed mode transmit buffer first address
0x14	SPIx_RX_START_ADDR	SPI0/1 high-speed mode receive buffer first address
0x18	SPIx_NUM	SPI0/1 high-speed mode data buffer address number

Base address: 0x500A 0000

Address offset	Register	Description
0x48	SPI_CLK_CFG	SPI input clock configuration register
0x4C	HSPEED_EN	SPI high-speed mode enable register
0x50	SPI0_IO_CTRL	SPI0 select enable register

12.2.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

7	6	5	4	3	2	1	0
UART4_CLKEN	UART3_CLKEN	UART2_CLKEN	UART1_CLKEN	UART0_CLKEN	SPI1_CLKEN	SPI0_CLKEN	Reserved
RW	RW	RW	RW	RW	RW	RW	

22	GPIO_CLKEN	GPIO module operation enable 1: Work

		0: Off, the default is 0
2	SPI1_CLKEN	SPI1 module operation enable 1: Work 0: Off, the default is 0
1	SPI0_CLKEN	SPI0 module operation enable 1: Work 0: Off, the default is 0

12.2.2. SPI configuration register (SRAM_SPI_SEL)

Address offset: 0x34

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:1	0
Reserved	SRAM_SPI_SEL
	RW

31:1	-	Reserved
0	SRAM_SPI_SEL	SPI selection when high-speed SPI0/1 communicates directly with SRAM for reading and writing 0: SPI0 1: SPI1

12.2.3. SPI0/1 configuration register 1 (SPIx_CFG1)

Address offset: 0x00

Reset value: 0x0000 0011

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:11	10	9	8	7	6	5	4	3	2	1	0
Reserved	MULTI_SLAVE_EN	HSPEED_OVERFLOW_IE	HSPEED_IE	RX_IE	SPI_EN	TX_IE	MSTR	CPOL	CPHA	LSBFE	CS_N
	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:11	-	Reserved
10	MULTI_SLAVE_EN	Multi-slave function enable

		1: SPI is used as a slave and used in a multi-slave environment (MISO is the output port when the chip select is pulled low, and the input port when the chip select is pulled high) 0: SPI as a slave, used in a single-slave environment (MISO is fixed as an output port)
9	HSPEED_OVERFLOW_IE	High-speed communication SRAM address overflow flag, when the communication address has reached the maximum value of SRAM and the communication is not completed, the interrupt is generated 1: The interrupt is valid 0: Disable interrupt (use polling) Note: It needs to be turned on at the same time as the high-speed communication completion interrupt enable to prevent the SRAM address from overflowing and the system enters the hardfault interrupt
8	HSPEED_IE	High-speed communication completion interrupt enable 1: The interrupt is valid 0: Disable interrupt (use polling) Note: first turn off the transmit interrupt enable TX_IE, then turn on the high-speed communication to complete the interrupt enable
7	RX_IE	Receive interrupt enable-this is the SPI receive buffer full (SPRF) interrupt enable 1: The interrupt is valid 0: Disable interrupt (use polling)
6	SPI_EN	SPI enable 1: Module enable is turned on 0: Module enable is closed
5	TX_IE	Transmit interrupt enable-this is the SPI transmit buffer empty (SPTEF) interrupt enable 1: The interrupt is valid 0: Disable interrupt (use polling) Note: In high-speed mode, you need to turn off the transmit interrupt enable, otherwise the high-speed communication will generate high-speed completion interrupt and send empty interrupt respectively
4	MSTR	Master-slave mode selection 1: Master mode 0: Slave mode
3	CPOL	SCLK active level selection 1: Active low 0: Active high
2	CPHA	SCLK phase selection 1: Send data on the first valid clock edge 0: Sample data at the first valid clock edge

1	LSBFE	LSB first (shifter direction) 1: SPI serial data transmission starts from the lowest bit 0: SPI serial data transmission starts from the most significant bit
0	CS_N	Chip select signal Note: The IO corresponding to the chip select is a low-speed IO, and the output delay is large. When communicating as a master: When starting communication, you need to pull down the IO corresponding to the chip select first, and after a delay of 1us, pull CS_N low; end the communication When the time, the external IO needs to be pulled high first, after a 1us delay, then CS_N is pulled high

12.2.4. SPI0/1 configuration register 2 (SPIx_CFG2)

Address offset: 0x04

Reset value: 0x0000 0060

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:14	13	12	11:9	8	7	6	5	4:0
Res.	DMOD	MISO_DISABLE	SPID_SIZE	Res.	HSPEED_START	HALF_DUPLEX	BIDIR_SELECT	SPR
	RW	RW	RW		RW	RW	RW	RW

31:14	-	Reserved
13	DMOD	DSPI/SPI selection 0: Standard SPI mode 1: Two-wire SPI mode
12	MISO_DISABLE	SPI_MISO port can be used as normal IO port when SPI_EN=1 0: MISO 1: Ordinary IO
11:9	SPID_SIZE[2:0]	In general SPI communication, the control bit is SPID_SIZE[2:0], write/read the effective data width of the SPI_SPID register (corresponding interrupt flag, the setting interval will be extended with this configuration): Note: It must be updated under IDLE. If there is an error, it means that the data has not been sent. Please update SPID_SIZE in advance. 000: 8 bits ---only for single-line normal mode 001: 9 bits ---only for single-line normal mode 010: 10 bits ---only for single-wire normal mode 011: 11 bits ---only for single-wire normal mode 100: 12 bits ---only for single-line normal mode

		101: 16 bits ---only for single-line and dual-line normal mode 110: 24 bits --- Send odd numbers of data in high-speed mode 111: 32 bits --- Send an even number of data in high-speed mode
8	-	Reserved
7	HSPEED_START	The high-speed SPI communication mode is turned on, and the hardware is automatically pulled down after the work is completed 1: High-speed SPI communication mode is on 0: High-speed SPI communication mode is off In high-speed SPI mode, whether in slave or master mode, the chip select signal cannot be pulled high, which will cause the data sent by SPI to be lost Note: 1) Only one SPI high-speed communication is allowed at the same time 2) After HSPEED_OVERFLOW=1, SPI will stop working, the hardware pulls down HSPEED_START, and generates HSPEED_OVERFLOW interrupt to prevent SRAM address overflow from causing the system to enter hardfault (clear the corresponding overflow flag, reconfigure the corresponding register, and restart the high-speed mode)
6	HALF_DUPLEX	Half-duplex mode selection 1: Select half duplex mode 0: Select full duplex mode
5	BIDIR_SELECT	Half-duplex mode, sending and receiving direction selection 1: Send 0: Receive
4:0	SPR	SPI baud rate coefficient, the highest communication frequency of master: 8M, the highest communication frequency of slave receiving is 4M, and the highest communication frequency of sending is 4M Baudrate=F_{HCLK} / [(SPR+1)*2]

12.2.5. SPI status register (SPIx_STATE)

Address offset: 0x08

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Reserved	HSPEED_OVERFLOW	RECE_NUM			HSPEED_FLG	SPRF	OVERFLOW_RX	SPTEF
	RC_W0	RC_W0			RC_W0	RC_W0	RC_W0	R

31:8	-	Reserved
7	HSPEED_OVERFLO	High-speed communication SRAM address overflow flag, software writes 0 to

	W	clear (Once the SRAM address overflows, the system enters the hardfault interrupt, so the detection FLAG is added) Note: 1) Used to remind the user that the configured SPI_TX_START_ADDR, SPI_RX_START_ADDR, and SPI_NUM have reached the maximum address of SRAM 0x7FF during the incomplete high-speed communication 2) Once the flag is set to 1, after reconfiguring the register, it is necessary to add a delay of the length of SPI data, and then clear the HSPEED_OVERFLOW flag, otherwise it will generate a receiving full interrupt flag; then restart HSPEED_START=1 (because if In receiving mode, when receiving SRAM address = 0x7FF, it is found that COUNT<SPI_NUM, and when the hardware pulls down HSPEED_START, the next data is ready, so another extra data will be sent, but SRAM will not be accessed again)
6:4	RECE_NUM	The valid number of currently received data (byte/piece), used for the last data in slave mode to stop working without interruption, query the flag bit to determine the valid number of received data: Write 0 to clear 0. It is recommended to read this state twice at a reasonable interval and clear it to 0 to avoid losing valid data. Writing 1 is invalid. 001: RECE_BUF[7:0] 011: RECE_BUF[15:0] 111: RECE_BUF[23:0] Note: Only applicable to SPID_SIZE selection: 8/16/24/32
3	HSPEED_FLG	High-speed communication complete mark, software write 0 to clear
2	SPRF	Read buffer full mark, software write 0 to clear
1	OVERFLOW_RX	In the normal communication mode, when the receiving overflow is caused by not reading in time, OVERFLOW_RX=1, the signal will not generate interruption, only mark In high-speed SPI communication mode, it is invalid (when the number of received data is equal to the configured {SPI_NUM_H, SPI_NUM_L}, the work will end, SPRF will be set, and a full interrupt will be generated)
0	SPTEF	Send buffer empty mark, write into SPID hardware to clear automatically. In the SPI idle state, the first data written to SPID will be directly stored in the shift register, and the second data written will be loaded into the transmit buffer, and SPTEF will be automatically pulled low

12.2.6. SPI0/1 buffer operation register (SPIx_SPID)

Address offset: 0x0C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPID[31:16]															

RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPID[15:0]															
RW															

31:0	SPID[31:0]	Reading this register will return the data read from the receive data buffer: When reading the SPI status register first, judging SPRF=1, and then reading the SPID, the SPRF flag can be cleared and the data in the receive data buffer can be read correctly. Writing to this register will write data into the transmit data buffer: Read the SPI status register first, and judge SPTEF = 1 (transmit buffer empty flag), and then write the SPID register (transmit data buffer). When the SPI is configured as a master, writing data to the transmit data buffer will initiate an SPI transfer. After SPRF is set and before another transmission is completed, data can be read from SPID at any time. If you fail to read data from the receive data buffer before the end of the new transmission, it will cause the receive overflow, and the newly transmitted data will be lost.

12.2.7. SPI0/1 high-speed mode transmit buffer start address (SPIx_TX_START_ADDR)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:12	-	Reserved
11:0	TXSA	The starting address (IP physical address, word unit) to fetch data from SRAM during SPI fast transmission (configure the address first, and then turn on the fast mode) Note 1: If the fast communication mode is not activated, the readout value is the register configuration value; if the fast communication mode is activated, the readout value is the real-time count value Note 2: SPI_TX_START_ADDR, SPI_NUM must be in Word (32Bit) as the

		unit, otherwise an error will occur

12.2.8. SPI0/1 high-speed mode receive buffer first address (SPIx_RX_START_ADDR)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:12	-	Reserved
11:0	RXSA	The start address of the data stored in SRAM when SPI is received

12.2.9. SPI0/1 high-speed mode data buffer address number (SPIx_NUM)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				NUM[12:0]											
				RW											

31:13	-	Reserved
12:0	NUM[12:0]	SPI fast read/write data number (unit: Word), not working when SPI_NUM=0, SPI_NUM_MAX= 0x1000 Note: 2K Word -1> the configuration value must be greater than 2

12.2.10. SPI input clock configuration register (SPI_CLK_CFG)

Address offset: 0x48

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:2	1	0

Reserved	SPI1_CLK_CFG	SPI0_CLK_CFG
	RW	RW

31:2	-	Reserved
1	SPI1_CLK_CFG	SPI1 input clock selection configuration 1: Use digital internally generated clock (master mode exists) 0: Use analog input signal
0	SPI0_CLK_CFG	SPI0 input clock selection configuration 1: Use digital internally generated clock (master mode exists) 0: Use analog input signal

12.2.11. SPI high-speed mode enable register (HSPEED_EN)

Address offset: 0x4C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:9	8	7	6	5	4	3	2	1	0
Reserved	HSSEL8	HSSEL7	HSSEL6	HSSEL5	HSSEL4	HSSEL3	HSSEL2	HSSEL1	HSSEL0
	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:9	-	Reserved
8:0	HSSEL[8:0]	Bit0: SPI communication port PA11 configuration high-speed mode enable Bit1: SPI communication port PA12 configuration high-speed mode enable Bit2: SPI communication port PA13 configuration high-speed mode enable Bit3: SPI communication port PB12 configuration high-speed mode enable Bit4: SPI communication port PB13 configuration high-speed mode enable Bit5: SPI communication port PB14 configuration high-speed mode enable Bit6: SPI communication port PC8 configuration high-speed mode enable Bit7: SPI communication port PC9 configuration high-speed mode enable Bit8: SPI communication port PC10 configuration high-speed mode enable The corresponding bits of HSSEL[8:0] are: 1: High-speed mode; 0: Normal mode

12.2.12. SPI0 select enable register (SPI0_IO_CTRL)

Address offset: 0x50

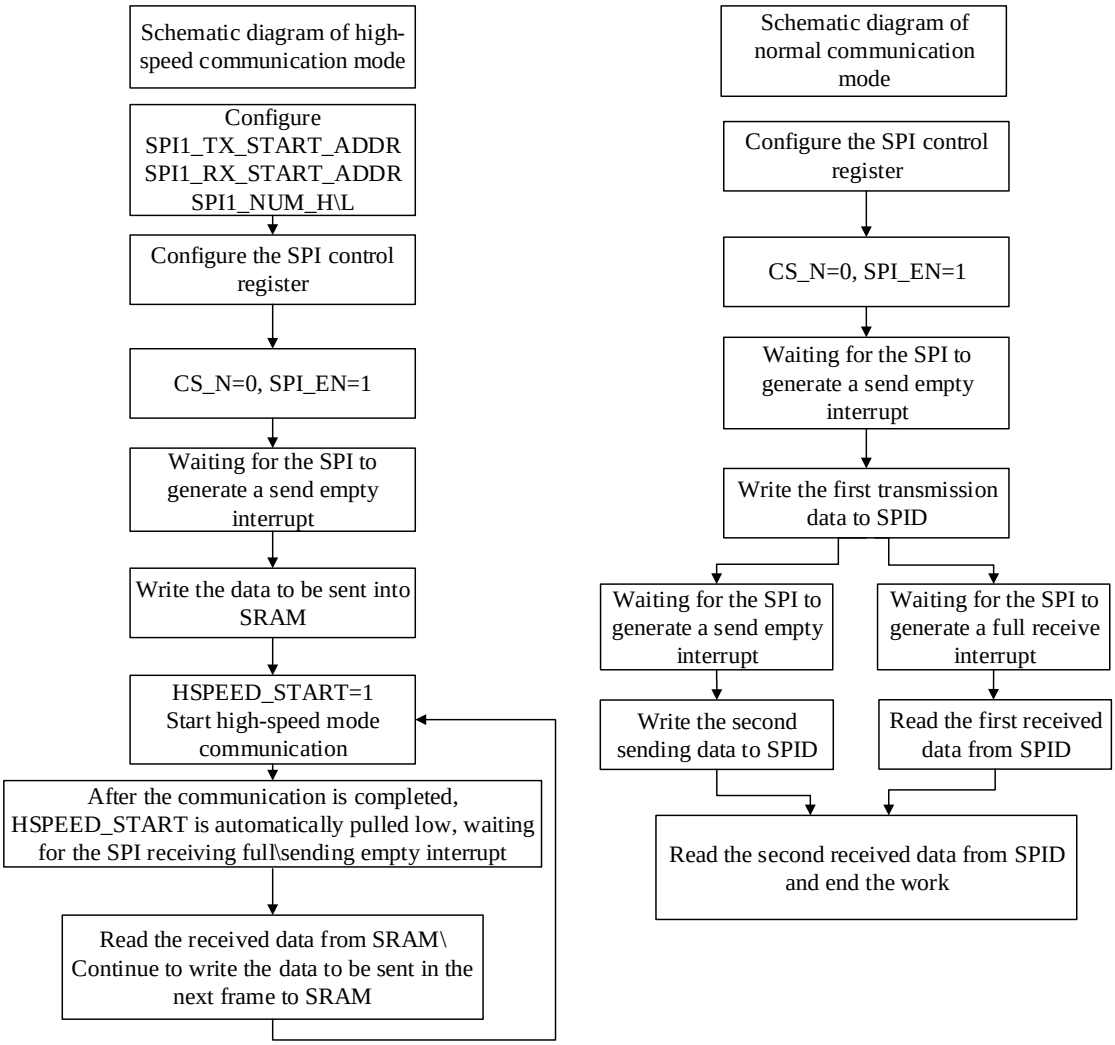
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:1	0
Reserved	SPI0_SEL
	RW

31:1	-	Reserved
0	SPI0_SEL	SPI0 port selection enable 0: PB12/PB13/PB14/PB15 port select SPI0 function 1: PA10/PA11/PA12/PA13 port select SPI0 function

12.3. SPI work flow chart



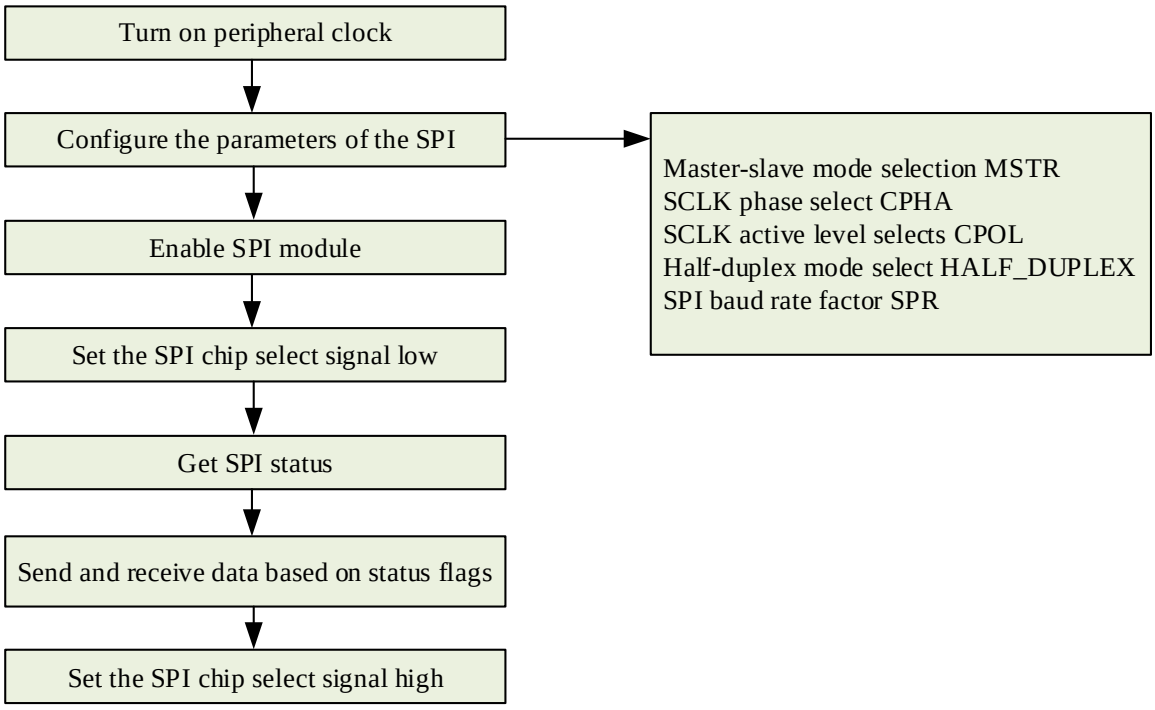
SPI work flow diagram

Note:

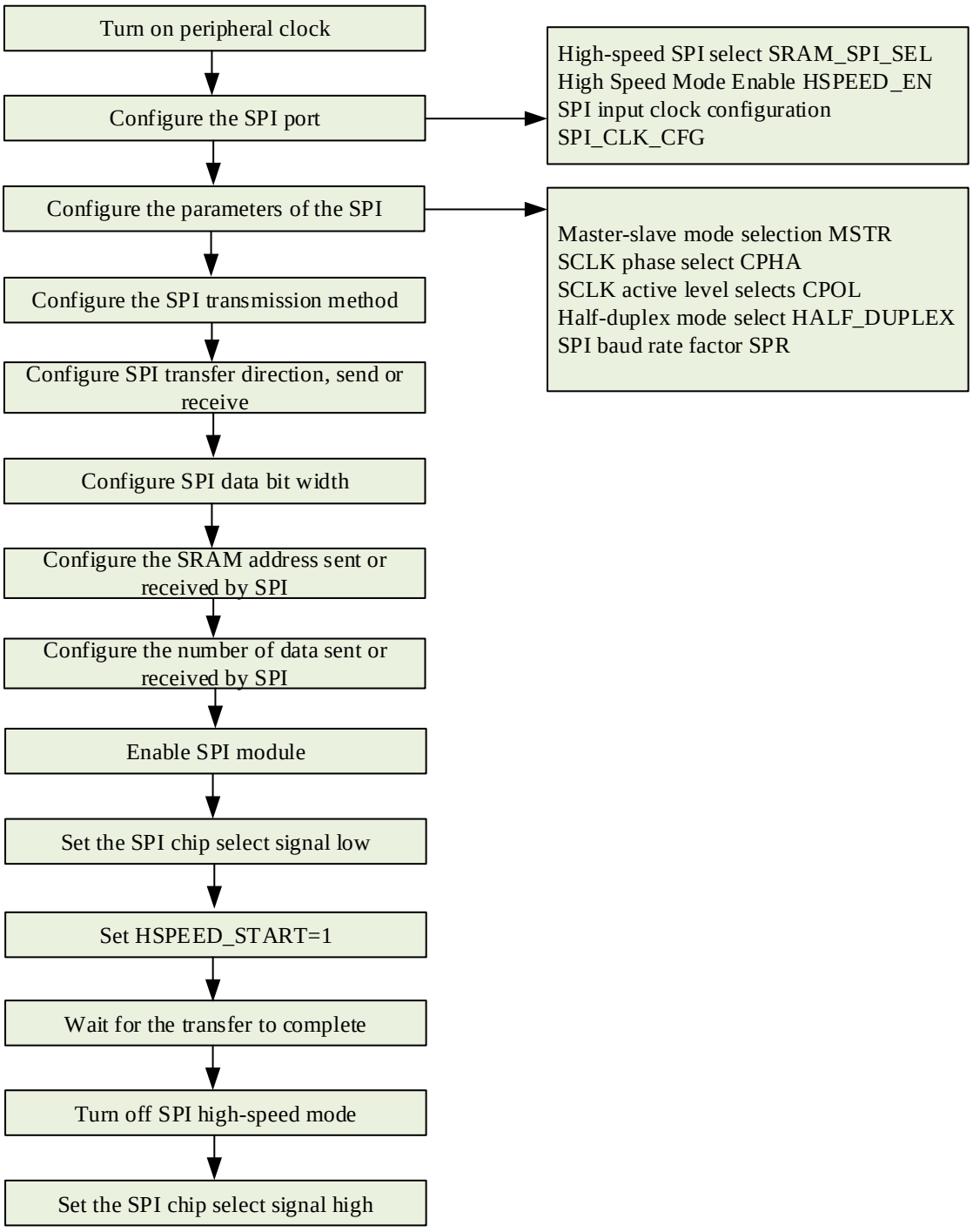
1. Configure CPOL and CPHA when chip select is high, otherwise SCLK has glitches (master, slave).
2. In the high-speed mode, hspeed_start will be automatically pulled low after the work is completed. At this time, the master cannot send SCLK anymore, otherwise an indeterminate state will occur.
3. In slave mode, after the chip select is pulled low, SPI_EN cannot be turned off. Otherwise, when the SPI EN is reopened and the chip select becomes low again, the internally generated SCLK will have a glitch. That is, while SPI is selected, SPI_EN cannot be turned off.
4. In the slave mode, if the chip select is always 0, if you need to switch CPOL/CPHA/LSBFE midway, the slave can only switch after the master raises the chip select.
5. In the high-speed mode, if an odd number of data is sent in each frame, the chip select signal needs to be pulled up once between each frame.

12.4. SPI configuration process

12.4.1. Common way configuration process



12.4.2. High speed mode configuration flow



13 Universal asynchronous receiver transmitter (UART)

13.1. Features

There are 5 UART modules in the BF7807AMXX series. Features of UART interface in UART system:

- Support full-duplex, half-duplex serial communication
- With independent double-buffered receiver, single-buffered transmitter
- Programmable baud rate (13-bit analog-to-digital frequency divider)
- Interrupt-driven or polling operation:
  - Delivery completed
  - Receive full
  - Receive overflow, parity check error, frame error
- Support hardware parity verification generation and inspection
- Programmable 8-bit or 9-bit character length
- Choose 1 or 2 STOP bit
- Support multi-processor mode
- Optional digital filter for receiving port
- Support TXD/RXD independent enable, and support pin interchange
- Support wake-up in standby mode

13.2. Function description

13.2.1. Baud rate generation

Baud rate generation modulus Baud_Mod = UART_BD [12:0]}.

Baud rate calculation formula: When Baud_Mod=0, the baud rate clock is not generated, when Baud_Mod>1, the baud rate = BUSCLK/ (16xBaud_Mod). BUSCLK uses the frequency division clock of the system clock source and is fixed at 48M.

Every time the baud rate register is configured, the internal counter will be cleared to regenerate the baud rate signal.

Communication requires the transmitter and receiver to use the same baud rate.

The allowable baud rate deviation range for communication: 8/(11*16)=4.5%.

13.2.2. Transmitter function

The transmitter is enabled by setting the TE bit in the UARTx_CON1 register. The entire sending process must be carried out when the module is enabled.

Sending data flow: Start sending by writing the UARTx_BUF value, set the sending interrupt after sending the stop bit, and clear the interrupt flag by software, and wait for the next write. The

idle state of the transmitter output pin (TXD) defaults to a logic high state.

By writing data into the data register (UARTx_BUF), the data will be directly saved to the sending data buffer and the sending process will be started. In the subsequent complete sending process, the data buffer is locked, and the configuration write data register is invalid until the sending is completed after the stop bit, the transmission interrupt flag is set, and UARTx_BUF is written again to restart a new transmission.

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

Both sending and receiving are in little-endian mode (LSB first).

13.2.3. Receiver function

The receiver is enabled by setting the RE bit in the UARTx_CON1 register. The entire receiving process must be carried out when the module is enabled.

Receiving data flow: When the receiving enable is valid, the data is received at any time, the receiving interrupt is set after the stop bit is received, and the software clears the interrupt flag.

The currently received data will have a detection mechanism, which can detect three types of errors: receiving overflow, frame error, and parity error, all of which require software to clear the flag. It is recommended that after detecting the receiving interrupt, read the status flag, read the data BUF, and finally clear all the received data status flags (UARTx_STATE[3:0]).

The data character consists of a logic 0 start bit, 8 (or 9) data bits (LSB first) and a logic 1 stop bit (1bit). After receiving the stop bit into the receive shifter, if the receive data register is not full (RXFUF = 0), the data characters are transferred to the receive data register, and the receive data register is full status flag (RXFUF = 1) is set. If you have set the receive data register full (RXFUF) at this time, set the receive overflow flag (RXOVF), and the new data will be lost. Because the receiver is double-buffered, the program has a full character time for reading after setting the receive data register full (RXFUF) and before reading the data in the receive data buffer to avoid receiver overflow. When the program detects that the receive data register is full (RXFUF = 1), it obtains data from the receive data register by reading UART_BUF.

13.2.4. Receiver sampling method

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

The receiver then samples at each bit time of RT8, RT9 and RT10, including the start bit and stop bit, to determine the logic level of the bit. The logic level is the logic level of the vast majority of samples taken during the bit time. When the falling edge is positioned, the logic level is 0 to

ensure that this is the real start bit, not noise. If at least two of these three samples are 0, the receiver assumes that it is synchronized with the receiver character and starts Shift receives 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 keeps looking for a falling edge. If an edge is detected, the sample clock resynchronizes the bit time. In this way, when noise or baud rate is not matched, the reliability of the receiver can be improved.

13.2.5. 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 function of this mechanism is to use hardware detection to eliminate the software overhead of processing unimportant information characters. Allows the receiver to ignore characters in messages used for different receivers.

In this application system, all receivers estimate the address character of each message (the 9th bit = 1). Once it is determined that the information is intended for different receivers, subsequent data characters (the 9th bit = 0) will not be received.

Configuration process: Configure receiving enable, configure multi-processor mode, receive address data (the 9th bit = 1), receive and generate an interrupt, the application confirms whether the address matches, and if it matches, the configuration closes the multi-processor mode, and all subsequent data (the 9th bit = 0) can be received and interrupted, until the next time the address data is received, and the address does not match, the multi-processor mode is turned on, and all subsequent data will not be received until the next address data, and loop in turn application.

13.2.6. Standby mode wake-up function

In idle mode 0, the UART clock is turned on, the module supports normal reception, and the receive interrupt can interrupt the wake-up core work, provided that the receive interrupt configuration is enabled.

In idle mode 1, the UART clock is turned off, the module does not support normal reception, but supports wake-up interrupt wake-up core work, provided that wake-up enable, receive enable and module enable are configured. When the receiving port inputs a low level, the wake-up process starts, and the UART interrupt processing function (wake-up status flag bit) is executed after the system wakes up.

13.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: UART0: 0x5003_0000; UART1: 0x5003_0100; UART2: 0x5003_0200;
UART3: 0x5003_0300; UART4: 0x5003_0400; **x=0/1/2/3/4**

Address offset	Register	Description
0x00	UARTx_BD	Baud rate control register
0x04	UARTx_CON1	Mode control register 1
0x08	UARTx_CON2	Mode control register 2
0x0C	UARTx_STATE	Status flag register
0x10	UARTx_BUF	Data register
0x14	UARTx_STATE_CLR	Status flag clear register
0x18	UARTx_SLEEP	Wake up control register

Base address: 0x500A 0000

Address offset	Register	Description
0x40	UARTx_IO_SEL	UARTx IO port control register

13.3.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

7	6	5	4	3	2	1	0
UART4_CLKEN	UART3_CLKEN	UART2_CLKEN	UART1_CLKEN	UART0_CLKEN	SPI1_CLKEN	SPI0_CLKEN	Reserved
RW	RW	RW	RW	RW	RW	RW	

22	GPIO_CLKEN	GPIO module operation enable 1: Work 0: Off, the default is 0
7	UART4_CLKEN	UART4 module operation enable 1: Work 0: Off, the default is 0
6	UART3_CLKEN	UART3 module operation enable

		1: Work 0: Off, the default is 0
5	UART2_CLKEN	UART2 module operation enable 1: Work 0: Off, the default is 0
4	UART1_CLKEN	UART1 module operation enable 1: Work 0: Off, the default is 0
3	UART0_CLKEN	UART0 module operation enable 1: Work 0: Off, the default is 0

13.3.2. Baud rate control register (UARTx_BD)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			BD[12:0]												
			RW												

31:13	-	Reserved
12:0	BD[12:0]	Baud rate modulus divisor register

13.3.3. Mode control register 1 (UARTx_CON1)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Res.	UART_EN	TE	RE	MULTI_M	STOP	DATA_M	PCE	PS
	RW	RW	RW	RW	RW	RW	RW	RW

31:8	-	Reserved
7	UART_EN	Module enable

		1: Module enable 0: The module is off
6	TE	Transmitter enable 1: Transmitter is enabled 0: Transmitter is off
5	RE	Receiver enable 1: The receiver is turned on 0: The receiver is off
4	MULTI_M	Multi-processor communication mode 1: Mode enable 0: Mode disabled
3	STOP	STOP bit width selection 1: 2 bits 0: 1 bit
2	DATA_M	Data mode selection 1: 9-bit mode 0: 8-bit mode
1	PCE	1: Parity check enabled 0: Parity check is disabled In 8-bit mode: parity check enable is invalid In 9-bit mode: When parity is enabled, the ninth bit is the calculated parity bit; When parity check is not enabled: the ninth bit is T8 written in
0	PS	Parity selection 1: Odd parity 0: Even parity

13.3.4. Mode control register 2 (UARTx_CON2)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	TXIEN	RXIEN	DFILEN
	RW	RW	RW

31:3	-	Reserved
2	TXIEN	Transmit interrupt enable 1: Interrupt enable

		0: Interrupt disabled (used in polling mode)
1	RXIEN	Receive interrupt enable 1: Interrupt enable 0: Interrupt disabled (used in polling mode)
0	DFILEN	Filtering is enabled, the input signal of the receiving port can be configured to select the filter function 1: Enable 0: Disable

13.3.5. Status flag register (UARTx_STATE)

Address offset: 0x0C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

31:7	6	5	4	3	2	1	0
Reserved	R8	T8	TXEMF	RXFUF	RXOVF	FEF	PEF
	R	R/RW	R	R	R	R	R

31:7	-	Reserved
6	R8	The 9th data of the receiver, read only
5	T8	The 9th data of the transmitter, cannot be written in 9-bit mode and parity check is enabled, and can be written and read in other cases
4	TXEMF	Send interrupt flag 1: The sending buffer is empty 0: The sending buffer is full, read only
3	RXFUF	Receive interrupt flag 1: The receive buffer is full 0: The receive buffer is empty, read-only
2	RXOVF	Receive overflow flag 1: Receive overflow (new data is lost) 0: No overflow, read only
1	FEF	Frame error flag 1: Frame error detected 0: No frame error detected, read only
0	PEF	Parity error flag 1: Receiver parity error 0: Receiver parity check is correct, read only

13.3.6. Data register (UARTx_BUF)

Address offset: 0x10

Reset value: 0x0000 00FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								BUF[7:0]							
								RW							

31:8	-	Reserved
7:0	BUF[7:0]	Data register Read returns the contents of the read-only receive data buffer, write into the write-only transmit data buffer

13.3.7. Status flag clear register (UARTx_STATE_CLR)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	TXEMFCL	RXFUFCL	RXOVFCL	FEFCL	PEFCL
	W	W	W	W	W

31:5	-	Reserved
4	TXEMFCL	Send interrupt flag to clear, write 1 to clear
3	RXFUFCL	Receive interrupt flag is cleared, write 1 to clear
2	RXOVFCL	Receive overflow flag to clear, write 1 to clear
1	FEFCL	Frame error flag is cleared, write 1 to clear
0	PEFCL	Clear parity error flag, write 1 to clear

13.3.8. Wake-up control register (UARTx_SLEEP)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	WASFCL	WASF	WAKEN
	W	R	RW

31:3	-	Reserved
2	WASFCL	Idle mode 1 wake-up state clear flag Write 1 to clear zero, write only
1	WASF	Idle mode 1 wake-up status flag, read only
0	WAEN	Idle mode 1 wake-up enable 1: Enable 0: Disable

13.3.9. UART port control register (UARTx_IO_SEL)

Address offset: 0x40

Reset value: 0x000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:14	13:12	11:10	9:8	7:6	5	4	3	2	1	0
Res.	UART4 _SEL	UART3 _SEL	UART2 _SEL	UART1 _SEL	UART0 _SEL	UART4 _ EC	UART3 _ EC	UART2 _ EC	UART1 _ EC	UART0 _ EC
	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:14	-	Reserved
13:12	UART4 _SEL	UART4 port selection enable 00: PA6/PA7 port select UART4 function 01: PB6/PB7 port select UART4 function 10: PB5/PB6 port select UART4 function 11: PD8/PD9 port select UART4 function
11:10	UART3 _SEL	UART3 port selection enable 00: PA0/PA1 port select UART3 function 01: PA14/PA15 port select UART3 function 10: PC5/PC6 port select UART3 function
9:8	UART2_SEL	UART2 port selection enable 00: PD0/PD1 port select UART2 function 01: PD2/PD3 port select UART2 function 10: PD5/PD6 port select UART2 function 11: PD7/PD5 port select UART2 function
7:6	UART1_SEL	UART1 port selection enable 00: PB14/PB15 port select UART1 function

		01: PC3/PC4 port select UART1 function 10: PC6/PC7 port select UART1 function
5	UART0_SEL	UART0 port selection enable 0: PB12/PB13 port select UART0 function 1: PA12/PA13 port select UART0 function
4	UART4_ EC	UART4 port TXD/RXD pin exchange 1: Pin exchange 0: The pins are not exchange
3	UART3_ EC	UART3 port TXD/RXD pin exchange 1: Pin exchange 0: The pins are not exchange
2	UART2_ EC	UART2 port TXD/RXD pin exchange 1: Pin exchange 0: The pins are not exchange
1	UART1_ EC	UART1 port TXD/RXD pin exchange 1: Pin exchange 0: The pins are not exchange
0	UART0_EC	UART0 port TXD/RXD pin exchange 1: Pin exchange 0: The pins are not exchange

13.4. UART configuration process

1. Turn on the peripheral clock;
2. Configure the UART port control register (UARTx_IO_SEL) to select the corresponding IO port;
3. Configure module enable, send enable, receive enable, mode selection: UARTx_CON1;
4. Configure the baud rate and turn on the interrupt enable: UARTx_BD, UARTx_CON2;
5. Write UARTx_BUF to start sending data. After detecting the sending interrupt, clear the sending interrupt flag TXEMF. Once the sending process is completed, wait for the next write to UARTx_BUF to start the sending process (it is not allowed to configure the next data during the sending process, including UARTx_BUF and T8);
6. Detect the receiving interrupt, first read the receiving status UARTx_STATE, then read R8 and UARTx_BUF, and finally clear the receiving status flag, once the receiving process is completed, wait for the next receiving interrupt;
7. If the configuration interrupt is not enabled, the program polls to perform the UART function, and also reads the status flag first, then reads R8 and UARTx_BUF, and finally clears the receiving status flag.

Note:

1. **When clearing the interrupt flag bit, note that only the clear bit (UARTx_STATE_CLR) corresponding to the flag bit to be cleared needs to be set to 1, and other bits need to be written to 0, which can prevent false clearing;**
2. **After turning off the module enable, all states are cleared, and you need to wait at least 0.5us before turning on the module enable;**
3. **It is not recommended to modify the configuration during communication: UARTx_BD and UARTx_CON1[3:0] (STOP /DATA_M/PCE/PS), otherwise the current frame communication will be invalid.**

14 Pulse width modulation module (PWM)

The BF7807AMXX contains 5 independent 16-bit PWM modules. The PWM module clock is divided by 1/2/4/8/16/32/64/128 of PLL48MHz.

14.1. PWM0/1

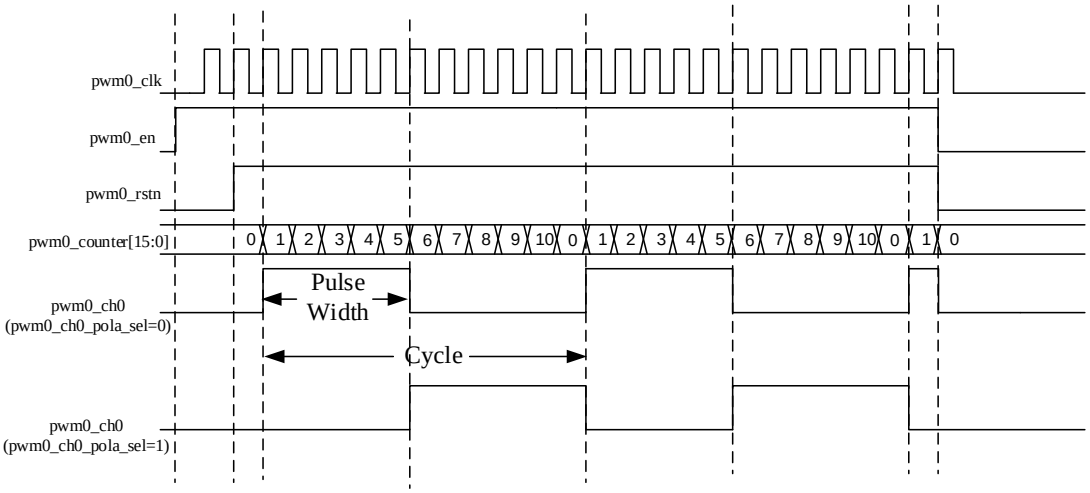
14.1.1. PWM0/1 overview

The period and pulse width of PWM0/1 pulse width modulation module can be configured through registers. Once the configuration value is updated after counting starts, the register value will be updated when the counter changes from (PWMx_MOD -1) to (PWMx_MOD), that is, the period and duty cycle are updated after a complete period, with buffer processing in the middle.

The PWM0/1 counter starts counting up from 0x0000. When PWMx_DUTY_CHx is counted, the output is reversed. This period of time is the pulse width. Continue counting until the count overflows when PWMx_MOD+1 is counted. If PWMx_CH_POLA_SEL=0, the PWM signal enters the low state when the output is inverted, and the PWM signal enters the high state when the count overflows. If PWMx_CH_POLA_SEL=1, the PWM signal enters the high state when the output is inverted, and the PWM signal enters the low state when the count overflows.

When the channel count register (PWMx_DUTY_CHx) is set to 0x0000, the duty cycle is 0%. When the channel count value register (PWMx_DUTY_CHx) is set to a value greater than the value set by the period configuration register (PWMx_MOD), a 100% duty cycle can be achieved. The counter is automatically reloaded and will not stop by itself. It will not stop until the register ENABLE = 0 and the counter is cleared.

When the count overflows, the interrupt flag will be set, and if the interrupt enable is configured, the interrupt response will be triggered.



PWM0 output waveform
(PWM0_DUTY_CH0=5, PWM0_MOD=10, duty_cycle=5/11)

14.1.2. PWM0/1 features

- Share a 16-bit counter, auto-reload, the counting clock is configured by register CLK_SEL
- Support idle mode 0 wake-up
- PWM0 supports 5 channels: PWM0A (or PWM0A1)/PWM0B/PWM0C/PWM0D/PWM0E. PWM0A, PWM0A1 support synchronous output or separate output
  - Each channel is individually enabled
  - Share counter
  - The duty cycle of each channel can be configured
  - The polarity of each channel can be configured
- PWM1 supports 5 channels: PWM1A (or PWM1A1)/PWM1B/PWM1C/PWM1D/PWM1E. PWM1A, PWM1A1 support synchronous output or separate output
  - Each channel is individually enabled
  - Share counter
  - The duty cycle of each channel can be configured
  - The polarity of each channel can be configured
- Configure output PWM waveform
  - Output period = (PWMx_MOD + 1) * T
  - Output pulse width = PWMx_DUTY_CHx* T
  - Duty cycle = PWMx_DUTY_CHx/ (PWMx_MOD + 1)
- When the configuration does not output PWM waveform, it is only used as a timer function
  - Timing period = (PWMx_MOD + 1) * T
- Support common frequency: 38kHz (infrared application)

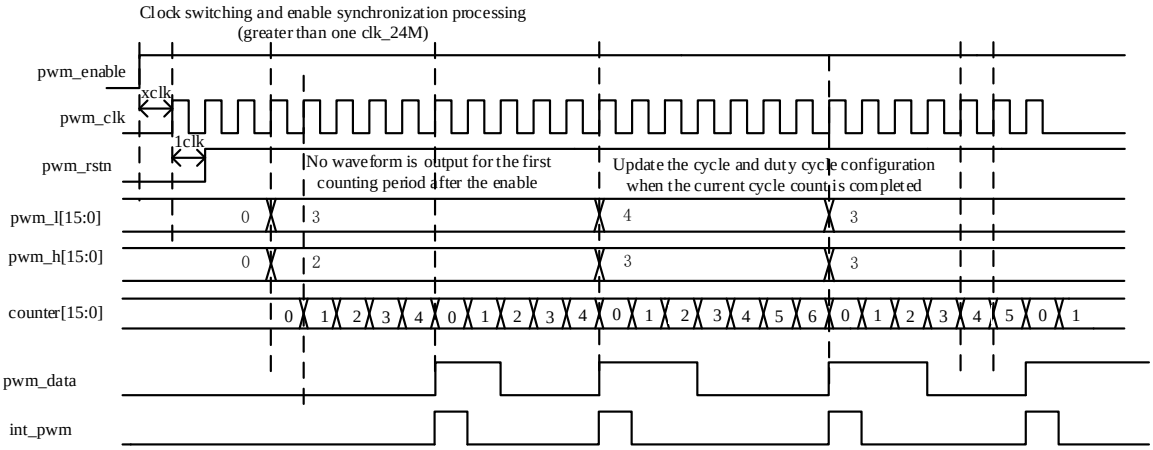
14.2. PWM2/3/4

14.2.1. PWM2/3/4 overview

The high and low time of PWM2/3/4 pulse width modulation module can be configured through registers. Once the configuration value is updated after the count is started, it will be updated again after a complete cycle, with buffering processing in the middle.

When the count overflows, the interrupt flag will be set, and if the interrupt enable is configured, the interrupt response will be triggered.

Note: When the duty cycle is 0 or 100%, the counter will not work and no interrupt will be generated.



14.2.2. PWM2/3/4 features

- 16-bit counter, automatic reload, counting clock configured by register register CLK_SEL
- Support idle mode 0 wake-up
- PWM2 supports 1 channel, PWM2A or PWM2A1. PWM2A, PWM2A1 support synchronous output or separate output
- PWM3 supports 1 channel, PWM3A or PWM3A1. PWM3A, PWM3A1 support synchronous output or separate output
- PWM4 supports 1 channel, PWM4A
- Configure output PWM waveform
  - Output period = (PWM_ H + PWM_L)*T
  - Duty cycle = PWM_H/ (PWM_L + PWM_H)
- When the configuration does not output PWM waveform, it is only used as a timer function
  - Timing period = (PWM_ H + PWM_L)*T (Neither PWM_ H nor PWM_L can be zero)
- Support common frequency: 38kHz (infrared application)

14.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: PWM0: 0x5005_0500; PWM1: 0x5005_0600;
PWM2: 0x5005_0700; PWM3: 0x5005_0800; PWM4: 0x5005_0900

PWM0/1 registers have the same function, x=0/1

Address offset	Register	Description
0x00	PWMx_MOD	PWM0/1 counting period configuration register
0x04	PWMx_CFG	PWM0/1 control register
0x08	PWMx_INT_CFG	PWM0/1 interrupt control register
0x0C	PWMx_DUTY_CH0	PWM0/1 channel 0 count value configuration register
0x10	PWMx_DUTY_CH1	PWM0/1 channel 1 count value configuration register
0x14	PWMx_DUTY_CH2	PWM0/1 channel 2 count value configuration register
0x18	PWMx_DUTY_CH3	PWM0/1 channel 3 count value configuration register
0x1C	PWMx_DUTY_CH4	PWM0/1 channel 4 count value configuration register
0x20	PWMx_CH_EN	PWM0/1 channel enable register
0x24	PWMx_CH_POLA_SEL	PWM0/1 polarity selection register
0x28	PWMx_CH_CMOD	PWM0/1 duty cycle mode selection register

PWM 2/3/4 registers have the same function, x= 2/3/4

Address offset	Register	Description
0x00	PWMx_TIME	PWM 2/3/4 level control register
0x04	PWMx_CFG	PWM 2/3/4 control register
0x08	PWMx_INT_CFG	PWM 2/3/4 interrupt control register

Base address: 0x500A 0000

Address offset	Register	Description
0x3c	PWM_OUT_EN	PWM output enable register

14.3.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16

LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8
TIMER1_CLKEN	TIMER0_CLKEN	PWM4_CLKEN	PWM3_CLKEN	PWM2_CLKEN	PWM1_CLKEN	PWM0_CLKEN	IIC_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

22	GPIO_CLKEN	GPIO module operation enable 1: Work 0: Off, the default is 0
13	PWM4_CLKEN	PWM4 module operation enable 1: Work 0: Off, the default is 0
12	PWM3_CLKEN	PWM3 module operation enable 1: Work 0: Off, the default is 0
11	PWM2_CLKEN	PWM2 module operation enable 1: Work 0: Off, the default is 0
10	PWM1_CLKEN	PWM1 module operation enable 1: Work 0: Off, the default is 0
9	PWM0_CLKEN	PWM0 module operation enable 1: Work 0: Off, the default is 0

14.3.2. PWM0 registers

Base address: 0x5005_0500

14.3.2.1. PWM0 counting period configuration register (PWM0_MOD)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOD[15:0]															

RW

31:16	-	Reserved
15:0	MOD[15:0]	PWM0 counting period configuration register Configure the PWM0 output period

14.3.2.2. PWM0 control register (PWM0_CFG)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CLK_SEL			ENABLE
												RW			RW

	-	Reserved
3:1	CLK_SEL	PWM0 module clock selection register 000: 48MHz 001: 24MHz 010: 12MHz 011: 6MHz 100: 3MHz 101: 1.5MHz 110: 0.75MHz 111: 0.375MHz
0	ENABLE	PWM0 module enable 1: PWM0 function is enabled 0: PWM0 function disabled

14.3.2.3. PWM0 interrupt control register (PWM0_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

15:3	-	Reserved
2	INT_CLR	Count overflow interrupt status flag clear register, write only Ways to clear the interrupt flag: a) System reset b) Write 1 to clear INT_FLAG c) Turn off PWM0 enable
1	INT_FLAG	Count overflow interrupt status flag register 1: End of counting period 0: Count incomplete, read only
0	INT_EN	Count overflow interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

14.3.2.4. PWM0 channel 0 count value configuration register (PWM0_DUTY_CH0)

Address offset: 0x0C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH0[15:0]															
RW															

31:16	-	Reserved
15:0	DCH0[15:0]	PWM0 channel 0 count value configuration register Configure PWM0A (or PWM0A1) output duty cycle

14.3.2.5. PWM0 channel 1 count value configuration register (PWM0_DUTY_CH1)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH1[15:0]															
RW															

31:16	-	Reserved

15:0	DCH1[15:0]	PWM0 channel 1 count value configuration register Configure PWM0B output duty cycle

14.3.2.6. PWM0 channel 2 count value configuration register (PWM0_DUTY_CH2)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH2[15:0]															
RW															

31:16	-	Reserved
15:0	DCH2[15:0]	PWM0 channel 2 count value configuration register Configure PWM0C output duty cycle

14.3.2.7. PWM0 channel 3 count value configuration register (PWM0_DUTY_CH3)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH3[15:0]															
RW															

31:16	-	Reserved
15:0	DCH3[15:0]	PWM0 channel 3 count value configuration register Configure PWM0D output duty cycle

14.3.2.8. PWM0 channel 4 count value configuration register (PWM0_DUTY_CH4)

Address offset: 0x1C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

DCH4[15:0]
RW

31:16	-	Reserved
15:0	DCH4[15:0]	PWM0 channel 4 count value configuration register Configure PWM0E output duty cycle

14.3.2.9. PWM0 channel enable register (PWM0_CH_EN)

Address offset: 0x20

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	CH4_EN	CH3_EN	CH2_EN	CH1_EN	CH0_EN
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	CH4_EN	PWM0E output enable 1: Enable; 0: Disable
3	CH3_EN	PWM0D output enable 1: Enable; 0: Disable
2	CH2_EN	PWM0C output enable 1: Enable; 0: Disable
1	CH1_EN	PWM0B output enable 1: Enable; 0: Disable
0	CH0_EN	PWM0A (or PWM0A1) output enable 1: Enable; 0: Disable

14.3.2.10. PWM0 polarity selection register (PWM0_CH_POLA_SEL)

Address offset: 0x24

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	CH4_PS	CH3_PS	CH2_PSL	CH1_PS	CH0_PS
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	CH4_PS	PWM0E polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
3	CH3_PS	PWM0D polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
2	CH2_PS	PWM0C polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
1	CH1_PS	PWM0B polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
0	CH0_PS	PWM0A (or PWM0A1) polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high

14.3.2.11. PWM0 duty cycle mode selection register (PWM0_CH_CMOD)

Address offset: 0x28

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:4	3	2	1	0
Reserved	CH4_CMOD	CH3_CMOD	CH2_CMOD	CH1_CMOD
	RW	RW	RW	RW

31:4	-	Reserved
3	CH4_CMOD	PWM0E duty cycle mode selection register 1: Select PWM0A (or PWM0A1) duty cycle 0: Select own channel duty cycle
2	CH3_CMOD	PWM0D duty cycle mode selection register 1: Select PWM0A (or PWM0A1) duty cycle 0: Select own channel duty cycle
1	CH2_CMOD	PWM0C duty cycle mode selection register 1: Select PWM0A (or PWM0A1) duty cycle 0: Select own channel duty cycle
0	CH1_CMOD	PWM0B duty cycle mode selection register 1: Select PWM0A (or PWM0A1) duty cycle

		0: Select own channel duty cycle

14.3.3. PWM1 registers

Base address: PWM1: 0x5005_0600

14.3.3.1. PWM1 counting period configuration register (PWM1_MOD)

Address offset: 0x00
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOD [15:0]															
RW															

31:16	-	Reserved
15:0	MOD [15:0]	PWM1 counting period configuration register Configure the PWM1 output period

14.3.3.2. PWM1 control register (PWM1_CFG)

Address offset: 0x04
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CLK_SEL			ENABLE
												RW			RW

	-	Reserved
3:1	CLK_SEL	PWM1 module clock selection register 000: 48MHz 001: 24MHz 010: 12MHz 011: 6MHz 100: 3MHz 101: 1.5MHz 110: 0.75MHz

		111: 0.375MHz
0	ENABLE	PWM1 module enable 1: PWM1 function is enabled 0: PWM1 function disabled

14.3.3.3. PWM1 interrupt control register (PWM1_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

15:3	-	Reserved
2	INT_CLR	Count overflow interrupt status flag clear register, write only Ways to clear the interrupt flag: a) System reset b) Write 1 to clear INT_FLAG c) Turn off PWM1 enable
1	INT_FLAG	Count overflow interrupt status flag register 1: End of counting period 0: Count incomplete, read only
0	INT_EN	Count overflow interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

14.3.3.4. PWM1 channel 0 count value configuration register (PWM1_DUTY_CH0)

Address offset: 0x0C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH0[15:0]															
RW															

31:16	-	Reserved

15:0	DCH0[15:0]	PWM1 channel 0 count value configuration register Configure PWM1A (or PWM1A1) output duty cycle

14.3.3.5. PWM1 channel 1 count value configuration register (PWM1_DUTY_CH1)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH1[15:0]															
RW															

31:16	-	Reserved
15:0	DH1[15:0]	PWM1 channel 1 count value configuration register Configure PWM1B output duty cycle

14.3.3.6. PWM1 channel 2 count value configuration register (PWM1_DUTY_CH2)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DH2[15:0]															
RW															

31:16	-	Reserved
15:0	DH2[15:0]	PWM1 channel 2 count value configuration register Configure PWM1C output duty cycle

14.3.3.7. PWM1 channel 3 count value configuration register (PWM1_DUTY_CH3)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

DCH3[15:0]
RW

31:16	-	Reserved
15:0	DCH3[15:0]	PWM1 channel 3 count value configuration register Configure PWM1D output duty cycle

14.3.3.8. PWM1 channel 4 count value configuration register (PWM1_DUTY_CH4)

Address offset: 0x1C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCH4[15:0]															
RW															

31:16	-	Reserved
15:0	DCH4[15:0]	PWM1 channel 4 count value configuration register Configure PWM1E output duty cycle

14.3.3.9. PWM1 channel enable register (PWM1_CH_EN)

Address offset: 0x20

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	CH4_EN	CH3_EN	CH2_EN	CH1_EN	CH0_EN
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	CH4_EN	PWM1E output enable 1: Enable; 0: Disable
3	CH3_EN	PWM1D output enable 1: Enable; 0: Disable
2	CH2_EN	PWM1C output enable 1: Enable; 0: Disable
1	CH1_EN	PWM1B output enable

		1: Enable; 0: Disable
0	CH0_EN	PWM1A (or PWM1A1) output enable 1: Enable; 0: Disable

14.3.3.10. PWM1 polarity selection register (PWM1_CH_POLA_SEL)

Address offset: 0x24

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	CH4_PS	CH3_PS	CH2_PS	CH1_PS	CH0_PS
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	CH4_PS	PWM1E polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
3	CH3_PS	PWM1D polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
2	CH2_PS	PWM1C polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
1	CH1_PS	PWM1B polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high
0	CH0_PS	PWM1A (or PWM1A1) polarity selection 1: Count value overflow makes the output low; 0: Count value overflow makes the output high

14.3.3.11. PWM1 duty cycle mode selection register (PWM1_CH_CMOD)

Address offset: 0x28

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:4	3	2	1	0
Reserved	CH4_CMOD	CH3_CMOD	CH2_CMOD	CH1_CMOD

	RW	RW	RW	RW

31:4	-	Reserved
3	CH4_CMOD	PWM1E duty cycle mode selection register 1: Select PWM1A (or PWM1A1) duty cycle 0: Select own channel duty cycle
2	CH3_CMOD	PWM1D duty cycle mode selection register 1: Select PWM1A (or PWM1A1) duty cycle 0: Select own channel duty cycle
1	CH2_CMOD	PWM1C duty cycle mode selection register 1: Select PWM1A (or PWM1A1) duty cycle 0: Select own channel duty cycle
0	CH1_CMOD	PWM1B duty cycle mode selection register 1: Select PWM1A (or PWM1A1) duty cycle 0: Select own channel duty cycle

14.3.4. PWM2 registers

14.3.4.1. PWM2 level control register (PWM2_TIME)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PWM2_H															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWM2_L															
RW															

31:16	PWM2_H	PWM2 high level control register
15:0	PWM2_L	PWM2 low level control register

14.3.4.2. PWM2 control register (PWM2_CFG)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

Reserved	CLK_SEL	ENABLE
	RW	RW

31:4	-	Reserved
3:1	CLK_SEL	PWM2 module clock selection register 000: 48MHz 001: 24MHz 010: 12MHz 011: 6MHz 100: 3MHz 101: 1.5MHz 110: 0.75MHz 111: 0.375MHz
0	ENABLE	PWM2 module enable 1: PWM2 function is enabled 0: PWM2 function disabled

14.3.4.3. PWM2 interrupt control register (PWM2_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

15:3	-	Reserved
2	INT_CLR	Interrupt status flag clear register, write only Ways to clear the interrupt flag: a) System reset b) Write 1 to clear INT_FLAG c) Turn off PWM2 enable
1	INT_FLAG	Interrupt status flag register, read only 1: End of counting period 0: Counting is not completed
0	INT_EN	Interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

14.3.5. PWM3 registers

14.3.5.1. PWM3 level control register (PWM3_TIME)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PWM3_H															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWM3_L															
RW															

31:16	PWM3_H	PWM3 high level control register
15:0	PWM3_L	PWM3 low level control register

14.3.5.2. PWM3 control register (PWM3_CFG)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CLK_SEL			ENABLE
												RW			RW

31:4	-	Reserved
3:1	CLK_SEL	PWM3 module clock selection register 000: 48MHz 001: 24MHz 010: 12MHz 011: 6MHz 100: 3MHz 101: 1.5MHz 110: 0.75MHz 111: 0.375MHz
0	ENABLE	PWM3 module enable 1: PWM3 function is enabled

		0: PWM3 function disabled

14.3.5.3. PWM3 interrupt control register (PWM3_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

15:3	-	Reserved
2	INT_CLR	Interrupt status flag clear register, write only Ways to clear the interrupt flag: a) System reset b) Write 1 to clear INT_FLAG c) Turn off PWM3 enable
1	INT_FLAG	Interrupt status flag register, read only 1: End of counting period 0: Counting is not completed
0	INT_EN	Interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

14.3.6. PWM4 registers

14.3.6.1. PWM4 level control register (PWM4_TIME)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PWM4_H															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWM4_L															
RW															

31:16	PWM4_H	PWM4 high level control register

15:0	PWM4_L	PWM4 low level control register

14.3.6.2. PWM4 control register (PWM4_CFG)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CLK_SEL			ENABLE
												RW			RW

31:4	-	Reserved
3:1	CLK_SEL	PWM4 module clock selection register 000: 48MHz 001: 24MHz 010: 12MHz 011: 6MHz 100: 3MHz 101: 1.5MHz 110: 0.75MHz 111: 0.375MHz
0	ENABLE	PWM4 module enable 1: PWM4 function is enabled 0: PWM4 function disabled

14.3.6.3. PWM4 interrupt control register (PWM4_INT_CFG)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

15:3	-	Reserved
2	INT_CLR	Interrupt status flag clear register, write only Ways to clear the interrupt flag:

		a) System reset b) Write 1 to clear INT_FLAG c) Turn off PWM4 enable
1	INT_FLAG	Interrupt status flag register, read only 1: End of counting period 0: Counting is not completed
0	INT_EN	Interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

14.3.7. PWM output enable register (PWM_OUT_EN)

Address offset: 0x3C
Reset value: 0x000 0000
Different PWM module outputs of the same port have priority: PWM0 prior to PWM1

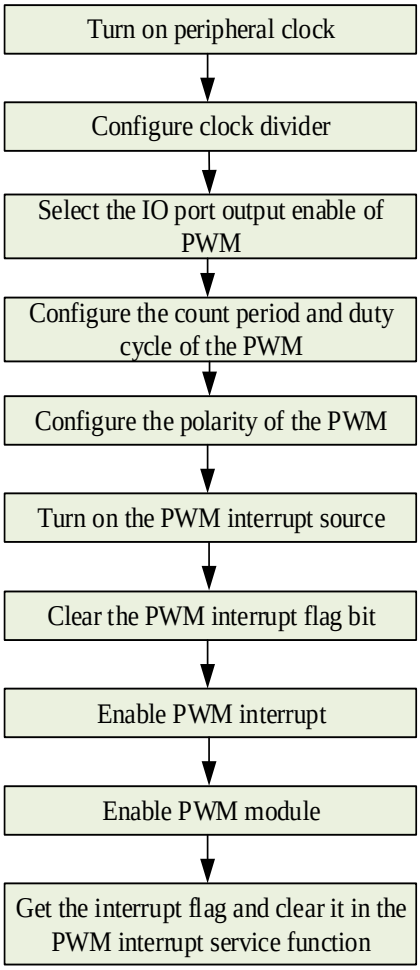
31:17	16
Reserved	PWM4A
	RW

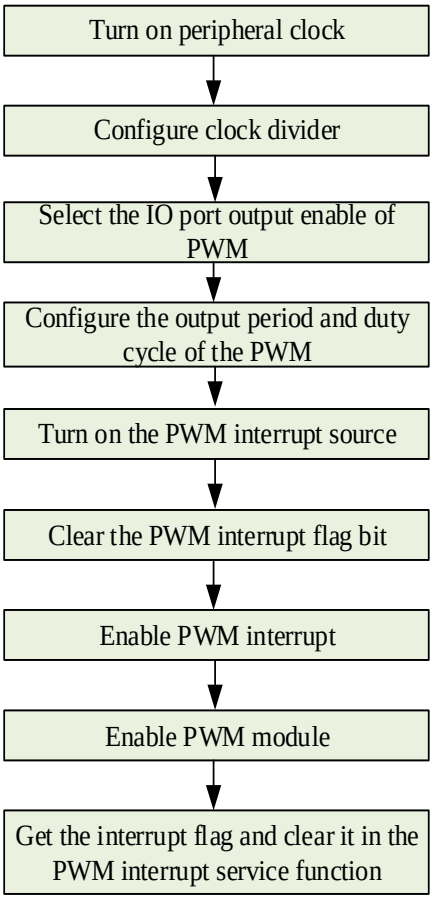
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWM3A1	PWM3A	PWM2A1	PWM2A	PWM1A1	PWM1E	PWM1D	PWM1C	PWM1B	PWM1A	PWM0A1	PWM0E	PWM0D	PWM0C	PWM0B	PWM0A
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

16	PWM4	PWM4A port output enable 1: Output, 0: No output
15:14	PWM3x[1:0]	Bit[0]: PWM3A port output enable Bit[1]: PWM3A1 port output enable The corresponding bits of PWM3 are: 1: Output, 0: No output
13:12	PWM2x[1:0]	Bit[0]: PWM2A port output enable Bit[1]: PWM2A1 port output enable The corresponding bits of PWM2 are: 1: Output, 0: No output

11:6	PWM1x[5:0]	Bit[0]: PWM1A port output enable Bit[1]: PWM1B port output enable Bit[2]: PWM1C port output enable Bit[3]: PWM1D port output enable Bit[4]: PWM1E port output enable Bit[5]: PWM1A1 port output enable The corresponding bits of PWM1 are: 1: Output, 0: No output
5:0	PWM0x[5:0]	Bit[0]: PWM0A port output enable Bit[1]: PWM0B port output enable Bit[2]: PWM0C port output enable Bit[3]: PWM0D port output enable Bit[4]: PWM0E port output enable Bit[5]: PWM0A1 port output enable The corresponding bits of PWM0 are: 1: Output, 0: No output

14.4. PWM configuration process





Left: PWM0/1 configuration flow

Right: PWM2/3/4 configuration process

Note: The recommended PWM frequency is shown in the table below

PWM clock(Hz)	Output frequency(Hz)
48M	733 ~ 480 k
24M	367 ~ 240 k
12M	184 ~ 120 k
6M	92 ~ 60 k
3M	46 ~ 30 k
1.5M	23 ~ 15 k
0.75M	12 ~ 7.5 k
0.375M	6 ~ 3.75 k

15 Touch key (CDC)

15.1. Function description

Any channel of this module can flexibly configure registers, including detection rate, detection accuracy, pull-up current value, etc. The detection is performed in the way of point scanning, that is, the software only gives one scanning channel address and the corresponding pull-up current value configuration at a time, and sends out an interrupt after scanning.

Scanning process: To start the scan, the software needs to set CDC_START = 1. After one scan, the result is stored in the data register (CDC_DATA), and then an interrupt is triggered, and CDC_START is automatically cleared. Setting CDC_START = 0 by software will stop the current scan immediately, and the relevant signals inside the module will be reset.

A series of registers are used to realize the application of various functions. The relationship between the capacitance detection correlation quantity and the SFR value is as follows:

The count value is proportional to RESO, RB resistance, and current source value, and inversely proportional to VTH_SEL. In the case of ensuring complete charging and discharging, it is proportional to the charging and discharging frequency set by PRS_DIV.

The channel touch change is proportional to RESO and RB, and inversely proportional to VTH_SEL. In the case of ensuring complete charging and discharging, the charging and discharging frequency set by PRS_DIV is proportional to the touch change amount.

The signal-to-noise ratio of touch detection is proportional to VTH_SEL, PULL_I_SEL, and inversely proportional to CDC_DS. When charging and discharging are incomplete, the charging and discharging frequency set by PRS_DIV is inversely proportional to the signal-to-noise ratio.

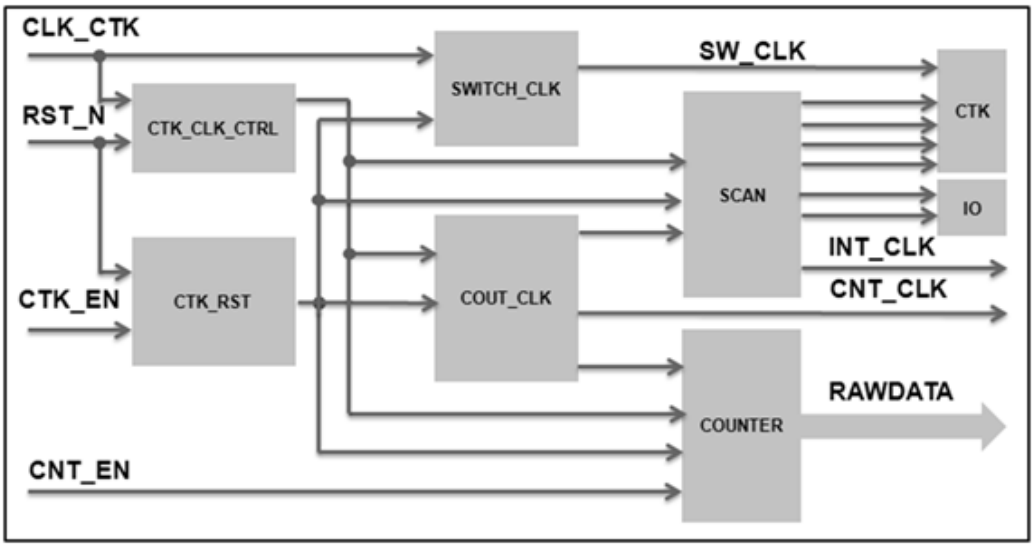
The time of single key detection is related to RESO and CDC_DS.

Note: When configuring parameters, ensure that the keys are fully charged and discharged.

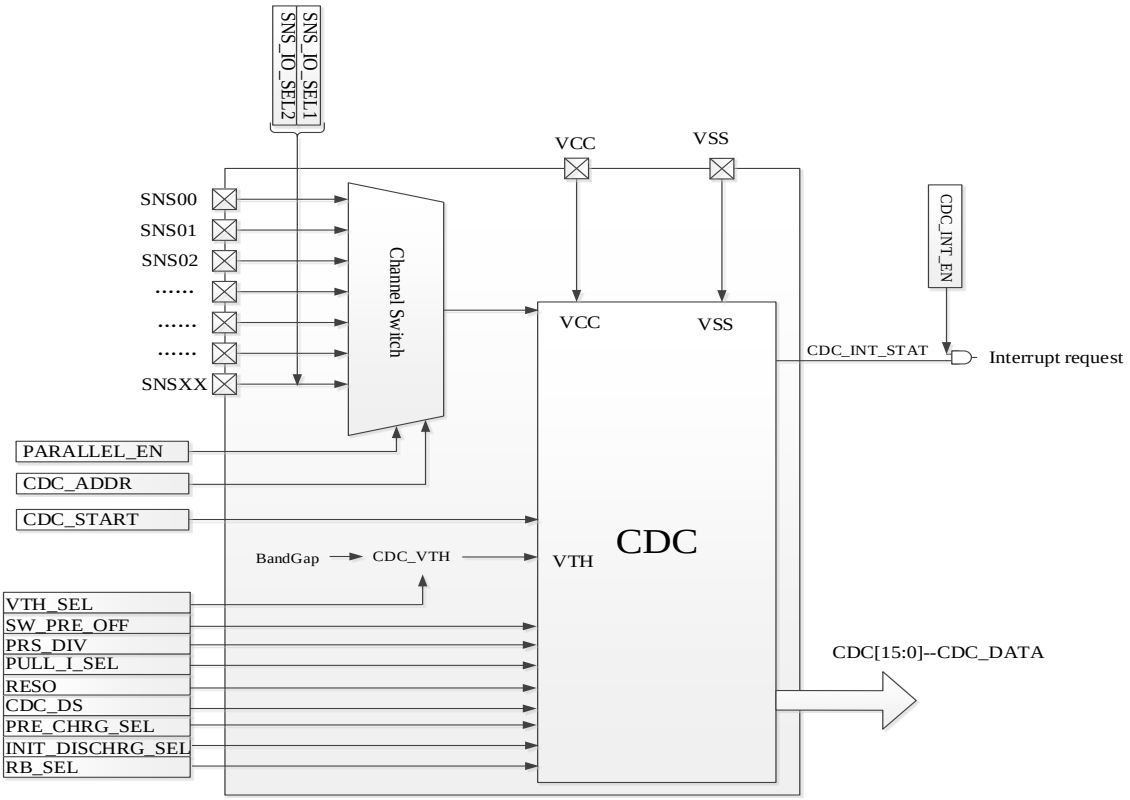
15.1.1. Features

- 3 modes of CDC charge and discharge clock are select able
  - Fixed frequency division of system clock 6M~369k
  - PRS 1.5M normal distribution
  - PRS 1.5M evenly distributed
- CDC counting clock: 24M, 12M, 6M, 4M optional
- Counting bit width 9~16 bits optional
- Only supports asynchronous scanning mode
- Support multi-channel parallel connection
- Support wake-up in idle mode 0

15.1.2. Block diagram



CDC module structure diagram



CDC structure block diagram

15.1.3. Sensitivity parameter

A set of parameters with a better signal-to-noise ratio can be obtained through the sensitivity parameter configuration, thereby improving the accuracy of key judgment.

1. **RESO:** 0~7 touch key capacitance scan resolution, counter digits: **(RESO + 9) bits,** the larger the touch key capacitance scan resolution, the greater the amount of Rawdata downward change, and the noise introduced will follow increase, and vice versa.
2. **VTH_SEL:** 0~7, the smaller the reference voltage, the greater the amount of change in Rawdata, and the noise introduced will also increase, and vice versa.
3. **CDC_DS:** Detection speed **0: 24M, 1: 12M, 2: 6M, 3: 4M,** the smaller the detection speed, the slower the rawdata sampling time, and vice versa. It is recommended that the default 24M is the fastest and the detection speed is at least 2 times the PRS clock.
4. **RB_SEL:** RB resistance selection: **2: 60k; 3: 80k; other: reserved;** the larger the resistance, the greater the amount of change in Rawdata, and the noise introduced will also increase, and vice versa.
5. **PRS_DIV:** Front-end charge and discharge clock frequency selection register:
   000000 ~ 111101: Fixed frequency: F=F48M/2/(PRS_DIV+4), that is (6M~369k)
   11 1110: Highest frequency 3M, lowest frequency 1M, center frequency 1.5M, normal distribution
   11 1111: The highest frequency is 3M, the lowest frequency is 1M, and the center frequency is 1.5M, evenly distributed
   The larger the PRS clock is, the larger the variation of Rawdata will be, and the noise introduced will also increase, and vice versa.
6. **PULL_I_SEL:** Pull-up current source
   Current source size=255.5-0.5*{PULL_I_SEL}, the smaller the current source, the smaller the count value.

Note:

1. **Rawdata is the real-time raw count value of the touch key capacitance counter.**
2. **In actual applications, it is necessary to view the data through the programming and debugging software and compare the parameters to obtain a set of parameters with a good signal-to-noise ratio.**
3. **The relationship between chip supply voltage and reference voltage: VCC-VTH>0.5V.**

15.2. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register
0x10	ANA_CFG	Analog module switch register

Base address: 0x5006_0000

Address offset	Register	Description
0x00	CDC_START	CDC scan open register
0x04	CDC_SCAN_CFG1	CDC scan configuration register 1
0x08	CDC_SCAN_CFG2	CDC scan configuration register 2
0x0C	CDC_ADDR_CFG	CDC address configuration register
0x10	CDC_INT_CFG	Interrupt configuration register
0x14	CDC_ANA_CFG	Analog register
0x18	SENSOR_IO_SEL1	SENSOR port selection enable register 1
0x1C	SENSOR_IO_SEL2	SENSOR port selection enable register 2
0x20	PULL_I_SEL	Pull-up power configuration register
0x24	CDC_DATA	Scan result register

15.2.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

19	CDC_CLKEN	CDC module operation enable 1: Work 0: Off, the default is 0

15.2.2. Analog module switch register (ANA_CFG)

Address offset: 0x10

Reset value: 0x0000 0007

15:5	4	3	2	1	0
Reserved	XTAL_HFR_SEL	XTAL_SEL	PD_XTAL	PD_CDC	PD_ADC
	RW	RW	RW	RW	RW

1	PD_CDC	CDC work control register 0: The CDC module works normally 1: The CDC module does not work

15.2.3. CDC scan start register (CDC_START)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:1	-	Reserved
0	START	CDC scan open register: 1: CDC scan is turned on; 0: CDC scan stop START=0→1(⎍), start CDC scan, after one scan, the hardware will automatically clear to 0. If you want to start the next CDC scan, you must wait for the last conversion to complete when CDC_START is 0, and then the software will set it to 1 to start the next CDC scan. If CDC_START is cleared to 0 during the CDC scan process, the scan will end immediately

15.2.4. CDC scan configuration register 1 (CDC_SCAN_CFG1)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									SW_PRE_OFF	PRS_DIV					
									RW	RW					

31:7	-	Reserved
6	SW_PRE_OFF	Front-end charge and discharge clock switch control 1: Close sw_clk 0: Open sw_clk
5:0	PRS_DIV	Front-end charge and discharge clock frequency selection register

		000000 ~ 111101: Fixed frequency: F=F48M/2/(PRS_DIV+4), that is (6M~369k) 11 1110: Highest frequency 3M, lowest frequency 1M, center frequency 1.5M, normal distribution 11 1111: The highest frequency is 3M, the lowest frequency is 1M, and the center frequency is 1.5M, evenly distributed

15.2.5. CDC scan configuration register 2 (CDC_SCAN_CFG2)

Address offset: 0x08

Reset value: 0x0000 0070

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:7	6	5	4	3	2	1	0
Reserved	RESO			DS		PRE_CHRG_SEL	INIT_DISCHRG_SEL
	RW			RW		RW	RW

31:7	-	Reserved
6:4	RESO	Counter bit selection register 000: 9 bits; 001: 10 bits; 010: 11 bits; 011: 12 bits; 100: 13 bits; 101: 14 bits; 110: 15 bits; 111: 16 bits
3:2	DS	Count clock frequency selection register 00: 24M 01: 12M 10: 6M 11: 4M
1	PRE_CHRG_SEL	Pre-charge time selection 0: 20us 1: 40us
0	INIT_DISCHRG_SEL	Pre-discharge time selection 0: 2us 1: 10us

15.2.6. CDC address configuration register (CDC_ADDR_CFG)

Address offset: 0x0C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									PARALLEL_EN	ADDR					
									RW	RW					

31:7	-	Reserved
6	PARALLEL_EN	SNS channel parallel enable register 1: Multi-channel parallel 0: Single channel
5:0	ADDR	The address of the detection channel, the channel number can be configured from 0 to 59 000000: SNS00 000001: SNS01 000010: SNS02 000011: SNS03 000100: SNS04 000101: SNS05 000110: SNS06 000111: SNS07 … 111010: SNS58 111011: SNS059

15.2.7. Interrupt configuration register (CDC_INT_CFG)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	CLR_INT_STAT	INT_STAT	INT_EN
	W	R	RW

31:3	-	Reserved
2	CLR_INT_STAT	CDC interrupt status clear register, write 1 to clear interrupt status bit, write 0 invalid
1	INT_STAT	CDC interrupt status register 0: CDC scan is not completed 1: CDC scan completed
0	INT_EN	CDC interrupt enable register 0: Interrupt is not enabled; 1: Interrupt enable

15.2.8. Analog register (CDC_ANA_CFG)

Address offset: 0x14

Reset value: 0x0000 002F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										RB_SEL			VTH_SEL		
										RW			RW		

31:6	-	Reserved
5:3	RB_SEL	RB resistance size selection 010: 60k 011: 80k Other: Reserved When using, you need to read the RB80k calibration value from the address [0x203D2] of the chip NVR2: Scale normalized sensitivity using CBYTE[0x203D2] (k) / CDC_RB_ADJ (k) In the sensitivity debugging stage, the CDC_RB_ADJ program is fixed equal to the read value of the CBYTE[0x203D2] address of the debugging chip, and the normalized ratio is 1. Different chips read different calibration values of RB80k, and the normalized ratio is calculated according to the formula CBYTE[0x203D2] (k) / CDC_RB_ADJ (k)
2:0	VTH_SEL	VTH voltage selection signal 000: 1.8V 001: 2.1V 010: 2.5V 011: 2.8V 100: 3.2V 101: 3.5V 110: 3.9V 111: 4.2V

15.2.9. SENSOR port selection enable register 1 (SENSOR_IO_SEL1)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SNS 31	SNS 30	SNS 29	SNS 28	SNS 27	SNS 26	SNS 25	SNS 24	SNS 23	SNS 22	SNS 21	SNS 20	SNS 19	SNS 18	SNS 17	Res.
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SNS 15	SNS 14	SNS 13	SNS 12	SNS 11	SNS 10	SNS 09	SNS 08	SNS 07	SNS 06	SNS 05	SNS 04	SNS 03	SNS 02	SNS 01	SNS 00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:0	SNS[31:0]	SENSOR's IO selection register, each bit corresponds to a sensor 1: Select the SENSOR function 0: SENSOR function is not selected Bit[16]: Reserved

15.2.10. SENSOR port selection enable register 2 (SENSOR_IO_SEL2)

Address offset: 0x1C

Reset value: 0x0000 0000

31:28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	SNS59	SNS58	SNS57	SNS56	SNS55	SNS54	SNS53	SNS52	SNS51	SNS50	SNS49	SNS48
	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SNS47	SNS46	SNS45	SNS44	SNS43	SNS42	SNS41	SNS40	SNS39	SNS38	SNS37	SNS36	SNS35	SNS34	SNS33	SNS32
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:28	-	Reserved
27:0	SNS[27:0]	IO selection register of SENSOR, each bit corresponds to a sensor 1: Select the SENSOR function 0: SENSOR function not selected

15.2.11. Pull-up power configuration register (PULL_I_SEL)

Address offset: 0x20

Reset value: 0x0000 0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							PUI[8:0]								
							RW								

31:9	-	Reserved
8:0	PUI[8:0]	CDC pull-up current source size selection switch

15.2.12. Scan result register (CDC_DATA)

Address offset: 0x24

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

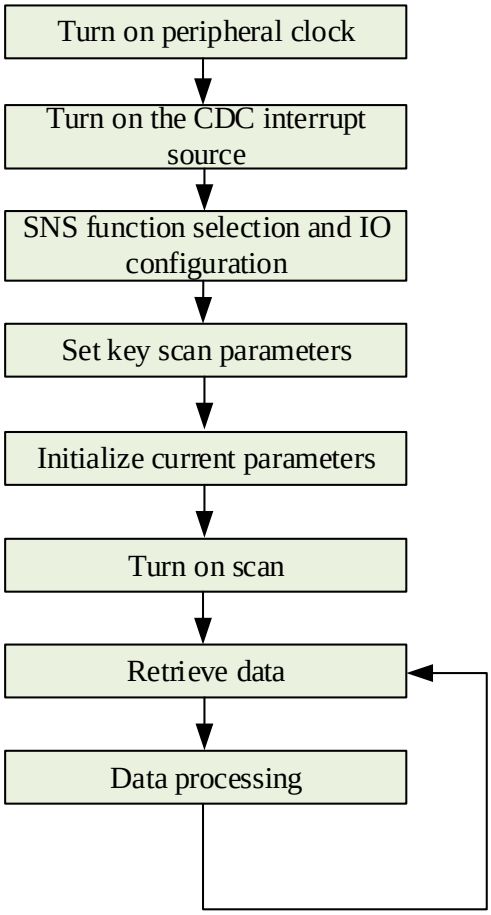
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															

R

31:16	-	Reserved
15:0	DATA[15:0]	CDC scan channel data

15.3. Touch key configuration process

Touch key scanning is a query or interrupt mode. First, configure the touch key parameters; then, turn on the touch key scan; finally, the Touch key interrupts to obtain and save the touch key data, and the software algorithm processes the data and determines the output of the keys.



16 Analog-to-digital conversion module (ADC)

16.1. Module description

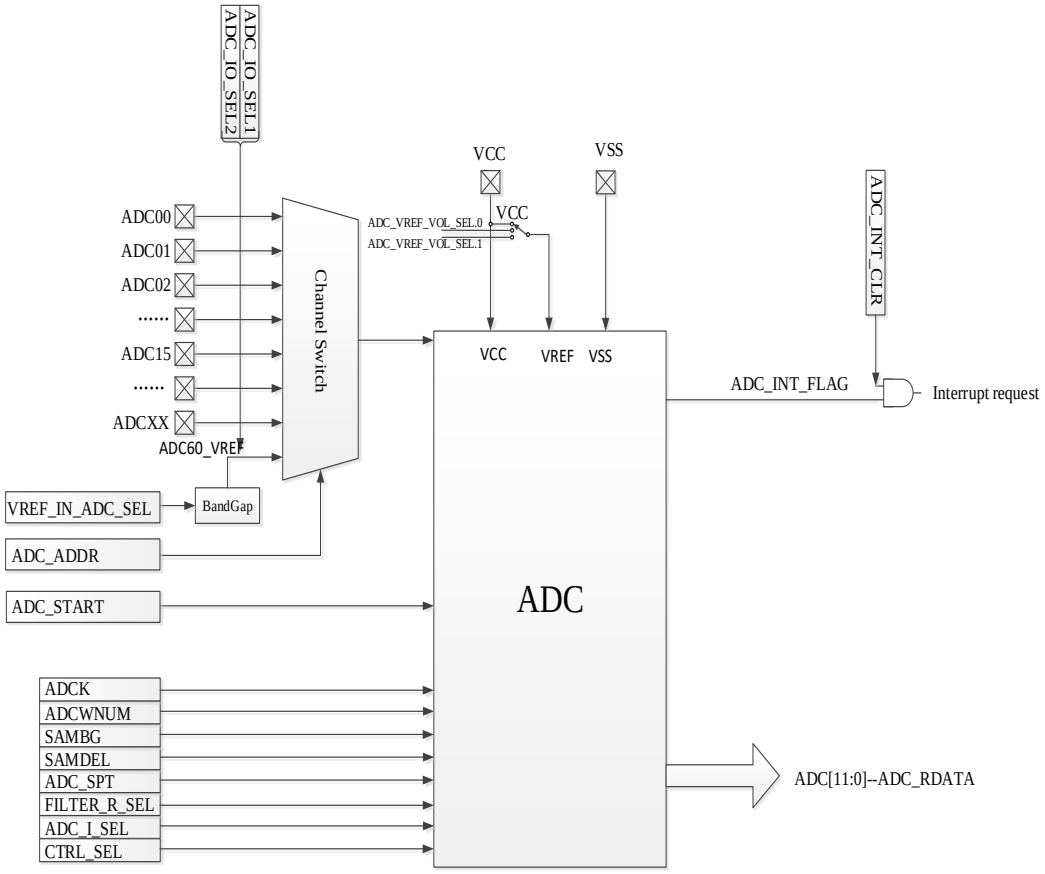
The analog-to-digital converter ADC converts a continuous analog signal into a discrete digital signal. ADC can implement analog-to-digital conversion on any channel selected by software. When ADC_START = 0, the ADC module is disabled. Configure ADC_START = 1 once to perform an ADC scan. After the conversion is completed, the result is saved in the data register (ADC_RDATA), and then an interrupt is triggered.

ADC_START write 1 to start scanning. After one scan, the hardware will automatically set it to 0. To start the next scan, the software needs to set it to 1 again. If ADC_START = 0 is set during the scanning process, the scanning will be stopped immediately and the relevant signals inside the module will be reset.

16.1.1. Features

- 12 Bit resolution linear successive approximation ADC
- Single conversion mode
- Sampling time and conversion speed are configurable
- Support wake-up in idle mode 0
- Reference voltage: VCC/2V/4V

16.1.2. ADC module block diagram



ADC structure block diagram

16.1.3. ADC conversion time

The conversion clock can be selected through the configuration register ADC_CFG1[2:0], there are 8 options:

ADC_CLK: 8MHz 6MHz 4MHz 3MHz
2MHz 1.5MHz 1MHz 0.5MHz

As shown in the table, the ADC conversion time formula:

Formula	Instruction
T_{AD} = T1 +T2+T3	ADC conversion time
T1 = (ADC_SPT+1) *4*T_{ADC_CLK}	ADC sampling time
T2 = (3+ADCWNUM+SAMDEL)*T_{ADC_CLK}	Conversion interval and sampling delay time
T3 = (2*1+12)*T_{ADC_CLK}	The time for the sampled signal to be converted into data

ADCWNUM: Selection of distance conversion interval after sampling
(3+ADCWNUM)*ADC_CLK;

SAMDEL: Sampling delay time selection, 00: 0*(ADC_CLK); 01: 2*(ADC_CLK);
10: 4*(ADC_CLK); 11: 8*(ADC_CLK).

16.1.4. Reference voltage

The BF7807AMXX series has 3 reference sources: VCC/2V/4V

- **When VCC is selected as the ADC reference voltage:**

When the power supply voltage fluctuates greatly or drops, the VCC voltage value can be inversely calculated by formula (1):

ADCINNER_Data/VREF_IN_ADC_SEL = 4096/VCC (1)

Note:

- ADCINNER_Data: ADC internal channel data;
- VREF_IN_ADC_SEL: The chip calibration value needs to be read;

The Vin voltage value can be inversely calculated by formula (2):

Vin_Data/Vin = 4096/VCC (2)

Note:

- Vin_Data: ADC input channel data;
- Vin: Input voltage;

The Vin voltage value can be obtained by formula (3):

Vin = (Vin_Data/ADCINNER_Data)*VREF_IN_ADC_SEL (3)

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 two data acquisitions is as short as possible;

- **When selecting ADC_VREF_VOL_SEL 2V/4V reference voltage, the frequency must be ≤ 3MHz:**

  The Vin voltage value can be inversely calculated by formula (4):

  Vin_Data/Vin=4096/ ADC_VREF_VOL_SEL (4)

  Note:
  - Vin_Data: ADC input channel data;
  - Vin: Input voltage (0~ ADC_VREF_VOL_SEL);
  - VREF_IN_ADC_SEL: Need to read chip calibration value.

  Note: When 4V reference voltage is selected, VCC is greater than 4.5V.

- The chip calibration value is stored in the NVR2 (information block) storage area. The NVR2 needs to be unlocked to read the calibration value, which cannot be changed by the user. For details, please refer to the "NVR2 read" reference chapter:

  CBYTE[0x203C0] = ADC internal channel input voltage calibration value high eight bits;
  CBYTE[0x203C1] = ADC internal channel input voltage calibration value lower eight bits;
  Read the chip information address ADC internal channel input voltage 1.362V calibration value;

  CBYTE [0x203C2] = ADC internal channel input voltage calibration value high eight bits;
  CBYTE [0x203C3] = ADC internal channel input voltage calibration value high eight bits;
  Read the chip information address ADC internal channel input voltage 2.253V calibration value;

  CBYTE[0x203C4] = ADC internal channel input voltage calibration value high eight bits;
  CBYTE[0x203C5] = ADC internal channel input voltage calibration value lower eight bits;
  Read the chip information address ADC internal channel input voltage 3.111V calibration value;

  CBYTE [0x203C6] = ADC internal channel input voltage calibration value high eight bits;
  CBYTE [0x203C7] = ADC internal channel input voltage calibration value high eight bits;
  Read the chip information address ADC internal channel input voltage 4.082V calibration value;

  CBYTE [0x203C8] = ADC internal channel input voltage calibration value high eight bits;
  CBYTE [0x203C9] = ADC internal channel input voltage calibration value high eight bits;
  Read the calibration value of the chip information address ADC_VREF2V;

  CBYTE [0x203CA] = ADC internal channel input voltage calibration value high eight bits;
  CBYTE [0x203CB = ADC internal channel input voltage calibration value high eight bits;
  Read the calibration value of the chip information address ADC_VREF4V.

16.1.5. Single conversion mode

In single conversion mode, only one conversion is performed after the ADC is started. ADC_START write 1 to start scanning. After one scan, the hardware will automatically set to 0. If you want to start the next scan, you need to set it to 1 again by software. If ADC_START = 0 is set during the scanning process, the scanning will stop immediately and the relevant internal signals of the module will be reset.

Single conversion steps:

1. ADC channel is configured as ADC function
2. Enable the PD_ADC bit of the ANA_CFG register to 1, enable the ADC module
3. Set ADC conversion parameters
   i. Set ADC sampling time
   ii. Set the ADC sampling delay time
   iii. Set the ADC conversion interval time
   iv. Set the clock ADCK for ADC conversion
4. Select ADC reference voltage
5. Select ADC scan channel
6. Configure ADC_START =1, start scanning
7. Get results
8. If you need to convert other channels, repeat steps 5~7

Note: ADC_START is not allowed to be configured during scanning. At the same time, if the IO port corresponding to the channel address is not enabled as an ADC port, the ADC_START register cannot be configured to be pulled high, and scanning cannot be started.

16.2. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register
0x10	ANA_CFG	Analog module switch register

Base address: 0x5007_0000

Address offset	Register	Description
0x00	ADC_START	ADC scan enable register
0x04	ADC_ADDR	ADC channel address selection register
0x08	ADC_SPT	ADC sampling time configuration register
0x0C	ADC_CFG1	ADC configuration register 1
0x10	ADC_CFG2	ADC configuration register 2
0x14	ADC_RDATA	ADC scan result register
0x18	ADC_INT_CFG	ADC interrupt configuration register
0x1C	ADC_IO_SEL1	ADC selection enable register 1
0x20	ADC_IO_SEL2	ADC selection enable register 2

16.2.1. Peripheral module clock control register (RCU_EN)

This register is a register that allows or prohibits the provision of clock to the ADC.

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

20	ADC_CLKEN	ADC module work enable 1: Work 0: Off, the default is 0

16.2.2. Analog module switch register (ANA_CFG)

Address offset: 0x10

Reset value: 0x0000 0007

15:5	4	3	2	1	0
Reserved	XTAL_HFR_SEL	XTAL_SEL	PD_XTAL	PD_CDC	PD_ADC
	RW	RW	RW	RW	RW

0	PD_ADC	Analog ADC shutdown control register 0: ADC module works normally 1: ADC module does not work

16.2.3. Analog module switch register (ADC_START)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:1	-	Reserved
0	START	ADC scan enable register START is set to 1 from 0, the ADC starts to scan, after one scan, the START hardware is automatically set to 0

16.2.4. ADC channel address selection register (ADC_ADDR)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:6	-	Reserved
5:0	ADDR	ADC channel address selection register, used to control the selection of the current scan channel 000000: Scan channel ADC00 000001: Scan channel ADC01 000010: Scan channel ADC02 000011: Scan channel ADC03 000100: Scan channel ADC04 000101: Scan channel ADC05 000110: Scan channel ADC06 000111: Scan channel ADC07 … 111010: Scan channel ADC58 111011: Scan channel ADC59 111100: ADC_VREF input channel

16.2.5. ADC sampling time configuration register (ADC_SPT)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:8	-	Reserved
7:0	SPT	ADC sampling time configuration register Sampling time: T1 = (ADC_SPT+1)*4*Tadc_clk

16.2.6. ADC configuration register 1 (ADC_CFG1)

Address offset: 0x0C

Reset value: 0x0000 0F80

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ADCWNUM					SAMBG	SAMDEL		Res.	ADCK		
				RW					RW	RW			RW		

31:12	-	Reserved
11:7	ADCWNUM	Distance conversion interval after sampling: (3+ADCWNUM)*ADC_CLK 00000: Reserved 00001: Reserved 00010: 5*ADC_CLK 00011: 6*ADC_CLK 00100: 7*ADC_CLK ... 11110: 33*ADC_CLK 11111: 34*ADC_CLK
6	SAMBG	Sampling timing and comparison timing interval selection 0: Interval 0 1: Interval 1*(ADC_CLK)
5:4	SAMDEL	Sampling delay time selection

		00: 0*ADC_CLK 01: 2*ADC_CLK 10: 4*ADC_CLK 11:8*ADC_CLK
3	-	Reserved
2:0	ADCK	ADC_CLK frequency division selection: When ADC_VREF output is selected, the frequency must be ≤3MHZ 000: 8MHz 001: 6MHz 010: 4MHz 011: 3MHz 100: 2MHz 101: 1.5MHz 110: 1MHz 111: 0.5MHz

16.2.7. ADC configuration register 2 (ADC_CFG2)

Address offset: 0x10

Reset value: 0x0000 0010

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:9	8	7	6	5	4	3	2	1	0
Res.	VREF_SEL	VREF_VOL_SEL	VREF_IN_ADC_SEL		CTRL_SEL		I_SEL[1:0]		FILTER_R_SEL
	RW	RW	RW		RW		RW		RW

31:9	-	Reserved
8	VREF_SEL	Select the source of the output signal 0: Select VCC as the output signal 1: Select the output of the ADC_VREF module as the output signal
7	VREF_VOL_SEL	ADC_VREF output mode selection signal, ADC_CLK frequency must be ≤3MHZ 0: 2V as ADC reference voltage 1: 4V as ADC reference voltage (**VCC is greater than 4.5V**)
6:5	VREF_IN_ADC_SEL	Voltage ADC60_VREF input to ADC60 00: 1.362V 01: 2.253V 10: 3.111V 11: 4.082V

4:3	CTRL_SEL	ADC comparator offset cancellation selection signal, the default value is 10 00/01: To remove the offset first and then sample 10/11: Offset elimination and sampling at the same time 10: The switch of the first-stage comparator is finally disconnected 11: All switches are disconnected at the same time
2	I_SEL_1	ADC bias current size selection register Select BUFFER bias current 0: 5uA 1: 4uA
1	I_SEL_0	Select comparator bias current 0: 5uA 1: 4uA
0	FILTER_R_SEL	Input signal filter selection 0: No RC filter 1: Add RC filter

16.2.8. ADC scan result register (ADC_RDATA)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				RDATA[11:0]											
				R											

31:12	-	Reserved
11:0	RDATA[11:0]	ADC scan result register

16.2.9. ADC interrupt configuration register (ADC_INT_CFG)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	INT_CLR	INT_FLAG	INT_EN
	W	R	RW

31:3	-	Reserved
2	INT_CLR	Interrupt status flag clear register Write 1 to clear INT_FLAG, write only
1	INT_FLAG	Interrupt status flag register 1: The interrupt is valid 0: Interrupt is invalid, read only
0	INT_EN	Interrupt enable register 1: Interrupt enable 0: Interrupt disabled (used in polling mode)

16.2.10. ADC selection enable register 1 (ADC_IO_SEL1)

Address offset: 0x1C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC31	ADC30	ADC29	ADC28	ADC27	ADC26	ADC25	ADC24	ADC23	ADC22	ADC21	ADC20	ADC19	ADC18	ADC17	Res.
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC15	ADC14	ADC13	ADC12	ADC11	ADC10	ADC09	ADC08	ADC07	ADC06	ADC05	ADC04	ADC03	ADC02	ADC01	ADC00
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:0	ADC[31:0]	Pin control enable, each bit corresponds to an ADC Bit [0]: ADC00 Bit [1]: ADC01 … Bit [31]: ADC31 1: Select ADC function; 0: Do not select ADC function Bit [16]: Reserved

16.2.11. ADC selection enable register 2 (ADC_IO_SEL2)

Address offset: 0x20

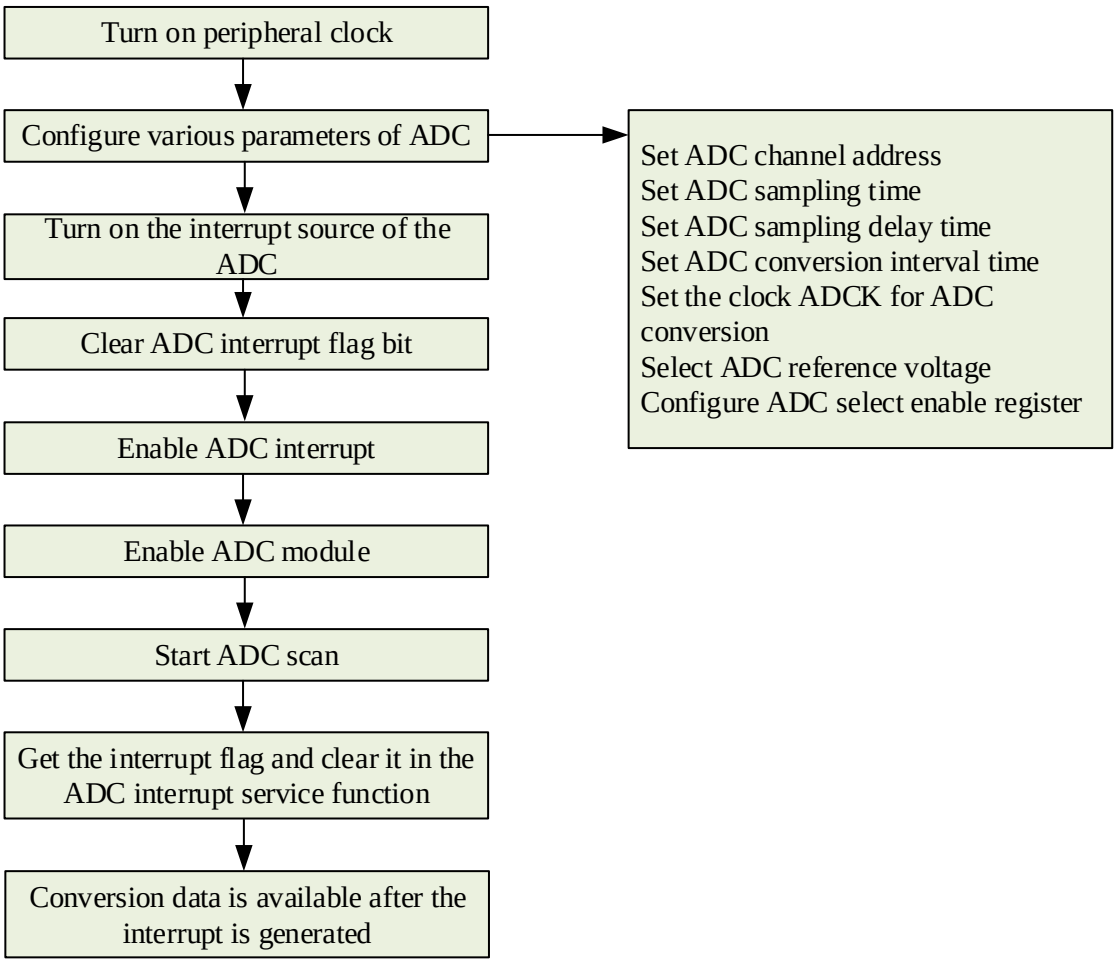
Reset value: 0x0000 0000

31:28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved	ADC59	ADC58	ADC57	ADC56	ADC55	ADC54	ADC53	ADC52	ADC51	ADC50	ADC49	ADC48
	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC47	ADC46	ADC45	ADC44	ADC43	ADC42	ADC41	ADC40	ADC39	ADC38	ADC37	ADC36	ADC35	ADC34	ADC33	ADC32
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:28	-	Reserved
27:0	ADC[27:0]	Pin control enable, each bit corresponds to an ADC ADC_IO_SEL2[0]: ADC32 ADC_IO_SEL2[1]: ADC33 … ADC_IO_SEL2[27]: ADC59 1: Select ADC function; 0: Do not select ADC function

16.3. ADC configuration process



17 Low voltage detection (LVDT)

17.1. LVDT function description

The BF7807AMXX series supports low voltage alarm function, which can effectively monitor the dynamic changes of voltage. Support 8 voltage levels, respectively: 2.7V/3.0V/3.3V/3.6V/3.8V/4.0V/4.2V/4.4V (preset point step-down interrupt, hysteresis 0.1V generates corresponding step-up interrupt). When the voltage monitoring is configured with the above threshold, a voltage drop below this threshold will trigger a low-voltage interrupt, and the system can handle the low-voltage interrupt appropriately according to application needs.

17.2. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x1C	LVDT_CTRL	LVDT control register
0x20	LVDT_STATE	LVDT status register

17.2.1. LVDT control register (LVDT_CTRL)

Address offset: 0x1C

Reset value: 0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Reserved	VTH_SEL			DELAY_SEL		PD_LVDT	PO_INTEN	BO_INTEN
	RW			RW		RW	RW	RW

31:8	-	Reserved
7:5	VTH_SEL	LVDT threshold selection 000: 2.7V 001: 3.0V 010: 3.3V 011: 3.6V 100: 3.8V 101: 4.0V 110: 4.2V 111: 4.4V For the specific corresponding threshold voltage, see the table "Threshold and

		delay selection"
4:3	DELAY_SEL	LVDT power-down delay configuration 00: Power-down delay 1 01: Power-down delay 2 10: Power-down delay 3 11: Power-down delay 4
2	PD_LVDT	LVDT control register 1: Off 0: On, off by default
1	PO_INTEN	LVDT low voltage boost interrupt enable 1: Enable 0: Disable
0	BO_INTEN	LVDT low voltage step-down interrupt enable 1: Enable 0: Disable

17.2.2. LVDT status register (LVDT_STATE)

Address offset: 0x20

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														POF	BOF
														RC_W1	RC_W1

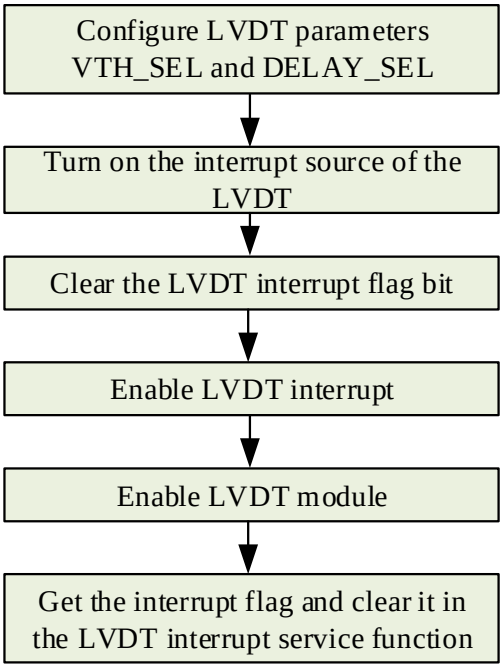
31:2	-	Reserved
1	POF	LVDT boost interrupt flag 1: Effective 0: Invalid When this interrupt occurs, the hardware sets Bit, and the software writes 1 to clear 0
0	BOF	LVDT buck interrupt flag 1: Effective 0: Invalid When this interrupt occurs, the hardware sets Bit, and the software writes 1 to clear 0

17.3. Threshold and delay selection

Threshold selection	Delay selection	LVDT			
		Power down threshold (V)	Recovery threshold (V)	Hysteresis (mV)	Delay (μs)
000	00	2.7	2.8	124.8	7.3
	01	2.7	2.8	125.2	14.3
	10	2.7	2.8	125.9	28.5
	11	2.7	2.8	127.3	56.7
001	00	3.0	3.1	114.4	8.0
	01	3.0	3.1	114.8	15.8
	10	3.0	3.1	115.5	31.4
	11	3.0	3.1	117.1	62.5
010	00	3.3	3.4	92.9	8.6
	01	3.3	3.4	93.4	17.1
	10	3.3	3.4	94.2	34.2
	11	3.3	3.4	95.9	68.1
011	00	3.6	3.7	109.5	9.2
	01	3.6	3.7	110	18.2
	10	3.6	3.7	110.9	36.4
	11	3.6	3.7	112.7	72.6
100	00	3.8	3.9	123.2	9.5
	01	3.8	3.9	123.7	19.0
	10	3.8	3.9	124.6	37.9
	11	3.8	3.9	126.5	75.6
101	00	4.0	4.1	135.6	9.8
	01	4.0	4.1	136	19.4
	10	4.0	4.1	137	39.0
	11	4.0	4.1	139	77.9
110	00	4.2	4.3	124.3	10.0
	01	4.2	4.3	124.8	20.0
	10	4.2	4.3	125.8	40.1
	11	4.2	4.3	127.8	80.0
111	00	4.4	4.5	82.3	10.3
	01	4.4	4.5	83.3	20.5
	10	4.4	4.5	84.3	41
	11	4.4	4.5	86.4	82

Table Threshold and delay selection

17.4. LVDT configuration process



18 LED/LCD driver module

The module can be configured with two drive modes: LED matrix drive mode and LCD drive mode. Select LCD drive or LED drive configuration by register DP_CON[2], and only support one mode of work at the same time.

All drive modes, the total IO port switch is configurable, the scanning mode is configurable, the software controls the LED scanning to start, the interrupt mode scans once to interrupt and stop, and the cycle mode automatically starts the next frame scan after scanning for one frame without interruption. If you want to stop, you need the software to turn off the scan enable.

18.1. LED matrix

18.1.1. Features

- Support up to 8 COM x 16 SEG LED drive matrix
- Single COM conduction time setting file: 8-bit register, configurable range is 16μs-4.096ms, step is 16μs
- Single COM port conduction duty cycle can be configured: 1/8~8/8 can be configured
- The number of COMs supported is 1-8, the registers are configurable
- Each IO port is switched to COM port/SEG port through register configuration, register COM_IO_SEL determines the number of COM ports scanned, and register SEG_IO_SEL determines the number of SEG ports

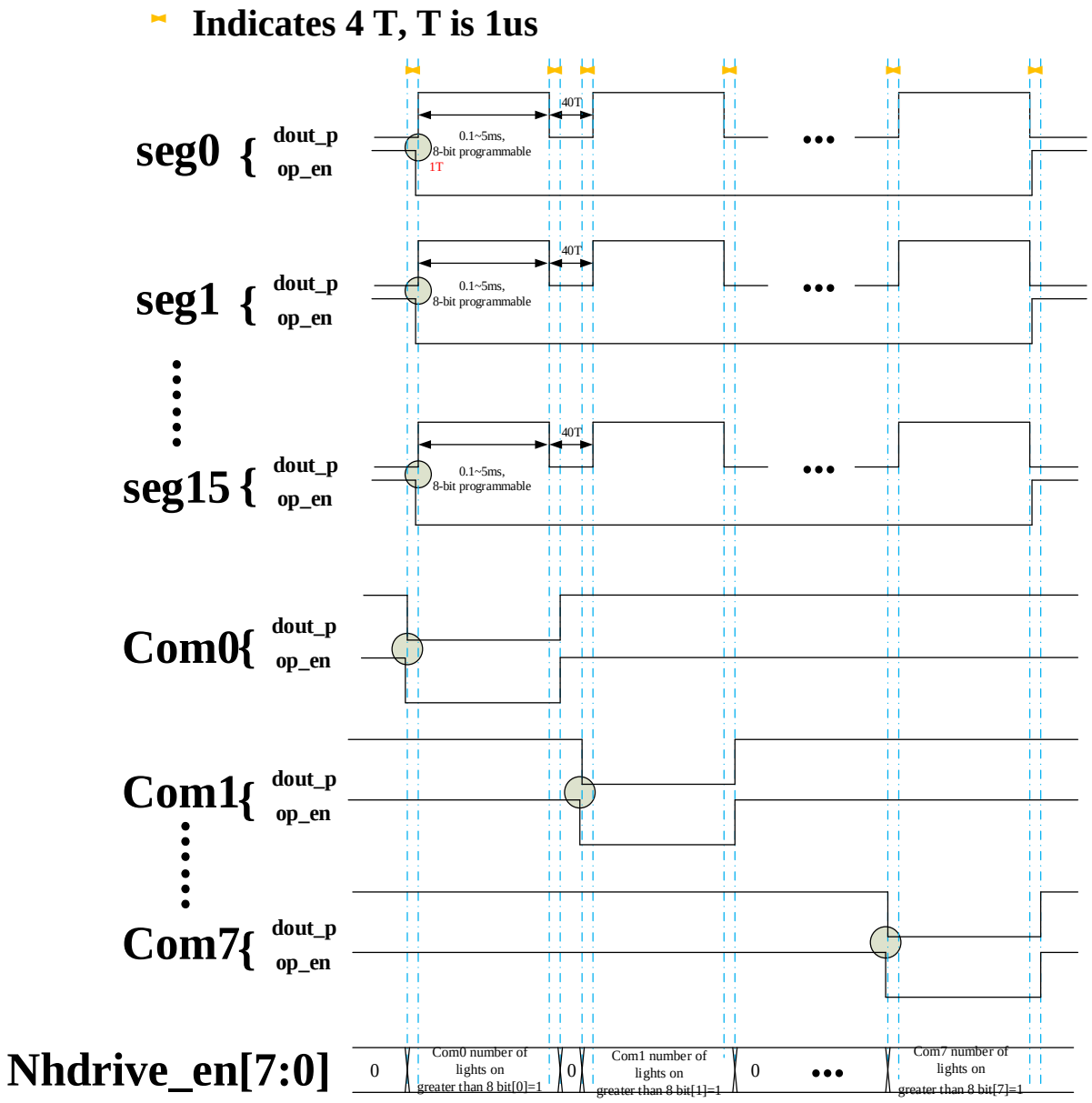
In the pin definition of BF7807AMXX, COML: COM of LED matrix, SEGL: SEG of LED matrix.

18.1.2. LED matrix drive description

The LED matrix circuit consists of a controller, a counter, a duty cycle comparator.

In LED matrix mode, PA port is used as COM port and PC/PD port is used as SEG port. The corresponding SEG port output data of each COM port is configured by the register LED_LCD_BUFx to determine whether to light up (1 means light, 0 means no light), the hardware code only needs to directly output data to the IO port according to the following sequence.

The timing diagram is as follows:



Timing diagram of LED matrix mode

Each bit of the signal Nhdrive_en[7:0] corresponds to the drive capability of a COM port. There are two drive options, which are determined by the register NHDRIV_EN_SEL. When the high-current IO port function is selected, the fixedly selected COM port is the high-current port, and the corresponding Nhdrive_en is 1.

Parameter	Number of driving lights	Nhdrive_en corresponding bit
NHDRIV_EN_SEL=1	1~8	0
	Greater than 8	1
NHDRIV_EN_SEL=0	1~4	0
	Greater than 4	1

18.1.3. Show configuration address

LED matrix drive mode corresponds to the display configuration:
SEGx means to choose whether to light up, 0: No light, 1: Light.

Address	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	0x4C	0x48	0x44	0x40	0x3C	0x38	0x34	0x30
bit[0]	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0
bit[1]	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1
bit[2]	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2
bit[3]	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3
bit[4]	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4
bit[5]	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5
bit[6]	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6
bit[7]	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7
bit[8]	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8
bit[9]	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9
bit[10]	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10
bit[11]	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11
bit[12]	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12
bit[13]	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13
bit[14]	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14
bit[15]	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15

18.2. LCD matrix

18.2.1. Features

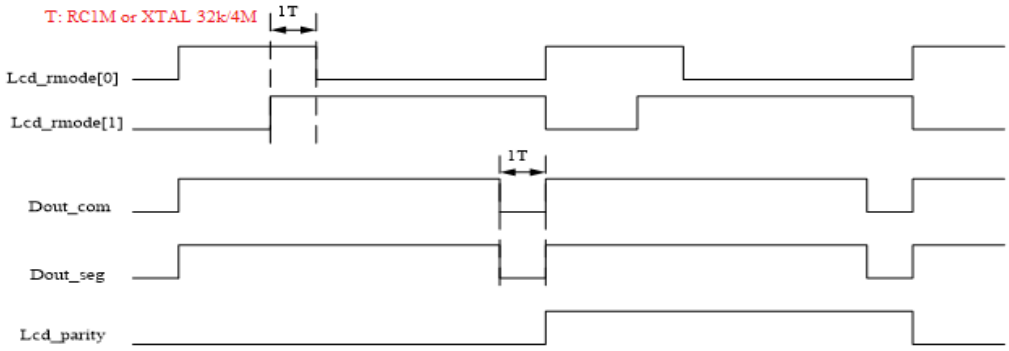
- Supports multiple drive duty cycles
  - 1/4 duty cycle, 1/3 bias (4 COM x 24 SEG)
  - 1/8 duty cycle, 1/4 bias (8 COM x 24 SEG)
  - 1/4 duty cycle, 1/3 bias (4 COM x 28 SEG)
  - 1/5 duty cycle, 1/3 bias (5 COM x 27 SEG)
  - 1/6 duty cycle, 1/3 bias (6 COM x 26 SEG)
  - 1/6 duty cycle, 1/4 bias (6 COM x 26 SEG)
- Support 2 drive modes: Traditional resistance mode (fast charging mode, slow charging mode), automatic switching mode between fast and slow charging
- Support 3 kinds of bias resistance: 60k/225k/900k
- Support 3 kinds of work clock: Internal low-speed clock 32768Hz, external crystal oscillator 32768Hz/4MHz, RC1MHz
  - When LCD selects external crystal oscillator and RC1M, the lighting time of a single COM can be configured, and the configuration range is 0.064~4.096ms, and the step is 64us;
  - LCD selects internal low-speed clock 32768Hz, LCD turn-on frequency is fixed at 64Hz (8COM configuration).
- Support LCD contrast control, 0.531VDD~1.000VDD, 16-level contrast adjustment
- The COM port is determined by the duty cycle configuration, and the SEG port is freely configured by the register

18.2.2. LCD matrix driver description

The LCD drive circuit consists of a controller, a counter, and two comparators.

In LCD matrix mode, the number of COM ports scanned is completely controlled by the duty cycle configuration of the drive mode. The output data of the SEG port corresponding to each COM port is configured by the register LED_LCD_BUFx to determine whether to light (1 means light, 0 means no lights up), the hardware code needs to directly output data to the IO port control circuit according to the following sequence.

The sequence diagram is as follows:



LCD timing diagram

Analog IO implements the following truth table:

Bias voltage selection LCD_BIAS_SEL 0: 1/3 bias; 1: 1/4 bias;

Odd and even frame selection LCD_PARITY 0: Odd frame; 1: Even frame;

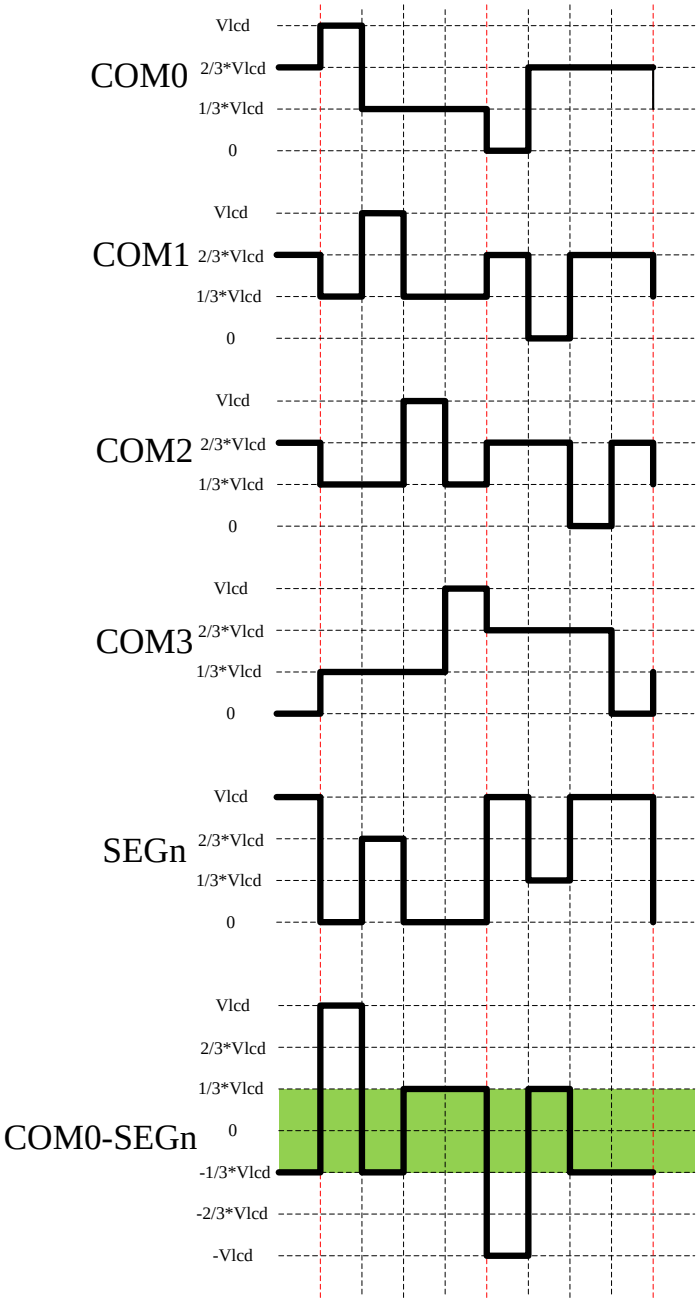
Resistance string selection LCD_RMODE 001: 20k; 010: 75k; 100: 300k;

Data selection DOUT_PB (for example), compatible with the previous data line, the output function of the corresponding IO port is invalid (OP_EN_N=1).

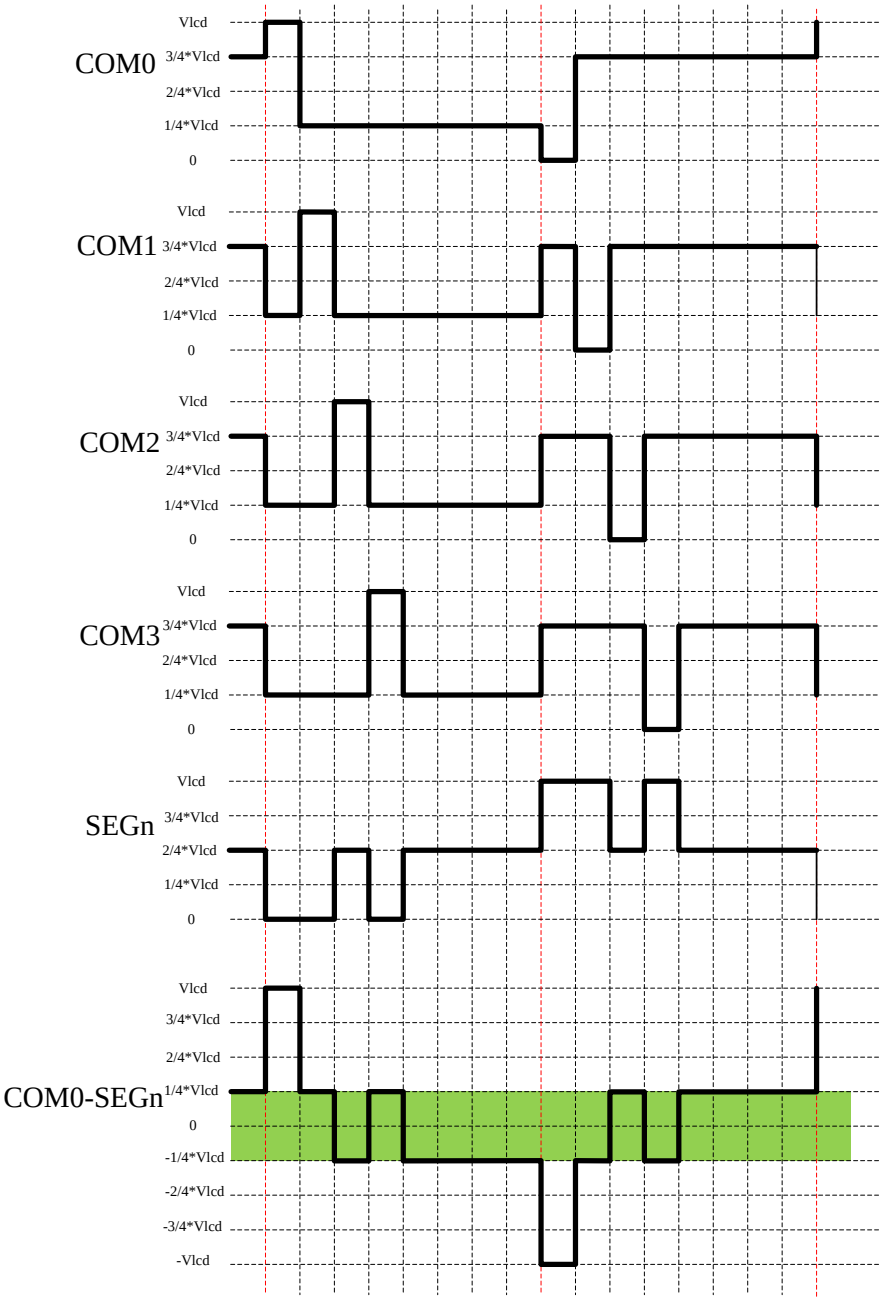
COM truth table			
LCD_BIAS_SEL	LCD_PARITY	DOUT_PB	Output voltage value
0	0	0	1/3VLCD
0	0	1	VLCD
0	1	0	2/3VLCD
0	1	1	VSS
1	0	0	1/4VLCD
1	0	1	VLCD
1	1	0	3/4VLCD
1	1	1	VSS
SEG mouth truth table			
LCD_BIAS_SEL	LCD_PARITY	DOUT_PB	Output voltage value
0	0	0	2/3VLCD
0	0	1	VSS
0	1	0	1/3VLCD
0	1	1	VLCD
1	0	0	2/4VLCD
1	0	1	VSS
1	1	0	2/4VLCD
1	1	1	VLCD

LCD configuration truth table

This realizes the bias voltage division sequence on the PAD, as shown in the figure below:



LCD timing diagram (1/4 duty cycle, 1/3 bias)



LCD timing diagram (1/8 duty cycle, 1/4 bias)

18.2.3. Display configuration

The LCD drive mode corresponds to the display configuration:
SEGx means to choose whether to light up, 0: No light, 1: Light.

Address	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	0x4C	0x48	0x44	0x40	0x3C	0x38	0x34	0x30
bit[0]	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0
bit[1]	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1
bit[2]	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2
bit[3]	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3
bit[4]	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4
bit[5]	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5
bit[6]	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6
bit[7]	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7
bit[8]	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8
bit[9]	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9
bit[10]	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10
bit[11]	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11
bit[12]	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12
bit[13]	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13
bit[14]	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14
bit[15]	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15
bit[16]	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16
bit[17]	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17
bit[18]	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18
bit[19]	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19
bit[20]	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20
bit[21]	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21
bit[22]	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22
bit[23]	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23
bit[24]			SEG24	SEG24	SEG24	SEG24	SEG24	SEG24
bit[25]			SEG25	SEG25	SEG25	SEG25	SEG25	SEG25
bit[26]				SEG26	SEG26	SEG26	SEG26	SEG26
bit[27]					SEG27	SEG27	SEG27	SEG27

18.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register
0x0C	XTAL_HS_SEL	Comparator hysteresis voltage selection register in crystal oscillator
0x10	ANA_CFG	Analog module switch register

Base address: 0x5008_0000

Address offset	Register	Description
0x00	SCAN_START	LCD, LED scan open register
0x04	DP_CON	LCD, LED control register
0x08	LCD_DP_MODE	LCD control register
0x0C	SCAN_WIDTH	LED cycle configuration register
0x10	DP_CON1	LCD, LED control register 1
0x14	LED_INT_CFG	LED interrupt configuration register
0x18	LCD_INT_CFG	LCD interrupt configuration register
0x20	LCD_IO_SEL	LCD SEG0-23 port selection configuration register
0x24	COM_IO_SEL	COM port selection configuration register
0x28	SEG_IO_SEL	LED SEG0-15 port selection configuration register
0x2C	LCED_BUF0	SEG port data register 0
0x30	LCED_BUF1	SEG port data register 1
0x34	LCED_BUF2	SEG port data register 2
0x38	LCED_BUF3	SEG port data register 3
0x3C	LCED_BUF4	SEG port data register 4
0x40	LCED_BUF5	SEG port data register 5
0x44	LCED_BUF6	SEG port data register 6
0x48	LCED_BUF7	SEG port data register 7

18.3.1. LED/LCD shared registers

18.3.1.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

23	LED_LCD_CLKEN	LCD/LED module operation enable 1: Work 0: Off, the default is 0

18.3.1.2. LCD, LED scan on register (SCAN_STAR)

Address offset: 0x00

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:1	-	Reserved
0	SCAN_START	LCD, LED scan open register 1: Scan on 0: Scan off SCAN_START is set to 1 from 0, the LCD/LED starts to scan, and is automatically cleared by hardware after the scan is over

18.3.1.3. LCD, LED control register (DP_CON)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									IO_ON	DUTY_SEL			DPSEL	SCAN_MODE	COM_MOD
									RW	RW			RW	RW	RW

31:7	-	Reserved
6	IO_ON	LCD/LED scanning corresponds to the total control bit of all IO ports 0: Close IO 1: Open IO
5:3	DUTY_SEL	LED row and column drive mode single COM port conduction duty cycle configuration register 000: 1/8 duty cycle 001: 2/8 duty cycle

		010: 3/8 duty cycle 011: 4/8 duty cycle 100: 5/8 duty cycle 101: 6/8 duty cycle 110: 7/8 duty cycle 111: 8/8 duty cycle LCD drive mode duty cycle configuration register 000: 1/4 duty cycle, 1/3 bias (4 COM X 24 SEG) COM port: COM0-3; SEG port: SEG0-23 001: 1/8 duty cycle, 1/4 bias (8 COM X 24 SEG) COM port: COM0-7; SEG port: SEG0-23 010: 1/4 duty cycle, 1/3 bias (4 COM X 28 SEG) COM port: COM0-3; SEG port: SEG0-23, COM4-7 shared as SEG24-27 011: 1/5 duty cycle, 1/3 bias (5 COM X 27 SEG) COM port: COM0-4; SEG port: SEG0-23, COM5-7 shared as SEG25-27 100: 1/6 duty cycle, 1/3 bias (6 COM X 26 SEG) COM port: COM0-5; SEG port: SEG0-23, COM6-7 shared as SEG26-SEG27 101: 1/6 duty cycle, 1/4 bias (6 COM X 26 SEG) COM port: COM0-5; SEG port: SEG0-23, COM6-7 shared as SEG26-SEG27 Others: 1/4 duty cycle, 1/3 bias (4 COM X 24 SEG) COM port: COM0-3; SEG port: SEG0-23
2	DPSEL	LCD, LED selection control bit 0: Select LCD driver, LED driver is invalid 1: Select LED driver, LCD driver is invalid
1	SCAN_MODE	LCD, LED scan mode configuration 1: Cycle scan mode 0: Interrupt scan mode
0	COM_MOD	High current IO port drive enable 1: The COM port function is locked and works as a high-current IO port 0: The COM port function is not locked and can be configured to other functions When the COM port locks the high-current IO port, configure the GPIO register to output the drive timing. When it is valid, all the following LED/LCD scan configurations are invalid.

18.3.1.4. LED, LCD cycle configuration register (SCAN_WIDTH)

Address offset: 0x0C
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								SCAN_WIDTH[7:0]							
								RW							

31:8	-	Reserved
7:0	SCAN_WIDTH[7:0]	In the LED matrix drive mode, the corresponding single COM port scan time: period=(scan_width+1)*16us, support configuration range 0.016~4.096ms In LCD drive mode, the corresponding single COM port scan time: period=(scan_width+1)*64us, support configuration range 0.064~4.096ms Note: In this mode, this register is only applicable to LCD selection clock RC1M/XTAL mode. In clock LIRC mode, the LCD fixed frame frequency is 64Hz (8*24)

18.3.1.5. LCD, LED control register 1 (DP_CON1)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

31:6	5	4	3	2	1	0
Reserved	NHDRIV_EN_SEL	PD_LCD_POWER	VOL			
	RW	RW	RW			

31:6	-	Reserved
5	NHDRIV_EN_SEL	nhdriver_en maximum seg lighting selection in LED mode 0: Set to 1 when the number of seg lights on the com port is greater than 4 1: Set to 1 when the number of seg lights on the com port is greater than 8
4	PD_LCD_POWER	LCD contrast control enable bit 0: Close LCD contrast control 1: Open LCD contrast control
3:0	VOL	LCD contrast control bit 0000: VLCD = 0.531VDD 0001: VLCD = 0.563VDD 0010: VLCD = 0.594VDD 0011: VLCD = 0.625VDD 0100: VLCD = 0.656VDD 0101: VLCD = 0.688VDD

		0110: VLCD = 0.719VDD 0111: VLCD = 0.750VDD 1000: VLCD = 0.781VDD 1001: VLCD = 0.813VDD 1010: VLCD = 0.844VDD 1011: VLCD = 0.875VDD 1100: VLCD = 0.906VDD 1101: VLCD = 0.938VDD 1110: VLCD = 0.969VDD 1111: VLCD = 1.000VDD

18.3.1.6. SEG port data register 0 (LED_LCD_BUF0)

Address offset: 0x2C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF0[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF0[15:0]															
RW															

31:28	-	Reserved
27:0	BUF0[27:0]	Data of SEG port of LCD and LED, corresponding to COM0 Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.7. SEG port data register 1 (LED_LCD_BUF1)

Address offset: 0x30

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF1[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF1[15:0]															
RW															

31:28	-	Reserved
27:0	BUF1[27:0]	Data of SEG port of LCD and LED, corresponding to COM1

		Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.8. SEG port data register 2 (LED_LCD_BUF2)

Address offset: 0x34

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF2[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF2[15:0]															
RW															

31:28	-	Reserved
27:0	LED_LCD_BUF2[27:0]	Data of SEG port of LCD and LED, corresponding to COM2 Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.9. SEG port data register 3 (LED_LCD_BUF3)

Address offset: 0x38

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF3[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF3[15:0]															
RW															

31:28	-	Reserved
27:0	BUF3[27:0]	Data of SEG port of LCD and LED, corresponding to COM3 Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.10. SEG port data register 4 (LED_LCD_BUF4)

Address offset: 0x3C

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF4[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF4[15:0]															
RW															

31:28	-	Reserved
27:0	BUF4[27:0]	Data of SEG port of LCD and LED, corresponding to COM4 Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.11. SEG port data register 5 (LED_LCD_BUF5)

Address offset: 0x40

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF5[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF5[15:0]															
RW															

31:28	-	Reserved
27:0	BUF 5[27:0]	Data of SEG port of LCD and LED, corresponding to COM5 Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.12. SEG port data register 6 (LED_LCD_BUF6)

Address offset: 0x44

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF6[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF6[15:0]															
RW															

31:28	-	Reserved
27:0	BUF6[27:0]	Data of SEG port of LCD and LED, corresponding to COM6 Register DP_CON[2] selects whether to drive LCD or LED

18.3.1.13. SEG port data register 7 (LED_LCD_BUF7)

Address offset: 0x48

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				BUF7[27:16]											
				RW											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUF7[15:0]															
RW															

31:28	-	Reserved
27:0	BUF7[27:0]	Data of SEG port of LCD and LED, corresponding to COM7 Register DP_CON[2] selects whether to drive LCD or LED

18.3.2. LED registers

18.3.2.1. LED interrupt configuration register (LED_INT_CFG)

Address offset: 0x14

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	CLR_INT_STAT	INT_STAT	INT_EN
	W	R	RW

31:3	-	Reserved
2	CLR_INT_STAT	LED interrupt status clear register, write 1 to clear interrupt status bit, write 0 invalid
1	INT_STAT	LED interrupt status register 0: LED scan is not completed 1: LED scan completed
0	INT_EN	LED interrupt enable register 0: Interrupt is not enabled 1: Interrupt enable

18.3.2.2. COM port selection configuration register (COM_IO_SEL)

Address offset: 0x24

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:8	7	6	5	4	3	2	1	0
Reserved	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
	RW	RW	RW	RW	RW	RW	RW	RW

31:8	-	Reserved
7:0	COM[7:0]	COM port selection configuration register, bit[0]~bit[7] correspond to PA0~PA7 ports 1: Select COM port mode 0: Select IO port mode Note: This register is valid when selecting LED row-column matrix mode, valid when selecting high-current IO port driver enable, and invalid in other cases.

18.3.2.3. LED SEG0-15 port selection configuration register (SEG_IO_SEL)

Address offset: 0x28

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:16	-	Reserved
15:0	SEG[15:0]	LED_SEG0-15 port selection configuration register, bit[0]~bit[15] correspond to PC4~PC15, PD0~PD3 ports respectively 1: Select SEGMENT port mode 0: Select IO port mode

18.3.3. LCD registers

18.3.3.1. The hysteresis voltage selection register of the comparator in the crystal oscillator (XTAL_HS_SEL)

Address offset: 0x0C
Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:2	1	0
Reserved	HS_SEL[1:0]	
	RW	

31:2	-	Reserved
1:0	HS_SEL[1:0]	The hysteresis voltage selection of the comparator in the crystal oscillator 00: 300mV 01: 400mV 10: 500mV 11: 600mV

18.3.3.2. Analog module switch register (ANA_CFG)

Address offset: 0x10
Reset value: 0x0000 0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:5	4	3	2	1	0
Reserved	XTAL_HFR_SEL	XTAL_SEL	PD_XTAL	PD_CDC	PD_ADC
	RW	RW	RW	RW	RW

31:5	-	Reserved
4	XTAL_HFR_SEL	Analog high frequency crystal oscillator circuit selection (corresponding to different current threshold configuration words) 0: 4MHz 1: 8MHz
3	XTAL_SEL	Analog crystal oscillator circuit frequency selection 0: 32768Hz 1: 4MHz/8MHz
2	PD_XTAL	Analog crystal oscillator circuit (32768Hz/4MHz/8MHz) control register

		1: Off 0: On, off by default

18.3.3.4. LCD control register (LCD_DP_MODE)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									CKSEL		RSEL	FCSEL		RMOD	
									RW		RW	RW		RW	

31:7	-	Reserved
6:5	CKSEL	LCD clock selection register 10/11: Select RC1M 01: Select XTAL 00: Select LIRC
4	RSEL	LCD bias resistance selection control bit 0: LCD bias resistance is 225k 1: The total LCD bias resistance is 900k
3:2	FCSEL	Charge time control bit 00: 1/8 LCD COM cycle 01: 1/16 LCD COM cycle 10: 1/32 LCD COM cycle 11: 1/64 LCD COM cycle
1:0	RMOD	Drive mode selection bits 00: Traditional resistive mode (slow charge mode), the sum of bias resistors is 225k/900k When RSEL=0, the sum of LCD bias resistors is 225k When RSEL=1, the sum of LCD bias resistors is 900k 01: Traditional resistive mode (fast charge mode), the sum of bias resistors is 60k 10/11: Fast and slow charging automatic switching mode, the sum of bias resistance automatically switches between 60k and 225k/900k

18.3.3.5. LCD interrupt configuration register (LCD_INT_CFG)

Address offset: 0x18

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:3	2	1	0
Reserved	CLR_INT_STAT	INT_STAT	INT_EN
	W	R	RW

31:3	-	Reserved
2	CLR_INT_STAT	LCD interrupt status clear register Writing 1 to clear the interrupt status bit, writing 0 is invalid
1	INT_STAT	LCD interrupt status register 0: LCD scan is not completed 1: LCD scan completed
0	INT_EN	LCD interrupt enable register 0: Interrupt is not enabled 1: Interrupt enable

18.3.3.6. LCD SEG0-23 port selection configuration register (LCD_IO_SEL)

Address offset: 0x20

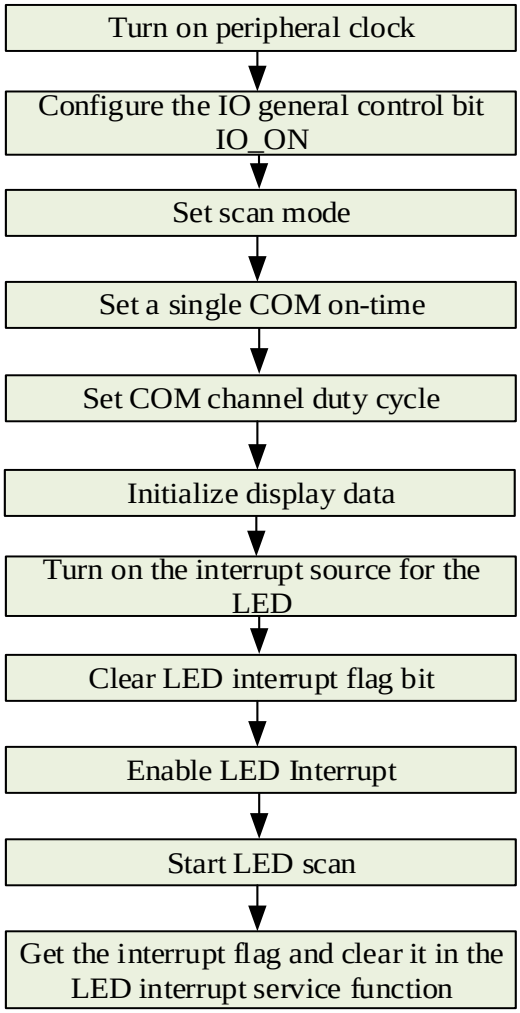
Reset value: 0x0000 0000

31:28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16
	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

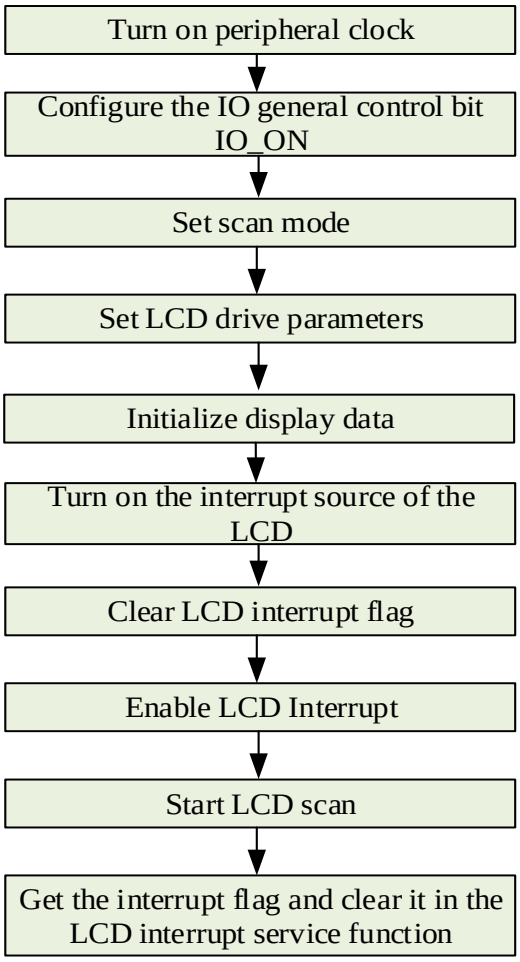
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

31:28	-	Reserved
27:0	IO_SEL[27:0]	Bit[0]~bit[23] are LCD_SEG0-23 port selection configuration registers, corresponding to PA8~PA15, PB8~PB15, PC3~PC4, PC8~PC13 ports; Bit[24]~bit[27] are LCD_SEG25-28 port selection configuration registers, corresponding to PA4~7 ports when multiplexing 1: Select SEGMENT port mode 0: Select IO port mode

18.4. Configuration process



Left: LED configuration process



Right: LCD configuration process

19 Cyclic redundancy check module (CRC)

The cyclic redundancy check module uses a polynomial generator to generate a CRC code from an 8-bit/16-bit/32-bit data. BF7807AMXX adopts look-up table method to realize CRC calculation.

The CRC calculation unit has a single 32-bit read/write data register (CRC_DR), which is used to input new data (write access) and save the result of the previous CRC calculation (read access). Each write operation of the data register will perform another CRC calculation on the previous CRC value and the new value. The CRC calculation is done for the entire 32-bit data word or byte by byte, depending on the data writing format (word-by-word, right-aligned halfword, and aligned byte for access).

19.1. Features

- Support 3 kinds of CRC polynomials
  - Use CRC-32 by default: 0x04C11DB7
    Polynomial: $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$
  - CRC-16: 0x1021
    Polynomial: $x^{16}+x^{12}+x^{5}+1$
  - CRC-8: 0x07
    Polynomial: $x^{8}+x^{2}+x+1$
- Handling 8-bit, 16-bit, 32-bit data size
  - 32-bit data size, CRC calculation is completed within 4 HCLK clock cycles
  - 16-bit data size, CRC calculation is completed within 2 HCLK clock cycles
  - 8-bit data size, CRC calculation is completed within 1 HCLK clock cycle
- The initial value and xor value can be configured

19.2. CRC data format conversion

Register operations can be performed in byte, halfword and word formats. Only related to CRC_POL_SEL when reading.

Data format conversion description: Write data

	REV_IN=0	REV_IN=1	REV_IN=2	REV_IN=3
8-bit write	Does not affect bit order	Byte-wise bit-reversal	Byte-wise bit-reversal	Byte-wise bit-reversal
16-bit write	Does not affect bit order	Byte-wise bit-reversal	Perform bit reversal by halfword	Perform bit reversal by halfword
32-bit write	Does not affect bit order	Byte-wise bit-reversal	Perform bit reversal by halfword	Perform bit reversal by word

Data format conversion description: Read data

CRC-8		CRC-16		CRC-32	
REV_OUT=0	REV_OUT=1	REV_OUT=0	REV_OUT=1	REV_OUT=0	REV_OUT=1
Does not affect bit order	Byte-wise bit-reversal	Does not affect bit order	Perform bit reversal by halfword	Does not affect bit order	Perform bit reversal by word

19.3. Registers

Base address: 0x5000 0000

Address offset	Register	Description
0x04	RCU_EN	Peripheral module clock control register

Base address: 0x5009_0000

Address offset	Register	Description
0x00	CRC_DR	CRC data register
0x04	CRC_CR	CRC control register
0x08	CRC_POL_SEL	CRC polynomial selection
0x0C	CRC_INIT	CRC initial value
0x10	CRC_XOR	CRC xor value

19.3.1. Peripheral module clock control register (RCU_EN)

Address offset: 0x04

Reset value: 0x0000 0001

23	22	21	20	19	18	17	16
LED_LCD_CLKEN	GPIO_CLKEN	CRC_CLKEN	ADC_CLKEN	CDC_CLKEN	WDT_CLKEN	TIMER3_CLKEN	TIMER2_CLKEN
RW	RW	RW	RW	RW	RW	RW	RW

21	CRC_CLKEN	CRC module operation enable 1: Work 0: Off, the default is 0

19.3.2. CRC data register (CRC_DR)

Address offset: 0x00

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR[31:16]															
RW															

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

DR[15:0]
RW

31:0	DR[31:0]	This register is used to write new data to the CRC calculation unit Read the register to read the previous CRC calculation result If the data size is less than 32 bits, the lower bit is valid Can be written and read in byte, half-word, and word format CRC data writing and calculation results multiplex this register, the default value of read register is the initial value of CRC calculation

19.3.3. CRC control register (CRC_CR)

Address offset: 0x04

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											CF	REV_OUT	REV_IN		RESET
											R	RW	RW		RS

31:5	-	Reserved
4	CF	CRC calculation flag: When writing the data register, the hardware pulls the flag high, and the hardware clears the flag after the calculation is completed. When the flag is high, new data cannot be written.
3	REV_OUT	It is used to control the inversion of the bit order of the output data, which is performed by the whole word 0: Does not affect the bit order 1: Bit-reversed output format
2:1	REV_IN	Used to control the reversal of the bit order of input data 00: Does not affect the bit order 01: Perform bit reversal by byte 10: Perform bit reversal by halfword 11: Perform bit reversal by word
0	RESET	Set by software, clear by hardware Used to reset the CRC module

19.3.4. CRC polynomial selection (CRC_POL_SEL)

Address offset: 0x08

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

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

31:2	-	Reserved
1:0	POL_SEL	For selecting polynomials: 00: 32-bit polynomial, 0x04C11DB7 01: 16-bit polynomial, 0x1021 10: 8-bit polynomial, 0x07 11: Reserved

19.3.5. CRC initial value (CRC_INIT)

Address offset: 0x0C

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INIT [31:16]															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INIT [15:0]															
RW															

31:0	INIT [31:0]	CRC initial value When 8/16-bit polynomial is selected, the lower bits are valid Initial value cannot be configured during CRC calculation

19.3.6. CRC xor value (CRC_XOR)

Address offset: 0x10

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
XOR [31:16]															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
XOR [15:0]															
RW															

31:0	XOR [31:0]	CRC XOR value When 8/16-bit polynomial is selected, the lower bits are valid XOR value cannot be configured during CRC calculation

19.4. Configuration process

1. Configure and turn on the CRC module clock;
2. Configure related registers (polynomial selection/initial value/exclusive or value/bit inversion);
3. Configure RESET to reset the calculation module;
4. After the read flag bit CF is 0, write the data register CRC_DR to calculate the CRC check result;
5. When the read flag bit CF is 0, the CRC check result can be read out after a delay of one system clock cycle.

Note:

1. The configuration process must be configured in order.
2. When dealing with 8-bit/16-bit data size, the data register CRC_DR should be converted to 8-bit/16-bit.
3. Configure the RESET bit of the CRC_CR register to 1, reset the CRC module, reset the hardware to zero, and the pulse width is one system clock cycle.
4. The polynomial selection register, xor value register, and bit-reversal control register cannot be configured in real time during CRC calculation.

20 Programming and debugging

20.1. SWD debug interface

The BF7807AMXX family uses two pins as shown in the table below. PB12/PB13 is the SWD download and debugging function port by default. When configuring SW communication, the internal pull-up resistor of PB12/PB13 is turned on by default, and the pull-up is valid.

Note: When the PB12/PB13 is configured as a common IO port, it will affect the use of the SWD function. If the BF7807AM64-LJTA/LJTX wants to restore the SWD function, the chip can be reset through the external reset pin. The BF7807AM44-LJTX does not support external reset. It is recommended that PB12/PB13 not be multiplexed as common IO or other functions.

Pin name	Description	Corresponding pin
SWCLK	Clock signal	PB12
SWDIO	Data input/output	PB13

20.2. Register

Base address: 0x500A 0000

Address offset	Register	Description
0x30	SW_IO_EN	SWD port selection enable register

20.2.1. SWD port selection enable register (SW_IO_EN)

Address offset: 0x30

Reset value: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15:1	0
Reserved	SW_SEL
	RW

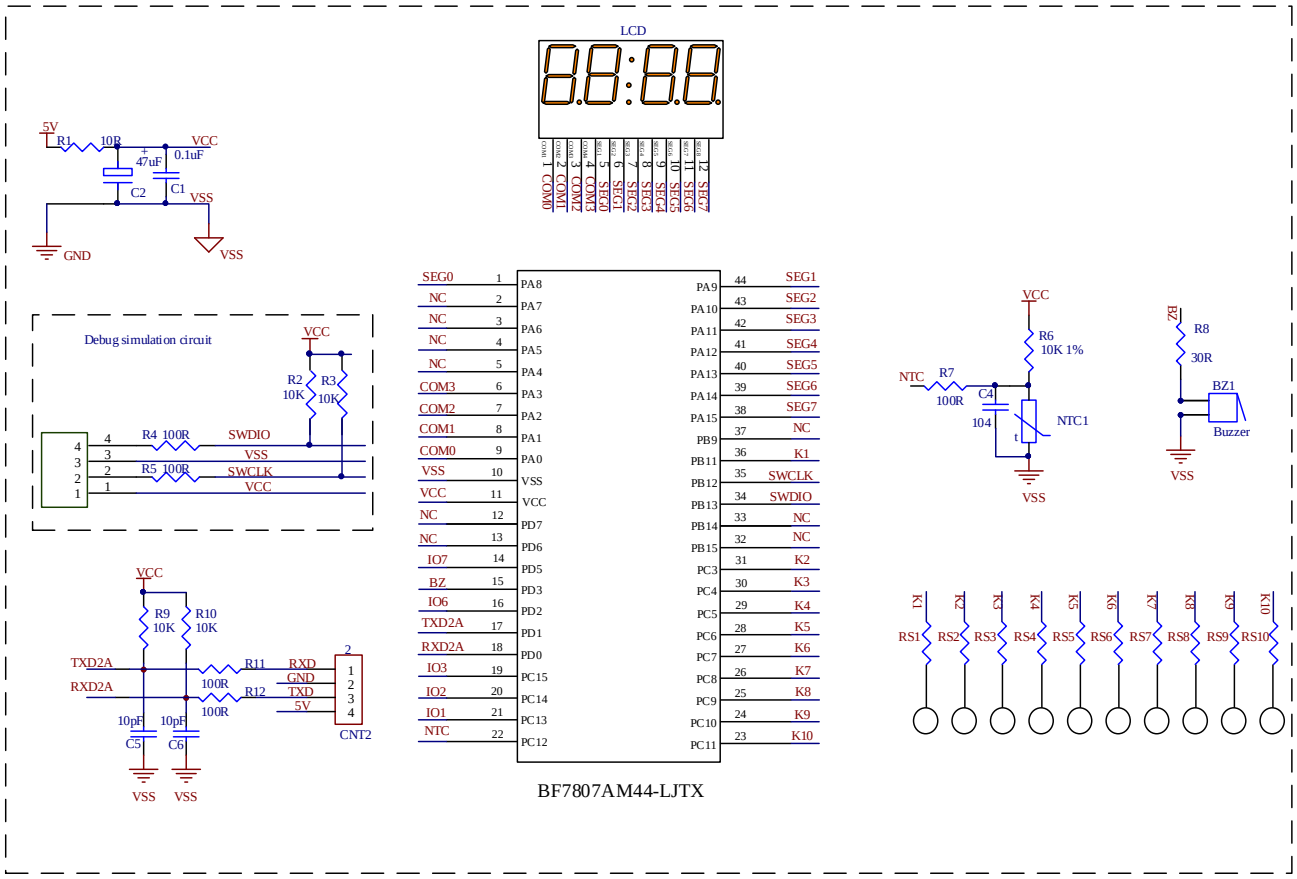
31:1	-	Reserved
0	SW_SEL	SWD port selection enable 1: PB12/PB13 is SWD function 0: PB12/PB13 are GPIO functions

20.3. PGC, PGD burning

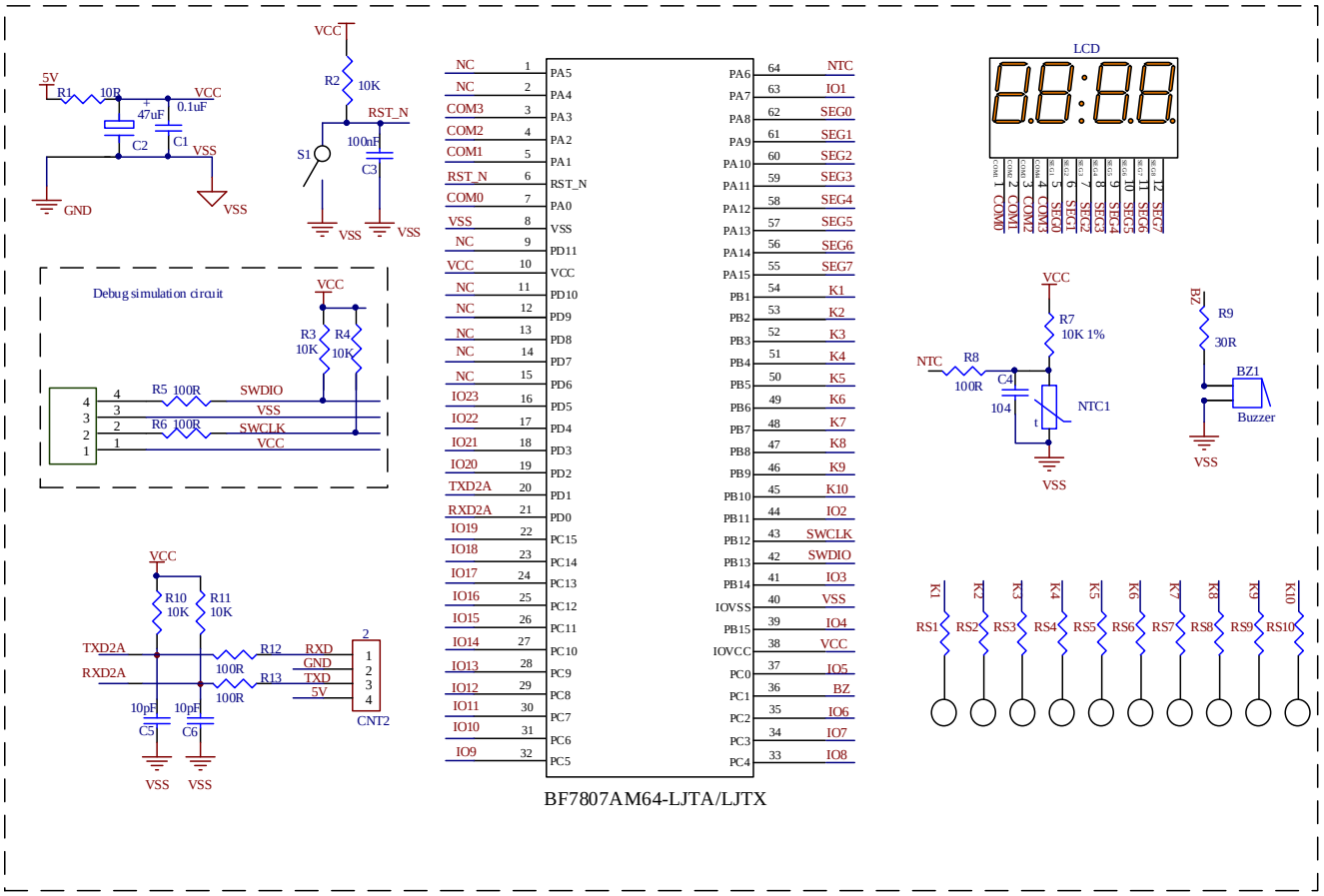
The BF7807AMXX supports PGC0 (PGC1), PGD0 (PGD1) programming, using our dedicated MP100 programming tool. For details, see "BYD MP100 user manual for mass production and burning tool".

21 Reference application circuit

21.1. BF7807AM44-LJTX reference circuit



21.2. BF7807AM64-LJTA/LJTX reference circuit

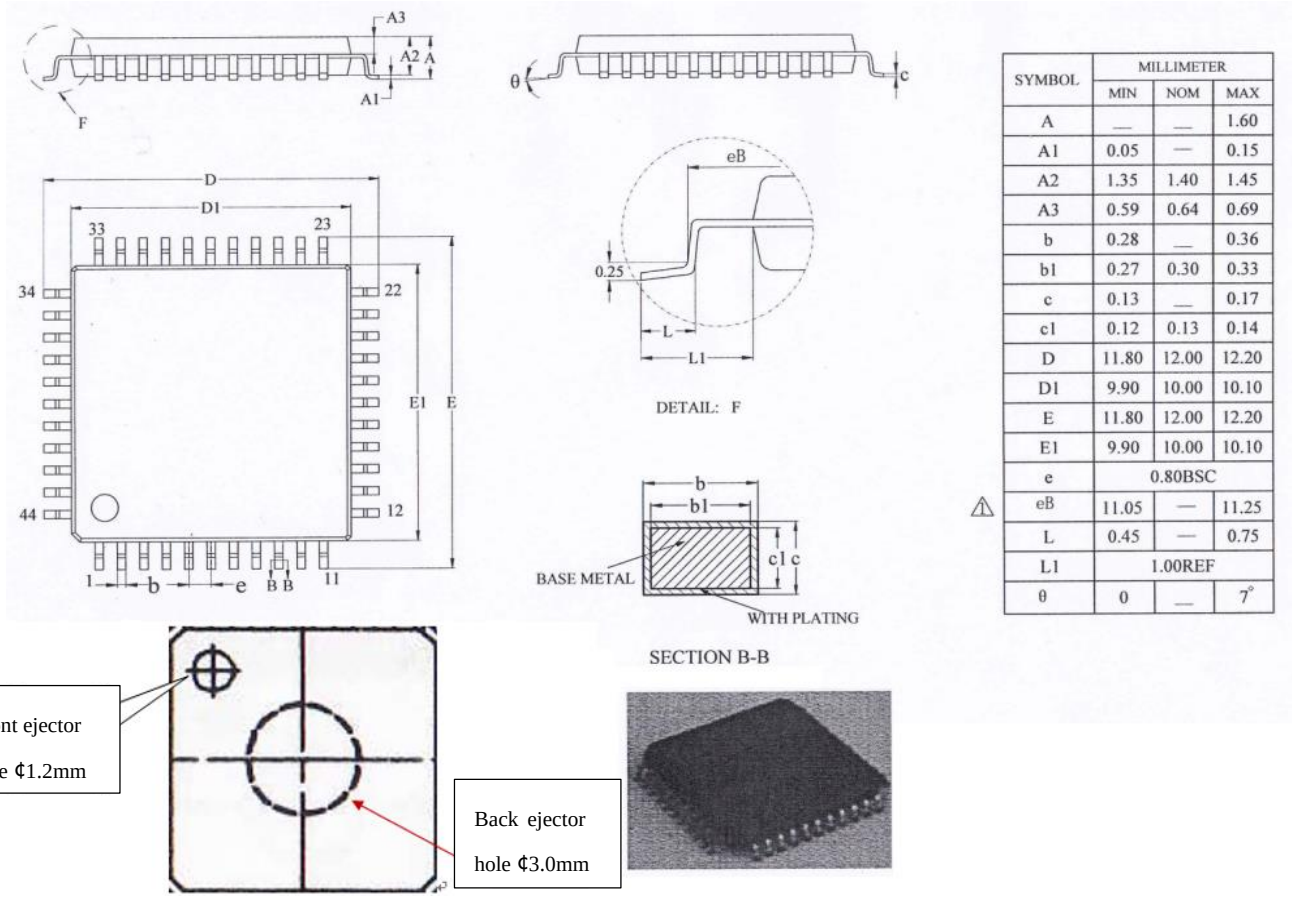


Note: The above reference circuit is for reference design only.

1. **RSX channel resistance is recommended 1k~8.2k, conventional 4.7k.**
2. **Debug the emulation circuit. If there is a pull-up resistor on the emulator or on the adapter board, there is no need to connect the pull-up resistor.**
3. **The resistance on the power supply is recommended to be 0~10 Ω, which can improve the anti-interference ability of the EMS (ESD) test. The resistance on the power supply is replaced with a magnetic bead, and the EMI (RE) test item can increase the test margin. The recommended parameter is 600Ω@100MHz.**

22 Package information

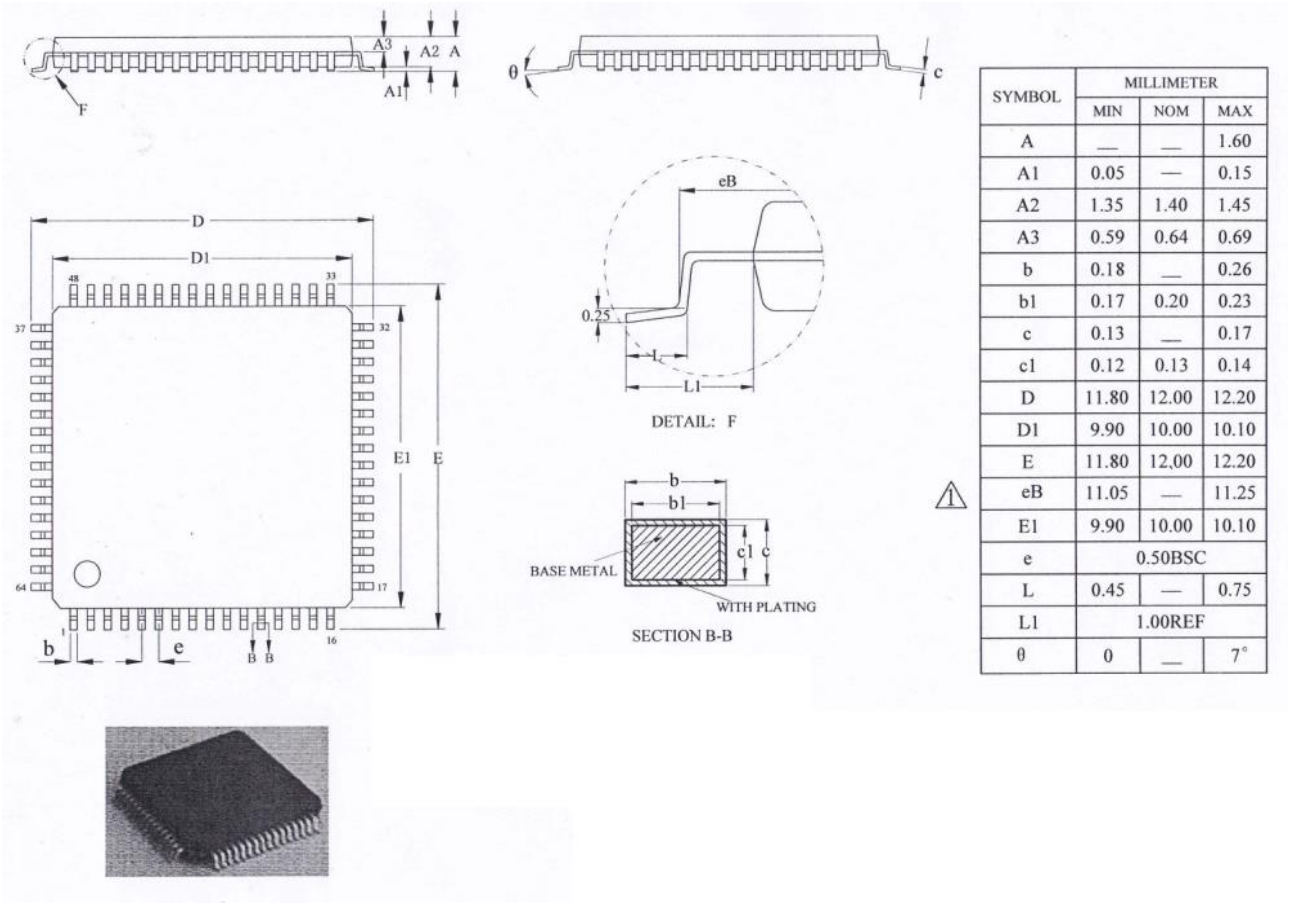
22.1. LQFP44-LJTX



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.28	—	0.36
b1	0.27	0.30	0.33
c	0.13	—	0.17
c1	0.12	0.13	0.14
D	11.80	12.00	12.20
D1	9.90	10.00	10.10
E	11.80	12.00	12.20
E1	9.90	10.00	10.10
e	0.80BSC		
⚠ eB	11.05	—	11.25
L	0.45	—	0.75
L1	1.00REF		
θ	0	—	7°

BF7807AM44-LJTX package infographic

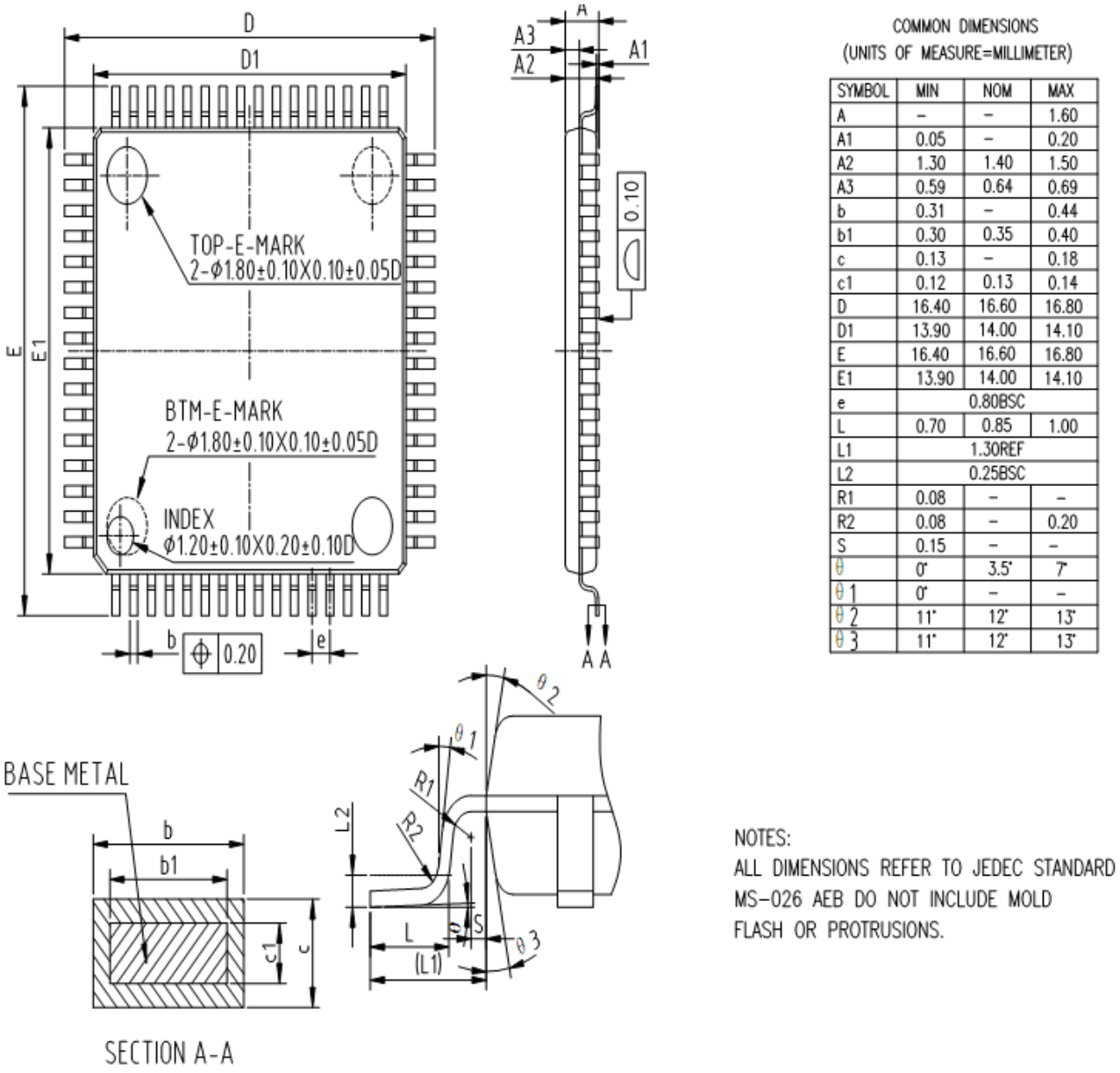
22.2. LQFP64-LJTA



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.18	—	0.26
b1	0.17	0.20	0.23
c	0.13	—	0.17
c1	0.12	0.13	0.14
D	11.80	12.00	12.20
D1	9.90	10.00	10.10
E	11.80	12.00	12.20
eB	11.05	—	11.25
E1	9.90	10.00	10.10
e	0.50BSC		
L	0.45	—	0.75
L1	1.00REF		
θ	0	—	7°

BF7807AM64-LJTA package infographic

22.3. LQFP64-LJTX



COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	–	–	1.60
A1	0.05	–	0.20
A2	1.30	1.40	1.50
A3	0.59	0.64	0.69
b	0.31	–	0.44
b1	0.30	0.35	0.40
c	0.13	–	0.18
c1	0.12	0.13	0.14
D	16.40	16.60	16.80
D1	13.90	14.00	14.10
E	16.40	16.60	16.80
E1	13.90	14.00	14.10
e		0.80BSC	
L	0.70	0.85	1.00
L1		1.30REF	
L2		0.25BSC	
R1	0.08	–	–
R2	0.08	–	0.20
S	0.15	–	–
θ	0°	3.5°	7°
θ1	0°	–	–
θ2	11°	12°	13°
θ3	11°	12°	13°

NOTES:
ALL DIMENSIONS REFER TO JEDEC STANDARD MS-026 AEB DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS.

BF7807AM64-LJTX package infographic

23 Device electronic signature

Electronic signatures are stored in the NVR2 (information block) storage area. The NVR2 needs to be unlocked to read the unique identification code (UID) of the chip. For details, please refer to the "NVR2 read" reference chapter.

- Used as serial number
- Used as a password, when writing flash memory, use this unique identifier in combination with the software encryption and decryption algorithm to improve the security of the code in the flash memory

Under no circumstances can the user modify this identity.

The unique product identifier can be read by byte (8 bits), half word (16 bits) and word (32 bits) according to different usages of users.

Address	UID(96/128 bits)
0x203A0	ID1
	ID2
	ID3
	ID4
0x203A4	ID5
	ID6
	ID7
	ID8
0x203A8	ID9
	ID10
	ID11
	ID12
0x203AC	ID13
	ID14
	ID15
	ID16

Appendix A

Appendix 1: List of register-related abbreviations

Abbreviation	Detailed description
Read/Write (RW)	Software can read and write this bit
Read only (R)	Software can only read this bit
Write only (W)	Software can only write to this bit, and reading this bit will return the reset value
Read/Clear write1 (RC_W1)	Software can read this bit or clear it by writing a 1. Writing 0 has no effect on the value of this bit
Read/Clear write0 (RC_W0)	Software can read this bit or clear it by writing a 0. Writing a 1 has no effect on the value of this bit
Read/Set (RS)	Software can read this bit or set it. Writing 0 has no effect on the value of this bit
Reserved (Res.)	Reserved bit, write operation is prohibited, otherwise it may cause chip exception

Appendix 2: Storage environment and conditions

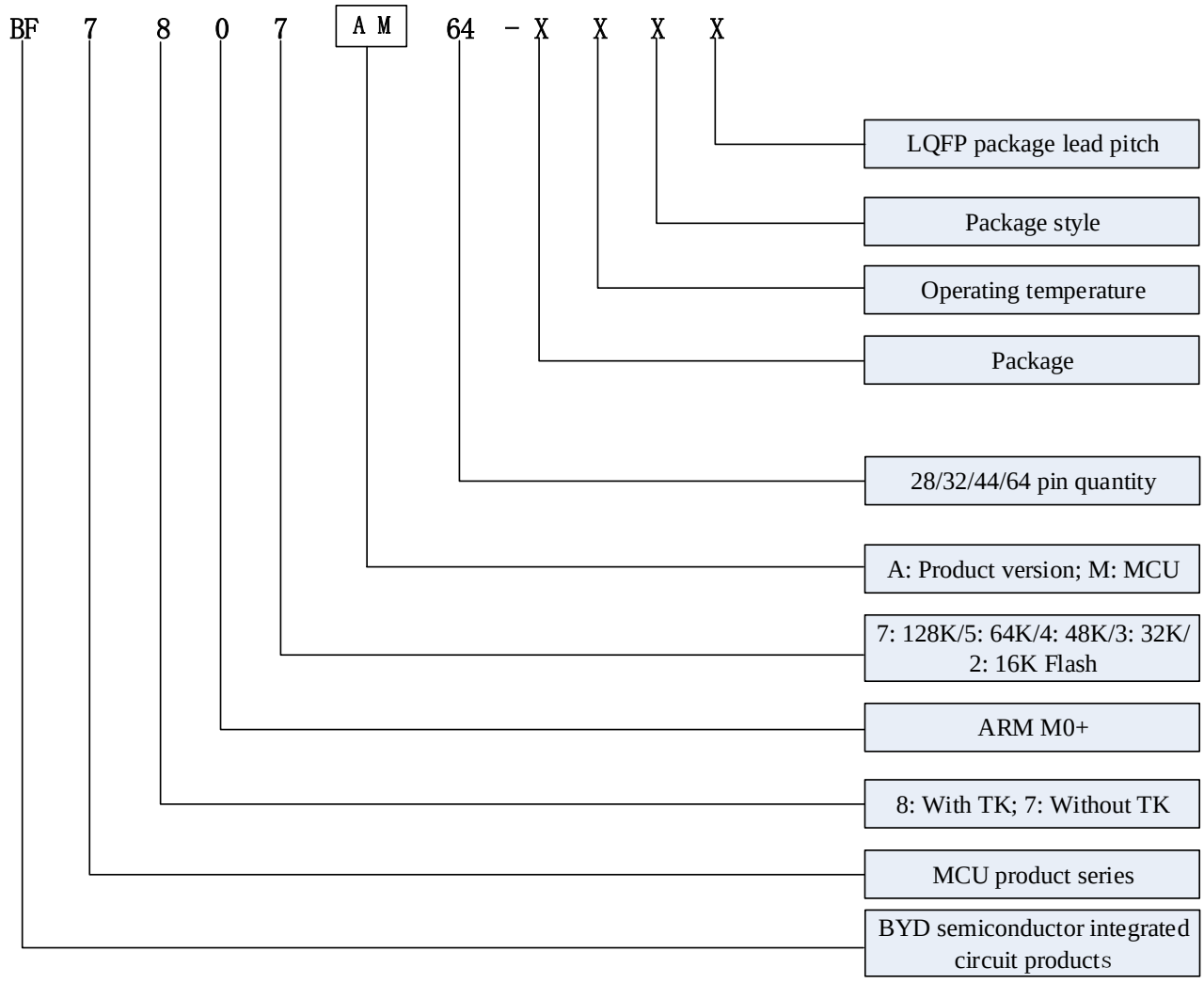
Constant temperature and humidity: Temperature: 15-25 ℃, relative humidity: 30%-60%, 12 months shelf life without vacuum packaging.

MSL3 moisture sensitivity level: Reference standard IPC/JEDEC J-STD-020 production operations.

Order information

Package	Operating temperature		Package style	LQFP package lead pitch
S: SOP	Car grade	A: -40℃~+150℃	B: Taping	A: e=0.50mm
A: SSOP		B: -40℃~+125℃	L: Material pipe	B: e=0.65mm
T: TSSOP		C: -40℃~+105℃	T: Tray	X: e=0.80mm
M: MSSOP		D: -40℃~+85℃	-	-
L: LQFP	Industrial grade	K: -40℃~+85℃	-	-
Q: QFN		J: -40℃~+105℃	-	-
B: BGA		L: -40℃~+125℃	-	-
D: DIP	Consumer grade	P: -25℃~+70℃	-	-
-		Q: 0℃~+70℃	-	-

Example:



Revision record

Revision date	Revised content	Revisionist	Remark
2022-05-18	First edition	ZQ	V1.0

Disclaimer

1. The information in this document can be modified and updated from time to time without notifying the user. See the revision record for details. Please contact FAE or the agent for the latest version.
2. The BYD Semiconductor Co., Ltd. will do its utmost to ensure the high quality and high stability of the company's products. Nevertheless, 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 conditions. When using our 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 components, 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, measurement tools, industrial machinery, household appliances, etc.). This company's product cannot and is prohibited from being applied to some special equipment that requires extremely high stability and quality, so as to avoid accidents such as casualties. Products that cannot be used include nuclear energy control equipment, aircraft and aviation devices, transportation equipment, traffic signal equipment, combustion control equipment, medical equipment, and all safety equipment, etc. The company is not responsible for the losses and injuries caused by users in the non-product applications listed above.