RN8318_RN8615_RN8613_RN8611 V2
User Manual
Rev1.1

Revision History

Version	Author	Date	Reasons for changes	Major Changes
V1.0	Renergy	2023-02-14	First release	NA
V1.1	Renergy	2023-05-05	Modification	1. Added note on page 8 about limitation for external voltage in VBAT domain; 2. Added DSP module description on page 15; 3. Updated SPI pins description on page 23; 4. Added IIC clock frequency settings chart on page 165; 5. Added note on page 146 about baud rate of UART and 38K infrared modulation when system clock running under 32.768KHz.

CONTENTS

1 Overview

1.1 Introduction

The MCU chips RN8318 V2, RN8613 V2, RN8615V2, and RN8611 V2 are the second generation of MCU chips developed by Renergy, with improved performance and expanded functionality based on the first generation products, while also maintaining backward compatibility with the first generation products.

RN8318 V2： The package is LQFP128L and is pin-to-pin compatible with V1, with register compatibility. However, software library functions need to be upgraded to V2 version. The SRAM is expanded to 96KB+4KB, with optimized power consumption and RTC performance. The device also includes hardware acceleration units for ECC, AES, HASH, and TRNG. It also incorporates three SPI and SPI/UART DMA channels. This MCU is typically used in the applications of the National Grid 21st Edition three-phase meter and overseas three-phase meters.

RN8615 V2: Packaged in LQFP100L, it is pin-to-pin compatible with V1 and register-compatible. Software library functions need to be upgraded to V2 version. The SRAM is expanded to 96KB+4KB, with optimized power consumption and RTC performance. The device also includes hardware acceleration units for ECC, AES, HASH, and TRNG. It also incorporates three-way SPI and SPI/UART DMA channels. This MCU is typically used in overseas single-phase utility meters.

RN8613 V2: Packaged in LQFP100L, it is pin-to-pin compatible with V1 and register-compatible. Software library functions need to be upgraded to V2 version. The FLASH is expanded to 512KB and the SRAM is expanded to 48KB, with optimized power consumption and RTC performance. The device also includes three-way SPI and SPI/UART DMA channels. This MCU is typically used in the National Grid 21st Edition single-phase meter.

RN8611 V2: Packaged in LQFP64L, this is a new model with a FLASH capacity of 512KB and a SRAM capacity of 96KB. This MCU is typically used in a single-phase IoT meter separation solution, as well as terminal data acquisition boards.

1.2 Features

Fundamental Features:

- Highly integrated: 32bit ARM M0 CPU+maximum 512KB Flash/32KB SRAM+independent power supply hardware temperature compensation RTC+LCD controller.
- Wide voltage range: Under typical conditions, voltage range of 2.3V~5.5V ensures the normal operation of CPU;
  Under typical conditions, voltage range of 2.7V~5.5V ensures the accuracy of RTC.
  Under typical conditions, voltage range of 1.8V~5.5V is maintained for the perpetual calendar time.
- High performance:

✓ Under the condition of using 32.768KHz crystal oscillator and 32.768KHz crystal oscillator with an external high-frequency crystal oscillator, the CPU can work at a maximum frequency of 29.4912MHz (32.768KHz, 1.8432MHz, 7.3728MHz, 14.7456MHz, and 29.4912MHz are optional).

✓ The internal high-frequency RCH can work at a maximum of 29.5MHz, and its accuracy is guaranteed within ±1% over the full temperature range. It can be used as a backup clock.

- Low power:
  - ✓ The power consumption of CPU subsystem at 32KHz is better than 18μA (with cache);
  - ✓ The total power consumption of the chip in sleep mode is about 7μA (RTC automatic temperature compensation; Ram to keep; CPU and digital peripherals do not lose power; Interrupt wake up);
  - ✓ The typical power consumption in VBAT domain is about 1.8μA.

  - ✓ The power consumption of LCD is about 5μA, and the constant chip's is 13uA (resistance series voltage division mode, including 6COM display power consumption);
  - ✓ The power consumption of LCD is about 10μA, and the constant chip's is 18uA (charge pump mode, including 8COM display power consumption).
- High precison:
  - ✓ The RTC has a second pulse error of ＜±5ppm over the full temperature range (-40°C~85°C), and the minimum calibration scale is 0.0339ppm.
- Processor-related:
  - ✓ ARM Cortex-M0 core;
  - ✓ RN8318/RN8615/RN8613/RN8611：512KB FLASH memory;
  - ✓ RN8318/RN8615：96KB+4KB SRAM；
  - ✓ RN8613：48KB SRAM；
  - ✓ RN8611：96KB SRAM；
- Single-cycle multiplier(32bit*32bit);
- System timer is embedded in the CM0;
- Support for external interrupts and other ways to wake up;
- Complete hardware and software for the integrated development environment is provided.
- DSP：
  - ✓ Conversion between integers and floating-point numbers;
  - ✓ Floating-point addition, subtraction, multiplication, and division;
  - ✓ Single butterfly operation (for complex numbers) and continuous butterfly operation with DMA;
  - ✓ The entire process of radix-2 FFT, with point numbers of 64, 128, 256, 512, and 1024;
  - ✓ Automatic data movement operation for bit-reversal, with point numbers of 4, 8, 16, 32, 64, 128, 256, 512, and 1024;
  - ✓ Sine and cosine calculation;
  - ✓ Root-mean-square calculation;
  - ✓ Arctangent calculation;
  - ✓ Single IIR calculation and IIR calculation with DMA;
  - ✓ FIR filtering operation;
  - ✓ Linear interpolation;
  - ✓ Lagrange interpolation;

- RTC：
  - ✓ Hardware's automatic temperature compensation, precision and power consumption meet the national standard, and CPU is not required to participate in the temperature compensation;
  - ✓ Temperature Sensor: Provide accurate temperature, temperature measurement accuracy is ±1℃ in the range of -25℃ ~70℃;
  - ✓ RTC perpetual calendar and automatic temperature compensation circuit are powered by VBAT pin independently;
  - ✓ The typical power consumption in VBAT domain is better than 1.8μA;
  - ✓ The opening temperature compensation time is about 2ms, and the typical power consumption of temperature is 250μA. The average power consumption is 0.016μA when it is opened once in 30 seconds.

- LCD：

  - ✓ RN8318：4*40、6*38、8*36；
  - ✓ RN8615/RN8613：4*34、6*32、8*30；
  - ✓ Supports ChargePUMP and internal resistor column voltage divider modes, both of which are hardware compatible. If lower display power consumption is required, the resistor string voltage divider mode can be selected.
  - ✓ The power sonxumption of LCD module is better than 5μA (resistor series voltage division scheme).
- D2F Energy Integration Unit: Provides 12 D2F integrators, of which 3 support pulse output.
- M2M Memory Transfer Unit: 1 M2M module, which can realize memory data transfer.
- Encryption: (Not compatible with V1 version, application program needs to be modified)
  - ✓ Hardware true random number generator meets U.S. NIST FIPS140-2 standard;
  - ✓ AES hardware acceleration unit;
  - ✓ ECC hardware acceleration unit;
  - ✓ RSA hardware acceleration unit;
  - ✓ HASH hash algorithm hardware acceleration unit;
  - ✓ RN8613 and RN8611 do not support encryption;
  - ✓ Encryption documentation refers to Renergy application notes.

Other peripherals：

- High speed GPIO, supports interface of peripheral devices with different voltage:
  - ✓ RN8318：108 IO.
  - ✓ RN8615/RN8613：83 IO.
  - ✓ RN8611：50 IO.
- 12bit ADC: Temperature sensor / battery voltage detection / generic ADC time division multiplexing;
- Voltage detecting LVD: chip supply voltage detection; external voltage detection.
- Two low power comparators CMP: external voltage detection.
- Extended timer: two 32bit timers, four simple 16bit timers.
- UART: 6 sets, automatic baud rate, infrared modulation under 32.768kHz frequency, UART wakes up, level reversal, DMA available.
- 7816 ports：2.
- I2C：1.
- SPI：4 sets, DMA available.
- Hardware watchdog.
- Button Interrupts: 8, pin multiplexing.
- External interrupts: 8, pin multiplexing.

1.3 Renergy MCU model division

Table 1-1 MCU Product Model List

MCU series	FLASH	RAM	Encrypting	Packaging
RN8318	512KB	96KB+4KB	√	LQFP128L
RN8615	512KB	96KB+4KB	√	LQFP100L
RN8613	512KB	48KB	×	LQFP100L
RN8611	512KB	96KB	×	LQFP64L

1.4 MCU Product Resource Comparison Table

Table 1-2 MCU Product Resource Comparison

Model		RN8318	RN8615	RN8613	RN8611
CPU		Cortex-M0			
Typical Freq.		14.7456M			
Max Freq.		29.4912M			
FLASH		512KB			
RAM		96KB+4KB		48KB	96KB
SEA Encryption		√	√	-	-
TRNG		√	√	-	-
Timers	32bit Timer	2	2	2	2
	16bit Timer	4	4	4	4
	systick	1	1	1	1
RTC		1	1	1	1
WDT		1	1	1	1
KBI		8	4	4	3
INTC		8	4	4	5
SPI		4	4	4	4
UART		6	6	6	6
I2C		1	1	1	1
7816		2	2	2	2
GPIO		108	83	83	50
DMA		√			
D2F		√			
M2M		√			
LCD		4*40/6*38/8*36	4*34/6*32/8*30		-
LCD booster		√	√	√	-
CMP		2	2	2	1
LVD		1	1	1	-
SAR-ADC		7	7	7	2
TempSensor		√			

1.5 Pin Configuration

1.5.1 RN8318 LQFP128L Pin Configuration

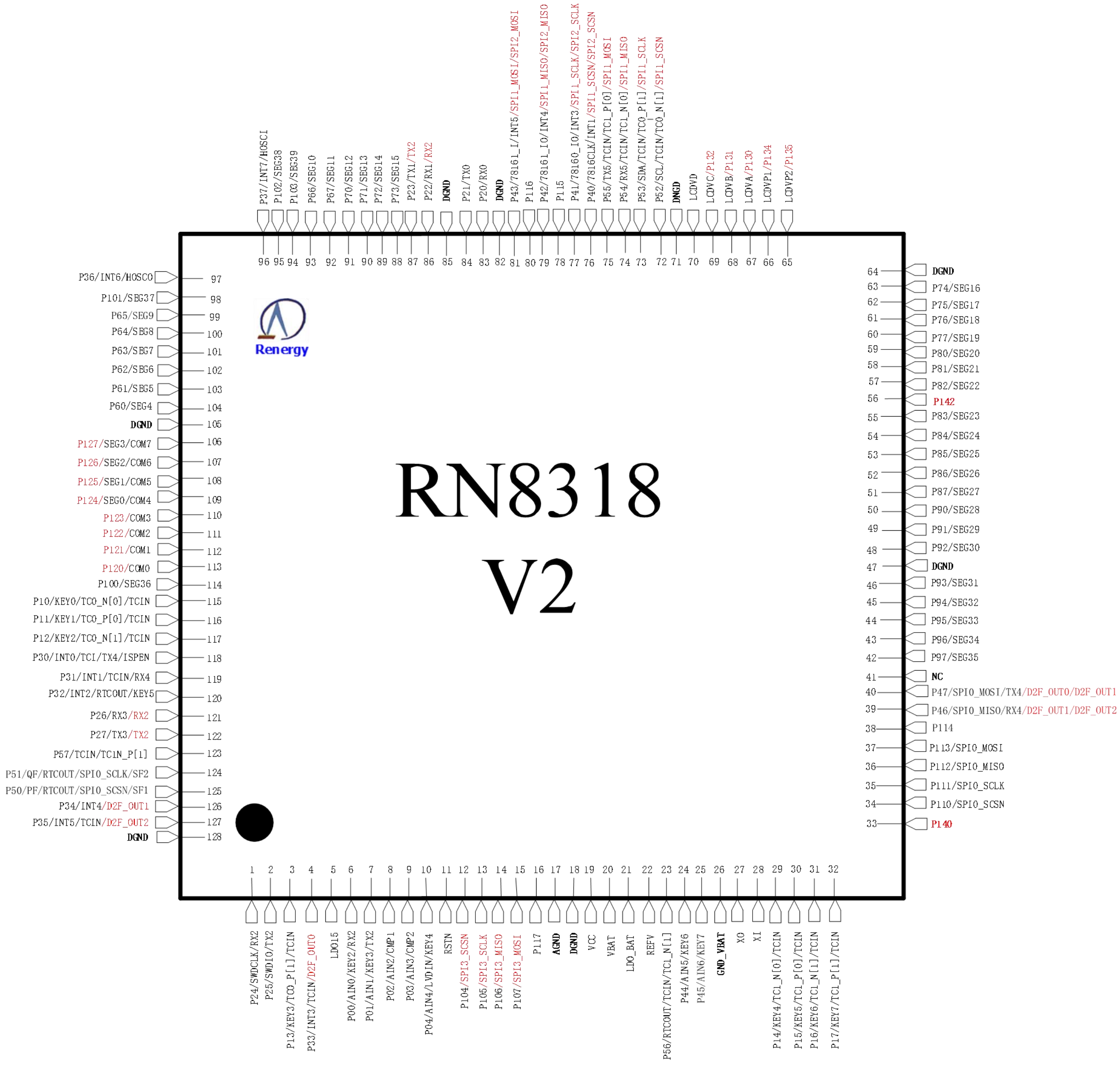


Fig 1.1. RN8318 Pin Configuration

Note:

1. VBAT domain pins: VBAT, LDO_VBAT, REFV, P56, P44, P45, GND_VBAT, XO and XI.
2. P56/P44/P45 output a high level equal to the VBAT voltage.
3. The external voltage applied to the P56/P44/P45 pins must not exceed the VBAT voltage.
4. The pin in red is the pin that is different from RN8318 V1 and RN8312 V2.

1.5.2 RN8613/8615 LQFP100L Pin Configuration

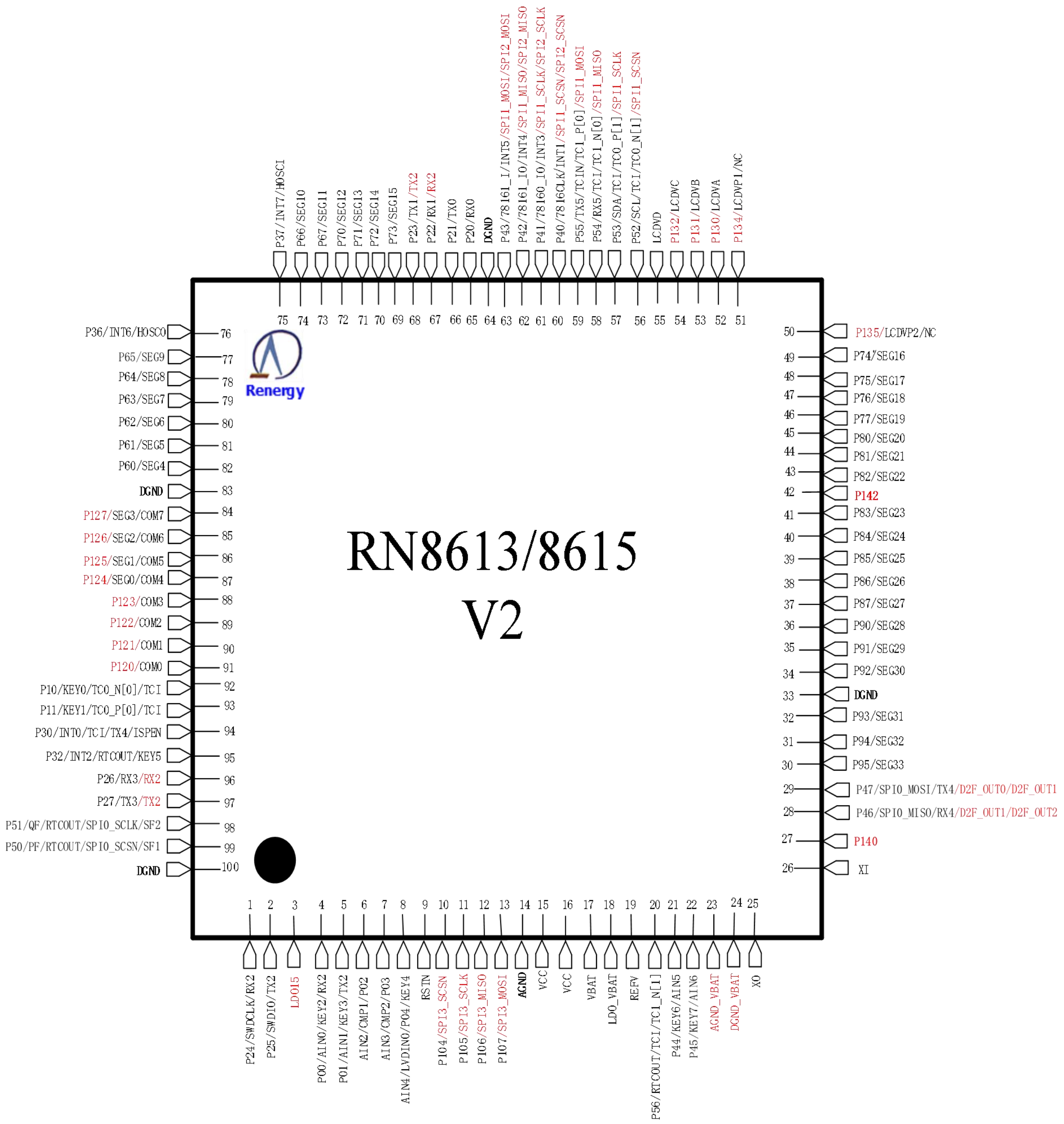


Fig 1.2. RN8613/RN8615 Pin Configuration

Note:

1. VBAT domain pins: VBAT, LDO_VBAT, REFV, P56, P44, P45, AGND_VBAT, DGND_VBAT, XO and XI.
2. Note that the pin P56/P44/P45 output a high level equal to the VBAT voltage.
3. Note that the external voltage applied to the P56/P44/P45 pins must not exceed the VBAT voltage.
4. The pin in red is the pin that is different from RN8613/8615 V1 and RN8613/8615 V2.

1.5.3 RN8611 LQFP64L Pin Configuration

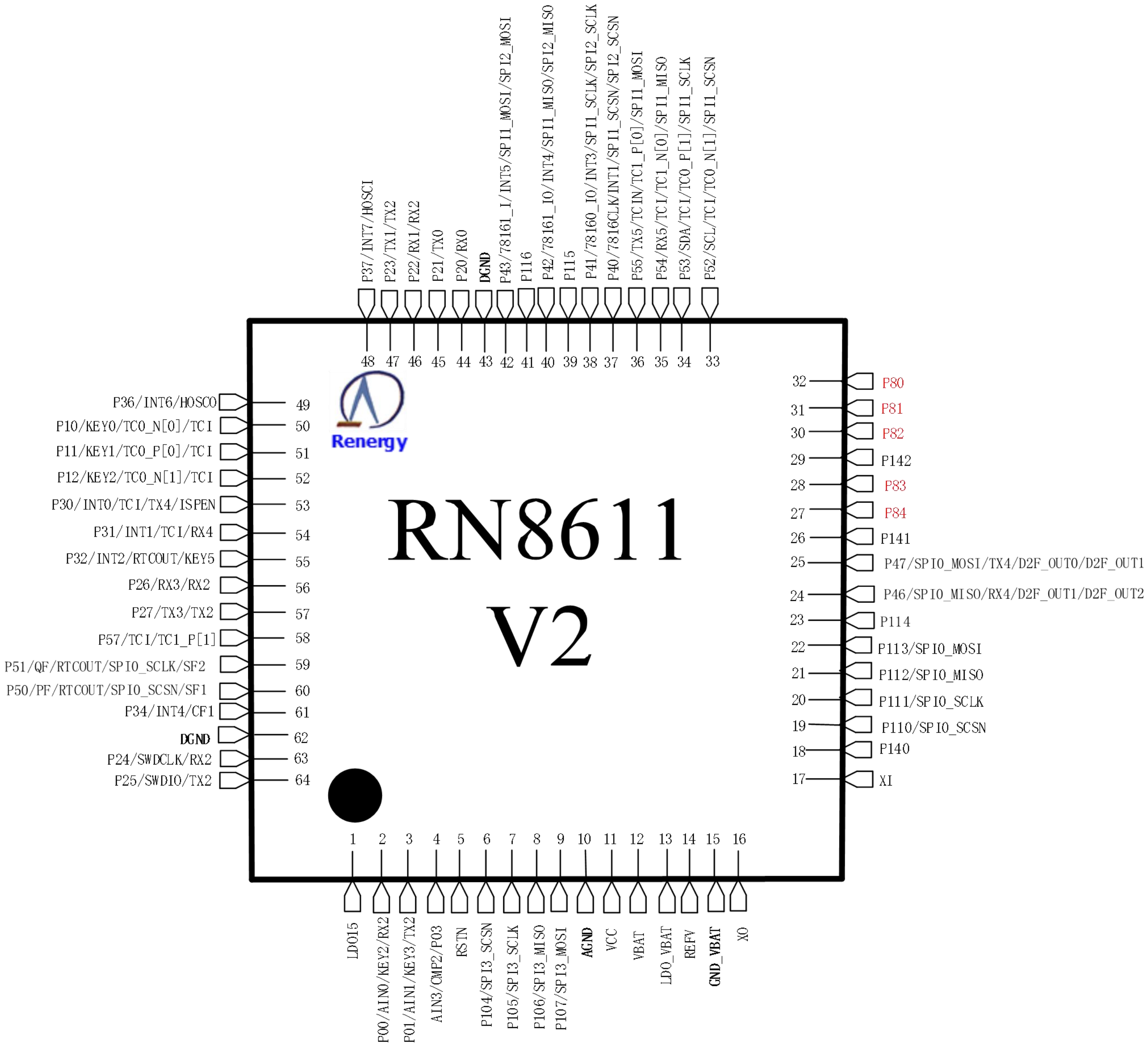


Fig 1.3. RN8611 Pin Configuration

Note:

1. The VBAT domain pins include: VBAT, LDO_VBAT, REFV, GND_VBAT, XO and XI.

2. Red pins P80, P81, P82, P83, and P84 are all of open-drain structure and require external pull-up resistors when used as outputs.

1.5.4 Pin Category

Table 1-3 Pin Category

	Cell Type				Input Option				Osc Option	Lcd Option	Output Option	
Pin Type	A	B	I	O	U	D	S	L	X	G	D	Load Capability
PABULD3	√	√			√			√			√	3mA
PABUS3	√	√			√		√					3mA
PBDS3		√				√	√				Normally on	3mA
PBDSG3		√				√	√			√	Normally on	3mA
PBULD3		√			√			√			√	3mA
PBULD6		√			√			√			√	6mA
PBUS6		√			√		√					6mA
PIU			√		√							
PIUX			√		√				√			

Abbreviation:

A – Analog; B – Bipolar; I – Input; O – Output; U – Pull up; D – Pull Down; S – Schmitt; L – TTL; X – Crystal Oscillator; G – SEG; D – Open Drain;

1.5.5 Pin Definition

Table 1-4 Pin Definition

Pin Number			Pin Function	Type	Descriptions
RN8318	RN8615 RN8613	RN8611			
LQFP128L	LQFP100L	LQFP64L			
1	1	63	P24	PBULD3	GPIO
			SWDCLK		SWD clock
			RX2		UART2 receiver
2	2	64	P25	PBULD3	GPIO
			SWDIO		SWD data IO
			TX2		UART2 transmitter
3			P13	PBULD3	GPIO
			KEY3		Key 3 input
			TC0_P[1]		Positive output of Channel 1, Timer 0
			TCIN		Timer input
4			P33	PBUS6	GPIO
			INT3		External interrupt input 3
			TCIN		Timer input
			D2F_OUT0		Output of power integral module D2F0
5	3	1	LDO15	Power Source	Output of internal LDO (V1:1.8V, V2:1.5V); required external filtering

					capacitors (0.1uF in parallel with 1 uF);
6	4	2	P00	PABUS 3	GPIO
			AIN0		Input of SAR-ADC external channel 0
			KEY2		Key 2 input
			RX2		UART2 receiver
7	5	3	P01	PABUS 3	GPIO
			AIN1		Input of SAR-ADC external channel 1
			KEY3		Key 3 input
			TX2		UART2 transmitter
8	6		P02	PABUS 3	GPIO
			AIN2		Input of SAR-ADC external channel 2
			CMP1		Input of comparator 1
9	7	4	P03	PABUS 3	GPIO
			AIN3		Input of SAR-ADC external channel 3
			CMP2		Input of comparator 2
10	8		P04	PABUS 3	GPIO
			AIN4		Input of SAR-ADC external channel 4
			LVDIN		Input of low voltage detection
			KEY4		Key 4 input
11	9	5	RSTN	PIU	Input of PIN reset
12	10	6	P104	PBUS6	GPIO
			SPI3_SCSN		SPI3 chip select
13	11	7	P105	PBUS6	GPIO
			SPI3_SCLK		SPI3 clock
14	12	8	P106	PBUS6	GPIO
			SPI3_MISO		SPI3 master input & slaver output
15	13	9	P107	PBUS6	GPIO
			SPI3_MOSI		SPI3 master output & slaver input
16			P117	PBULD 3	GPIO
17	14	10	AGND	Ground	Analog Ground
18			DGND	Ground	Digital Ground
	15		VCC	Power	Main power source input
19	16	11	VCC	Power	Main power source input, required a 4.7uf capacitor in parallel with a 0.1uf capacitor for decoupling
20	17	12	VBAT	Power	3.6V power source，required ex-ternal RC filtering; R/C：10Ω/1uF
21	18	13	LDO_VBAT	Power	Output of internal LDO in VBAT domain (V1:2.0V,V2:1.8V); required external filtering capacitors of 0.22uF

22	19	14	REFV	Power	Reference input of SARADC
23	20		P56	PBULD3	GPIO Input voltage should not exceed VBAT power supply voltage as input pin；Output voltage is equal to VBAT power supply voltage as output pin.
			RTCOUT		RTCOUT output
			TC1_N[1]		Negative output of Channel 1, Timer 1
			TCIN		Timer input
24	21		P44	PABULD3	GPIO Input voltage should not exceed VBAT power supply voltage as input pin；Output voltage is equal to VBAT power supply voltage as output pin.
			AIN5		Input of SAR-ADC external channel 5; input voltage should not exceed VBAT power supply voltage. A VBAT voltage higher than 3.6V would be reqired when the input voltage of AIN5 is equal to 3.6V.
			KEY6		Key 6 input
25	22		P45	PABULD3	GPIO Input voltage should not exceed VBAT power supply voltage as input pin；Output voltage is equal to VBAT power supply voltage as output pin.
			AIN6		Input of SAR-ADC external channel 6; input voltage should not exceed VBAT power supply voltage. A VBAT voltage higher than 3.6V would be reqired when the input voltage of AIN6 is equal to 3.6V.
			KEY7		Key 7 input
26	23	15	AGND_VBAT	Ground	Analog ground in VBAT domain
	24		DGND_VBAT	Ground	Digital ground in VBAT domain
27	25	16	XO	Clock	32.768KHz-no source crystal oscillator output
28	26	17	XI	Clock	32.768KHz-no source crystal oscillator input; No need for external resistor and capacitor between XO and XI. The crystal oscillator should be isolated by ground. Recommend to use crystal oscillator of which the capacitance is 12.5pF. The high-speed flipping function of the IO pins near XO and XI are not avaiable.
29			P14	PBULD	GPIO

			KEY4	3	Key 4 Input
			TC1_N[0]		Negative output of Channel 0, Timer 1
			TCIN		Timer Input
30			P15	PBULD 3	GPIO
			KEY5		Key 5 Input
			TC1_P[0]		Positive output of Channel 0, Timer 1
			TCIN		Timer Input
31			P16	PBULD 3	GPIO
			KEY6		Key 6 Input
			TC1_N[1]		Negative output of Channel 1, Timer 1
			TCIN		Timer Input
32			P17	PBULD 3	GPIO
			KEY7		Key 7 Input
			TC1_P[1]		Positive output of Channel 1, Timer 1
			TCIN		Timer Input
33	27	18	P140	PBULD 3	Digital Ground for Version 1； GPIO for Version 2, can be grounded.
34		19	P110	PBULD 3	GPIO
			SPI0_SCSN		SPI0 Chip Select
35		20	P111	PBULD 3	GPIO
			SPI0_SCLK		SPI0 Clock
36		21	P112	PBULD 3	GPIO
			SPI0_MISO		SPI0 master input & slaver output
37		22	P113	PBULD 3	GPIO
			SPI0_MOSI		SPI0 master output & slaver input
38		23	P114	PBULD 3	GPIO
39	28	24	P46	PBULD 6	GPIO
			SPI0_MISO		SPI0 master input & slaver output
			RX4		UART4 receiver
			D2F_OUT1		Output of D2F1, source drive capability: 3mA for Version 1， 6mA for Version 2
			D2F_OUT2		Output of D2F2, source drive capability: 3mA for Version 1， 6mA for Version 2
40	29	25	P47	PBULD 6	GPIO
			SPI0_MOSI		SPI0 master output & slaver input
			TX4		UART4 transmitter
			D2F_OUT0		Output of D2F0, source drive capability: 3mA for Version 1， 6mA for Version 2
			D2F_OUT1		Output of D2F1, source drive capability: 3mA for Version 1， 6mA for Version 2
41			NC	-	Not connected
42			P97	PBDSG	GPIO

			SEG35	3	SEG pin for driving LCD
43			P96	PBDSG3	GPIO
			SEG34		SEG pin for driving LCD
44	30		P95	PBDSG3	GPIO
			SEG33		SEG pin for driving LCD
45	31		P94	PBDSG3	GPIO
			SEG32		SEG pin for driving LCD
46	32		P93	PBDSG3	GPIO
			SEG31		SEG pin for driving LCD
		26	P141	PBULD3	GPIO
47	33		DGND	Ground	Digital ground
48	34		P92	PBDSG3	GPIO
			SEG30		SEG pin for driving LCD
49	35		P91	PBDSG3	GPIO
			SEG29		SEG pin for driving LCD
50	36		P90	PBDSG3	GPIO
			SEG28		SEG pin for driving LCD
51	37		P87	PBDSG3	GPIO
			SEG27		SEG pin for driving LCD
52	38		P86	PBDSG3	GPIO
			SEG26		SEG pin for driving LCD
53	39		P85	PBDSG3	GPIO
			SEG25		SEG pin for driving LCD
54	40	27	P84	PBDSG3	GPIO
			SEG24		SEG pin for driving LCD
55	41	28	P83	PBDSG3	GPIO
			SEG23		SEG pin for driving LCD
56	42	29	P142	PBULD3	Digital Ground for Version 1； GPIO for Version 2, can be grounded.
57	43	30	P82	PBDSG3	GPIO
			SEG22		SEG pin for driving LCD
58	44	31	P81	PBDSG3	GPIO
			SEG21		SEG pin for driving LCD
59	45	32	P80	PBDSG3	GPIO
			SEG20		SEG pin for driving LCD
60	46		P77	PBDSG3	GPIO
			SEG19		SEG pin for driving LCD
61	47		P76	PBDSG3	GPIO
			SEG18		SEG pin for driving LCD
62	48		P75	PBDSG3	GPIO
			SEG17		SEG pin for driving LCD
63	49		P74	PBDSG	GPIO

			SEG16	3	SEG pin for driving LCD
64			DGND	Ground	Digital ground
65	50		P135	PBDS3	GPIO for Version 2, not available for Version 1
			LCDVP2		Analog output， it requires a capacitor of 100nF between LCDVP1 and LCDVP2. It can be floating for resistor string mode.
66	51		P134	PBDS3	GPIO for Version 2, not available for Version 1
			LCDVP1		See LCDVP2
67	52		P130	PBDS3	GPIO for Version 2, not available for Version 1
			LCDVA		LCD voltage output， required an external capacitor of 470nF
68	53		P131	PBDS3	GPIO for Version 2, not available for Version 1
			LCDVB		LCD voltage output， required an external capacitor of 470nF
69	54		P132	PBDS3	GPIO for Version 2, not available for Version 1
			LCDVC		LCD voltage output， required an external capacitor of 470nF
70	55		LCDVD	PBDS3	LCD voltage output， required an external capacitor of 470nF
71			DGND	Ground	Digital ground
72	56	33	P52	PBULD 3	GPIO
			SCL		I2C clock
			TC0_N[1]		Negative output of Channel 1, Timer 0
			TCIN		Timer input
			SPI1_SCSN		SPI1 chip select
73	57	34	P53	PBULD 3	GPIO
			SDA		I2C data
			TC0_P[1]		Positive output of Channel 1, Timer 0
			TCIN		Timer input
			SPI1_SCLK		SPI1 clock
74	58	35	P54	PBULD 3	GPIO
			RX5		UART5 receiver
			TC1_N[0]		Negative output of Channel 0, Timer 0
			TCIN		Timer input
			SPI1_MISO		SPI1 master input & slaver output
75	59	36	P55	PBULD 3	GPIO
			TX5		UART5 transmitter
			TC0_P[0]		Positive output of Channel 0, Timer 0

			SPI1_MOSI		SPI1 master output & slaver input
			TCIN		Timer input
76	60	37	P40	PBULD3	GPIO
			7816CLK		7816 clock output
			INT1		External interruption 1 input
			SPI1_SCSN		SPI1 chip select
			SPI2_SCSN		SPI2 chip select
77	61	38	P41	PBULD3	GPIO
			78160_IO		78160 input/output
			INT3		External interruption 3 input
			SPI1_SCLK		SPI1 clock
			SPI2_SCLK		SPI2 clock
78		39	P115	PBULD3	GPIO
79	62	40	P42	PBULD3	GPIO
			78161_IO		78161 input/output
			INT4		External interruption 4 input
			SPI1_MISO		SPI1 master input & slaver output
			SPI2_MISO		SPI2 master input & slaver output
80		41	P116	PBULD3	GPIO
81	63	42	P43	PBULD3	GPIO
			78161_I		78161 input
			INT5		External interruption 5 input
			SPI1_MOSI		SPI1 master output & slaver input
			SPI2_MOSI		SPI2 master output & slaver input
82	64	43	DGND	Ground	Digital ground
83	65	44	P20	PBULD3	GPIO
			RX0		UART0 receiver
84	66	45	P21	PBULD3	GPIO
			TX0		UART0 transmmiter
85			DGND	Ground	Digital ground
86	67	46	P22	PBULD3	GPIO
			RX1		UART1 receiver
			RX2		UART2 receiver
87	68	47	P23	PBULD3	GPIO
			TX1		UART1 transmmiter
			TX2		UART2 transmmiter
88	69		P73	PBDSG3	GPIO
			SEG15		SEG pin for driving LCD
89	70		P72	PBDSG3	GPIO
			SEG14		SEG pin for driving LCD
90	71		P71	PBDSG	GPIO

			SEG13	3	SEG pin for driving LCD
91	72		P70	PBDSG3	GPIO
			SEG12		SEG pin for driving LCD
92	73		P67	PBDSG3	GPIO
			SEG11		SEG pin for driving LCD
93	74		P66	PBDSG3	GPIO
			SEG10		SEG pin for driving LCD
94			P103	PBDSG3	GPIO
			SEG39		SEG pin for driving LCD
95			P102	PBDSG3	GPIO
			SEG38		SEG pin for driving LCD
96	75	48	P37	PIUX	GPIO
			INT7		External interruption 7 input
			HOSCI		High-frequency crystal oscillator input, required a resistor of 10MΩ between HOSCO and HOSCI, and a capacitor of 15pF in parallel to ground.
97	76	49	P36	PIUX	GPIO
			INT6		External interruption 6 input
			HOSCO		High-frequency crystal oscillator input, required a resistor of 10MΩ between HOSCO and HOSCI, and a capacitor of 15pF in parallel to ground.
98			P101	PBDSG3	GPIO
			SEG37		SEG pin for driving LCD
99	77		P65	PBDSG3	GPIO
			SEG9		SEG pin for driving LCD
100	78		P64	PBDSG3	GPIO
			SEG8		SEG pin for driving LCD
101	79		P63	PBDSG3	GPIO
			SEG7		SEG pin for driving LCD
102	80		P62	PBDSG3	GPIO
			SEG6		SEG pin for driving LCD
103	81		P61	PBDSG3	GPIO
			SEG5		SEG pin for driving LCD
104	82		P60	PBDSG3	GPIO
			SEG4		SEG pin for driving LCD
105	83		DGND	Ground	Digital ground
106	84		P127	PBDSG3	GPIO for Version 2, not available for Version 1
			SEG3		SEG pin for driving LCD
			COM7		COM pin for driving LCD
107	85		P126	PBDSG	GPIO for Version 2, not available for

				3	Version 1
			SEG2		SEG pin for driving LCD
			COM6		COM pin for driving LCD
108	86		P125	PBDSG 3	GPIO for Version 2, not available for Version 1
			SEG1		SEG pin for driving LCD
			COM5		COM pin for driving LCD
109	87		P124	PBDSG 3	GPIO for Version 2, not available for Version 1
			SEG0		SEG pin for driving LCD
			COM4		COM pin for driving LCD
110	88		P123	PBDSG 3	GPIO for Version 2, not available for Version 1
			COM3		COM pin for driving LCD
111	89		P122	PBDSG 3	GPIO for Version 2, not available for Version 1
			COM2		COM pin for driving LCD
112	90		P121	PBDSG 3	GPIO for Version 2, not available for Version 1
			COM1		COM pin for driving LCD
113	91		P120	PBDSG 3	GPIO for Version 2, not available for Version 1
			COM0		COM pin for driving LCD
114			P100	PBDSG 3	GPIO
			SEG36		SEG pin for driving LCD
115	92	50	P10	PBULD 3	GPIO
			KEY0		Key 0 input
			TC0_N[0]		Negative output of Channel 0, Timer 0
			TCIN		Timer input
116	93	51	P11	PBULD 3	GPIO
			KEY1		Key 1 input
			TC0_P[0]		Positive output of Channel 0, Timer 0
			TCIN		Timer input
117		52	P12	PBULD 3	GPIO
			KEY2		Key 2 input
			TC0_N[1]		Negative output of Channel 1, Timer 0
			TCIN		Timer input
118	94	53	P30	PBUS6	GPIO
			INT0		External interruption 0 input
			TX4		UART4 transmmiter
			TCIN		Timer input
			ISPEN		Once reset, BOOTROM will inspect the state of this pin. If it is low, the system

					will enter ISP.
119		54	P31	PBUS6	GPIO
			INT1		External interruption 1 input
			RX4		UART4 receiver
			TCIN		Timer input
120	95	55	P32	PBUS6	GPIO
			INT2		External interruption 2 input
			RTCOUT		RTC output
			KEY5		Key 5 input
121	96	56	P26	PBULD 3	GPIO
			RX3		UART3 receiver
			RX2		UART2 receiver
122	97	57	P27	PBULD 3	GPIO
			TX3		UART3 transmmiter
			TX2		UART2 transmmiter
123		58	P57	PBULD 3	GPIO
			TC1_P[1]		Positive output of Channel 1, Timer 1
			TCIN		Timer input
124	98	59	P51	PBUS6	GPIO
			QF		QF is the output of INT0 input
			RTCOUT		RTC output
			SPI0_SCLK		SPI0 clock
			SF2		SF2 is the output of INT7 input
125	99	60	P50	PBUS6	GPIO
			PF		PF is the output of INT2 input
			RTCOUT		RTC output
			SPI0_SCSN		SPI0 chip select
			SF1		SF1 is the output of INT6 input
126		61	P34	PBUS6	GPIO
			INT4		External interruption 4 input
			SF		SF is the output of INT4 input
			D2F_OUT1		Output of power-integral module D2F1
127			P35	PBUS6	GPIO
			INT5		External interruption 5 input
			TCIN		Timer input
			D2F_OUT2		Output of power-integral module D2F2
128	100	62	DGND	Ground	Digital ground

1.5.6 Application Recommendation

1. AGND refers to analog ground and DGND is digital ground. Please take this into consideration in the application for the isolation between analog and digital system.

2. Pin LDO15 should be decoupled by external capacitors. A combination of a 1μf capacitor in parallel with a 0.1μf capacitor is recommended.
3. REFV is the reference input of SAR_ADC, which should be connected with a 0.22μf capacitor externally.
4. VBAT is the 3.6V source input pin. RC filtering should be applied externally. It is recommended to use a 10Ω resistor and a 1μF capacitor.
5. VCC is the main voltage source input. The operating voltage range is between 2.3V and 5.5V. It should be externally decoupled by a 4.7μf capacitor and a 0.1μf capacitor in parallel.
6. LDO_VBAT is the output of LDO in VBAT domain, which should be connected with a 0.22μf capacitor externally.
7. A crystal oscillator of 32.768 KHz should be applied between XO and XI. The crystal oscillator should be isolated by ground. External resistor and capacitor are not required. The high-speed flipping function of the IO pins near XO and XI are not avaiable.
8. LCDVD, LCDVC, LCDVB and LCDVA are LCD voltage output, each pin should be connected with a 470nf capacitor externally.
9. A 100nf capacitor should be applied between LCDVP1 and LCDVP2 when LCD CHARGE PUMP mode is enabled. When internal resistor array mode is enabled, LCDVP1 should be floating and LVDVP2 can be floating or grounded.
10. When SEG/IO multi-functional pins are used as IO pins. The output should be pulled up externally since it is an open-drain structure.
11. The input voltage of Pin VBAT and AIN6 would be halved before it reaches SAR ADC since there are 2 internal resistors of 300KΩ used for voltage division.
12. P36 and P37 are input pins when they are used as IO pins. Output function are not available.
13. P56,P44 and P45 are in VBAT domain, the output voltage would be equal to VBAT when it is high.
14. The input of timer can be either TC0 or TC1, which depends on the timer register settings.
15. It is not allowed to define the same UART in different IO pins at the same time.
16. It is not allowed to define the same SPI in different IO pins at the same time.

2 Charateristics

2.1 Electrical Ratings

Table 1-5 Electrical Ratings

Item	**Symbol**	**MIN**	**TYP**	**MAX**	**Unit**	**Remark**
Normal Ratings (Operating Temperature：-40℃～+85℃)						
Main Supply Voltage	VCC	2.3	5/3.3	5.5	V	5V±5% or 3.3V±5%
CPU Minimum Operating Voltage	Vil		1.8/2.2		V	1.8V or 2.2V
VBAT Supply Voltage	VBAT1	2.5	3.6	5.5	V	Within the range of VBAT1, the accuracy of RTC is guaranteed.
	VBAT2	1.8	3.6	5.5	V	Within the range of VBAT2, the RTC calendar runs normally. No guarantee for RTC temperature offset.
Digital Current 1	DIdd1		1.5		mA	CPU running under 3.6864MHz (PLL)
Digital Current 2	DIdd2		2.5		mA	CPU running under 7.3728MHz (PLL)
Digital Current 3	DIdd3		4		mA	CPU running under 14.7456MHz (PLL)
VCC Sleep Current	SIdd		5.5		μ A	RAM remains；CPU and digital system on; WDT on；LDO on；wake up by interruption; Typical Condition: Vcc=3.6V; Tc= 25° C
VBAT Supply Current	SIdd		1.8		uA	Typical Condition: Vbat=3.6V;Tc=25° C; Extreme Condition: Vbat=3.6V;Tc=25° C;
Instant Temperature Offset Current	TPSIdd		250		uA	Typical Condition: Vbat=3.6V；Tc=25° C;
LDO15	V1P8	1.35	1.5	1.65	V	CPU core voltage, 10% tolerance within the range of operating temperature
Extreme Ratings (Operating Temperature：-40℃～+85℃)						
VBAT Supply Voltage	Vvcc	-0.3	--	＋7	V	
VBAT Supply Voltage	Vvbat	-0.3	--	＋7	V	

DV_{DD} to DGND		-0.3	--	+7	V	
DV_{DD} to AV_{DD}		-0.3		+0.3	V	
Output High Voltage	VOH		--	VCC+0.3	V	
Output Low Voltage	VOL	-0.3	--		V	
Input High Voltage	VIH		0.7VCC			CMOS
Input Low Voltage	VIL		0.3VCC			CMOS
Input High Voltage	VIH		0.4VCC			TTL
Input Low Voltage	VIL		0.2VCC			TTL
Source Current	Isource	5		10	mA	6mA type
Sink Current	Isink	7		15	mA	6mA type
Source Current	Isource	3		5	mA	3mA type
Sink Current	Isink	5		10	mA	3mA type
Analog Output Voltage to AGND	V_{INA}	-0.3	--	AV_{DD} +0.3	V	
Operating Temperature Range	T_A	-40	--	85	℃	
Storage Temperature Range	T_{stg}	-65	--	150	℃	
Juntion Temperature	T_J		125		℃	
Reference Voltage (VCC=3V~5.5V，T= -40℃～+85℃)						
		min.	typ.	max.	unit	Notes
Output Voltage	V_{ref}	1.245	1.25	1.255	V	
Temperature Coefficient	T_c		5	15	ppm/℃	
Analog System (Operating Temperature：-40℃～+85℃)						
Low Power Comparator CMP1/CMP2/LVDIN Threshold Voltage ViL1	CMP	1.23	1.28	1.33	V	Vih is 220mV higher than Vil by default.
Low Power Comparator CMP1/CMP2 Threshold Voltage ViL2	CMP	0.8	0.84	0.88	V	Vih is 140mV higher than Vil when the 0.9V hysteresis setting is applied.
SAR ADC Output Voltage Range	SAR-IN	0		REFV	V	REFV is the internal reference voltage, which is 1.26V typically.
LCD PUMP Output Voltage	LCDVD	4.85	5	5.15	V	5V charge pump
LCD Resistor String Output Voltage	LCDVD	3.135	3.3	3.465	V	3.3V resistor string
VBAT Measurement Range	VBATD	0	3.6	5	V	The measurement range from SAR ADC to VBAT
Single Temperature Offset Time	Ttps		2		ms	

Clock (Operating Temperature：-40℃～+85℃)						
Input Low-frequency Clock Range	XI		32.768		KHz	
Input High-frequency Clock Range	HOSI	7.3728	14.7456	29.4912	MHz	
Internal PLL Clock Frequency Range	PLL		14.7456	29.4912	MHz	
Internal High-frequency RCH Range	RCH		29.5		MHz	Used as default clock after reset. The frequency is 1.8MHz after frequency division. The tolerance of RCH is 1%.
Internal Low-frequency RCL Range	RCL	20	32.768	40	KHz	Used as WDT clock

2.2 Reliability Ratings

Table 1-6 Reliability Ratings

Item	Abbreviation	Condition	Value	Unit
Electro-Static Discharge	ESD	Human Body Model（HBM），according to JEDEC EIA/JESD22-A114，applied to all pins	4000	V
		Machine Model（MM），according to JEDEC EIA/JESD22-A115C，applied to all pins	200	V
		Charged Device Model（CDM），according to JEDEC EIA/JESD22-C101F, applied to all pins	500	V
Latch-UP	-	According to JEDEC STANDARD NO.78D NOVEMBER 2011, applied to all pins	200	mA
Moisture Sensitive Detection	MSD	According to IPC/JEDEC J-STD-020D.1	Level 3	/

3 System Control

3.1 Power Domain Block Diagram

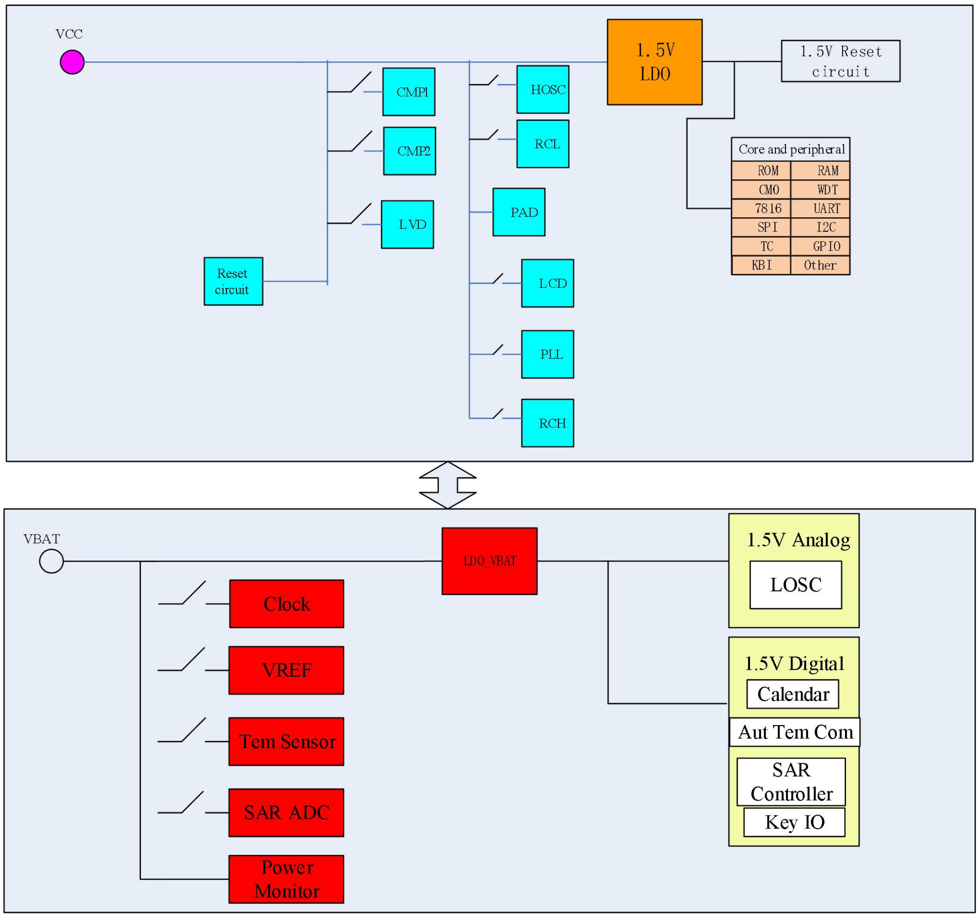


Figure 3-1 Internal Power Management

VCC and VBAT are powered independently, RTC related circuits (32768 crystal oscillator/perpetual calendar/automatic temperature compensation circuit, etc.), SAR-ADC measurement related circuits, P44/P45/P56 are powered by VBAT, CPU system and other peripherals are powered by VCC. In practical application, VCC or VBAT can be connected to supply power together or independently. The typical working range of VCC is 2.3V~5.5V, and the typical working range of VBAT is 2V~5.5V.

Note that the three IO ports P44, P45 and P56 are powered by VBAT, the output high level is equal to VBAT, the input high level cannot be higher than VBAT, and meet the requirements of VIH level (0.7 * VBAT). If only VBAT is powered on, VCC does not power on to initialize VBAT domain. Port P56 outputs 1Hz by default. Once VCC finishes power on to initialize VBAT domain, port P56 returns to default high resistance status.

Because some configuration registers (crystal oscillator parameter setting, etc.) of RTC module need to be configured by CPU during power on initialization (bootloader calls parameters in flash option bytes, and these parameters are written by customers in mass production programming), it is necessary to ensure that the configuration parameters can be written normally. If VBAT power on is too slow, when VCC power on initializes the configuration of VBAT, VBAT is not ready, so the temperature compensation parameters cannot be written normally. It is recommended that CPU software manages VBAT domain configuration register effectively, and calls Reenergy library function to do secondary configuration for relevant registers, so as to ensure that configuration parameters are written correctly.

3.2 Clock Source

- Two external clock:

LOSC: external 32.768 KHz crystal oscillator, used for RTC clock and CPU clock under low frequency operation, never shut down. The 32.768 KHz crystal oscillator does not need external capacitance and resistance, and the chip has been built-in. It is recommended to select 12.5pF external crystal oscillator.

HOSC: external high-frequency crystal oscillator can support external 7.3728MHz, 14.7456MHz, and 29.4912MHz crystal oscillator.

- Three internal clocks:

RCH: internal high-frequency RC clock (typical value is 3.2MHz). After the CPU is powered on and reset, the clock defaults to RCH; RCH can choose 1/2 frequency division or not.

RCL: internal low frequency RC clock, used for WDT clock, CPU clock and LCD clock under battery power supply.

PLL: internal PLL clock, multiplying 32.768 KHz to 7.3728MHz or 14.7456MHz.

- Low frequency operation mode can be selected as LOSC or RCL.
- LCD clock can be selected as LOSC or RCL.

The above four clock sources can be used for CPU main system clock.

In the operation mode, the clock source can be PLL or high frequency crystal oscillator.

CPU switches from low frequency clock to high frequency clock by instructions. The system master clock can be switched between RC, PLL (or HOSC) and LOSC clocks. In order to ensure the clock accuracy, PLL or external high frequency clock shall be selected as the main clock of the system under normal operation mode.

For clock switching, users must call Renergy library function. Users should not write to the OSC_ CTL1 (0x0) and SYS_Mode (0x4) registers in the application program. If write operation is performed on OSC_CTL2(0x10), only the bit bit to be operated shall be changed, and the value of other register bits shall not be changed.

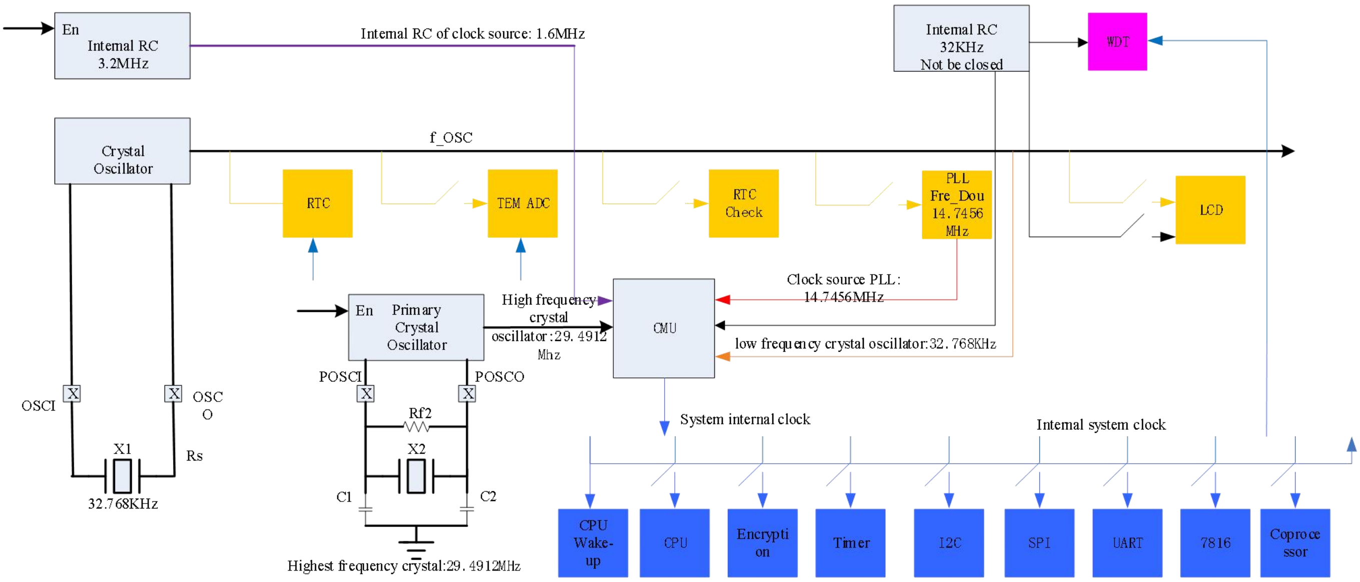


Figure 3-2 Clock Source

3.3 Clock Switching

Include switchings as follows:

1. Default is RCH after reset;
2. PLL and RCH switching, completed by CPU instructions;
3. PLL and LOSC/RCL switching, completed by CPU instructions;
4. LOSC/RCL and RCH switching, completed by CPU instructions;

Please call the library function provided by Renergy to complete clock switching.

If external high-frequency crystal HOSC is selected as system main clock, OSC_CTL2 register should be configured before calling library function.

If PLL is selected as system main clock at first power-on, calling the clock switching library function need to wait for 32 KHz crystal oscillator start-up (start-up time is approximately 0.5s).

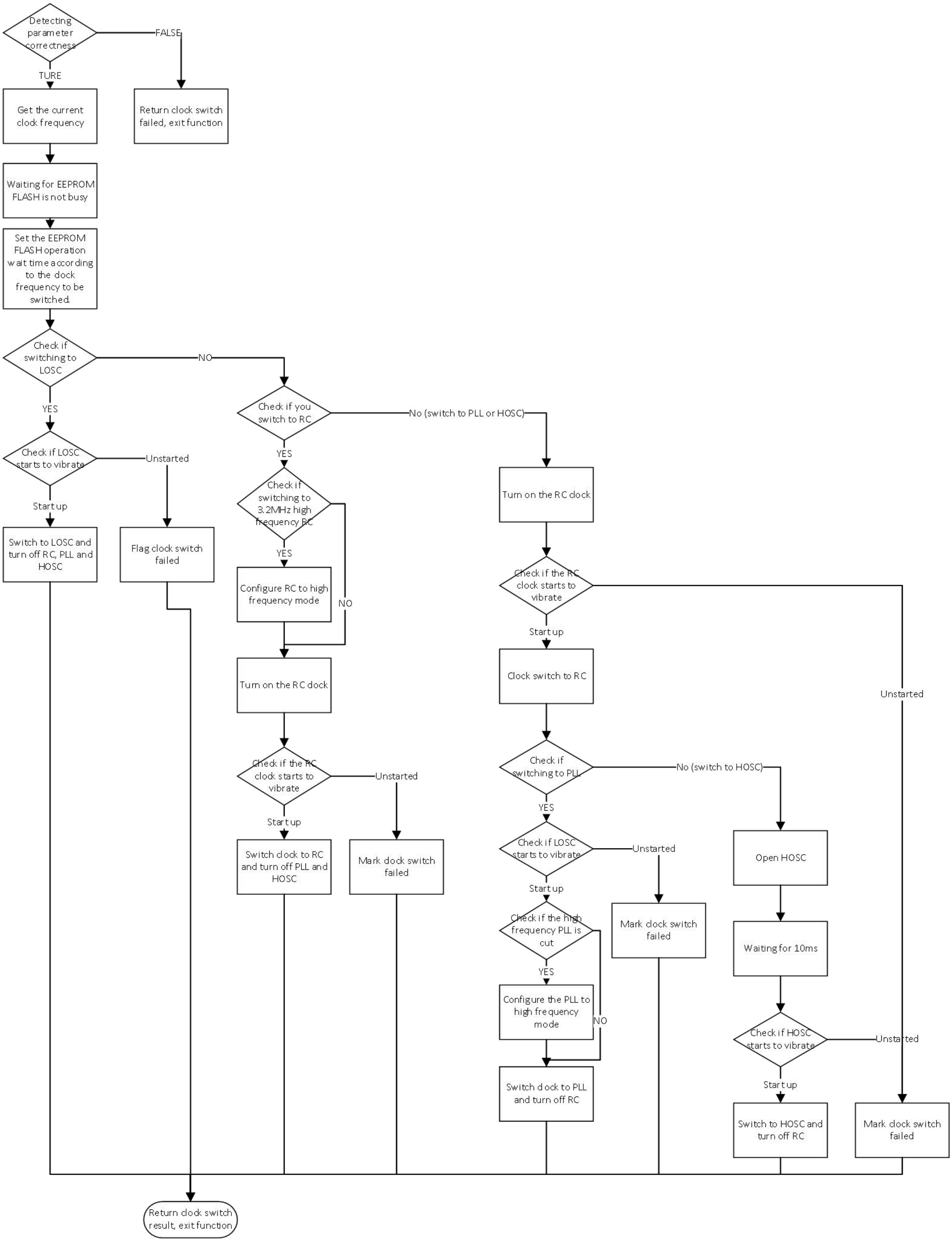


Figure 3-3 Library function clock switching diagram

3.4 MCU Low-power Mode

The low power modes of M0 are Sleep and DeepSleep. The differences between these two instructions are:

1. After the CPU runs the Sleep instruction, it will only turn off the CPU core clock, and the peripheral clock needs a register to turn off (see MODE0 and MODE1);

2. After the CPU runs the DeepSleep command, it will not only turn off the CPU core clock, but also automatically turn off most of the peripheral clocks (see MODE0 and MODE1);

It is recommended that the software not distinguish between Sleep and DeepSleep, use Sleep directly, and other peripheral clocks are closed by program.

In addition to the low-power mode of M0, the MCU provides a flexible mechanism to achieve the user's different power consumption modes:

1. The CPU can switch between the high frequency clock mode HCM, the low frequency clock mode LCM and the system default mode RCH by calling the library function mode;
2. The clock of the CPU and peripherals can be turned off;
3. In the lowest power mode (CPU sleep, SRAM and digital peripherals are not powered down, RTC operation) power consumption is about 7uA;

Default status of main module after power on:

Main Module	Default Status
1.8V Digital domain (powered from VCC)	
M0 core	Open, closable clock, never power down
Interrupt system	Open, closable clock, never power down
SRAM	Open, closable clock, never power down
ROM	Open, closable clock, never power down
FLASH	Open, automatically power off after CPU sleep
WDT	Open, can not be closed
Other peripherals	Open, closable clock, never power down
VCC domain	
3.3V LDO	Close, closable power supply
1.8V LDO	Open, can not be closed
RCL	Open, can not be closed
RCH	Open, can not be closed
LCD	Close, closable power supply
LVD	Close, closable power supply
Comparator CMP1	Open ,closable power supply
Comparator CMP2	Close, closable power supply
Reset circuit	Always open
PLL	Close, closable power supply
VBAT domain	
RTC	Open, calendar can not be closed, no reset
Temperature measurement ADC	Timing on
LOSC	Always open

The default off modules can be turned on or off under the three clocks.

3.5 Reset

3.5.1 External Pin Reset

The external pin RSTN has a built-in 50K ohm pull-up resistor and the input level are CMOS level. The internal filter time is 1ms, and resetting occurs when the external input is low for more than 1ms.

3.5.2 Power-on and Power-off Reset

Three power-on reset (POR) circuits and two brown-out reset (BOR) circuits are built to monitor VCC, LDO18 (1.8V LDO) and VBAT, respectively. Among them, there is only POR circuit in VBAT.

The circuit is always active and cannot be turned off to ensure that the system will operate normally when the threshold is exceeded (about 2.6V); below the threshold (about 2.35V), the device is in reset. It is no longer necessary to use an external reset circuit.

This product also has a built-in programmable voltage monitor LVD that monitors VCC and compares it to a set threshold that generates an interrupt when VCC is below or above the threshold.

3.5.3 Software Reset

The Cotex M0 has a built-in SCB_AIRCR register. Simply set the SYSRESETREQ bit of this register to cause a reset of the entire chip system. The reset effect is equivalent to the external PIN reset. See the M0 documentation for details.

3.5.4 Watchdog Reset

If the watchdog is not fed within the specified time, or an illegal command is used to feed the dog, the built-in hardware watchdog will reset the entire chip, and the reset effect is equivalent to the external PIN reset.

3.6 Power-down Detection

The chip provides the following means to do power-down detection, application software in the detection of power-down should be processed in time to complete the necessary work, turn off peripherals, so that the CPU into low-frequency operation mode or sleep mode.

1. CMP1/CMP2

CMP1/CMP2 is a low power consumption comparator, which detects the voltage input to the IO port, detects the power supply voltage of the LDO front end that supplies power to the chip, and is also the only basis for power-on judgment; the power consumption of this module is less than 1uA, and in practical applications, it can be always turned on. The customer software can use CMP1/CMP2 interrupt or flag bit inquiry to make power-up judgment.

2. LVD

The LVD module can detect the input voltage of the external pin LVDIN (external pin is required, and the power consumption is larger than that of CMP1/CMP2), and can also detect the power supply voltage of the chip (no external pin input is required, and the VCC power supply is detected internally in the chip, and multiple thresholds can be set). The client software can use interrupt or flag bit inquiry to judge the upper and lower power.

3.7 Register Description

The base address of system control module:

Module Name	**Physical Address**	**Mapping Address**
SYSC	0x40034000	0x40034000
Register Name	**Address Offset**	**Description**
OSC_CTL1	0x0	System OSC control register1
SYS_MODE	0x4	System mode switch register

SYS_PD	0x8	System power down control register
ADC_CTL	0xC	ADC control register
OSC_CTL2	0x10	System OSC control register2
SYS_RST	0x14	System reset register
MAP_CTL	0x18	Address Mapping control register
MOD0_EN	0x1C	Module enable 0 register
MOD1_EN	0x20	Module enable 1 register
INTC_EN	0x24	Module enable 1 register
KBI_EN	0x28	INTC enable register
CHIP_ID	0x2C	Device ID register
SYS_PS	0x30	System control register password protection bit
IRFR_CTL	0x34	Infrared clock division factor in RCH mode
TRIM_CFG1	0x78	Clock TRIM configuration register
TRIM_START	0x7C	Clock TRIM result register
FAB_UID0	0xF4	Chip unique code register 0
FAB_UID1	0xF8	Chip unique code register 1

The 0x00~0x28, 0x34/0x78/0x7C registers can be written only when SYS_PS (0x30)=8’h82.

3.7.1 System OSC control register1 OSC_CTL1 (0x00)

Offset address 0x00

Bit	**Name**	**Description**	**Write/Read Flag**	**Reset Value**
31:16	---	Reserved	R	0
15:11	CLOCK_FLAG	System clock open flag: if clock opened, this bit is 1: {HOSC, RCL, RCH, PLL, LOSC}	R	01101
10:8	SYSCLK_STAT	System main clock frequency 000: 7.3728MHz; 001: 3.6864MHz; 010: 1.8432MHz; 011: 32.768KHz; 100: 14.7456MHz; 101: 29.4912MHz; others:reserved	R	010
7	PLL_LOCK	PLL Lock Status 0: Unlocked 1: Locked	R	0
6	PLL_HOSC_ON	System operating at external high frequency or internal PLL clock, this bit is 1; System operating at other clocks, this bit is 0.	R	0
5	IRCH_ON	System operating at internal high frequency clock, this bit is 1;	R	1

		System operating at other clocks, this bit is 0.		
4	LOSC_ON	System operating at external low frequency clock, this bit is 1; System operating at other clocks, this bit is 0	R	0
3:2	PLL_HOSC_DIV	System master clock divide select:(Only valid for high frequency clock mode) 00: PLL, HOSC as CPU master clock; 01: Two Division of PLL, HOSC as CPU master clock; 10: Four division of PLL, HOSC as CPU master clock; 11: Eight division of HOSC (clock select as 14MHz and 29MHz) as CPU master clock. If the HOSC is 7.3728MHz or the PLL is selected, it is represented as four-way; Note: Can only be changed under RC or LC mode. Note: These registers only decide division factor, and specific system master frequency is determined by the frequency division factor and current clock source.	R/W	01
1	IRCH_PD	1.6MHz internal RC enable bit: 0: open; 1: close.	R/W	0
0	PLL_PD	PLL module enable bit 0: open; 1: close.	RW	1

It is recommended to call Renergy library functions, but not recommended to write OSC_CTL1 (0x0) register in applications.

3.7.2 System Mode Setting Register SYS_MODE (0x04)

Offset address 0x04

Bit	**Name**	**Description**	**Read /Write Flag**	**Reset Value**
31:6	---	Reserved	R	0
5	FLASH_BUSY	Flash busy status, can not enter the mode switch: 0: idle 1: busy	R	0
4	Reserved	Reserved	R	0
3:0	MODE	Write D, set enter high frequency mode HCM, bit2 read is 1; Write E, set enter RC mode RCM, bit1 read is1; Write F, set enter 32.768KHz mode LCM, bit0 read is1. The register read value is: {0, HCM, RCM, LCM}.	R/W	2

Note: The instructions of current mode status should be read from LOSC_ON, IRCH_ON, PLL_HOSC_ON (OSC_CTL register bit4~6) these three status. But not from this register. This register represents the mode switch command is written, which does not mean that has been switched to the desired mode.

It is recommended to call Renergy library functions. And it is not recommended write SYS_MODE (0x4) register in applications.

3.7.3 System Power Down Control Register SYS_PD (0x08)

Offset address 0x08

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:17	---	Reserved	R	0
16	vsel_bor5	BOR5 threshold selection signal, vih/vil =0:2.5v/2.2v, default =1:2.1v/1.8v	R/W	0
15	Reserved	Reserved	R/W	0
14	Reserved	Reserved	R/W	0
13	Reserved	Reserved	R/W	0
12	Reserved	Reserved	R/W	0
11	hysen_cmp2	Cmplp2 internal hysteresis comparator hysteresis switch 0x1: hysteresis on 0x0: hysteresis off	R/W	0
10	hysen_cmp1	Cmplp1 internal hysteresis comparator hysteresis switch 0x1: hysteresis on 0x0: hysteresis off	R/W	0
9	PWD_CMP2R	CMP2 internal 600K resistance sampling switch 0: CMP2 internal resistance sampling is turned on, the peripheral circuit needs to pay attention to the internal 600K to ground resistance, the comparator Vil typical value is 1.28V, hysteresis is 0.22V, do not configure bit11 to 1 at this time; 1: CMP2 internal resistance sampling is off, the comparator threshold is 0.9V, no hysteresis by default; bit11 can be set to 1 with a hysteresis of 0.14V, and the typical value of Vil is 0.84V;	R/W	0
8	PWD_CMP1R	CMP1 internal 600K resistance sampling switch 0: CMP1 internal resistance sampling is turned on, the peripheral circuit needs to pay attention to the internal 600K to ground resistance, the comparator Vil typical value is 1.28V, hysteresis is 0.22V, do not configure bit10 as 1 at this time; 1: CMP1 internal resistance sampling is off, the comparator threshold is 0.9V, no hysteresis by default; bit10 can be set to have a hysteresis of 0.14V, and the typical value of Vil is 0.84V;	R/W	0
7	Reserved	Reserved	R/W	0

6	Reserved	Reserved	R/W	0
5	CMP2_PD	Comparator 2 power switch 0:on 1:off GPIO mux register needs to be configured in advance.	R/W	0
4	CMP1_PD	Comparator 1 power switch 0:on 1:off GPIO mux register needs to be configured in advance.	R/W	1
3	LVD_PD	LVD power switch 0:on 1:off GPIO mux register needs to be configured in advance.	R/W	1
2:0	Reserved	Reserved	R	0

3.7.4 System OSC Control Register 2 OSC_CTL2 (0x10)

Offset address 0x10

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset value**
31:19	Reserved	Reserved		
18:16	RCH_FREQ	000: 1.8MHz; 001: 3.6MHz; 010: 7.3MHz; 011: 14.7Mhz; 100: 29.5Mhz; others:reserved Note: Call the library function to select the chip operating frequency and do not change the value of this bit in the application.	R/W	0
15	RCL_LOSC_FLT_SEL	Filter clock source selection 0: Filter clock selection LOSC 1: Filter clock selection RCL	R/W	0
14	Reserved	Reserved	R	0
13	RCL_LCD	=0: LCD selects LOSC external low frequency crystal as the clock source; =1: LCD selects RCL internal low-frequency crystal as the clock source;	R/W	0
12	RCL_STB	=0: Low frequency operation mode LCM selects LOSC external low frequency crystal oscillator as the clock source; =1: Low frequency operation mode LCM selects RCL internal low frequency crystal oscillator as the clock source; This register can only be modified in RC mode or high	R/W	0

		frequency mode.		
11:10	Reserved	Writable, internal test register; Do not change the default value of this register.	R/W	00
9	Reserved	Reserved	R	0
8	Reserved	Reserved	R	0
7:5	PLL_FREQ	The PLL frequency is fixed at 14.7456Mhz, and the frequency is selected by digital frequency division: 000: 7.3728MHz; 001: 14.7456Mhz; Other: reserved Call the library function to select the chip operating frequency and do not change the value of this bit in the application.	R/W	000
4	PLL_HOSC_SEL	System master clock selection at full speed: 0: Select the PLL output as the system master clock; 1: Select the spare high frequency crystal as the system master clock. This register can only be configured in RC mode and low frequency mode..	R/W	0
3	HOSC_PD	External high frequency oscillator enable bit: 0:on 1:off	R/W	1
2:0	HOSC_ FREQ	000：7.3728MHz 001：14.7456MHz 010： reserved 011：29.4912MHz	R/W	000

System clock configuration truth table :(PLL and HOSC mode):

PLL_HOSC_SEL	PLL_FREQ	Clock	PLL_HOSC_DIV =000	PLL_HOSC_DIV =001	PLL_HOSC_DIV =010	PLL_HOSC_DIV =011
			no frequency division	second divided	divide by four	divide by eight
0	000	PLL	7.3728Mhz	3.6864MHz	1.8432MHz	Does not support eight-way
1	000	HOSC	7.3728Mhz	3.6864MHz	1.8432MHz	Does not support eight-way
0	001	PLL	14.7456Mhz	7.3728Mhz	3.6864Mhz	1.8432Mhz
1	001	HOSC	14.7456Mhz	7.3728Mhz	3.6864Mhz	1.8432Mhz
0	010	PLL	29.4912Mhz	14.7456Mhz	7.3728Mhz	3.6864Mhz
1	011	HOSC	29.4912Mhz	14.7456Mhz	7.3728Mhz	3.6864Mhz

3.7.5 System Reset Register SYS_RST (0x14)

Offset address 0x14

(10 to 5 bits of this register can only be reset by power-on reset)

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:11	Reserved	Reserved	R	0
10	boi_vbat_ie	VBAT domain power-down interrupt enable bit: =0 interrupt disabled =1 interrupt enabled	R/W	0
9	boi_vbat_flag	=0, VBAT domain is powered normally. =1, VBAT domain has been powered down Write 1 clear	R/W	0
8	MCU_RST	CPU reset flag(occur software reset or LOCK UP reset): =1 represents that the reset occurred, =0 represents un happen. Write 1 to clear	R/W	0
7	WDT_RST	WDT reset flag: =1 represents that the reset occurred; =0 represents un happen. Write 1 to clear	R/W	0
6	PIN_RST	External pin reset flag: =1 represents that the reset occurred, =0 represents un happen. Write 1 to clear	R/W	0
5	POWEROK_RST	Power reset flag Source power on/down reset flag =1 represents that the reset occurred, =0 represents un happen. Write 1 to clear	R/W	1
4:3	Reserved	Reserved	R	0
2	LOCKUP_ENRST	LOCKUP enable reset (CPU occurs twice Hard Fault will cause LOCKUP, if enable this bit, can cause system reset): 0: LOCKUP doesn’t cause system reset 1: LOCKUP cause system reset	R/W	0
1	Reserved	Reserved	R	0
0	Reserved	Reserved	R	0

3.7.6 System Mapping Control Register SYS_MAPCTL (0x18)

Offset address 0x18

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:3	Reserved	Reserved	R	0
2:0	REMAP	Address mapping: 000: FLASH mapping to 0 address (normal mode); 001: FLASH maps in 1/2 capacity address (only 512KB capacity supports this function) or FLASH and EEPROM mapped address interchange (only supported by	R/W	000

		RN8612/8610B); 010: FLASH and SRAM mapping address interchange; 011: BOOTROM maps at address 0; 100: FLASH maps at 1/2 capacity address (only 512KB capacity supports this function); Others: Reserved.		

3.7.7 Module Enable 0 Register MOD0_EN (0x1c)

Offset address 0x1C

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:21	---	Reserved	R	0
20	SIMP_TC_EN	SIMP_TC module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
19	---	Reserved	R	0
18	SPI3_EN	SPI3 module clear/enable, clock gating, cm0 enters deepsleep mode and synchronously turns off this clock: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
17	SPI2_EN	SPI2 module clear/enable, clock gating, cm0 enters deepsleep mode and synchronously turns off this clock: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
16	SPI1_EN	SPI1 module clear/enable, clock gating, cm0 enters deepsleep mode and synchronously turns off this clock: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
15	SPI_EN	SPI module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
14	I2C_EN	I2C module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
13	ISO7816_EN	ISO7816 module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
12	UART38K _EN	UART38K infrared modulated clock clear/enable: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
11	UART3_EN	UART3 module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
10	UART2_EN	UART2 module clear/enable, clock gating:	R/W	0

		0: Clock stop, module clear; 1: Clock start, module enable.		
9	UART1_EN	UART1 module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
8	UART0_EN	UART0 module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
7	UART5_EN	UART5 module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
6	UART4_EN	UART4 module clear/enable, clock gating: 0: Clock stop, module clear; 1: Clock start, module enable.	R/W	0
5	TC1_EN	TC1 module clear/enable, clock gating: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
4	TC0_EN	TC0 module clear/enable, clock gating: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
3	Reserved	Reserved	R/W	0
2	Reserved	Reserved	R	0
1	Reserved	Reserved	R	0
0	Reserved	Reserved	R	0

3.7.8 Module 1 Enable Register MOD1_EN (0x20)

Offset address 0x20

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:16	Reserved	Reserved	R	0
15	M2M_EN	M2M module clear/enable, clock gating, cm0 enters deepsleep mode and synchronously turns off this clock: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
14	---	Reserved	R/W	0
13	D2F_EN	D2F module (electric energy integrator) clock: 0: Clock stop; 1: Clock start.	R/W	0
12	--	Reserved	R	0
11	CMPLVD_EN	CMP/LVD module clear/enable, apb clock gating: 0: Clock stop; 1: Clock start.	R/W	0

10	RTC_SAR_EN	RTC/SAR apb clock gating, cm0 enters deepsleep and synchronously turns off this clock: 0: Clock stop; 1: Clock start.	R/W	1
9	WDT_EN	WDT apb clock gating, cm0 enters deepsleep and synchronously turns off this clock: 0: Clock stop; 1: Clock start.	R/W	1
8	Reserved	Reserved	R	0
7	Reserved	Reserved	R	0
6	LCD_EN	LCD module clear/enable, clock gating: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
5	GPIO_EN	GPIO module clear/enable, clock gating, cm0 enters deepsleep and synchronously turns off this clock: 0: Clock stop, module clear 1: Clock start, module enable	R/W	0
4	---	Reserved	R	0
3	---	Reserved	R	0
2	---	Reserved	R	0
1	---	Reserved	R	0
0	---	Reserved	R	0

3.7.9 INTC Enable Register INTC_EN (0x24)

Offset address 0x24

Bit	Name	Description	Read/Write Flag	Reset Value
31:9	Reserved	Reserved	R	0
8	INTC_EN	INTC apb module clock gating, cm0 enters deepsleep and synchronously turns off this clock: 0: clock stop 1: clock start	R/W	0
7	INTC7_EN	INTC7 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
6	INTC6_EN	INTC6 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
5	INTC5_EN	INTC5 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
4	INTC4_EN	INTC4 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0

3	INTC3_EN	INTC3 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
2	INTC2_EN	INTC2 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
1	INTC1_EN	INTC1 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
0	INTC0_EN	INTC0 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0

3.7.10 KBI Enbale Register KBI_EN (0x28)

Offset address 0x28

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:9	Reserved	Reserved	R	0
8	KBI_EN	KBI apb module clock gating, cm0 enters deepsleep and synchronously turns off this clock: 0: clock stop 1: clock start	R/W	0
7	KBI7_EN	KBI7 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
6	KBI6_EN	KBI6 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
5	KBI5_EN	KBI5 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
4	KBI4_EN	KBI4 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
3	KBI3_EN	KBI3 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
2	KBI2_EN	KBI2 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
1	KBI1_EN	KBI1 module clear/enable, clock gating: 0: clock stop 1: clock start	R/W	0
0	KBI0_EN	KBI0 module clear/enable, clock gating:	R/W	0

		0: clock stop 1: clock start		

3.7.11 Chip ID Register CHIP_ID (0x2C)

Bit	Name	Description	Read/Write Flag	Reset Value
31:24	---	Reserved	R	0
15:0	CHIP_ID	CHIP_ID : V1:8213 V2:8220	R	xxxx

3.7.12 System Control Password Register SYS_PS (0x30)

Bit	Name	Description	Read/Write Flag	Reset Value
31:8	---	Reserved	R	0
7:0	SYS_PSW	When SYS_PSW = 0x82, the 0x00~0x28 registers can be written; When SYS_PSW = other values, the 0x00~0x28 registers are not writable; This register reads the written value. It is recommended to disable the register as soon as the write operation is complete..	R/W	00

3.7.13 Infrared Configuration Register IRFR_CTL (0x34)

Bit	Name	Description	Read/Write Flag	Reset Value
31:6	---	Reserved	R	0
5:0	IRFR_CYCLE	The password for this register is 0x82. In RCH mode, the infrared clock division factor is based on RCH of 3.6864MHZ. ... 0x19: infrared output clock 36.9K; 0x18: infrared output clock 38.4K; 0x17: infrared output clock 40K; ... The formula is: IRFR value = RCH measured frequency / 4/38K	R/W	0x18

3.7.14 Clock Correction Configuration Register TRIM_CFG1 (0x78)

Bit	Name	Description	Read/Write Flag	Reset Value
31:29	---	Reserved	R	0
28	CAL_OV_IE	Calibrated Clock Counter Overflow Flag Interrupt Enable: 0: interrupt disabled;	R/W	0

		1: interrupt enabled.		
27	CAL_DONE_IE	Clock Calibration Completion Flag Interrupt Enable 0: interrupt disabled; 1: interrupt enabled.	R/W	0
26	CAL_CLK_SEL	Corrected clock source selection: 0: RCH; 1: RCL	R/W	0
25:24	REF_CLK_SEL	Reference clock source selection: 00: LOSC; 01: HOSC; 10: RCH; 11: PLL;	R/W	11
23:20	---	Reserved	R	0
19:0	REF_CLK_CNT	Reference clock count	R/W	0x10000

3.7.15 Clock Correction Start Register TRIM_START TRIM_START (0x7C)

Bit	Name	Description	Read/Write Flag	Reset Value
31:28	---	Reserved	R	0
27	STOP	Clock calibration termination bit: 0: no operation; 1: Terminate the clock calibration; Note: Write a 0 to this bit to restart the clock calibration after termination.	R/W	0
26	START	Clock calibration start bit: 0: no operation; 1: Start clock calibration; Note: This bit is automatically cleared when the clock calibration is completed or terminated.	R/W	0
25	CAL_OV	Corrected clock counter overflow flag: 0: no overflow; 1: overflow; Note: Write 1 to clear.	R/W	0
24	CAL_DONE	Clock calibration completion flag: 0: not completed; 1: completed; Note: Write 1 to clear.	R/W	0
23:20	---	Reserved	R	0
19:0	CAL_CLK_CNT	The count value returned by the calibrated clock	R	0

Examples:

1. Select the reference clock as LOSC and select the clock to be calibrated as RCH;
2. Select the reference clock counter REF_CLK_CNT to 0x1000 and the count time to 0.125S;
3. Start the clock correction operation, query the flag bit or wait for the system to control the interrupt

generation;

4. Assume that the count value returned by the corrected clock is CAL_CLK_CNT=0x 61A80, and the decimal is 400000;
5. Then the measured RCH frequency value is:

(CAL_CLK_CNT/REF_CLK_CNT)*32768Hz

= (400000/4096)*32768Hz

=3200000Hz

=3.2MHz

3.7.16 DMA Priority Configuration Register DMA_PRI (0x80)

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:28	WKEY	WKEY is 0~25bit write operation password protection bit, the password is 0xE. It must ensure that the upper 4 bits of data written are 0xE and SYS_PSW is 8'h82 at the same time.	R	0
27:26	Reserved	Reserved	R	0x0
25:24	DMA_CH12_PRI	Channel 12: CPU Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
23:22	DMA_CH11_PRI	Channel 11 : M2M DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
21:20	DMA_CH10_PRI	Channel 10 : DSP DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
19:18	DMA_CH9_PRI	Channel 9 : UART5 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
17:16	DMA_CH8_PRI	Channel 8: UART4 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
15:14	DMA_CH7_PRI	Channel 7: UART3 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
13:12	DMA_CH6_PRI	Channel 6: UART2 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
11:10	DMA_CH5_PRI	Channel 5: UART1 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
9:8	DMA_CH4_PRI	Channel 4: UART0 DMA Access SRAM Priority Configuration	R/W	0x0

		Priority configuration same as CH0		
7:6	DMA_CH3_PRI	Channel 3 : SPI3 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
5:4	DMA_CH2_PRI	Channel 2 : SPI2 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
3:2	DMA_CH1_PRI	Channel 1 : SPI1 DMA Access SRAM Priority Configuration Priority configuration same as CH0	R/W	0x0
1:0	DMA_CH0_PRI	Channel 0 : SPI0 DMA Access SRAM Priority Configuration 0x3:Highest; 0x2:Higher 0x1:Lower; 0x0:Lowest The channel with lower number has higher priority.	R/W	0x0

Notes:

1、 There are 13 hosts accessing SRAM on the system: SPI0-DMA、SPI1-DMA、SPI2-DMA、SPI3-DMA、UART0-DMA、UART1-DMA、UART2-DMA、UART3-DMA、UART4-DMA、UART5-DMA、DSP-DMA、M2M-DMA 、ARM-M0 CPU;

2、When different hosts access the same SRAM, there will be competition, and the system will determine according to the priority configuration. The host channel with higher priority will get the access right of SRAM first.

a) The DMA_PRI register defines the priority of host access to SRAM with the following priority policies: configurable priority and channel number absolute priority.

b) Each channel priority is configured with 2 bits, and 4 priority levels can be configured. The priority order is 3>2>1>0, and the default is 0.

c) The channel with lower number has higher priority.

3.7.17 Chip Unique Code Register 0 FAB_UID0 (0xF4)

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:0	FAB_UID0	Form a chip unique identification code with FAB_UID 1.	R	0

3.7.18 Chip Unique Code Register 1 FAB_UID1 (0xF8)

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:0	FAB_UID1	Form a chip unique identification code with FAB_UID 0.	R	0

4 CPU Architecture

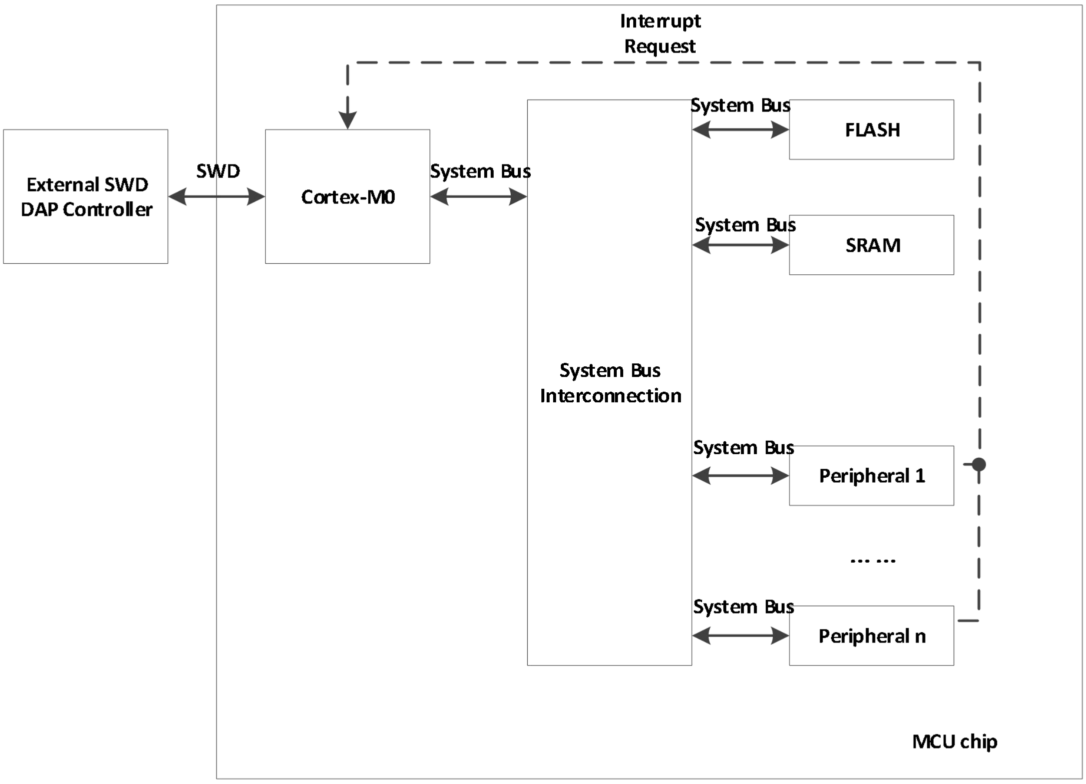
4.1 Overview

There are two ways (two master devices) which can access to MCU built-in device.

- Cortex-M0:
  - Comand Access and Data Access;
  - Access to all slave devices
- Exsternal SWD Controller (JLINK or similar function devices):
  - Interface Debug and Resource Access;
  - Access to all slave devices.

MCU built-in slave device resources include memory (FLASH, EEPROM and SRAM) and various peripherals (UART, Timer, Watchdog, etc.).

Figure 4- 1 MCU device physical interconnection architecture



4.2 Cortex-M0 Processor

Cortex-M0 processor is a 32-bit processor which is designed for the embedded systems, and has the following characteristics:

- Easy-to-use programming model
- High code density, with 32-bit properties
- Tool and binary code upward compatible with cortex-m processor series, easy to upgrade and expand

- A low-power sleep mode integrated
- Highly efficient code execution allows that the processor clock is more low, or extend the time for sleep mode
- 32-bit single-cycle hardware multiplier
- Zero jitter interrupt
- Interrupt timing determination, interrupt processing with high efficiency
- Interrupt / exception nesting and seizing
- 24-bit system beats counter
- Four interrupt priorities
- Two observation points, four hardware breakpoints
- Serial debug interface (SWD), to achieve the internal status of the processor highly visible and controllable
- CM0 embedded system timer, please refer to the ARM documentation for details.

4.2.1 Interrupt Configuration

32 interrupts in total, including 8 external interrupts, Interrupt 0 ~ Interrupt7.

For details of interrupts, such as priority masking registers, nested vector interrupt controllers (NVIC), refer to the arm-m0 manual..

Table 4- 1Interrupt/exception inventory and its configuration information

Abnormal number	Interrupt number	Vectors' name	Address	Priority	Activate way
-	-	MSP initial value	0x00	-	-
1	-	Reset	0x04	-3, Highest	Asynchronous with processor
2	-14	Non-shielded interrupt	0x08	-2	Asynchronous with processor
3	-13	Hardware failure	0x0C	-1	Synchronous with processor
4-10	-12~-6	Reserved	0x10 ~0x28	-	
11	-5	System calls	0x2C	Configurable	Synchronous with processor
12-13	-4~-3	Reserved	0x30 ~0x34	-	
14	-2	PendSV	0x38	Configurable	Asynchronous

					with processor
15	-1	System beat counter	0x3C	Configurable	Asynchronous with processor
16	0	System control	0x40	Configurable	Asynchronous with processor
17	1	CMP1, CMP2, LVD	0x44	Configurable	Asynchronous with processor
18	2	Reserved	0x48	-	
19	3	RTC	0x4C	Configurable	Asynchronous with processor
20	4	D2F	0x50	-	
21	5	Multiplexing ADC	0x54	Configurable	Asynchronous with processor
22	6	UART0	0x58	Configurable	Asynchronous with processor
23	7	UART1	0x5C	Configurable	Asynchronous with processor
24	8	UART2	0x60	Configurable	Asynchronous with processor
25	9	UART3	0x64	Configurable	Asynchronous with processor
26	10	SPI0	0x68	Configurable	Asynchronous with processor
27	11	I2C	0x6C	Configurable	Asynchronous with processor
28	12	7816_0/SPI3	0x70	Configurable	Asynchronous with processor
29	13	7816_1/SPI2	0x74	Configurable	Asynchronous with processor

30	14	TC0	0x78	Configurable	Asynchronous with processor
31	15	TC1	0x7C	Configurable	Asynchronous with processor
32	16	UART4	0x80	Configurable	Asynchronous with processor
33	17	UART5	0x84	Configurable	Asynchronous with processor
34	18	Watchdog	0x88	Configurable	Asynchronous with processor
35	19	KBI	0x8C	Configurable	Asynchronous with processor
36	20	LCD	0x90	Configurable	Asynchronous with processor
37	21	Encrypted SEA	0x94	Configurable	Asynchronous with processor
38	22	Reserved	0x98	-	
39	23	SPI1	0x9C	-	
40	24	External interrupt 0 /INTn (INT0~7 combined)	0xA0	Configurable	Asynchronous with processor
41	25	External interrupt 1 /SIMP_TC1	0xA4	Configurable	Asynchronous with processor
42	26	External interrupt 2 /SIMP_TC2	0xA8	Configurable	Asynchronous with processor
43	27	External interrupt 3 /SIMP_TC3	0xAC	Configurable	Asynchronous with processor
44	28	External interrupt 4	0xB0	Configurable	Asynchronous

		/SIMP_TC4			with processor
45	29	External interrupt 5/M2M	0xB4	Configurable	Asynchronous with processor
46	30	External interrupt 6	0xB8	Configurable	Asynchronous with processor
47	31	External interrupt 7	0xBC	Configurable	Asynchronous with processor

4.3 MCU Memory Mapping

Please refer to "Figure 4-2 MCU memory mapping" for the memory mapping of MCU.

High-speed peripherals of MCU include:

- GPIO
- Encripted SEA

Low-speed peripherals of MCU include:

- 6 UART ports
- 2 32-bit timers
- 4 16-bit timers
- 4 SPI ports
- 1 I2C ports
- 1 KEY controllers
- 1 universal ADC port
- 1 watchdog unit
- 1 system control unit
- 2 7816 ports
- 1 RTC port
- 1 interrupt port
- 1 LCD controller
- 1 D2F energy integration unit
- 1 M2M memory handling module

4.3.1 MCU Memory Mapping

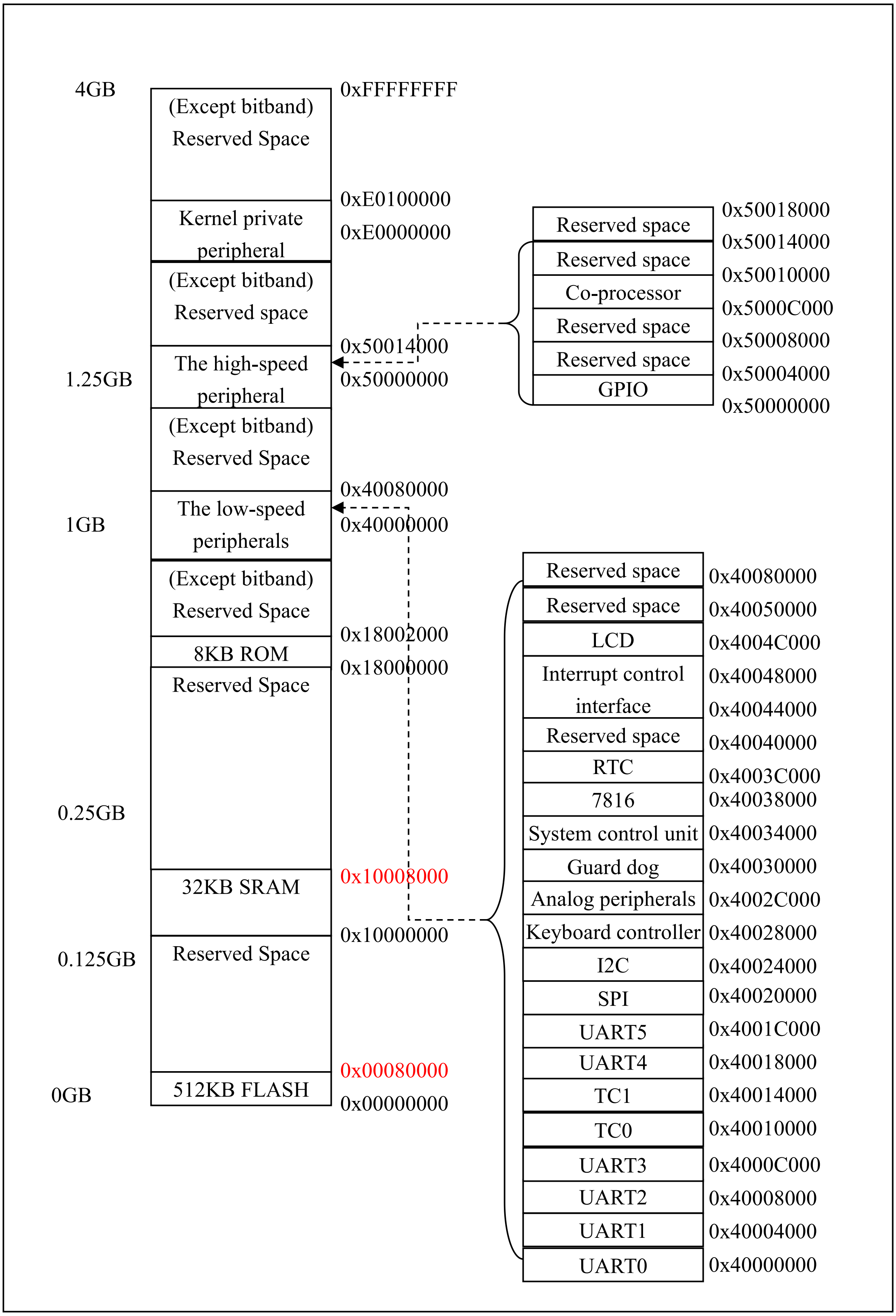


Figure 4-2 MCU memory mapping

4.3.2 Memory Re-mapping

The MCU supports swapping base addresses for two storage areas, including FLASH and SRAM.Switching is done by configuring the REMAP register in the system controller.The address allocation of devices other than FLASH and SRAM is not affected by the memory re-mapping.

Table 4-2Store the re-mapped configuration

Memory	REMAP	Mapping address
FLASH	0	0x00000000
	1	0x00000000
	2	0x10000000
	3	0x18000000
SRAM	0	0x10000000
	1	0x10000000
	2	0x00000000
	3	0x10000000

- Bitband function

The system supports bitband function of the following address space:

0x10000000~0x10007FFF mapped to 0x12000000~0x120FFFFF
0x40000000~0x4004FFFF mapped to 0x42000000~0x423FFFFF;
0x50000000~0x50007FFF mapped to 0x52000000~0x5203FFFF;

Access to the bitband region is equivalent to access to the special location in the external register.

Btband address corresponding to the y bit of storage unit with address x:

$$Z = (X \ \& \ 0xFC000000) + 0x02000000 + (Y << 2) + ((X << 5) \ \& \ 0x03FFFFFF)$$

4.3.3 SRAM

- System SRAM

The maximum capacity of on-chip SRAM is 96KB, the address is 0x1000_0000~0x1001_8000, the RAM runs at the same frequency as the processor, supports random access of 8-bit, 16-bit or 32-bit data, and can be used as code or data storage.

- Algorithm SRAM

The chip contains 4KB SRAM as the encryption module algorithm RAM. When the encryption module is not working, it can also be used as an ordinary RAM and accessed by the CPU. The address is 0x1001_8000~0x1001_9000.

WDT, external pin, software reset, etc. will not clear SRAM data, but it should be noted that BOOTROM uses the address space of 92KB~96KB. Once the system is reset, the CPU will execute the boot program from BOOTROM, and the data in this address space will be occupied. Please note this feature when using this address space.

4.3.4 FLASH

Maximum 512KB FLASH on chip:

- At least 100,000 erasures;
- The data will be stored for at least 20 years;
- The vault contains 32 blocks, each block contains 32 pages, each page contains 512Bytes;
- 8-bit, 16-bit and 32-bit random reads;
- Page erase, block erase, page programming, specific operations need to call library functions of Renergy (nvm.a(IAR)/nvm.lib(KEIL));
- FLASH will automatically turn off or on when low power consumption is applied;

The FLASH operation function interface provided by library function (nvm.a(IAR)/nvm.lib(KEIL) is as follows:

uint8_t flashPageErase(uint32_t pg)
uint8_t flashSectorErase(uint32_t sec)
uint8_t flashProgram(uint32_t dst_addr, uint32_t src_addr, uint32_t len)

4.4 Interrupt Application

SOC header file #include < rn8x1x.h > in the header file, you can use SOC interrupt, rn821x.h file contains cortex-m0 defined part of the header file, core_cmfunc.h, core_cmfunc.h, core_cmInstr. H.All the above documents can be found in the header file provided by the Renergy.

Close interrupt enable:__disable_irq();

Enable total interruption:__enable_irq();

Interrupt operation

The interrupt program of each module can be written completely in C language, and users do not need to consider the problems of pushing and pushing. The interrupt operation steps are as follows:

Take the KBI interrupt as an example:

1、Total interruption of enable:__enable_irq();

2、Configure modules that need to generate interrupts, such as the KBI module, and set KBI_MASK to enable interrupts.

3、Enable KBI interrupt: find the interrupt number in rn821x.h file and start the interrupt, for example, KBI interrupt number is KBI_IRQn, start KBI interrupt is NVIC_EnableIRQ(KBI_IRQn), if you need to set the interrupt priority, use void NVIC_SetPriority(IRQn_t IRQn, uint32_t priority).

4、For different interrupts, the function name has been fixed and can be found in startup_rn821x.s. For example, the KBI interrupt service program function named KBI_HANDLER can be written as:

```
void KBI_HANDLER(void)
                  {
                     /* Start adding user code. Do not edit comment generated here */
                  }
```

5、Close interrupt enable: void NVIC_DisableIRQ0 (IRQn_t IRQn).

5 RTC

5.1 Overview

The BaseAddr is 0x4003C000;

RTC module provides real-time clock, oscillator temperature compensation, calendar, alarm clock, clock pulse output, etc.

The real-time clock tracks time with independent registers for hour, minute and second. The calendar includes year, month, day and week register, with leap year leap month automatic correction function. Clock pulse output provides multiple frequencies for clock calibration and alarm.

Integrated temperature sensor provides digital results of temperature measurements, which is powered by VBAT.

5.2 Theatures

- Accurate temperature measurement, accuracy up to ±1℃ between -25℃～70℃
- Initial calibration of RTC at room temperature
- Automatic completion of RTC temperature compensation operation without CPU participation
- Low-power consumption
- Frequency adjustment accuracy up to 0.0339ppm
- Oscillator with high stability
- RTC does not turn off in different modes and still works normally at low power consumption
- Provide clock and calendar functions: output registers include seconds, minutes, hours, dates, months, years and weeks, etc.
- Automatic leap year and leap month adjustment function, timing range 100 years (00-99)
- 1 crystal stop interrupt, 1 alarm clock interrupts, 2 timer periodic interrupts, 5 time interrupts
- Output uncorrected frequency 32768Hz
- Output corrected frequency 1Hz/4Hz/8Hz/16Hz/(1/30Hz)
- Added RTC quartic curve warm compensation. For detailed instructions, please refer to Renergy Application Notes.

5.3 Register Description

Basic address of RTC module

Name	**Physical Address**	**Mapping Address**
RTC	BaseAddr: 0x4003C000	Base1

Register offset of RTC module

Name	**Offset**	**Description**
RTC Registers		
RTC_CTL	Offset+0x00	RTC control register
RTC_SC	Offset+0x04	Register of second, write protection
RTC_MN	Offset+0x08	Register of minute, write protection

RTC_HR	Offset+0x0C	Register of hour, write protection
RTC_DT	Offset+0x10	Register of day, write protection
RTC_MO	Offset+0x14	Register of month, write protection
RTC_YR	Offset+0x18	Register of year, write protection
RTC_DW	Offset+0x1C	Register of week, write protection
RTC_CNT1	Offset+0x20	Register of counter 1
RTC_CNT2	Offset+0x24	Register of counter 2
RTC_SCA	Offset+0x28	Register of alarm (second)
RTC_MNA	Offset+0x2C	Register of alarm (minute)
RTC_HRA	Offset+0x30	Register of alarm (hour)
RTC_IE	Offset+0x34	Register of RTC interruption
RTC_IF	Offset+0x38	Register of RTC status
RTC_TEMP	Offset+0x3C	Register of current temperature, Readable/writable,write protection
RTC_TEMP2	Offset+0xF8	Register 2 of current temperature, 12bit, read only
RTC_DOTA0	Offset+0x48	Register of RTC initial deviation correction
LOSC_CFG1	Offset+0x6C	Register of 32768 oscillator settings
RTC_CALPS	Offset+0xCC	Register of RTC secondary compensation, write protection, write 8’hA8 to enable Register T0~T9
RTC_CAL_T0	Offset+0xD0	T0~T9 are 8bit registers. Based on hardware automatic temperature compensation, the error of RTC is compensated twice. The scale is 0.25ppm. Two's complement format. Compensation temperature range: T < -30 ℃
RTC_CAL_T1	Offset+0xD4	Compensation temperature range:-30<= T <-20℃
RTC_CAL_T2	Offset+0xD8	Compensation temperature range:-20<= T < -10℃
RTC_CAL_T3	Offset+0xDC	Compensation temperature range:-10<= T < 0℃

RTC_CAL_T4	Offset+0xE0	Compensation temperature range: 0<= T <= 10℃
RTC_CAL_T5	Offset+0xE4	Compensation temperature range: 35< T <=45℃
RTC_CAL_T6	Offset+0xE8	Compensation temperature range: 45<T <= 55℃
RTC_CAL_T7	Offset+0xEC	Compensation temperature range: 55< T <=65℃
RTC_CAL_T8	Offset+0xF0	Compensation temperature range: 65< T <=75℃
RTC_CAL_T9	Offset+0xF4	Compensation temperature range: 75< T < =85℃
LOSC_CFG1	Offset+0x6C	Register of LOSC configuration
VBAT_IOMODE	Offset+0x88	Register of IO pin configuration in VBAT domain, for P44, P45 and P56, refer to Register Description for more details
VBAT_IOWEN	Offset+0x8C	Register of VBAT_IOMODE write enable configure this register before writing VBAT_IOMODE. Please refer to Register Description for more details
P44N0_TIME0	Offset+0x90	Register of P44 port freeze time at the first falling edge
P44N0_TIME1	Offset+0x94	Register of P44 port freeze time at the first falling edge
P44N0_TIME2	Offset+0x98	Register of P44 port freeze time at the first falling edge
P44P0_TIME0	Offset+0x9C	Register of P44 port freeze time at the first rising edge
P44P0_TIME1	Offset+0xA0	Register of P44 port freeze time at the first rising edge
P44P0_TIME2	Offset+0xA4	Register of P44 port freeze time at the first rising edge
P44N1_TIME0	Offset+0xA8	Register of P44 port freeze time at the second falling edge
P44N1_TIME1	Offset+0xAC	Register of P44 port freeze time at the second falling edge
P44N1_TIME2	Offset+0xB0	Register of P44 port freeze time at the second falling edge
P44P1_TIME0	Offset+0xB4	Register of P44 port freeze time at the second rising edge
P44P1_TIME1	Offset+0xB8	Register of P44 port freeze time at

		the second rising edge
P44P1_TIME2	Offset+0xBC	Register of P44 port freeze time at the second rising edge
P45N0_TIME0	Offset+0x100	Register of P45 port freeze time at the first falling edge
P45N0_TIME1	Offset+0x104	Register of P45 port freeze time at the first falling edge
P45N0_TIME2	Offset+0x108	Register of P45 port freeze time at the first falling edge
P45P0_TIME0	Offset+0x10C	Register of P45 port freeze time at the first rising edge
P45P0_TIME1	Offset+0x110	Register of P45 port freeze time at the first rising edge
P45P0_TIME2	Offset+0x114	Register of P45 port freeze time at the first rising edge
P45N1_TIME0	Offset+0x118	Register of P45 port freeze time at the second falling edge
P45N1_TIME1	Offset+0x11C	Register of P45 port freeze time at the second falling edge
P45N1_TIME2	Offset+0x120	Register of P45 port freeze time at the second falling edge
P45P1_TIME0	Offset+0x124	Register of P45 port freeze time at the second rising edge
P45P1_TIME1	Offset+0x128	Register of P45 port freeze time at the second rising edge
P45P1_TIME2	Offset+0x12C	Register of P45 port freeze time at the second rising edge

5.3.1 RTC Control Register RTC_CTL (0x00)

Offset Address 0x00

Bit	Name	Description	Read/Write Flag	Reset Value
31:12	---	Reserved	R	0
11	---	Reserved	R/W	0
10	Cal_busy	RTC correction calculation busy flag =1 RTC is performing calibration calculations; =0 RTC correction calculation completed. Read only	R	0
9	Wr_busy	RTC Register Write Busy; Write multiple registers without waiting. Read after write, wait for the busy bit to change from 1 to 0 before reading back the correct value. Therefore, it is recommended that applications wait for 300us before reading	R	0

		after writing the perpetual calendar.		
8	WRTC	Allowable RT register group writing: 0: disable RTC register write operation; 1: allow RTC register write operation. Note: this bit is valid for RTC register group 00 ~ 1C/3C/C4, and for RTC < CTL [7:0]. There are two ways to write the time register of the perpetual calendar: Write follows the sequence of "month, day, hour, minute, second". When the second register is written, the time starts to accumulate from the time of writing. Note that this method may flip the minute before the second is written, so read it out for verification after writing; Write follows the sequence of "seconds, minutes, months, days", write the second register first, and the perpetual calendar counter is cleared. As long as other values are written in one second, it can ensure successful writing. Note 1: the hardware has made a legal judgment on "year month day", which can not be written in the order of "day month year", only "year month day" can be written continuously. Note 2: the time registers of the perpetual calendar must be written in the above order, and the complete 6 registers (month, day, hour, minute and second) must be written in a time write, instead of just updating the part (such as only writing minutes and seconds).	R/W	0
7:6	TSE	Allowable bit of temperature sensor 00: prohibit automatic temperature compensation 01: start automatic temperature compensation. Conduct periodic temperature compensation according to the settings of TCP. 10: start the user's temperature compensation mode 0, the temperature register can be changed, and the user fills in the temperature value, and the user starts the temperature compensation every time he writes the temperature register; 11: start users' temperature compensation mode 1, the temperature register cannot be changed, and the temperature compensation operation will be started every time the temperature register is written. The value of the temperature register is measured by MCU. Note: this register only works with power on reset.	R/W	00
5:3	TCP	Temperature Compensation Cycle: 000:2s 001:10s Default 010:20s 011:30s	R/W	001

		100:1min 101:2min 110:5min 111:10min		
02:00	FOUT	000:Output Disable 001:1Hz Outputs 010:1/30Hz Outputs 011:32768Hz Outputs 100:16Hz Outputs 101:8Hz Outputs 110:4Hz Outputs 111:Reserved This register only works on power-on reset.	R/W	000

5.3.2 Register of second RTC_SC (0x04)

Offset Address: 0x04

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:07	Reserved	Reserved	R	0
06:00	SC	Store the second value of the clock. BCD Format, SC [6:4] is the tens of the second value, SC[3:0] is the units of the second value. The second value ranges from 0~59.	R/W	-

5.3.3 Register of minute RTC_MN (0x08)

Offset Address: 0x08

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:07	Reserved	Reserved	R	0
06:00	MN	Store the minute value of the clock. BCD Format, **MN [6:4]** are the tenth of minute value, **MN [3:0]** are the bits of minute value. The Rang of Minute Value is 0~59.	R/W	-

5.3.4 Register of hour RTC_HR (0x0c)

Offset Address: 0x0C

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:06	Reserved	Reserved	R	0
05:00	HR	Store the hour value of the clock. BCD Format, **HR [5:4]** are the tenth of hour value, **HR[3:0]** are the bits of hour value. The Rang of Hour Value is 0~23.	R/W	-

5.3.5 Register of date RTC_DT (0x10)

Offset Address 0x10

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:06	Reserved	Reserved	R	0
05:00	DT	Store the date value of the clock. BCD Format, **DT [5:4]** are the tenth of date value, **DT [3:0]** are the bits of date value. The Rang of date value is 1~31.	R/W	-

5.3.6 Register of month RTC_MO (0x14)

Offset Address 0x14

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:05	Reserved	Reserved	R	0
04:00	MO	Store the month value of the clock. BCD Format, **MO [5:4]** are the tenth of month value, **MO [3:0]** are the bits of month value. The rang of month value is 1~12.	R/W	-

5.3.7 Register of year RTC_YR (0x18)

Offset Address 0x18

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:08	Reserved	Reserved	R	0
07:00	YR	Store the year value of the clock. BCD Format, **YR [7:4]** are the tenth of year value, **YR [3:0]** are the bits of year value. The rang of year value is 0~99.	R/W	-

5.3.8 Register of week RTC_DW (0x1c)

Offset Address 0x1C

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:03	Reserved	Reserved	R	0
02:00	DW	Store the week corresponding to the current date. The counting cycle of **DW [2:0]** is 0-1-2-3-4-5-6-0-1-2-....	R/W	-

5.3.9 Register of RTC Timer 1 RTC_CNT1 (0x20)

Offset Address 0x20

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:09	Reserved	Reserved	R	0
08	CNT1PD	=0: 1 second timing from the perpetual calendar second interrupt. =1: turn off timer 1. When it is turned on again, the counter will start again (the scale is 1s, independent of the	R/W	0

		interruption of the perpetual calendar second)		
07:00	CNT	Timer 1 counter preset. An unsigned number in 1s. When the count value = (CNT + 1), set the RTCCNT1F (interrupt can be generated every 1 second at least and every 256 seconds at most). Note1: The timer is accurate after RTC correction. Note2: When RTC-IE-> RTC_1S_SEL is 0, i.e. pll_1hz is selected as the second interrupt source, the second interrupt is not synchronized with the update of the second register. The second interrupt may not be generated at the start of the second count, but at any time of the second count. When RTC-IE-> RTC_1S_SEL is 1, i.e. RTC second pulse is selected as the second interrupt source, the second interrupt is consistent with the second register update. It is recommended to select RTC-IE-> RTC_1S_SEL as 1.	R/W	0

5.3.10 Register of RTC Timer 1 RTC_CNT2 (0x24)

Offset Address 0x24

Bit	Name	Description	Read/Write Flag	Reset Value
31:10	Reserved	Reserved	R	0
09	CNT2_CLK_SEL	Colck source of CNT2_CLK. 0:32768Hz 1:RCL	R/W	0
08	CNT2PD	= 0: timer from internal fixed 1 / 256s interrupt. = 1: turn off timer 2. When it is turned on again, the counter will start again (scale 1 / 256s).	R/W	0
07:00	CNT	Timer 2 counter preset An unsigned number in 1/256s. When the count value = (CNT + 1), set the RTCCNT2F (interrupt can be generated once every 1 / 256 seconds at least and once every 1 second at most). Note: the timer is from 32768Hz crystal, uncorrected, with certain error.	R/W	0

5.3.11 Register of Second Alarm RTC_SCA (0x28)

Offset Address 0x28

Bit	Name	Description	Read/Write Flag	Reset Value
31:07	Reserved	Reserved	R	0
06:00	SCA	Second alarm clock value. BCD Format, **SCA [6:4]** is the tens of the second value, **SCA[3:0]** is the units of the second value. The second value	R/W	0

		ranges from 0~59.		

5.3.12 Register of Minute Alarm RTC_MNA (0x2c)

Offset Address 0x2C

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:07	Reserved	Reserved	R	0
06:00	MNA	Minute alarm clock value. BCD Format, MNA [6:4] are the tenth of minute value, MNA[3:0] are the bits of minute value. The Rang of Minute Value is 0~59.	R/W	0

5.3.13 Register of Hour Alarm RTC_HRA (0x30)

Offset Address 0x30

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:06	Reserved	Reserved	R	0
05:00	HRA	Hour alarm clock value BCD Format, HRA [5:4] are the tenth of hour value, HRA [3:0] are the bits of hour value. The Rang of Hour Value is0~23.	R/W	0

Note:04~30H register has no reset value.

5.3.14 RTC Interrupt Enable Register RTC_IE (0x34)

Offset Address 0x34

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:10	Reserved	Reserved	R	0
9	RTC_1S_SEL	Second interrupt source selection: =0: according to the system clock mode: select pll_1hz in high frequency mode and RTC second pulse in non high frequency mode. =1: always select the second pulse of RTC. Note: the update of pll_1hz and second register is not synchronized, and the second pulse of RTC is synchronized with the second register. It is recommended that the customer configure the register as 1 and select the second pulse of RTC.	R/W	0
08	IECLKEN	RTC interrupt generate clock enable; When anyone of the RTC [IE 8-0] is high, the interrupt module clock would open; When anyone of the RTC [IE 8-0] is low, the interrupt	R/W	0

		module clock would close;		
7	MOIE	Month Interrupt Enable 0 : Disable 1 : Enable	R/W	0
6	DTIE	Date Interrupt Enable 0 : Disable 1 : Enable	R/W	0
5	HRIE	Hour Interrupt Enable 0 : Disable 1 : Enable	R/W	0
4	MNIE	Minute Interrupt Enable 0 : Disable 1 : Enable	R/W	0
3	SCIE	Second Interrupt Enable 0 : Disable 1 : Enable Note: When RTC-IE-> RTC_1S_SEL is 0, i.e. pll_1hz is selected as the second interrupt source, the second interrupt is not synchronized with the update of the second register. The second interrupt may not be generated at the start of the second count, but at any time of the second count. When RTC-IE-> RTC_1S_SEL is 1, i.e. RTC second pulse is selected as the second interrupt source, the second interrupt is consistent with the second register update. It is recommended to select RTC-IE-> RTC_1S_SEL as 1.	R/W	0
2	RTCCNT2IE	RTC Timer2 Interrupt Enable 0 : Disable 1 : Enable	R/W	0
1	RTCCNT1IE	RTC Timer1 Interrupt Enable 0 : Disable 1 : Enable	R/W	0
0	ALMIE	Alarm Event Interrupt Enable 0 : Disable 1 : Enable	R/W	0

5.3.15 Register of RTC Interrupt Flag RTC_IF (0x38)

Offset Address 0x38

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:12	Reserved	Reserved	R	0
7	MOF	Month Interrupt Flag 0 : Month Counter not plus 1 1 : Month Counter plus 1 Note: Write 1 Cleared	R/W	0

6	DTF	Date Interrupt Flag 0 : Date Counter not plus 1 1 : Date Counter plus 1 Note: Write 1 Cleared	R/W	0
5	HRF	Hour Interrupt Flag 0 : Hour Counter not plus 1 1 : Hour Counter plus 1 Note: Write 1 Cleared	R/W	0
4	MNF	Minute Interrupt Flag 0 : Minute Counter not plus 1 1 : Minute Counter plus 1 Note: Write 1 Cleared	R/W	0
3	SCF	Second Interrupt Flag 0 : Second Counter not plus 1 1 : Second Counter plus 1 Note: Write 1 Cleared	R/W	0
2	RTCCNT2F	RTC Timer2 Interrupt Flag 0 : Timer1 Interrupt Unhappen 1 : Timer1 Interrupt Happen Note: Write 1 Cleared	R/W	0
1	RTCCNT1F	RTC Timer1 Interrupt Flag 0 : Timer1 Interrupt Unhappen 1 : Timer1 Interrupt Happen Note: Write 1 Cleared	R/W	0
0	ALMF	Alarm Event Flag, alarm event that matches with real-time clock occurs 0 : Alarm Event Unhappened 1 : Alarm Event Happened Note: Write 1 Cleared	R/W	0

5.3.16 Register of Current Temperature RTC_TEMP (0x3c)

Offset Address 0x3C

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:10	Reserved	Reserved	R	0

09:00	TEMP	The Present Temperature Value.
		Bit 9 is Sign Bit; Bit 8~2 are Integer Bits; Bit 1~0 are Decimal Place.
		// Temp[9] Temp[8:2] Temp[1] Temp[0]
		// Symbol -128℃~127℃ 0.5℃ 0.25℃
		Range: -128℃(0x200)~+127.75℃(0x1ff).
		Temperature conversion formula: if the symbol bit is 0, the temperature is TEMP/4. If the symbol bit is 1, the temperature is (2^10- TEMP)/4.
			R/W	-

		TSE=00: Disable temperature compensation automatic. This register RTC_TEMP is invalid, read the value is meaningless. TSE=01: Carry out automatic temperature compensation according to the cycle set by RTC_CTL_> TCP. At this time, RTC_TEMP register displays the temperature value of the temperature compensation period. The update period of RTC_TEMP register is the temperature compensation period set by RTC_CTL _> TCP. TSE=10: Start user warm compensation mode 0. At this time, RTC_TEMP register can be changed, and user fills in the temperature value. User starts the temperature compensation every time he writes the temperature register. TSE=11: Start user warm mode 1. At this time, RTC_TEMP register cannot be changed. Each time the temperature register is written, the temperature compensation operation is started. The value of the RTC_TEMP register is measured by the SOC.		

RTC automatic temperature compensation needs to define the following registers, and these register values are obtained in the mass production of customers.

1. Initial frequency deviation register RTC_DOTA0: modify the initial frequency deviation of the crystal; (each table needs to be obtained, and the library function provided by Renergy can complete the operation of this register).
2. Quadratic curve vertex temperature register RTC_XT0 (get crystal batch parameters, configure option bytes, write through programming interface).
3. Crystal temperature coefficient register RTC_ALPHA (get crystal batch parameters, configure option bytes, write through programming interface).

5.3.17 Register of Current Temperature 2 RTC_TEMP2 (0xF8)

Offset = 0xF8

Bit	Name	Description	Read/Write Flag	Reset Value
31:12	---	Reserved	R	0

11:00	TEMP2	Current temperature.
		Bit11 is sign bit; Bit10~4 are integer bits; Bit3~0 are decimal bits.
		// Temp[11] Temp[10:4] Temp[3] Temp[2] Temp[1] Temp[0]
		// Symbol -128℃~127℃ 0.5℃ 0.25℃ 0.125℃ 0.0625℃
		Range:-128℃~+127.75℃
			R	-

5.3.18 Register of initial frequency deviation correction RTC_DOTA0 (0x48)

Bit	Name	Description	Read/Write Flag	Reset Value
31:13	---	Reserved	R	0

12:0	DOTA0	The correction value DOTA0 is in signed twos complement form, bit12 is the sign bit: DOTA0 = round(σ_0*32), σ_0 is the initial frequency deviation, unit: ppm, can be expressed in the range: -128ppm ~127.96875ppm; Resolution of DOTA0 is 0.03125ppm. For example, the initial frequency deviation σ_0=-6ppm, correction value DOTA0= 0x1f 40. Note: Most of the 32768 crystals have a frequency deviation of ±20ppm typical and ±30ppm maximum)	R/W	0

Note: Only an up or down reset will restore the register to its default value; other resets will not.

When user calibrates initial frequency deviation of RTC, It is recommended to call Renergy library function “void RtcWriteDota(uint16_t dota);” , which not only calibrates DOTA0 register, but also effectively manages vbat domain configuration register, which can ensure that Vbat domain related parameters are written correctly, which is especially important for applications where Vbat power-on is slower than Vcc.

5.3.19 Register of LOSC configuration LOSC_CFG1 (0x6c)

Bit	Name	Description	Read/Write Flag	Reset Value
31:11	---	Reserved	R	0
10	VCC_BOI_FLAG	=0: normal; =1: Power failure occurred in VCC power domain; Write 1 to reset	R/W	0
9	LOSC_FAIL	=0: normal; =1: LOSC has stopped vibration Write 1 to reset	R	0
8	LOSC_WEN	=0: LOSC_PD is not writable =1: LOSC_PD writable	R/W	0
7:0	LOSC_PD	LOSC enable bit: =Others: open; =8'ha8: off. Note: it is also the external clock enable signal, = 1 enable external clock. LOSC_CFG1 = (1<<8); LOSC_CFG1 = (0xa8<<0);	R/W	0

Note: If an external 32.768 kHz clock is used, please inject it from XI and turn off the chip's oscillation circuit (i.e. turn off the LOSC enable bit).

5.3.20 Register of IO pin configuration in VBAT domain VBAT_IOMODE (0x88)

Bit	Name	Description	Read/Write Flag	Reset Value

31:12	---	Reserved	R	0
11:8	P45_KEY	Decap detection (P45) configuration =4’h5, Enable autolog function of Key P45; =4’hE, Clear Register 0xC0 ~ 0xEC; =others, disable key autolog. This function is compatible with the input function related to port P45 of GPIO chapter.	R/W	0
7:4	P56_RTCOUT	=4'ha, enable P56 port to output RTC_out function; =Other values, P56 functions are defined by GPIO chapter registers. The priority of this function is higher than the reuse configuration of P56 in GPIO chapter. When VBAT is powered on independently and VCC is not powered on, P56 outputs 1Hz by default; When VCC is powered on, CPU initialization will configure this register to 0.	R/W	0
3:0	P44_KEY	Decap detection (P44) configuration = 4’h5, Enable autolog function of Key P44 ; = 4’hE, Clear Register 0x90~ 0xBC; = Others, This function is compatible with the input function related to port P44 of GPIO chapter.	R/W	0

5.3.21 Register of write enable for IO pin configuration in VBAT domain VBAT_IOWEN (0x8C)

Bit	Name	Description	Read/Write Flag	Reset Value
31:8	---	Reserved	R	0
7:4	IOWEN	=4’hB, Enable VBAT_IOMODE register write operation; =others, disable VBAT_IOMODE register write operation;	R/W	0
3:2	---	Reserved	R	0
1	P45_KEYMODE	Bit1:Decap detection (P45) mode configuration	R/W	0

		0: Record the first two keypress events after IO keypress is enabled; 1: Record the latest two key events after IO key is enabled, and the latest two records do not distinguish the time sequence stored in cache, which shall be judged by software. KEYMODE should be configured before KEY function is enabled.		
0	P44_KEYMODE	IO pin mode selection: Bit0:Decap detection (P44) mode configuration; 0:Record the first two keypress events after IOEN is enabled; 1: Record the latest two keypress events after IOEN is enabled, and the latest two records do not distinguish the time sequence stored in the cache, which shall be judged by the software. KEYMODE should be configured before KEY function is enabled.	R/W	0

5.3.22 Registers of Quadratic Piecewise Compensation

5.3.22.1 Register of passwoed for Quadratic Piecewise Compensation RTC_CALPS (0xCC)

Bit	Name	Description	Read/Write Flag	Reset Value
31:8	---	Reserved	R	0
7:0	CALPS	Write 8’hA8 to enable Register T0~T9	R/W	0

5.3.22.2 Registers of Quadratic Piecewise Compensation RTC_CAL_T0~T9 (0xD0~0Xf4)

Bit	Name	Description	Read/Write Flag	Reset Value
31:8	---	Reserved	R	0
7:0	CAL_T0~T9	T0~T9 are 8bit registers, which make secondary compensation for RTC error on the basis of hardware automatic temperature compensation, and the scale is 0.25ppm; T0 Compensation temperature range: T < -30℃ T1 Compensation temperature range: -30 ℃ <= T <-20℃ T2 Compensation temperature range: -20 ℃ <= T <	R/W	0

		-10℃ T3 Compensation temperature range: -10 ℃ <= T < 0℃ T4 Compensation temperature range: 0 ℃ <= T <= 10℃ T5 Compensation temperature range: 35 ℃ < T < =45℃ T6 Compensation temperature range: 45 ℃ < T <= 55℃ T7 Compensation temperature range: 55 ℃ < T < =65℃ T8 Compensation temperature range: 65 ℃ < T < =75℃ T9 Compensation temperature range: T >75℃ Note: In actual design, 0.1875℃ should be added to the above range, but it is not required to be reflected in the application layer. The manual shall follow the above Description.		

5.3.23 Registers of Decap Detection

5.3.23.1 First freeze time of P44

Freezing time data 16 bits valid, all in BCD code format.

Address	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
0x90	P44N0_TIME0	P44 First falling edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0x94	P44N0_TIME1	P44 First falling edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0x98	P44N0_TIME2	P44 First falling edge freeze time year, month register [15:8]:year [7:0]:month	R	0
0x9C	P44P0_TIME0	P44 First rising edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0xA0	P44P0_TIME1	P44 First rising edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0xA4	P44P0_TIME2	P44 First rising edge freeze time year, month register [15:8]:year [7:0]:month	R	0

5.3.23.2 Second freeze time of P44

Freezing time data 16 bits valid, all in BCD code format.

地址	Name	Description	Read/Write Flag	Reset Value
0xA8	P44N1_TIME0	P44 Second falling edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0xAC	P44N1_TIME1	P44 Second falling edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0xB0	P44N1_TIME2	P44 Second falling edge freeze time year, month register [15:8]:year [7:0]:month	R	0
0xB4	P44P1_TIME0	P44 Second rising edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0xB8	P44P1_TIME1	P44 Second rising edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0xBC	P44P1_TIME2	P44 Second rising edge freeze time year, month register [15:8]:year [7:0]:month	R	0

5.3.23.3 First Frezzing Time of P45

Freezing time data 16 bits valid, all in BCD code format.

地址	Name	Description	Read/Write Flag	Reset Value
0x100	P45N0_TIME0	P45 First falling edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0x104	P45N0_TIME1	P45 First falling edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0x108	P45N0_TIME2	P45 First falling edge freeze time year, month register [15:8]:year [7:0]:month	R	0
0x10C	P45P0_TIME0	P45 First rising edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0x110	P45P0_TIME1	P45 First rising edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0

0x114	P45P0_TIME2	P45 First rising edge freeze time year, month register [15:8]:year [7:0]:month	R	0

5.3.23.4 Second Frezzing Time of P45

Freezing time data 16 bits valid, all in BCD code format.

地址	Name	Description	Read/Write Flag	Reset Value
0x118	P45N1_TIME0	P45 Second falling edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0x11C	P45N1_TIME1	P45 Second falling edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0x120	P45N1_TIME2	P45 Second falling edge freeze time year, month register [15:8]:year [7:0]:month	R	0
0x124	P45P1_TIME0	P45 Second rising edge freeze time minute, second register [15:8]:minute [7:0]:second	R	0
0x128	P45P1_TIME1	P45 Second rising edge freeze time day, hour register [15:8]:day [7:0]:hour	R	0
0x12C	P45P1_TIME2	P45 Second rising edge freeze time year, month register [15:8]:year [7:0]:month	R	0

5.4 RTC Clock Read/Write Steps

1. Set the 10th bit RTC_EN of the module enable 1 register MOD1_EN in the system control chapter to 1.
2. Clock reading: read the second, minute, hour and other time registers of RTC.
3. Clock writing:

Set the 8th WRTC of RTC_CTL to 1, and turn on the write enable operation.

Write according to the sequence of "month, day, hour, minute and second". When the second register is written, the time begins to accumulate from the time of writing. Note that this method may flip the minute before the second is written, so read it for verification after writing.

Or write according to the sequence of "seconds, minutes, months, days", write the second register first, and the perpetual calendar counter is cleared. As long as other values are written in one second, the successful writing can be guaranteed.

Note that the hardware has made a legitimacy judgment on "years months days", which can not be written in the order of "days months years", but can only write "years months days" continuously.

In order to enhance the reliability of software, it is recommended to read and confirm after writing.

5.5 RTC Calibration Step

The user only needs to calibrate the initial deviation of 32.768 KHz. Errors are written through the Renergy programmer interface or using library functions.

5.6 RTC Timer operation steps

Take timer 1 generating 1S interrupt as an example, the operation steps are as follows:

1. Set Module Enable 1 Register MOD1_EN Bit 10 RTC_EN to 1 Clock Start in the System Control section;
2. Turn on write enable operation by setting WRTC to bit 8 of RTC_CTL;
3. Set RTC-> CNT1 = 0x00; i.e. generate 1 interrupt every 1s;
4. Set RTC->IE = 0x02;RTC Timer 1 interrupt enabled;
5. Enable RTC interrupt, NVIC_EnableIRQ(RTC_IRQn);
6. Write interrupt service routines:

```
 void RTC_HANDLER (void)
 {
        if (RTC->IF&0x02)               // timmer 1
        {
        /* Start adding user code. Do not edit comment generated here */
        }
 }
```

7. After configuration is completed, a 1s interrupt can be generated.

6 WDT

There is a built-in hardware watchdog for detecting abnormal execution of programs.

6.1 Overview

The watchdog has the following characteristics:

- Overflow time can be: 16ms、32ms、128ms、512ms、1s、2s、4s、8s;
- Feeding window period can be set;

Watchdog reset occurs when any of the following situations occur:

- Watchdog timer counter overflow;
- Write data other than 0xBB to WDT_EN;
- Write data to WDT_EN during close of dog feeding window;
- Write data to WDT_EN via bitband space;

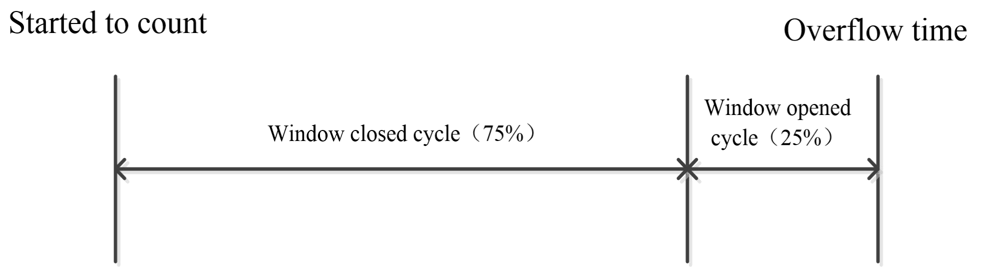
6.2 Configuration of Watchdog Timer

WDT is a hardware watchdog and cannot be configured directly through registers. It needs to be configured by setting "option byte". Watchdog configuration has interval interrupt, window open cycle, overflow time, CPU sleep settings, CPU debug settings and other options.

Name	Description	Default
Interval interrupt	0:Disable 1:Enable(reach 75% of spill events)	0
Window open cycle	0:25% 1:50% 2:75% 3:100% Write 0xBB to WDTE register while window is open, watchdog clears and recalculates; Writing 0xBB to the WDTE register during window closure generates an internal reset signal.	3
Overflowing time	0:16ms 1:32ms 2:128ms 3:512ms 4:1s 5:2s 6:4s 7:8s Note: Internal low-frequency RCL timing is used. The specific time is not accurate, but the overflow time can be guaranteed to be longer than the above standard time..	4
CPU sleep settings	0:Disable (WDT is not turned on when CPU is in sleep or deep sleep) 1:Enable (Turn on WDT when CPU is in sleep or deep sleep)	0
CPU debug	0:Disable (WDT is not turned on when CPU is in debug status)	0

settings	1:Enable (Turn on WDT when CPU is in debug status) Note: CPU in debug status means the user holds Cortex M0 through debug interface (PC pointer stops counting). Enabling this setting is not recommended if the chip is under development. Because if this setting is enabled, WDT will still count when the chip is in debug status, and an interrupt will be generated when overflow, which will cause debugging to fail.	

The window open cycle is defined as shown in the figure below, taking a window open cycle of 25% as an example.:



6.3 Register Description

WDT Register Base Address

Name	Physical Address	Mapping Address
WDT	0x40030000	0x40030000

WDT Register Offset Address

Name	Offset Address	Description
WDT_EN	0x0	enabled 寄存器

6.3.1 WDT_EN (0x0)

Bit	Name	Description	Read/Write Flag	Reset Value
31:9	---	Reserved	R	0
8	WR_BUSY	WDT busy Watchdog feeding is not related with WDT.	R	0
7:0	WDTE	Writing 0xBB clears the watchdog timer and restarts counting. Generation of the reset signal sets this register to 0x55	R/W	55

6.4 WDT Configuration Steps

1. Set Bit 9 of register MOD1_EN to 1 to turn on WDT APB clock;
2. WDT default configuration is start, timer overflow time is 1S, and window open cycle is 75%. WDT initializing configuration is not necessary.
3. Watchdog feeding : WDT->EN = 0xbb;
4. WDT will be turned off after sleep by default. The WDT clock in register MOD1_EN may not be turned off. WDT clock can be turned off only after WR_BUSY is 0 in bit 8 of WDT_EN.
5. When hardware emulation stops the program, the WDT count is also paused without affecting hardware

emulation.

6. Completed.

Recommendation:

Due to the extremely low power consumption of WDT, the additional power consumption increased by turning on WDT when CPU sleeps is less than 1uA. From the perspective of higher system reliability, it is recommended that customers turn on WDT when CPU sleeps, and use the second timer in RTC to wake up CPU for watchdog feeding operation.

7 LCD

MCU built-in segment code LCD controller;
Maximum 8COM*36SEG;
Charge pump and resistance voltage divider to drive.

7.1 Overview

LCD has the following characteristics:

- RN8318: 4*40、6*38、8*36;RN8615/RN8613/RN8612: 4*34、6*32、8*30;
- Two types of driving waveform: Type A and Type B;
- 1/3bias and 1/4bias;
- static, 1/2, 1/3, 1/4, 1/6, 1/8duty;
- 16-level contrast-adjustable LCD drive mode;
- After LCD module is turned off, all COM and SEG pins configured are automatically pulled down to ground. (Note: Please keep COM and SEG status unchanged when stopping display, do not switch pin multiplexing relationship to IO port, and do not change bias and duty configuration.).
- RN8318/RN8615/RN8613 provide two way for LCD Bias voltage dividing: charge pump and internal resistor string;
- RN8612 only provides internal resistor string.

7.1.1 Sweep Clock Frequency

The LCD unit is clocked from fosc or RCL (determined by the system-controlled OSC_CTL2 register) at a frequency of 32768 Hz. Fosc or RCL is divided as the LCD waveform sweep frequency flcd, which can be configured via the register LCD_CLKDIV [7:0].

Ffram = flcd/(number of coms). Generally, the refresh frequency of LCD screen is slightly higher than 60Hz.

Table 7-1Clock configuration of different duty cycles (The frequency marked in green is normally used)

LCD_CLKDIV	**Frequency**	**Static duty cycle**	**1/2 duty cycle**	**1/3 duty cycle**	**1/4 duty cycle**	**1/6 duty cycle**	**1/8 duty cycle**
0xff	64Hz	🟩 64Hz	32Hz	21.3Hz	16Hz	10.7Hz	8Hz
0x7f	128Hz	128Hz	🟩 64Hz	42.7Hz	32Hz	21.3Hz	16Hz

0x54	192.8Hz	192.8Hz	96.4Hz	64.3Hz	48.2Hz	32.1Hz	24.0Hz
0x3f	256Hz	256Hz	128Hz	85.3Hz	64Hz	42.7Hz	32Hz
0x2a	381.3Hz	381.3Hz	190.5Hz	127.0Hz	95.3Hz	63.5Hz	47.6Hz
0x1f	512Hz	512Hz	256Hz	170.7Hz	128Hz	85.3Hz	64Hz

7.1.2 Blink Mode

LCD supports two flashing modes: inner flashing and outer flashing. Both modes can be enabled simultaneously.

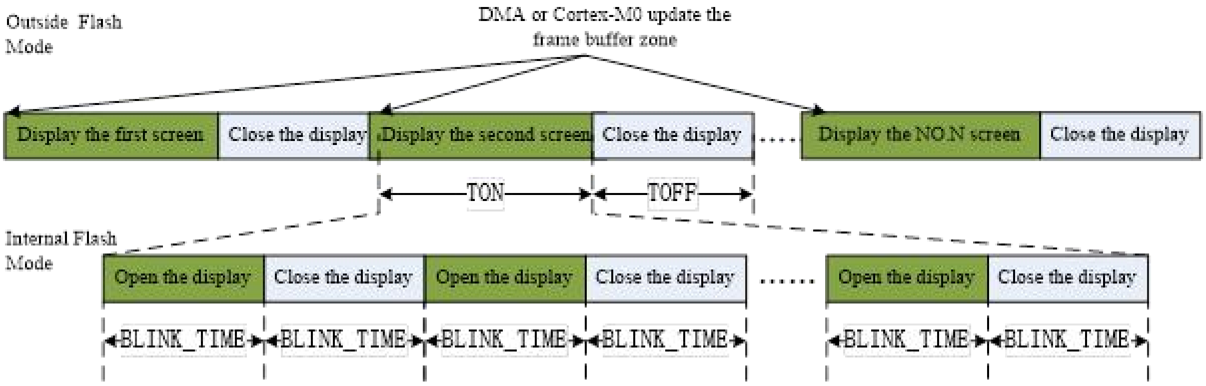


Figure 7-1 LCD Blink Mode

As shown in Figure 7-1, When LCD is enabled, it turns on the display for the length of time defined in the TON bit field of the LCD_BLINK register and then turns off the display for the length of time defined in the TON bit field of the LCD_BLINK register.

LCD can issue interrupt requests which can be used to update the frame uffer when the display is on and off.

7.1.2.1 Internal Blink Mode

Flashing mode can be inserted during display of length specified by TON bit field of LCD_BLINK register. The blinking interval is determined by the BLINK_TIME bit field of the LCD_BLINK register. When BLINK_TIME is 0, the internal flashing mode is disabled. TON must be an even multiple of BLINK_TIME when BLINK_TIME is not 0.

7.1.2.2 External Blink Mode

When TOFF in LCD_BLINK register is not 0, the blinking function is enabled. When Blink Mode is enabled, the blinking frequency is determined by TON and TOFF values in LCD_BLINK register.

7.1.3 LCD Driving Waveform

LCD drive waveforms depend on LCD Type, Duty number and Bias number set by software (LCD_CTL register).

MCU supports two kinds of driving waveforms, Type A and Type B.Type A driving is line inversion mode, that is, alternating positive and negative driving is completed once in 1 frame time. Type B driving is frame inversion, that is, it takes 2 frames to complete an alternating positive and negative driving. For details, please refer to the LCD drive waveform described in this chapter. When the number of drive Duty is higher, the display result of Type B drive mode will be better.

Select the duty cycle of the LCD output waveform according to the number of COM required for the application:

- 1 COMs: static duty cycle, COM0;
- 2 COMs: 1/2 duty cycle, COM0, COM1;
- 3 COMs: 1/3 duty cycle, COM0 ～COM2;
- 4 COMs: 1/4 duty cycle, COM0 ～COM3;
- 6 COMs: 1/6 duty cycle, COM0 ～COM5;
- 8 COMs: 1/8 duty cycle, COM0 ～COM7;

7.1.3.1 LCD Type A Driving Waveform

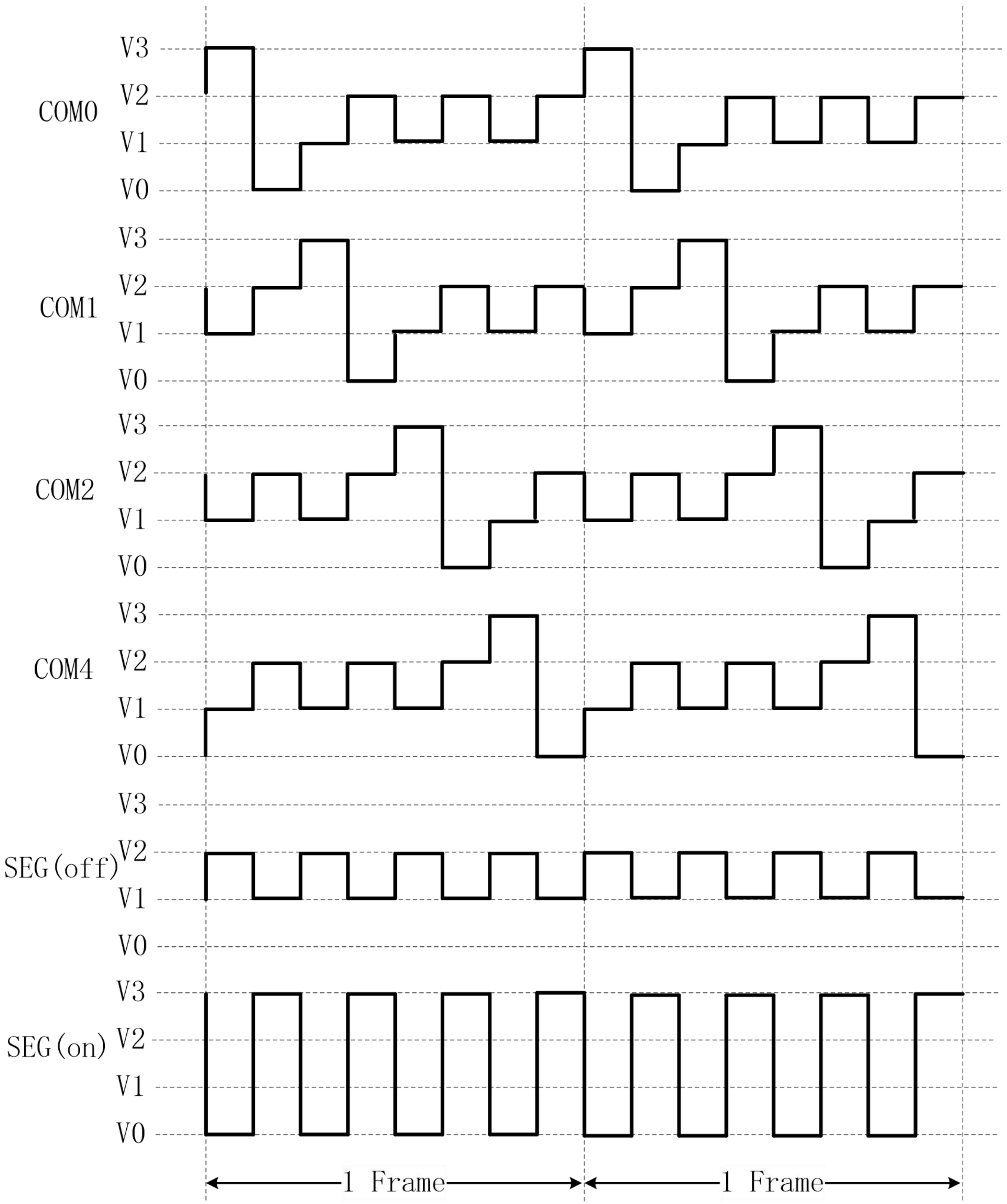


Figure 7-2 LCD driving waveform (1/4 Duty, 1/3 Bias, Type A)

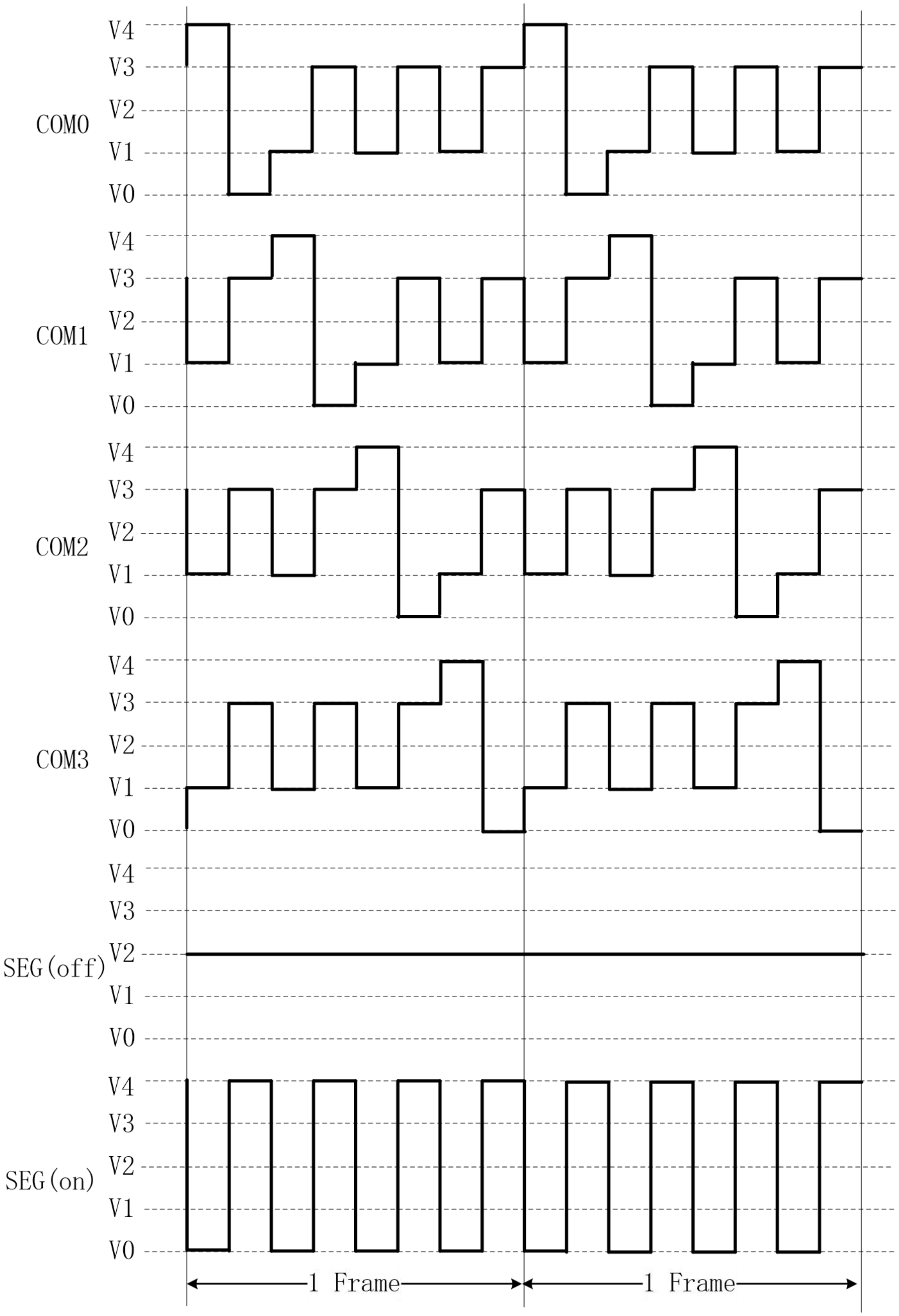


Figure 7-3 LCD driving waveform (1/4 Duty, 1/4 Bias, Type A)

7.1.3.2 Type B Driving Waveform

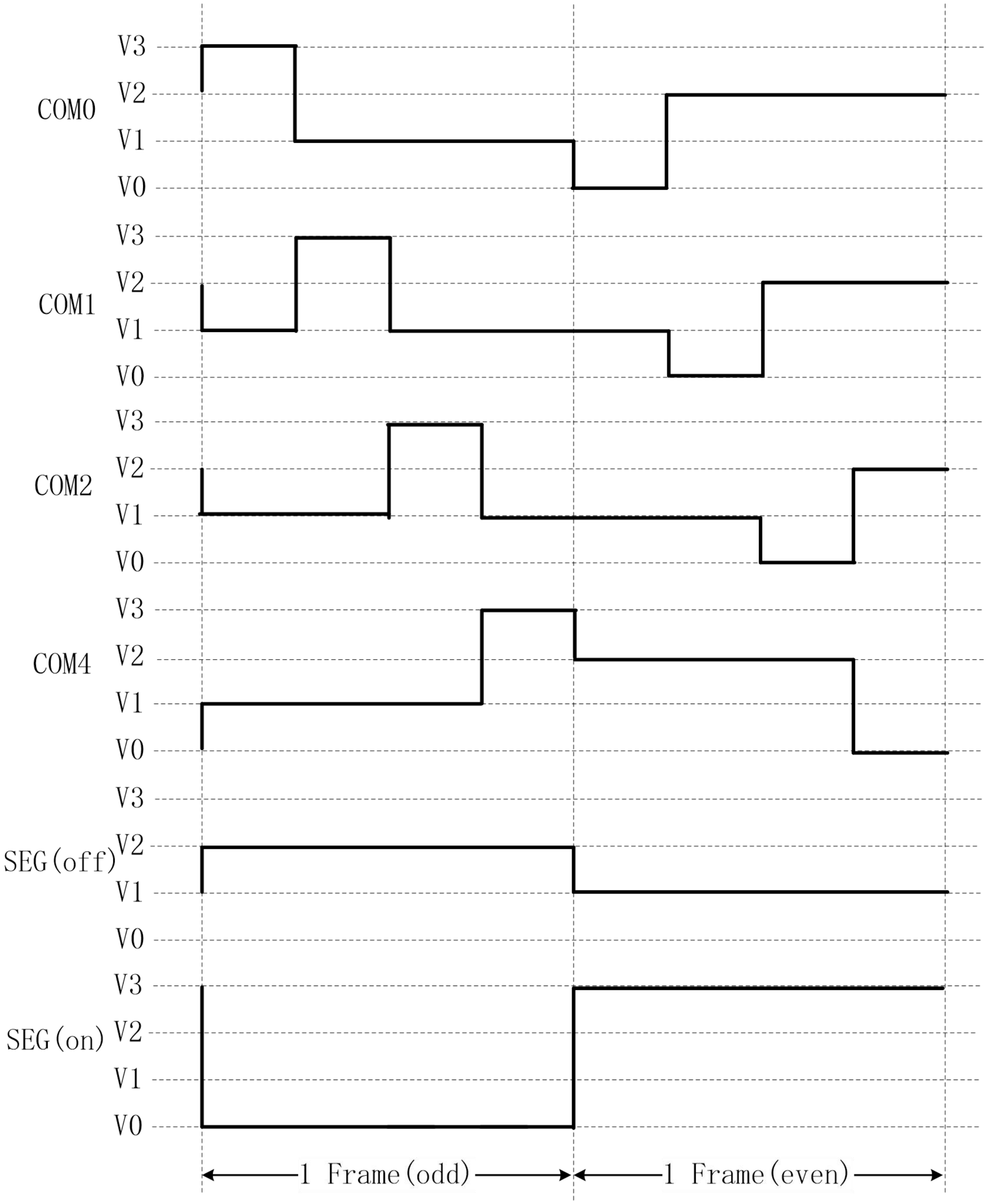


Figure 7-4 LCD driving waveform (1/4 Duty, 1/3 Bias, Type B)

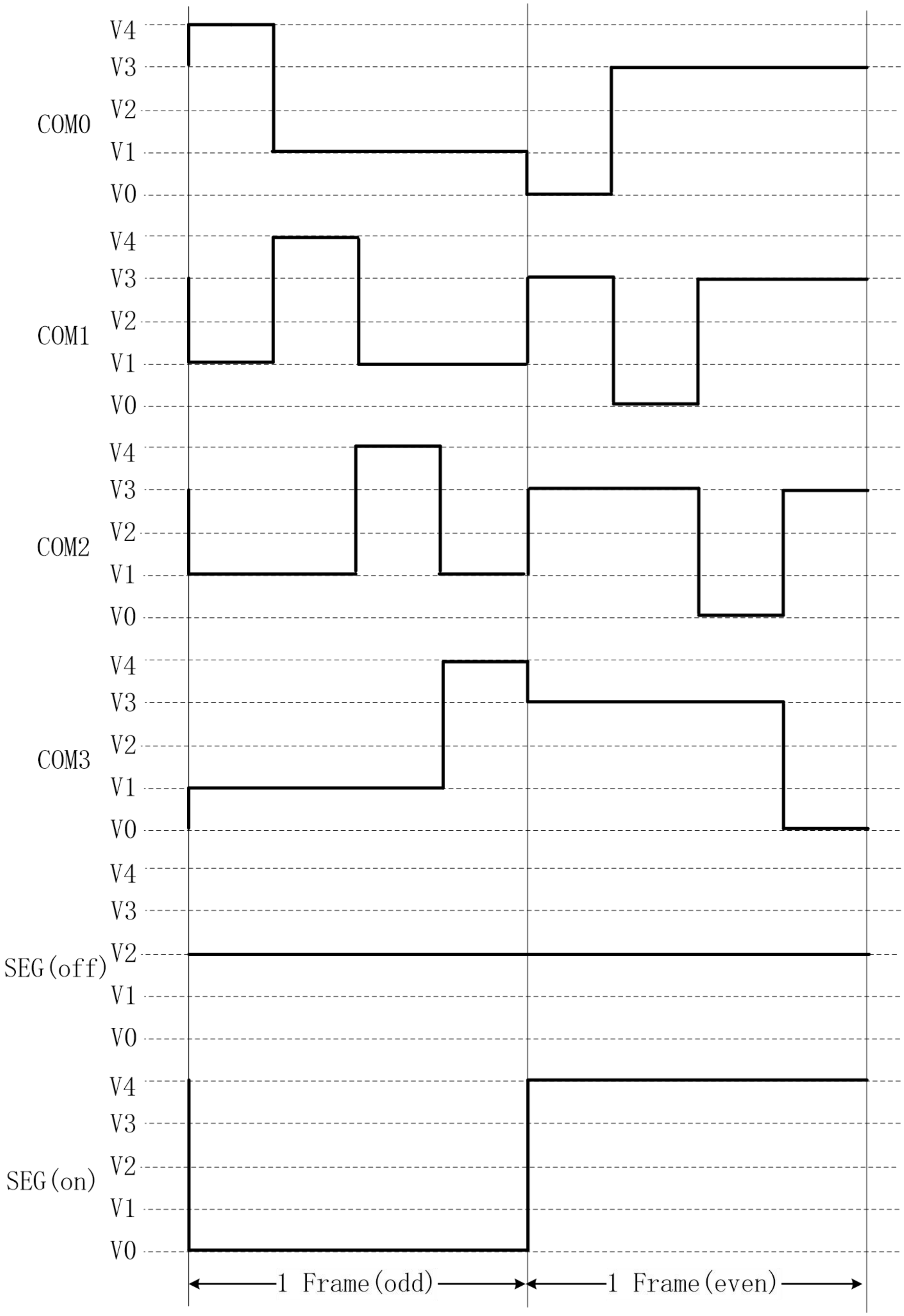


Figure 7-5 LCD driving waveform (1/4 Duty, 1/4 Bias, Type B)

7.1.4 Bias Voltage

7.1.4.1 Charge Pmup for LCD Bias Voltage

LCD bias voltage can be provided by Charge Pump. Charge Pump needs to generate 4 voltages (Va, Vb, Vc, Vd) in order to satisfy the application of 1/4 bias ratio. The voltage mode of the Charge Pump output varies for different bias ratio settings, as shown in Table 7-2.

Table 7-2 LCD Relationship between driving voltage and bias voltage ratio

Bias Voltage	Grayscale	Va	Vb	Vc	Vd	Vd(MAX)
1/3 bias	BIASLVL[5] = 0	Va = Vref*(32+BIASLVL[4:0])/63	Vb = Va	Vc = 2*Va	Vd = 3*Va	3.75 V
	BIASLVL[5] = 1	Va = Vref*(1 + BIASLVL[4:0]/63)	Vb = Va	Vc = 2*Va	Vd = 3*Va	5.59 V
1/4 bias	BIASLVL[5] = 0	Va = Vref*(32+BIASLVL[4:0])/63	Vb = 2*Va	Vc = 3*Va	Vd = 4*Va	5.0 V
	BIASLVL[5] = 1	Va = Vref*(1 + BIASLVL[4:0]/63)	Vb = 2*Va	Vc = 3*Va	Vd = 4*Va	6.032V

For example: Maximum Vd required for LCD is 5.2V. When a 1/4 bias ratio is selected, the LCD controller automatically clamps BIASLVL [5:0] to 6'h2d if BIASLVL [5:0] is set greater than 6'h2d.

LCD uses internal Vref as reference, Vref output typical value is 1.25V.

1/3 bias and 1/4 bias voltage sselection are shown in Figure 7-6.

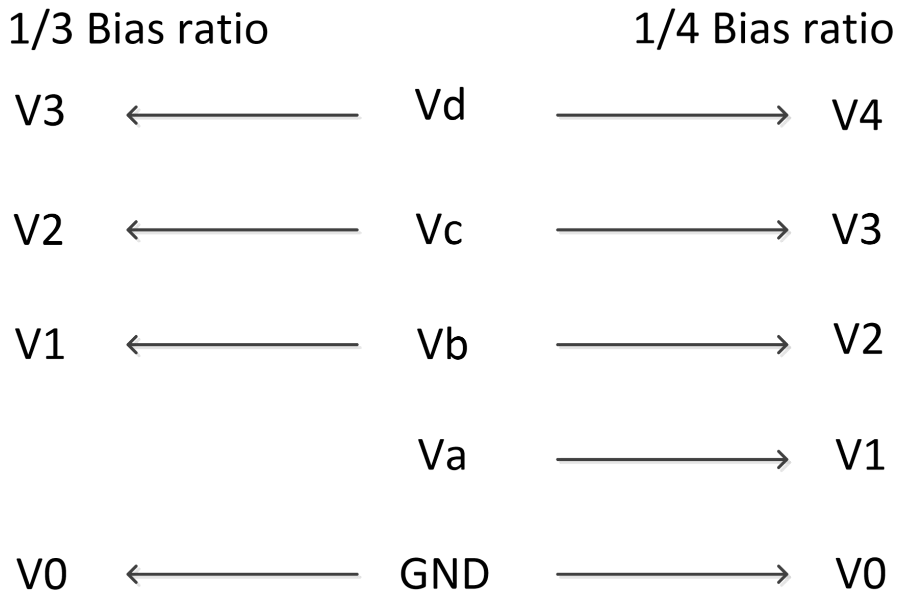


Figure 7-6 Bias voltage selection

7.1.4.2 Internal resistor string voltage division to provide LCD bias voltage

There is a built-in LDO, which has adjustable output 2.7~3.6V and adjustable step 60mV. It also supports 3.0V and 3.3V LCD. See LCD internal resistor string control register (LCD_RESCTL) for specific configuration.

LCD internal resistor string mode supports two modes, large/small resistor string time sharing on mode and small resistor/open circuit time sharing on mode. It is recommended to used low resistance/open mode, external 470nf capacitive filter is required.

Fast charge only works when the first start of the brush screen and has no effect on the normal display. The purpose of fast charge is to make the off-chip capacitor fully charged as soon as possible. It is recommended to enable this bit.

7.1.5 LCD BUFFER Mapping

The mapping relationship between register LCD_BUFx and LCD screens with different segment code specifications is as follows:

i. For 8 COMs, 36 LCD_BUF required, maximum 8*36 LCD screen

LCD_BUF[i] i=0~35 SEG MAX 36	SEG[i+4] COM7	SEG[i+4] COM6	SEG[i+4] COM5	SEG[i+4] COM4	SEG[i+4] COM3	SEG[i+4] COM2	SEG[i+4] COM1	SEG[i+4] COM0

ii. For 6 COMs, 38 LCD_BUF required, maximum 6*38 LCD screen

LCD_BUF[i] i=0~37 SEG MAX 38	-	-	SEG[i+2] COM5	SEG[i+2] COM4	SEG[i+2] COM3	SEG[i+2] COM2	SEG[i+2] COM1	SEG[i+2] COM0

iii. For 4 COMs, 3 COMs, 2 COMs and 1 COM, 20 LCD_BUF required, maximum 4*40 LCD screen

LCD_BUF[i] i=0~19 SEG MAX 40	SEG[2*i+1] COM3	SEG[2*i+1] COM2	SEG[2*i+1] COM1	SEG[2*i+1] COM0	SEG[2*i] COM3	SEG[2*i] COM2	SEG[2*i] COM1	SEG[2*i] COM0

7.2 Register Description

Base Address of LCD Module

Name	Physical Address	Mapping Address
LCD	0x40048000	0x40048000

Register Offset Address of LCD Module

Name	Offset Address	Description
LCD_CTL	Offset+0x0	Register of LCD control
LCD_STATUS	Offset+0x4	Register of LCD status
LCD_CLKDIV	Offset+0x8	Register of LCD clock control
LCD_BLINK	Offset+0xc	Register of LCD blink control
LCD_PS	Offset+0x10	Register of LCD fast charge time
LCD_RESCTL	Offset+0x14	Register of LCD internal resistor string
LCD_BUF[i]	Offset+0x20+i*1(i=0-37)	Register of LCD buffer (38 8-bit register in total)

7.2.1 Register of LCD Control LCD_CTL (0x00)

Offset Address 0x00

Bit	Name	Description	Read/Write Flag	Reset Value
31:13	Reserved	Reserved	R	0
12	PWD_PUMP	LCD PUMP switch: 0: turn on PUMP, LCD voltage from internal PUMP; 1: turn off PUMP, LCD voltage from resistor string; Note: For RN8612, this bit is Reserved.	R/W	0
11	TYPE	LCD drive type selection 0:Type A 1:Type B	R/W	0
10:5	BIASLVL	LCD bias voltage adjustment Control the Charge Pump to output voltages of different amplitudes to adjust the contrast of the LCD Note: For RN8612, this bit is Reserved.	R/W	0
4	BIAS	LCD Bias control 0:1/3Bias 1:1/4Bias	R/W	0
3:1	DUTY	LCD duty cycle control 000: static (COM0) 001: 1/2duty cycle (COM0~1) 010: 1/3duty cycle (COM0~2) 011: 1/4duty cycle (COM0~3) 100: 1/6duty cycle (COM0~5) 101: 1/8duty cycle (COM0~7) Other:Reserved	R/W	0
0	EN	LCD module enable 0:LCD module disabled 1:LCD module enabled Note: When the LCD module is turned off, the LCD enters standby mode and the associated analog circuits are also turned off to ensure low power consumption requirements..	R/W	0

7.2.2 Register of LCD Status LCD_STATUS (0x04)

Offset Address 0x04

Bit	Name	Description	Read/Write Flag	Reset Value
31:7	Reserved	Reserved	R	0
6	LCD_BUSY	LCD Busy Bit 0: Not busy 1: Busy Note: When LCD_BUSY is 1, LCD_CTRL (except EN Bit), LCD_CLKDIV, LCD_BLINK, LCD_PS registers cannot be modified.	R	0

5	Reserved	Reserved		
4	Reserved	Reserved		
3	IRQOFFEN	Display Off IRQ Enble Bit 0: disabled 1: enabled	R/W	0
2	IRQONEN	Display On IRQ Enble Bit 0: disabled 1: enabled	R/W	0
1	DOFF	Display Off Pending Bit 0: No interrupt event; 1: Set 1 when the display changes from bright to dark. Note: Write 1 to clear	R/W	0
0	DON	Display On Pending Bit 0: No interrupt event; 1: Set 1 when the display changes from dark to bright. Note: Write 1 to clear	R/W	0

7.2.3 Register of LCD Clock Control LCD_CLKDIV (0x08)

Offset Address 0x08

Bit	Name	Description	Read/Write Flag	Reset Value
31:8	Reserved	Reserved	R	0
7:0	CLKDIV	LCD Clock division factor LCD_CLK=fosc/(2*(CLKDIV+1)) (fosc is 32768Hz)	R/W	0

7.2.4 Register of LCD Blink Control LCD_BLINK (0x0c)

Offset Address 0x0c

Bit	Name	Description	Read/Write Flag	Reset Value
31:24	Reserved	Reserved	R	0
25:18	BLINK_TIME	Step is 0.25s, range 0~63.75s During TON display cycle, time of dark and display = 0.25*BLINK_TIME. Note: when it is 0, LCD always on without blink. When it is greater than 0, TON must be 2n times of BLINK_TIME (n is a integer greater than 0).	R/W	0
17:9	TOFF	Step is 0.25s, range 0~127.5s, set it >3s 0.25*TOFF	R/W	0
8:0	TON	Step is 0.25s, range 0~127.5s, set it >3s 0.25*TON	R/W	0

7.2.5 Register of LCD Fast Charge Time LCD_PS (0x10)

Register of LCD PUMP setup time

Offset=0x10

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:16	---	Reserved	R	0
15:0	PS	LCD PUMP Setup time Time = Tosc * (PS+4) (Tosc = 30.5uS)	R/W	0xccc

Note: This register is not available for RN8612.

7.2.6 Register of LCD Internal Resistor String LCD_RESCTL (0x14)

Offset Address 0x14

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:13	Reserved	Reserved	R	0
12:9	LDOS	LDO Output Level Selection Signal The output of generates LCDVD, and LCDVC、LCDVB、LCDVA by resistive voltage division; LDO output voltage level ranges from 2.7V to 3.6V, 16 steps in total. The step is 0.06V; LDOS Configuration / LDO Output Voltage (V): 0000: 2.7 0001: 2.76 0010: 2.82 0011: 2.88 0100: 2.94 0101: 3 0110: 3.06 0111: 3.12 1000: 3.18 1001: 3.24 1010: 3.3 1011: 3.36 1100: 3.42 1101: 3.48 1110: 3.54 1111: 3.6 The output of LDO is 3.0V by default.	R/W	0101
8	FCC	Fast charge control: 0: off 1: on; Internal resistor voltage divider is selected, an external 470nf capacitor is required, LCD module on, internal resistance adjusted to 5k, and kept for 100ms to complete the fast charging of capacitors.	R/W	1

7	RES_AO	Resistor string driving switch 0: resistor string part-time driving, driving signal generated by RES_DT and RES _ 1: resistor string full-time driving, RES_DT and RES_FT configuration disabled When TYPE A is selected, RES _AO is always 1.	R/W	1
6:4	RES_DT	Resistor string voltage division mode, 20k resistor drive time configuration within the cycle Driving time Td=(BIAS_DT[2:0]+1)* Tosc (Tosc 为 30.5uS) .	R/W	000
3:2	RES_FT	Resistor string voltage division mode, within the cycle, the 20k resistor. 00: drive once; 01: drive twice; 10: drive 3 times; 11: drove 4 times.	R/W	00
1	RSM	Selection signal of internal resistor voltage divider 0 = 20k resistor part-timely on, the 200k resistor is shorted; 1 = resistor switch mode, switch between 20k and 200k resistor.	R/W	0
0	IRSN	0 = Internal resistor voltage division mode 1 = Reserved Write 0 to enable internal resistor voltage division mode	R/W	1

Recommendation for LCD_RESCTL configuration:

Typical 3V screen, the value of register LCD->RES_CTRL should be 0xb14;

Typical 3.3V screen, the value of register LCD->RES_CTRL should be 0x1514.

7.2.7 Register of LCD Buffer LCD_BUF[i] (8 bits)

Offset Address0x20– 0x45

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:8	Reserved	Reserved	R	0
7:0	LCD_BUFx	Physical definition for the bits of SEG disaplay data 0: display unit off 1: display unit on	R/W	0

7.3 LCD Configuration Steps

1. Set Bit 6 of Register MOD1_EN 1 to 1 in order to turn on the clock of LCD control;
2. Set Register LCD_CTL to 0 in order to trun off LCD module;
3. Wait for 0 in Bit 6 of Register LCD_STATUS. If so, turn to the next step. Otherwise, wait for it. Usually it takes 16 ms.
4. Configure the Register LCD_CTL and LCD_RESCTL based on pratical application;
5. Copnfigure the frequency division factor in Register LCD_CLKDIV;

6. Configure the Register of LCD Blink Control, set it to **0x01** to disable blink;
7. Initializing configuration completed, send the data to Register LCD_BUF[i] directly.

8 Timer

There are two 32-bit built-in timers. Each timer runs independently. There is no resource shared between two timers. Synchronous operation is available.

The timers has the following functions:

- Interval timing
- Square wave output
- External/Internal event count
- Single pulse output
- PWM output
- Pulse width measurement

8.1 Overview

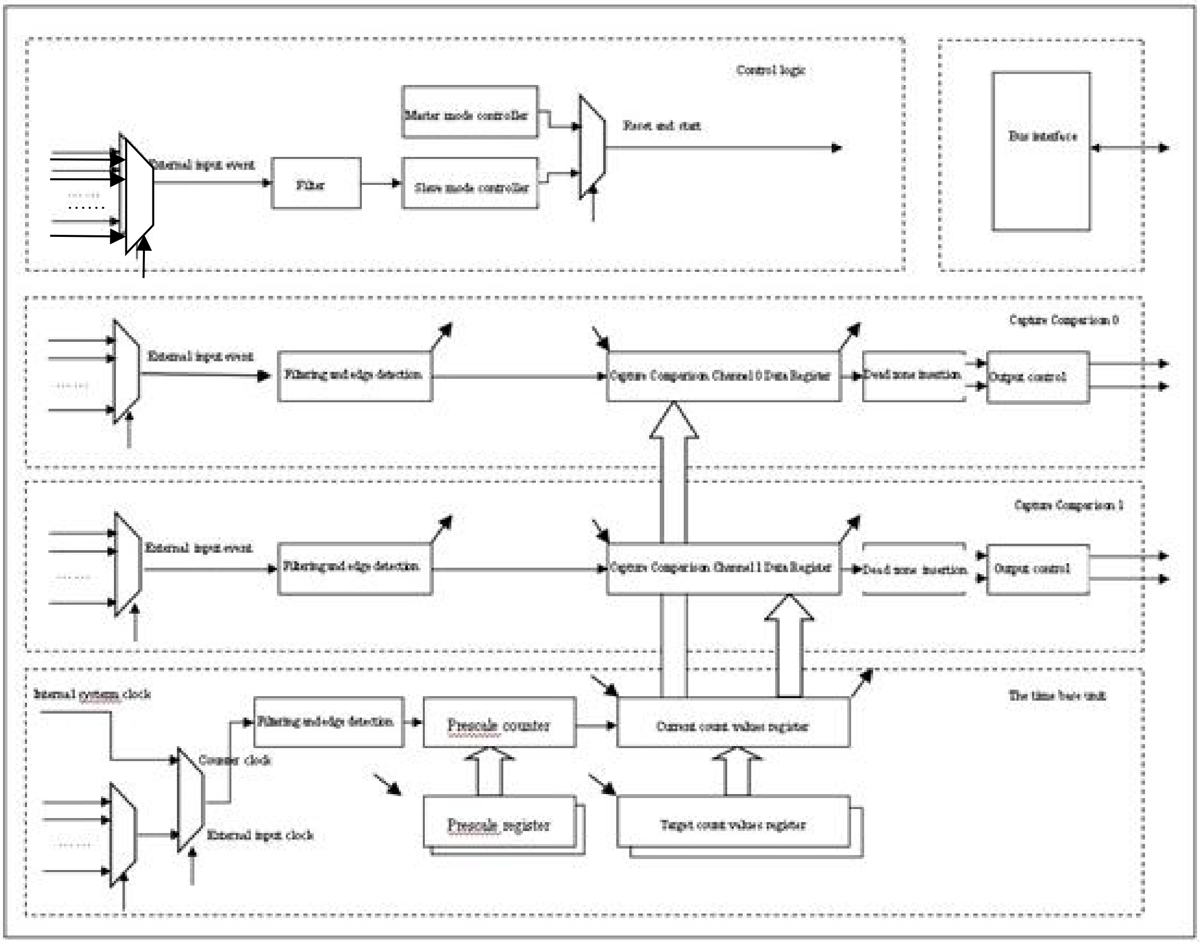
The timers has the following characteristics:

- Two 32-bit timers, each:
    - Has a 32-bit increment autoreload counter;
    - Has a 16-bit programmable frequency divider, frequency dividing factor ranges from 1 to 65535;
    - Supports dynamic access to count value;
    - Supports free running mode;
    - Supports single running;
- Each timer has 2 Capture/Compare channels, each channel can be configured as :
    - input capture;
    - output compare;
    - single pulse output;
    - complementary PWM:
        - Programmable dead zone length:
            - Dead zone lengths for both edges can be set independently;
            - Output polarity configurable;
        - Configurable failure handling:
            - output failure;
            - output clear;
            - output three-state;
- Supports slave mode:
    - External reset and restart;
    - External switch;
- Supports input capture:
    - Rising edge capture;
    - Falling edge capture;
    - Double edges capture;
    - Period measurement;
    - Pulse width measurement;
    - Optional filtering;
- Supports output compare:
    - Tri-State output;

  - Inverted output;
  - Fixed level output;
  - PWM output;
  - Compare register updated at any time;
- Supports interrupt:
  - Count overflow;
  - Input capture;
  - Output compare;

8.2 Function Block Diagram

The function block diagram of count timer is shown in the following figure. Each count timer has a 32-bit counter and four 32-bit capture/compare channel.



8.3 Register Description

Base Address

Name	Physical Address	Mapping Address
TC0	0x40010000	0x40010000
TC1	0x40014000	0x40014000

Offset Address of TC module

Name	Offset Address	Description
TC_CNT	0x0	Instructions of the current count values
TC_PS	0x4	Prescale Register
TC_DN	0xC	Target Count Value Register
TC_CCD0	0x14	Capture Comparison Channel 0 Data Register
TC_CCD1	0x18	Capture Comparison Channel 1 Data Register
TC_CCFG	0x1C	Clock Configuration Register
TC_CTRL	0x20	Control Register
TC_CM0	0x24	Capture Comparison Channel 0 mode Register
TC_CM1	0x28	Capture Comparison Channel 1 mode Register
TC_IE	0x2C	IER
TC_STA	0x30	Status Register

8.3.1.1.1 Register of current count value TC_CNT (0x00)

Bit	Name	Description	Read/Write Flag	Reset Value
31:0	CNT	Current count value	R	0

8.3.1.1.2 Register of Prescale TC_PS (0x04)

Bit	Name	Description	Read/Write Flag	Reset Value
31:16	-	Reserved	R	0
15:0	PS	Frequency division factor, value = (PS+1), 0 = no frequency division	R/W	0

8.3.1.1.3 Register of target count value TC_DN (0x0C)

Bit	Name	Description	Read/Write Flag	Reset Value
31:0	DN	Target count value, actual count period = DN+1	R/W	0

8.3.1.1.4 Data register of Channel 0 TC_CCD0 (0x014)

Bit	Name	Description	Read/Write Flag	Reset Value
31:0	CCD	Data	R/W	0

Note: The TC_CCD0 register is not writable when Channel 0 is configured for capture (i.e., CCM bit field of the

TC_CM0 register is 0)

8.3.1.1.5 Data register of Channel 1 TC_CCD1 (0x018)

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:0	CCD	Data	R/W	0

Note: The TC_CCD1 register is not writable when Channel 1 is configured for capture (i.e., CCM bit field of the TC_CM1 register is 0)

8.3.1.1.6 Register of clock configuration TC_CCFG (0x01C)

Bit	**Name**	**Description**	**Read/Write Flag**	**Reset Value**
31:24	-	Reserved	R	0
23:16	FLTOPT	External input clock filter parameter setting, set the number of clock cycles filtered	R/W	0
15	-	Reserved	R	0
14:13	ECLKMODE	External input clock mode: 00: rising edge 01: falling edge 10: dual edge 11: Reserved(Equivalent to dual edge)	R/W	0
12:8	CS	Extern input clock selection: 0: UART0 RXD 1: UART1 RXD 2: UART2 RXD 3: UART3 RXD 4: output outn[0] of another timer 5: output outp[0] of another timer 6: output outn[1] of another timer 7: output outp[1] of another timer 8: UART4 RXD 9: UART5 RXD 10:7816_0 input P41 11:7816_1 input P42 12:7816_1 input P43 13~15:Reserved 16:sf_out 17: qf_out 18: pf_out 19: rtc_out 20: p1[0]external IO pin 21: p1[1]external IO pin	R/W	0

		22: p1[2]、P5[2]external IO pin 23: p1[3]、P5[3]external IO pin 24: p1[4]、P5[4]external IO pin 25: p1[5]、P5[5]external IO pin 26: p1[6]、P5[6]external IO pin 27: p1[7]、P5[7]external IO pin 28: p3[0]external IO pin 29: p3[1]external IO pin 30: p3[3]external IO pin 31: p3[5]external IO pin		
7:2	-	Reserved	R	0
1	FLTEN	External clock filtering 0: disabled 1: enabled	R/W	0
0	CM	Clock source selection 0: internal system clock 1: external input clock (selected by CS)	R/W	0

8.3.1.1.7 Control Register TC_CR (0x020)

Bit	Name	Description	Read/Write Flag	Reset Value
31:29	-	Reserved	R	0
28	DBGSTBDIS	Counter debug: 0: disabled (Counters stop counting when CPU is in debug state) 1: enabled (Counters continue to count when CPU is in debug state) Note: CPU in debug state means that the user stops Cortex M0 (PC pointer stops counting) through debug interface.	R/W	0
27	--	Reserved	R/W	0
26	--	Reserved	R/W	0
25	--	Reserved	R/W	0
24	--	Reserved	R/W	0
23:21	--	Reserved	R	0
20	SLVGATELVL	Active level of gate in slave mode: 0: Active level is low 1: Active level is high	R/W	0
19:12	SLVFLTOPT	Input filter parameters of slave mode	R/W	0
11:10	SLVTRGMODE	Selction of slave mode: 00: Rising edge clears internal counter 01: Falling edge clears internal counter 10: Dual edge clears internal counter 11: Internal counter counts when external input signal	R/W	0

		is active		
9:5	SLVCHANSEL	Slection of external input event in slave mode: As same as the external input clock definition in CS bit field of clock configuration register (0x01 C).	R/W	0
4	OPS	Single-count mode: 0: disabled (counting does not stop after overflow, cycle counting); 1: enabled (counting stops after overflow)	R/W	0
3	SLVFLTEN	Filtering of external input events from mode: 0: disabled 1: enabled	R/W	0
2	SLVEN	Slave mode: 0: disabled 1: enabled	R/W	0
1	-	Reserved	R	0
0	START	Timer: 0: off 1: on	R/W	0

8.3.1.1.8 Register of capture/compare channel 0/1 TC_CM0/1 (0x024 and 0x028)

Bit	Name	Description	Read/Write Flag	Reset Value
31:30	-	Reserved	R	0
29	DFTLVL	Compare output default level: 0: low 1: high	R/W	0
28	EFELVL	Compare output effective level: 0: low 1: high	R/W	0
27:25	OM	Compare output mode: 000: No output (tri-state) 001: Set to effective level 010: Set to inactive level 011: Reversed 100: Force to effective level 101: Force to inactive level 110: PWM mode 1 111: PWM mode 2	R/W	0
24:20	CS	Capture external input event selection: As same as input clock definition defined in CS bit field of clock configuration register (0x01 C).	R/W	0
19	FLTEN	Capture external input event filter: 0: disabled	R/W	0

		1: enabled		
18:11	FLTOPT	Capture External Input Event Filter Parameters	R/W	0
10:9	CPOL	Capture external input event polarity selection: 00: rising edge 01: falling edge 10: dual edge 11:Reserved	R/W	0
8:3	DL	Compare output dead band length (PWM mode 1 and PWM mode 2 only, this bit is invalid in other modes)	R/W	0
2	DIEN	Compare Output Dead Zone Insertion:(PWM mode 1 and PWM mode 2 only, this bit is invalid in other modes) 0:disabled 1:enabled	R/W	0
1	CCM	Capture/Compare Mode Selection: 0: capture 1: compare	R/W	0
0	ENABLE	Channel: 0:disabled 1:enabled	R/W	0

8.3.1.1.9 Register of Interrupt TC_IE(0x2C)

Bit	Name	Description	Read/Write Flag	Reset Value
31:4	-	Reserved	R	0
3	SLVIE	Interrupt of slave mode: 0:disabled 1:enabled	R/W	0
2	CC1IE	Interrupt of Chanel 1: 0:disabled 1:enabled	R/W	0
1	CC0IE	Interrupt of Chanel 1: 0:disabled 1:enabled	R/W	0
0	OVIE	Overflow interrupt: 0:disabled 1:enabled	R/W	0

8.3.1.1.10 Register of Status TC_STA(0x30)

Bit	Name	Description	Read/Write Flag	Reset Value
31:4	-	Reserved	R	0
3	SLVF	Event flag of slave mode:(write 1, clear 0)	R/W	0

		0: No enents 1: Events		
2	CC1F	Event flag of Channel 1: (write 1, clear 0) 0: No enents 1: Events	R/W	0
1	CC0F	Event flag of Channel 1: (write 1, clear 0) 0: No enents 1: Events	R/W	0
0	OVF	Overflow event flag: (write 1, clear 0) 0: No enents 1: Events	R/W	0

8.4 Typical Application

8.4.1 Automative Operation Mode

Automative operation mode refers to the interval timing function.

Configure the registers in following steps:

1. Target count value register, i.e. timing duration, counted by count clock;
2. In the interrupt register, enable the overflow interrupt;
3. In the control register, enable the timer.
   The timer generates interrupts periodically with the target count value.

Common optional function configuration description:

1. The prescaler register value can be modified to change the frequency of the timer count clock.
2. Configurable clock configuration register, CM modifies configuration to external input clock, while modifying external input clock selection of CS bit configuration. External input clock frequency cannot be higher than internal system clock frequency divided by two.
3. Single count mode, timer stops after overflow; configuration control register OPS bit is 1 for single count mode.
4. External input clock source is the output of another timer, it can be connected to two timing cascade mode, can increase the bit width of the timer register.

8.4.2 Input Capture Mode, Pulse Width Measurement Function

The input capture mode is used to measure the pulse width.

Configure the registers in following steps:

1. Target count value register, counted by count clock, can be set to maximum value.
2. Capture compare channel 0/1 mode register setting, ENABLE channel 0/1enabled, CCM configured for capture mode, CPOL selects capture polarity, CS selects external input event；
3. In the interrupt register, enable the interrupt of Channel 0/1;
4. In the control register, enable the timer.

When the timer captures the capture polarity of an external input event, an interrupt is generated, and the current count value is stored in the capture compare channel data register. If two channels are used, one capturing the rising edge and one capturing the falling edge, the pulse width can be calculated from the data registers of both channels.

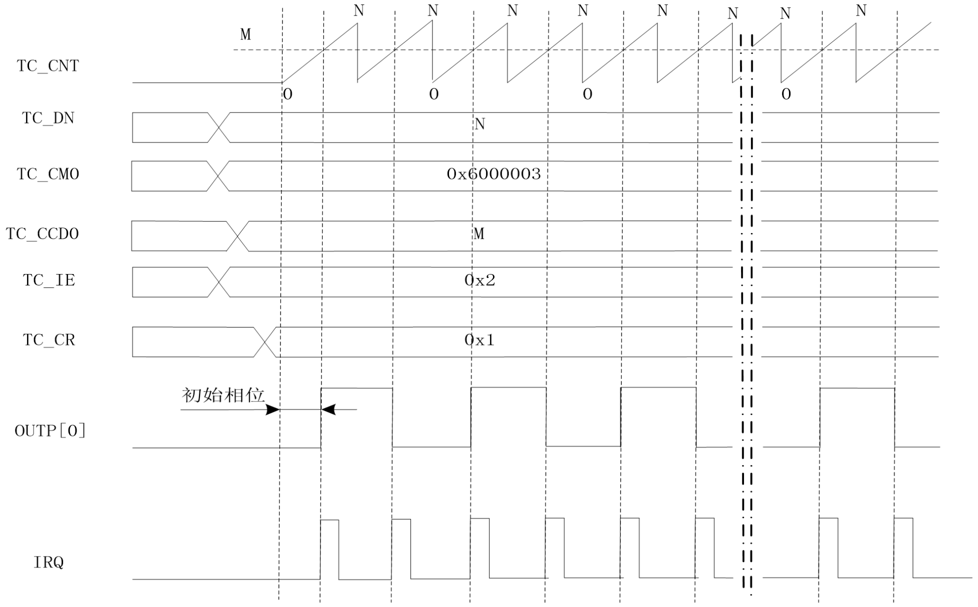
Common optional function configuration description:

1. The prescaler register value can be modified to change the timer count clock frequency.

2. Configurable clock configuration register, CM modifies the configuration to external input clock, while modifying the external input clock source configured by CS bit. The external input clock source frequency cannot be higher than the divide-by-two of the internal system clock frequency.
3. Filter function for external input events, enabled FLTEN filter function in capture compare channel mode register, set filter cycle number by configuring FLTOPT.
4. If the external input event is the output of another timer, it can be connected to cascade mode of two timers.

8.4.3 Comparision Output Moce, Square Wave Output Function

Square wave output function is to TC count clock frequency output function. Each timer has two output channels, each channel has P and N two output ports, P is the positive output port, N is the negative output port of P.



Configure the registers in following steps:

1. Target count value register, counted by count clock, can be set to maximum value.
2. Capture compare channel 0/1 mode register setting, ENABLE channel 0/1enabled, CCM configured for compare mode, DFTLVL configured for default level, EFELVL configured for active level, OM output configured for toggle function.
3. Sets the Capture Compare Channel 0/1 data register. (Set value not greater than target count value register)
4. In the interrupt register, enable the interrupt of Channel 0/1;
5. In the control register, enable the timer.

The value in the Target Count Register determines the period of the output square wave, and the value in the Capture Compare Channel 0/1 Data Register is the output flip point.

Common optional function configuration description:

1. Modify prescaler register value to change timer count clock frequency.
2. Configurable clock configuration register, CM modifies the configuration to external input clock, while modifying the external input clock source configured by CS bit. The external input clock source frequency cannot be higher than the divide-by-two of the internal system clock frequency.
3. External input clock filter function, enabled FLTEN filter function in clock configuration register, set filter cycle number by configuring FLTOPT.

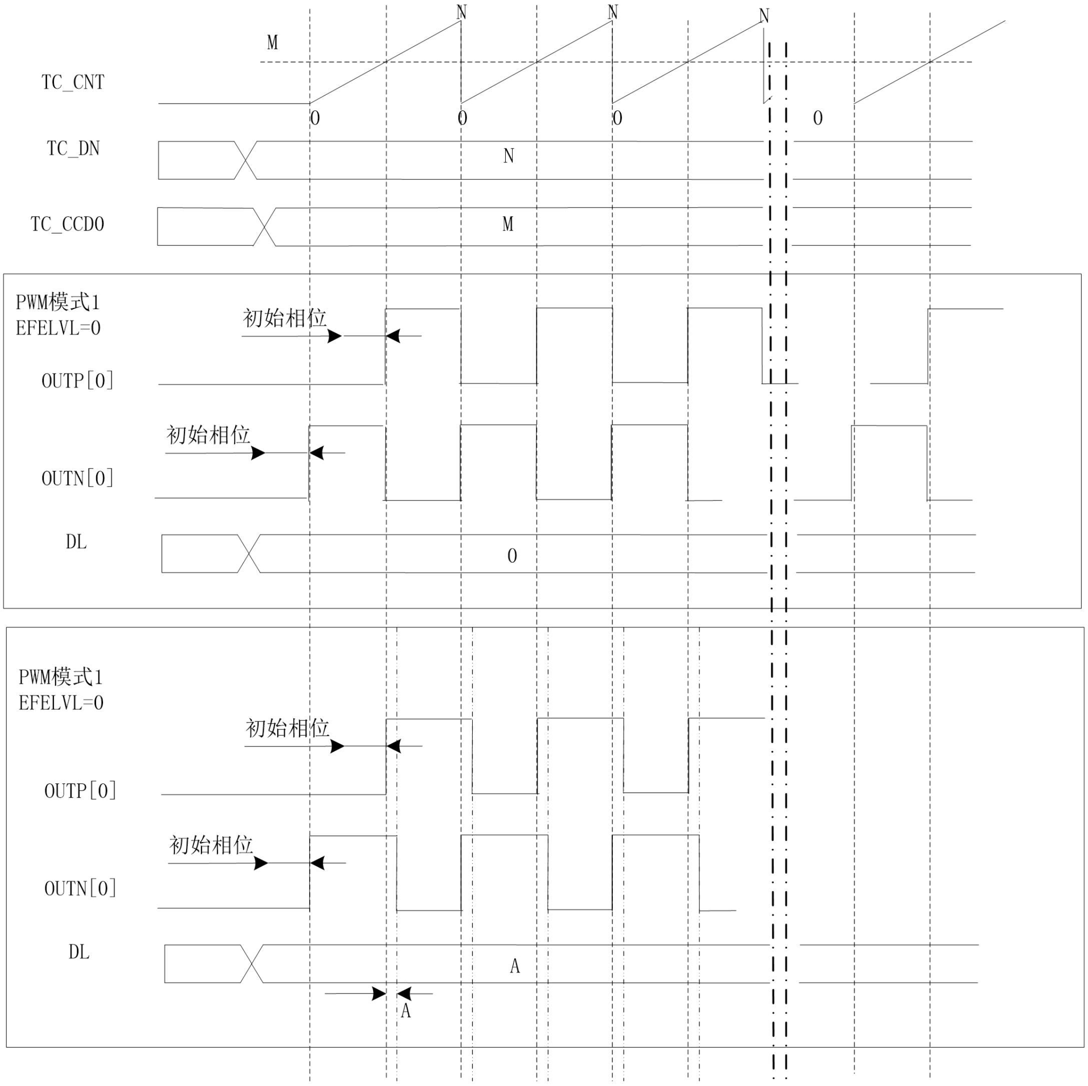
8.4.4 Comparision Output Mode, PWM Output Function

Pulse Width Modulation (PWM) mode generates a signal with a frequency determined by the TC_DN register and a duty cycle determined by the TC_CCDx register. There are two modes:

PWM Mode 1: If TC_CNT<TC_CCDx, the output is effective level, otherwise it is invalid level.

PWM Mode 2: If TC_CNT>=TC_CCDx, the output is effective level, otherwise it is invalid level.

Below is the typical application diagram of PWM mode 1.



Configure the registers in following steps:

1. Target count value register, counted by counting clock.
2. In the rregister of Channel 0/1 mode, enable Channel 0/1. CCM configured to be comparision mode. DFTLVL set to default voltage, EFELVL configured as effective, OM configured to PWM Mode 1 or PWM Mode 2.
3. In the Capture Compare Channel 0/1 Data Register, the data must be smaller than target counting value.
4. In the control register, enable the timer.

PWM mode 1/PWM mode 2 forward waveform is output at the P terminal of the channel, and the waveform opposite to the P terminal is output at the N terminal of the channel.

PWM Mode 1: The period is the target count register value plus 1, and the effective level period is the number of cycles of the channel data register value plus 1.

PWM Mode 2: The period is the target count register value plus 1, and the invalid level period is the number of cycles of the channel data register value plus 1.

Common optional function configuration description:

1. The prescaler register value can be modified to change the timer count clock frequency.
2. Configurable clock configuration register, CM modifies the configuration to external input clock, while modifying the external input clock source configured by CS bit. Configurable clock configuration register, CM modifies the configuration to external input clock, while modifying the external input clock source configured by CS bit.
3. Complementary outputs for dead band insertion, DIEN dead band insertion enabled, DL configuration dead band insertion length. Complementary outputs for dead band insertion, DIEN dead band insertion enabled, DL configuration dead band insertion length..

   When the effective level EFELVL is low: the falling edge of P and N output delay DL times period.

   When the effective level EFELVL is high: the rising edge of P and N output delay DL times period.

8.4.5 Slave Mode

From the original function of the mode, the external input event control internal counter reset and gating function is added.

Configure the registers in following steps:

1. Target count value register, counted by count clock.
2. In the interrupt register, enable the overflow interrupt.
3. In the control register, enable SLVEN, SLVTRGMODE, and SLVCHANSEL.
4. In the control register, enable timer.

By this to add an external reset function for internal CNT in free running mode.

Common optional function configuration description:

1. The prescaler register value can be modified to change the timer count clock frequency.
2. Configurable clock configuration register, CM modifies the configuration to external input clock, while modifying the external input clock source configured by CS bit. Configurable clock configuration register, CM modifies the configuration to external input clock, while modifying the external input clock source configured by CS bit.
3. Single count mode, timer stops after overflow; configuration control register OPS bit is 1 for single count mode.
4. External input clock source is the output of another timer, it can be connected to two timing cascade mode, which increase the bit width of the timer register.
5. In the control mode selection of slave mode, if the gate controlled mode is selected, SLVGATELVL configured as effective, the internal counter counts only when the slave mode input is gated active.
6. Slave mode configured as input capture mode, When the external input event and captured external input event of slave mode are configured to be the same, capture selects one edge and mode polarity selects the other edge, pulse width can be obtained directly from the capture compare channel data register.

8.5 Operation Steps

The steps to set an interval timer: Set TC0 to 1MS Interval Timed Interrupt

1. Configuration System Control Section Module enabled0 Register MOD0_EN Bit 4 enables TC0 clock. Configuration System Control Section Module enabled0 Register MOD0_EN Bit 4 enables TC0 clock.

2. Set TC0 module register:
   Clock configuration register TC_CCFG is configured to internal system clock, TC0->CCFG = 0;
   Configure Prescaler Register TC_PS;
   Configure Target Count Value Register TC_DN;

Configure the control register TC_CR, TC0->CTRL = 0x01; enable TC0 timer

Configure the interrupt register TC_IE, TC0->IE = 0x01; configured as overflow interrupt;

When the system clock is 3.6864MHz, TC0->PS = 255; TC0->DN = 13 after configuration, it can generate 1MS interrupt. (3.6864MHz/(255+1))/(13+1) = 1MS.

3. Enable TC0 interrupt, NVIC_EnableIRQ(TC0_IRQn);
4. Program interrupt service routines:

```
void TC0_HANDLER(void)
{
    /* Start adding user code. Do not edit comment generated here */
}
```

5. 1MS interrupt is generated when configuration is complete.

9 Analog Peripherals

9.1 Features

SAR ADC upgrade from 10bits to 12bits;

●When the voltage input to the VBAT pin and AIN6 pin is started, there are two 300K resistors inside to divide the voltage. The input voltage is reduced by half to the SAR ADC. The SAR ADC uses the 0.5x PGA to double the signal. Namely: 3.6V battery, the measured signal is about 0.9V. If the measurement is not started, the internal partial resistor will be turned off.

●9-channel analog multiplexeder, where AIN0 – AIN6 channels are used for external analog signal measurement, VBAT is used for battery voltage measurement, Temp Sensor is used to measure temperature sensor output, and temperature sensor has the highest priority.

●The input impedance of AIN0~AIN5 is about 3M ohms.

●In addition to AIN6, other AINs support up to 1.25V input. Gain support 0.5 times, 1 time, 1.5 times, 2 times, and add support for 0.25 times.

●When the ADC is not sampling, it automatically enters the power saving mode; each sampling is about 2ms from startup to completion.

The main features of one LVD circuit are as follows:

●The input of LVD can choose the power for the chip or an external input PIN;

●LVD threshold is adjustable, which is divided into 14 files from 2.3V to 4.9V;

●When selecting an external PIN as input, the threshold value is fixed at about 1.25V, and the internal resistance is about 1M ohm.

The main features of two comparator circuit CMP1 and CMP2 are as follows:

●External PIN input, the threshold is about 1.28V;

●The power consumption is better than 1uA. The comparator can be used to monitor the main power;

●Note that there are 600K pull-down resistors inside CMP1 and CMP2, and the internal resistor can be turned off by register (SYS_PD (0x08)).

9.2 Register

Base address of analog peripheral module

Module Name	**Physical Address**	**Address Mapping**
ANA	0x4002C000	0x4002C000

Register offset address of analog peripheral module

Register Name	Address Offset	Description
SAR_CTL	Offset+0x0	SAR-ADC Control Register
SAR_START	Offset+0x4	SAR-ADC Start Register
SAR_STAT	Offset+0x8	SAR-ADC Status Register
SAR_DAT	Offset+0xC	SAR-ADC Data Register
LVD_CTL	Offset+0x10	Comparator Control Register
LVD_STAT	Offset+0x14	Comparator Status Register
SAR_CTL1 (new)	Offset+0x18	SAR-ADC gain control register

SAR_DAT2 (new)	Offset+0x1C	SAR-ADC 12bit data register
ANA_PAD	Offset+0x3c	VBAT domain analog PAD configuration register
ANA_RST	Offset+0x40	VBAT domain software reset register

9.2.1 ADC Control Register SAR_CTL (0x00)

Offset Address 0x00

Bit	Name	Description	R/W Sign	Reset Value
31:17	Reserved	Reserved	R	0
16:12	REF_WAIT	The wait time from REF started to ADC started: The typical value of the minimum waiting time is 122us 5’d0: 122us 5’d31: (31+1)*122us Note: It is recommended that the client application configure this register to 0x10, which is 2074us.	R/W	0x5
11:7	SAR_WAIT	The wait time from SAR ADC started to sampling convert started: 5’d0: 30.5us 5’d31: (31+1)*30.5us Wait Time=(SAR_WAIT+1)*30.5us Note: Step of start ADC measurement: Start REF, wait time of REF_WAIT; Start ADC and Temperature Sensor, wait time of SAR_WAIT; Input clock and reset single, the sampling results obtained after 16 clock cycles. The above steps are achieved from hardware automatic control. Note 1: Due to the deviat The wait time from SAR ADC started to sampling convert started: 5’d0: 30.5us 5’d31: (31+1)*30.5us Wait Time=(SAR_WAIT+1)*30.5us Note: Step of start ADC measurement: Start REF, wait time of REF_WAIT; Start ADC and Temperature Sensor, wait time of SAR_WAIT; Input clock and reset single, the sampling results obtained after 16 clock cycles. The above steps are achieved from hardware automatic control. Note 1: Due to the deviation of the internal time base, if the software setting waits for the timeout and waits	R/W	0xE

		for the timeout, it is recommended to leave a margin of 3-5 times. Note 2: It is recommended that the client application configure this register to the default value of 0xE, which is 457.5us.ion of the internal time base, if the software setting waits for the timeout and waits for the timeout, it is recommended to leave a margin of 3-5 times. Note 2: It is recommended that the client application configure this register to the default value of 0xE, which is 457.5us.		
6	SAR_CH3	The SAR-ADC Channel is selected to form a 4-bit register with SAR_CH [2:0]. {SAR_CH3, SAR_CH[2:0]}=0xxx: See SAR_CH definition {SAR_CH3, SAR_CH[2:0]}=1000: Select AIN6 {SAR_CH3, SAR_CH[2:0]}=Other: Reserved	R/W	0
5	SAR_IE	SAR-ADC Interrupt Control: 1: Enable ADC interrupt output; 0: Disable ADC interrupt output.	R/W	0
4:3	SAR_PGA	SAR-ADC Gain Control: 00: 0.5 01: 1 10: 1.5 11: 2	R/W	0
2:0	SAR_CH	SAR-ADC Channel Choose 000: Temp Sensor 001:VBAT(For 3.6V battery, obtain 1.8V after 1/2 patrial pressure, PGA use 0.5 times, measure input 0.9V) 010:AIN0 011:AIN1 100:AIN2 101:AIN3 110:AIN4 111:AIN5 Note: Regardless of which channel is selected, automatic temperature measurement is the highest priority.	R/W	0
Remark: the above registers could write only when ST=0 in SAR_START.				

9.2.2 SAR-ADC Start Register SAR_START (0x04)

Offset Address 0x04

Bit	Name	Description	R/W Sign	Reset Value

31:01	Reserved	Reserved	R	0
0	ST	SAR-ADC Start Bit 0: SAR-ADC No Operation; 1: Start SAR-ADC sampling one time, automatically clear after the completion of the sampling. Note: Automatic temperature measurement controlled by the RTC is not controlled by the bit, and a higher priority than the configuration bits. When the ADC_START bit is 1, the software is forbidden to write 1 again to start the SAR-ADC measurement; after the last SAR-ADC conversion is completed, the bit is 0 after 100us (that is, after detecting 0, then delay 100us), the new can be started. SAR-ADC measurement; recommended timeout wait time = 2* (REF_WAIT + SAR_WAIT) + 3ms.	R/W	0

9.2.3 SAR-ADC Status Register SAR_STATUS (0x08)

Offset Address 0x08

Bit	Name	Description	R/W Sign	Reset Value
31:02	Reserved	Reserved	R	0
1	TPS_BUSY	Automatical temperature measurement bit, =1:Automatical temperature measurement in process; =0: No automatical temperature measurement. When TPS_BUSY =1, software write ADC_START register, hardware operations will work after TPS_BUSY is 0.	R	0
0	DREADY	ADC Date Ready Pending Bit 0:ADC The conversion result is not completed 1:ADC The conversion result is completed Note: 1. Write 1 clear; 2. Automatical temperature measurement controlled by the RTC is not indicated in this state;	R/W	0

9.2.4 ADC Data RegisterSAR_DAT (0x0c)

Offset address 0x0C

Bit	Name	Description	R/W Sign	Reset Value
15:10	Reserved	Reserved	R	0
9:0	SAR-DAT	The conversion result of ADC	R	0

9.2.5 LVD Control Register LVD_CTL (0x10)

Offset address 0x10

Bit	Name	Description	R/W Sign	Reset Value
31:8	Reserved	Reserved	R	0
7	CMP2IE	Comparator 2 enable interrupt: =0:Enable uninterrupted; =1:Enable interrupt;	R/W	0
6	CMP1IE	Comparator 1 enable interrupt: =0:Enable uninterrupted; =1:Enable interrupt;	R/W	0
5	LVDIE	LVD enable interrupt: =0:Enable uninterrupted; =1:Enable interrupt;	R/W	0
4	Reserved	Reserved	R	0
3:0	LVDS	Set LVD threshold voltage: 0000 2.7 0001 2.7 0010 2.7 0011 2.9 0100 3.1 0101 3.3 0110 3.5 0111 3.7 1000 3.9 1001 4.1 1010 4.3 1011 4.5 1100 4.7 1101 4.9 1110 Detect voltage of external pin LVDIN0 to compare with LBGR(1.25V); 1111 Reserved;	R/W	0

Note: LVD, Comparator 1 and Comparator 2 merge an interrupt vector; SAR is a single interrupt vector.

9.2.6 LVD Status Register LVD_STAT (0x14)

Offset address 0x14

Bit	Name	Description	R/W Sign	Reset Value
31:7	Reserved	Reserved	R	0
6	CMP2IIF	Comparator 2 interrupt flag =0:No interrupt is generated;=1:Generate an interrupt; Interrupt is generated when the input voltage is low relative to the threshold or becomes high, cleared by writing 1;	R/W	0
5	CMP1IIF	Comparator 1 interrupt flag =0:No interrupt is generated;=1:Generate an interrupt; Interrupt is generated when the input voltage is low relative to the threshold or becomes high, cleared by writing 1;	R/W	0
4	LVDIIF	LVD interrupt flag =0:No interrupt is generated;=1:Generate an interrupt; Interrupt is generated when the input voltage is low relative to the threshold or becomes high, cleared by writing 1;	R/W	0
3	Reserved	Reserved	R	0
2	CMP2IF	Comparator 2 status flag =0:Below threshold;	R	0

		=1:Above threshold, only read;		
1	CMP1IF	Comparator 2 status flag =0:Below threshold; =1:Above threshold; only read;	R	0
0	LVDIF	LVD status flag =0:Below threshold; =1:Above threshold, only read;	R	0

9.2.7 ADC Contorl Register1 SAR_CTL1 (0x18) (new)

SAR-ADC gain control register

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:15	---	Reserved	R	0x0
14:8	SAR_CONVERT	The waiting time for SAR ADC sampling conversion to digital sampling DOUT: 6'd0: Tsar_clk (Tsar_clk is SARInterface clock cycle, refer to ANA_ RCH register) … 6'd127: (127+1)* Tsar_clk i.e. Waiting time=(SAR_CONVERT +1)*Tsar_clk Note: SAR_CONVERT[6:0]>=(SAR_SAMPLE[2:0]*2+21)*2Tsar_clk	R/W	0x29
7	--	Reserved	R	0
6:4	SAR_SAMPLE	Clock number of Sampling cycle: 000: 1 clk 001: 2 clk 010: 3 clk …… 111: 8 clk	R/W	0x0
3	SAR_PGA_SEL	Selection instructions of SAR-ADC gain control register =0, choose SAR_CTL[4:3] as SAR-ADC gain control register; =1, choose SAR_CTL1[2:0] as SAR-ADC gain control register	R/W	0x0
2:0	SAR_PGA	SAR-ADC gain control bit 000: 0.5 times 001: 1 times 010: 1.5 times 011: 2 times 1xx: 0.25 times	R/W	0x0

9.2.8 ADC Data Register2 SAR_DAT2 (0x1C) (new)

ADC DAT register 2

Bit	**Name**	**Description**	**R/W Sign**	**Reset**

				Value
15:12	---	Reserved	R	0
11:0	SAR_DAT2	ADC converting result	R	0

9.2.9 AIN function configuration register ANA_PAD

ANA PAD register

Offset address = 0x3c

Bit	Name	Description	R/W Sign	Reset Value
31:3	---	Reserved	R	0
1	P45_AEN	SAR ADC Measuring input switch 0:close; 1:open:	R/W	1’b0
0	P44_AEN	SAR ADC Measuring input switch 0:close: 1:open:	R/W	1’b0

Note: If you want to use the AIN function of the P44/P45 port (SAR ADC measurement input), you must configure the ANA_PAD register (0x3C).

9.2.10 Local reset register ANA_RST

ANA RST register.

Offset address = 0x40

Bit	Name	Description	R/W Sign	Reset Value
31:16	---	Reserved	R	0
15:8	ps	Write password protection bit for [7:0] =0xa5, [7:0] can be written = other values, [7:0] cannot be written	R/W	0
7:0	ana_rst	Forced reset of all registers with reset on VBAT power domain, excluding perpetual calendar register =0xa8, reset all registers with reset terminal in VBAT power domain = other, does not work	R/W	0

Note: To reset the registers of the VBAT power domain, proceed as follows:

Open the password protection bit of the ANA RST register: ANA RST = 0xa500;

1. Put the VBAT power domain in the reset state: ANA RST = 0xa5a8;
2. Delay 300us;
3. Release the VBAT power domain reset state: ANA RST = 0xa500;
4. Disable the password protection bit of the ANA RST register: ANA RST = 0x0000

9.3 ADC voltage detection steps

1. Configure the system control chapter module enable 1 register MOD1_EN bit 11 to 1, turn on the SAR_EN clock.
2. Determine whether the first bit of SAR_ADC status register SAR_STATUS is 0. No automatic temperature measurement is performed. If it is 0, it will enter the next step. If it is 1, it will wait.
3. Configure the ADC control register SAR_CTL, configure the wait time and gain, and select the corresponding ADC channel for the channel.
4. The SAR-ADC start register SAR_START is configured to 1 to initiate ADC conversion.

5. Determine the 0th bit of the SAR-ADC status register SAR_STATUS and wait for the conversion to complete.

6. Read the ADC conversion data register SAR_DAT.

7. Calculation: The ADC reference voltage source is 1.25V, and the ADC DAT register value is 1024 when the ADC is full. The calculation formula is (ADC DAT *1.25) / 1024. When the voltage value exceeds the full scale of the ADC, the conversion value is 1024.

8. Conversion and calculation is completed.

9.4 VBAT voltage detection

1. Configure the system control chapter module enable 1 register MOD1_EN bit 11 to 1, turn on the SAR_EN clock.

2. Determine whether the first bit of SAR_ADC status register SAR_STATUS is 0. No automatic temperature measurement is performed. If it is 0, it will enter the next step. If it is 1, it will wait.

3. Configure the ADC control register SAR_CTL channel to be VBAT, configure the waiting time;

4. The SAR-ADC start register SAR_START is configured to 1 to initiate ADC conversion.

5. Determine the 0th bit of the SAR-ADC status register SAR_STATUS and wait for the conversion to complete.

6. Read the ADC conversion data register SAR_DAT.

7. Calculation: The ADC reference voltage source is 1.25V, and the ADC DAT register value is 1024 when the ADC is full. The 3.6V battery is directly connected to the VBAT pin, and the MADC->AD_CTRL is configured to 0x01. At this time, the gain is 0.5 times, then the voltage is the value is (ADC DAT *1.25*4) / 1024, where the gain is 0.5 times, the internal VBAT access has 1/2 partial pressure, so the actual voltage needs to be multiplied by 4.

8. Conversion and calculation are completed.

9.5 Low Voltage Detection Application

1. For the energy meter with battery, in order to save battery power, it is necessary to judge the external power state, and let the system enter sleep mode when the power is low. The power detection module needs to be always on. The CMP2/CMP1 comparator is a very low power comparator. This comparator can be used to detect the power supply voltage and operate in battery mode. The CMP input pin voltage is compared with the reference 1.25V and an interrupt is generated. And status flags.

2. Configure the system control chapter module enable 1 register MOD1_EN bit 11 to 1, turn on the SAR_EN clock.

3. Configure the system control section. The 5th position 0 of the system power-down control register SYS_PD turns on the CMP2 power supply.

4. Configure the LVD control register LVD_CTL to enable the CMP2 interrupt. Enable CMP2 interrupt NVIC_EnableIRQ (CMP_IRQn).

5. Write the interrupt service program:

```
Void CMP_HANDLER (void)
{
If(!(MADC->LVD_STAT & 0x04))
{
}
Else
{
}
MADC->LVD_STAT = 0x01ff;
}
```

Since LVD, CMP1, and CMP2 use the same interrupt service routine, the interrupt status needs to be judged based on LVD_STAT.

6. Complete.

The CMP hardware filtering time is 100 system clocks. It is recommended that the software confirm and filter the CMP status after the interrupt wakes up.

10 GPIO

10.1 Overview

- Contain PA,PB,PC three GPIO
- PA ports include 5 P0 ports,8 P1 ports, 8 P2 ports, 8 P3 ports
- PB ports include 8 P4 ports, 8 P5 ports, 8 P6 ports, 8 P7 ports
- PC ports include 8 P8 ports, 8 P9 ports ,8 P10 ports,8 P11 ports
- PD ports include 8 P12 ports, 5 P13 ports ,3 P14 ports
- GPIO is peripheral of AHB
- Support bitband operation;
- Note: The SEG/IO multiplexed pin of the RN8611/8613/8615/8318 is an open-drain structure when used as an IO port. The RN8612/RN8610B is a common structure.
- Supports the power pulse forwarding IOCNT function, See Section 10.4 for details

10.2 Register description

GPIO Register Base Address:

Module Name	Physical Address	Address Mapping
GPIO	0x50000000	0x50000000

GPIO Register Base Address:

Register Name	Address Offset	Description
PMA	0x00H	PA port mode register(input or output)
PA	0x04H	PA port data register
PCA0	0x08H	PA port reuse register 0
PCA1	0x0CH	PA port reuse register 1
PUA	0x10H	PA port pull-up selection register
PIMA	0x14H	PA port input mode configuration
PIEA	0x18H	PA port input enable selection
PMB	0x1CH	PB port mode register(input or output)
PB	0x20H	PB port data register
PCB	0x24H	PB port reuse register
PUB	0x28H	PUB port pull-up selection register

PIMB	0x2CH	PB port input mode configuration
PIEB	0x30H	PB port input enable selection
PMC	0x34H	PC port mode register(input or output)
PC	0x38H	PC port date register
PCC	0x3CH	PC port reuse register
PUC	0x40H	PUC port pull-up selection register
PIEC	0x44H	PC port input enable selection
PIMC	0x48H	PC port input mode register
PCB2	0X4CH	PB port multiplexes register 2
PMD	0x50H	PD port mode register(input or output)
PD	0x54H	PD port date register
PCD	0x58H	PD port reuse register
PUD	0x5CH	PUD port pull-up selection register
PCE	0x60H	SEGCOM port multiplexes register
PASET	0X64H	PA port data reset register, write 1 to this register, a correspond bit in PA port will be write 1;
PACLR	0X68H	PA port data clear register, write 1 to this register, a correspond bit in PA port will be cleared;
PBSET	0X6CH	PB port data reset register, write 1 to this register, a correspond bit in PB port will be write 1;
PBCLR	0X70H	PB port data clear register, write 1 to this register, a correspond bit in PB port will be cleared;
PCSET	0X74H	PC port data reset register, write 1 to this register, a correspond bit in PC port will be write 1;
PCCLR	0X78H	PC port data clear register, write 1 to this register, a correspond bit in PC port will be cleared;
PDSET	0X7CH	PD port data reset register, write 1 to this register, a correspond bit in PD port will be write 1;
PDCLR	0X80H	PD port data clear register, write 1 to this register, a correspond bit in PD port will be cleared;
PIED	0x84H	PD port input enable selection

PIMD	0x88H	PD port input mode register
P34_CFG	0X100	P34 port pulse forwarding function set register
P50_CFG	0X104	P50 port pulse forwarding function set register
P51_CFG	0X108	P51 port pulse forwarding function set register
IOCNT_CTL	0X10C	IOCNT pulse forwarding control register

Note: IO port type, see Chapter 1.4 Pin Arrangement.

Recommends using bitband function (see chapter 4.3.2) to access GPIO registers, facilitate the IO port register bit operations related.

Also can be SET / CLR register (0x64H ~ 0x78H) write GPIO data register;

If the IO port configuration options for the multiplexing function outside the IO port, mode register, the data register, input enable register is invalid, the pull-up selection, input mode selection is valid in all multiplex configuration.

10.2.1 PA port mode register PMA (input or output) (0x00)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PM37~PM30	=0 Output Mode =1 Input Mode PM37 and PM36 are read only, read 1, only can be input model;	R/W	FF
23:16	PM27~PM20	=0 Output Mode =1 Input Mode	R/W	FF
15:8	PM17~PM10	=0 Output Mode =1 Input Mode	R/W	FF
7:5	---	Reserved	R	0
4:0	PM04~PM00	=0 Output Mode =1 Input Mode	R/W	1F

10.2.2 PA port data register PA (0x04)

Bit	Name	Description	R/W Sign	Reset Value
31:30	**P37~P36**	**P36 and P37 data input register, read-only;**	**R**	**0**
29:24	P35~P30	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00
23:16	P27~P20	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00
15:8	P17~P10	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00

7:5	---	Reserved	R	0
4:0	P04~P00	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value If defined as an analog input, the input mode to read a value of 0.	R/W	00

10.2.3 PA port reuse 0 register PCA0 (0x08) (Modified)

When selected as an analog input, the input mode is automatically selected, PMA register is invalid.

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:30	UART2_SEL	Define UART2 multiplex configuration: =00：set P24/P25 to UART2 =01：set P00/P01 to UART2 =10：set P22/P23 to UART2 =11：set P26/P27 to UART2	R/W	0
29	SWD_SEL	=0:P24 and P25 don't select SWD, define by PC245(bit27); =1:P24 and P25 select SWD; P24 is multiplexed to SWDCLK P25 is multiplexed to SWDIO	R/W	1
28	PC267	Define the port P26 and P27 multiplex configuration: =0: select IO port; =1: select UART3 port. P26 is multiplexed to RX3 P27 is multiplexed to TX3	R/W	00
27	PC245	Define the port P24 and P25 multiplex configuration: =0: select IO port; =1: select UART2 port. P24 is multiplexed to RX2 P25 is multiplexed to TX2	R/W	00
26	PC223	Define the port P22 and P23 multiplex configuration: =0: select IO port; =1: select UART1 port. P22 is multiplexed to RX1 P23 is multiplexed to TX1	R/W	00
25	PC201	Define the port P20 and P21 multiplex configuration: =0: select IO port; =1: select UART0 port. P20 is multiplexed to RX0 P21 is multiplexed to TX0	R/W	00
24：9	PC17[1:0] PC16[1:0] PC15[1:0] PC14[1:0] PC13[1:0]	Define the port P10~P17 multiplex configuration: =00:select IO port; =01:select KEY input port; =10: select TC output port; =11: select TC input port.	R/W	00

	PC12[1:0] PC11[1:0] PC10[1:0]	The TC outputs corresponding to P17~P10 are {tc1_p[1], tc1_n[1], tc1_p[0], tc1_n[0], tc0_p[1], tc0_n[1], tc0_p[0], tc0_n[0]} respectively.		
8	KEY4_SEL	=0: The function of P04 is determined by PC04 bit; =1: P04 is selected as KEY4 (PC14 is selected as KEY4 with high priority);	R/W	0
7:4	PC05~PC02	Define the port P05~P02 multiplex configuration: =0:select IO port; =1:select analog input port	R/W	0
3:2	PC01	Define the port P01 multiplex configuration: =00:select IO port; =01:select analog input port; =10: Select KEY3 (PC13 selects KEY3 as high priority); =11: Select as TX2 ;	R/W	0
1:0	PC00	Define the port P00 multiplex configuration: =00:select IO port; =01:select analog input port =10: Select KEY2 (PC12 selects KEY2 as high priority); =11: Select as RX2 ;	R/W	0

10.2.4 PA port reuse 1 register PCA1 (0x0C) (Modified)

Bit	Name	Description	R/W Sign	Reset Value
31:18	---	Reserved	R	0
17	PC267_2	Define the port P26 and P27 multiplex configuration 2: =0:The function of P26 and P27 is determined by the PC267 register bit (PCA0.bit28) =1: P26 and P27 are selected as UART2 P26 is multiplexed to RX2 P27 is multiplexed to TX2	R/W	0
16	PC223_2	Define the port P22 and P23 multiplex configuration 2: =0:The function of P22 and P23 is determined by the PC223 register bit (PCA0.bit26) =1: P22 and P23 are selected as UART2 P22 is multiplexed to RX2 P23 is multiplexed to TX2	R/W	0
15:14	PC37[1:0]	Define the port P37 multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT7; =1x:select pin crystal POSCI Note: Just PC36 [1] and PC37 [1] in any one of the high, then choose to POSC	R/W	0
13:12	PC36[1:0]	Define the port P36 multiplex configuration: =00:select IO port;	R/W	0

		=01:select an external interrupt input port INT6; =1x:select pin crystal POSCO Note:Just PC36 [1] and PC37 [1] in any one of the high, then choose to POSC		
11:10	PC35[1:0]	Define the port P35 multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT5; =10: select input for TC; =11: Select the output D2F_OUT2 of the power integration module D2F	R/W	0
9:8	PC34[1:0]	Define the port P34 multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT4; =10:select as the apparent energy pulse output SF_OUT; =11: Select the output D2F_OUT1 of the power integration module D2F SF is a direct output after INT4 input, and can also be configured as a reverse output, or a frequency division output, the specific configuration of the reference pulse forwarding function. This function is used to directly forward the output after the input pulse from the metering chip for electrical energy accuracy correction.	R/W	0
7:6	PC33[1:0]	Define the port P33 multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT3; =10: select input for TC; =11: Select the output D2F_OUT0 of the power integration module D2F	R/W	0
5:4	PC32[1:0]	Define the port P32 multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT2; =10:select as RTC output RTC_OUT (the default choice for the RTC output) =11: select IO port.	R/W	10
3:2	PC31[1:0]	Define the port P31multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT1; =10: select input for TC; =11: select RX4;	R/W	0
1:0	PC30[1:0]	Define the port P30 multiplex configuration: =00:select IO port; =01:select an external interrupt input port INT0; =10:select TC input; =11:Reserved;	R/W	0

10.2.5 PA port pull-up select register PUA (0x10)

Note: When the IO port in output mode or analog PAD mode, regardless of how the configuration register PU, PIN not enable pull-up.

Bit	Name	Description	R/W Sign	Reset Value
31:24	PU37~PU30	Define port pull-up configuration: =0:Don't select pull-up; =1:Select pull-up;	R/W	00
23:16	PU27~PU20	Define port pull-up configuration: =0:Don't select pull-up; =1:Select pull-up; Note: P24 and P25 as SWD default pull-up enable.	R/W	30
15:8	PU17~PU10	Define port pull-up configuration: =0:Don't select pull-up; =1:Select pull-up;	R/W	00
7:5	--	Reserved	R	0
4:0	PU04~PU00	Define port pull-up configuration: =0:Don't select pull-up; =1:Select pull-up;	R/W	00H

10.2.6 PA port input mode configuration register PIMA (0x14)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PIL27~PIL20	Define port P20~P27 input buffer type: =0:CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1:TTL buffer, Vil=0.16VCC Vih=0.4VCC;	R/W	00
23:16	PIL17~PIL10	Define port P10~P17 input buffer type: =0:CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1:TTL buffer, Vil=0.16VCC Vih=0.4VCC;	R/W	00
15:8	PID27~PID20	Define port P20~P27 whether is N-ch open-drain output: =0:Normal mode; =1:N-ch open-drain output mode;	R/W	00
7:0	PID17~PID10	Define port P10~P17 whether is N-ch open-drain output: =0:Normal mode; =1:N-ch open-drain output mode;	R/W	00

10.2.7 PA port input enable register PIEA (0x18)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PIE37~PIE30	Input Enable: =1:No input enable; =0:Input enable; Note: The P30 is set to enter the electricity needs BOOTROM enabled, ISP is easy to detect.	R/W	FF

23:16	PIE27~PIE20	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
15:8	PIE17~PIE10	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
7:5	--	Reserved	R	0
4:0	PIE04~PIE00	Input enable: =1:No input enable; =0:Input enable;	R/W	3F

10.2.8 PB port mode register PMB (input or output) (0x1C)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PM77~PM70	=0 Output mode =1 Input mode	R/W	FF
23:16	PM67~PM60	=0 Output mode =1 Input mode	R/W	FF
15:8	PM57~PM50	=0 Output mode =1 Input mode	R/W	FF
7:0	PM47~PM40	=0 Output mode =1 Input mode	R/W	FF

When the IO port is set to 7816 port or SPI port, direction register does not work, it controls by the communication module itself.

10.2.9 PB port data register PB (0x20)

Bit	Name	Description	R/W Sign	Reset Value
31:24	P77~P70	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00
23:16	P67~P60	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00
15:8	P57~P50	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00
7:0	P47~P40	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00

10.2.10 PB port reuse register PCB (0x24)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PC77~PC70	PC77~PC70 define port multiplex configuration:	R/W	00

		=0: select IO port; =1: select LCD.		
23:16	PC67~PC60	PC67~PC60 define port multiplex configuration: =0: select IO port; =1: select LCD.	R/W	00
15:14	PC57~PC56	PC57~PC56 define port multiplex configuration: =0: select IO port; =1: Reserved Note: The RTCOUT multiplexing relationship of port P56 is determined by the RTC chapter register and is independent of the GPIO chapter multiplexing configuration register. When VCC has no power and VBAT is powered on, P56 outputs 1Hz by default. Once VCC is powered up, the 1Hz output will be turned off. Peripheral circuit design needs to pay attention to this problem.	R/W	0
13:12	PC55~PC54	Define the port P55 and P54 multiplex configuration: =0: select IO port; =1: select UART5 port. P54 is multiplexed to RX5 P55 is multiplexed to TX5	R/W	0
11:10	PC53~PC52	Define the port P53 and P52 multiplex configuration: =0: select IO port; =1: select I2C port P52 is multiplexed to SCL P53 is multiplexed to SDA	R/W	0
9	PC51	Define the port P51 multiplex configuration: =0: select IO port; =1: select an external interrupt input port INT0; P51/QF is a direct output after INT0 input, and can also be configured as a reverse output, or a frequency division output, the specific configuration of the reference pulse forwarding function. This function is used to directly forward the output after the input pulse from the metering chip for electrical energy accuracy correction.	R/W	0
8	PC50	Define the port P50 multiplex configuration: =0: select IO port; =1: select an external interrupt input port INT2; P50/PF is a direct output after INT2 input, and can also be configured as a reverse output, or a frequency division output, the specific configuration of the reference pulse forwarding function. This function is used to directly forward the output after the input pulse from the metering chip for electrical energy accuracy correction.	R/W	0
7:6	PC47~PC46	Define the port P47 and P46 multiplex configuration:	R/W	0

		=0: select IO port; =1: select SPI0 port P46 is multiplexed to SPI0_MISO P47 is multiplexed to SPI0_MOSI		
5:4	PC45~PC44	Define the port P45 and P44 multiplex configuration: =0: select IO port; =1: select SPI0 port（This function is not recommended because of different power domains. P50/P51 can be reused for this function） P44 is multiplexed to SPI0_SCSN P45 is multiplexed to SPI0_SCLK The AIN5/AIN6 multiplexing relationship of port P44/P45 is determined by onfiguration of ANA_PAD in the Analog Peripherals chapter and It has the highest priority.	R/W	0
3:0	PC43~PC40	Define the port P43~P40 multiplex configuration: =0: select IO port; =1: select 7816 port P40 is multiplexed to 7816_CLK P41 is multiplexed to 78160_IO P42 is multiplexed to 78161_IO P43 is multiplexed to 78161_I	R/W	0

10.2.11 PB port reuse register 2PCB2 (0x4CH) (Modified)

Bit	Name	Description	R/W Sign	Reset Value
31:30	PC57_2	P57 port multiplexing configuration 2: =00: The P57 port function is determined by the PC57 register bit (0x24). =01: P57 is selected as TCIN. =10: P57 is selected as TC1_P[1]; =11: Reserved	R/W	00
29:28	PC56_2	P56 port multiplexing configuration 2: =00: The P56 port function is determined by the PC56 register bit (0x24). =01: P56 is selected as TCIN. =10: P56 is selected as TC1_N[1]; =11: Reserved Note: The RTCOUT multiplexing relationship of port P56 is determined by the RTC chapter register and is independent of the GPIO chapter multiplexing configuration register. When VCC has no power and VBAT is powered on, P56 outputs 1Hz by default. Once VCC is powered up, the 1Hz output will be turned off. Peripheral circuit design needs to pay attention to this problem.	R/W	00

27:26	PC55_2	P55 port multiplexing configuration 2: =00: The P55 port function is determined by the PC55 register bit (0x24); =01: P55 is selected as TCIN. =10: P55 is selected as TC1_P[0]; =11: P55 is selected as SPI1_MOSI	R/W	00
25:24	PC54_2	P54 port multiplexing configuration 2: =00: The P54 port function is determined by the PC54 register bit (0x24); =01: P54 is selected as TCIN. =10 : P54 is selected as TC1_N[0]; =11: P54 is selected as SPI1_MISO	R/W	00
23:22	PC53_2	P53 port multiplexing configuration 2: =00: The P53 port function is determined by the PC53 register bit (0x24); =01: P53 is selected as TCIN. =10: P53 is selected as TC1_P[1]; =11: P53 is selected as SPI1_SCLK	R/W	00
21:20	PC52_2	P52 port multiplexing configuration 2: =00: The P52 port function is determined by the PC52 register bit (0x24); =01: P23 is selected as TCIN. =10: P52 is selected as TC0_N[1]； =11: P52 is selected as SPI1_SCSN Note 1: P40~P43, P52~P55 cannot be multiplexed as SPI1 at the same time. Note 2: P52 is only valid in slave mode of SPI1, do not set to 1 in master mode of SPI1.	R/W	00
19:18	PC51_2	P51 port multiplexing configuration 2: =00: The P51 port function is determined by the PC51 register bit (0x24); =01: P51 is selected as RTC_OUT. =10: P51 is selected as SPI_SCLK; =11: SF2; (INT7 pin output) SF2 is a direct output after INT7 input, and can also be configured as a reverse output, or a frequency division output, the specific configuration of the reference pulse forwarding function. This function is used to directly forward the output after the input pulse from the metering chip for electrical energy accuracy correction.	R/W	00
17:16	PC50_2	P50 port multiplexing configuration 2: =00: The P50 port function is determined by the PC50 register bit (0x24); =01: P50 is selected as RTC_OUT. =10: P50 is selected as SPI_SCSN;	R/W	00

		=11: SF1 (INT6 pin output) Note 1: P50 is only valid in slave mode of SPI0, do not set to 1 in master mode of SPI0.P50/P51/P46/P47 and P110~P113 cannot be multiplexed as SPI0 at the same time, and PCC (0x3C) register bit28 determines which SPI0 group is effective. Note 2: SF1 is a direct output after INT6 input, and can also be configured as a reverse output, or a frequency division output, the specific configuration of the reference pulse forwarding function. This function is used to directly forward the output after the input pulse from the metering chip for electrical energy accuracy correction.		
15:14	PC47_2	P47 port multiplexing configuration 2: =00: The P47 port function is determined by the PC47 register bit (0x24); =01: P47 is selected as TX4. =10: P47 is select as the output D2F_OUT0 of the power integration module D2F =11: P47 is select as the output D2F_OUT1 of the power integration module D2F	R/W	00
13:12	PC46_2	P46 port multiplexing configuration 2: =00: The P46 port function is determined by the PC46 register bit (0x24); =01: P46 is selected as RX4. =10: P46 is select as the output D2F_OUT1 of the power integration module D2F =11: P46 is select as the output D2F_OUT2 of the power integration module D2F	R/W	00
11:10	PC45_2	P45 port multiplexing configuration 2: =00: The P45 port function is determined by the PC45 register bit (0x24); =01: P45 is selected as KEY7 (PC17 has a higher priority). =1x;reserved The AIN6 multiplexing relationship of port P45 is determined by onfiguration of ANA_PAD in the Analog Peripherals chapter and It has the highest priority.	R/W	00
9:8	PC44_2	P44 port multiplexing configuration 2: =00: The P44 port function is determined by the PC44 register bit (0x24); =01: P44 is selected as KEY6 (PC16 has a higher priority). =1x;reserved The AIN5 multiplexing relationship of port P44 is determined by onfiguration of ANA_PAD in the Analog Peripherals chapter and It has the highest priority.	R/W	00
7:6	PC43_2	P43 port multiplexing configuration 2:	R/W	00

		=00: The P43 port function is determined by the PC43 register bit (0x24); =01: P43 is selected as INT5. (PC35 has a high priority) =10: P43 is select as SPI1_MOSI =11: P43 is select as SPI2_MOSI		
5:4	PC42_2	P42 port multiplexing configuration 2: =00: The P42 port function is determined by the PC42 register bit (0x24); =01: P42 is selected as INT4. (PC34 has a high priority) =10：P42 is select as SPI1_MISO =11：P42 is select as SPI2_MISO	R/W	00
3:2	PC41_2	P41 port multiplexing configuration 2: =00: The P41 port function is determined by the PC41 register bit (0x24); =01: P41 is selected as INT3. (PC33 has a high priority) =10：P41 is select as SPI1_SCLK =11：P41 is select as SPI2_SCLK	R/W	00
1:0	PC40_2	P40 port multiplexing configuration 2: =00: The P40 port function is determined by the PC40 register bit (0x24); =01: P40 is selected as INT1. (PC31 has a high priority) =10：P40 is select as SPI1_SCSN =11：P40 is select as SPI2_SCSN Note 1: P40~P43, P52~P55 cannot be multiplexed as SPI1 at the same time. Note 2: P40 is only valid in slave mode of SPI1, do not set to 1 in master mode of SPI1.	R/W	00

10.2.12 PB port pull-up/pull-down selection register PUB (0x28)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PU77~PU70	PU77~PU70 define P7 port whether is internal pull-down: =0:No pull-down; =1: Internal pull-down.	R/W	00
23:16	PU67~PU60	PU67~PU60 define P6 port whether is internal pull-down: =0:No pull-down; =1: Internal pull-down.	R/W	00
15:8	PU57~PU50	PU57~PU50 define P5 port whether is internal pull-down: =0:No pull-up; =1: Internal pull-up.	R/W	00
7:0	PU47~PU40	PU47~PU40 define P4 port whether is internal pull-down: =0: No pull-up; =1: Internal pull-up.	R/W	00

10.2.13 PB port input mode register PIMB (0x2C)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PIL57~PIL50	Define port P50~P57 input buffer type: =0:CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1:TTL buffer, Vil=0.16VCC Vih=0.4VCC; Which PIL51 and PIL50 read-only bit 0;	R/W	00
23:16	PIL47~PIL40	Define port P40~P47 input buffer type: =0:CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1:TTL buffer, Vil=0.16VCC Vih=0.4VCC;	R/W	00
15:8	PID57~PID50	Define port P50~P57 whether is N-ch open-drain output: =0:Normal mode; =1:N-ch open-drain mode; Which PID51 and PID50 read-only bit 0;	R/W	00
7:0	PID47~PID40	Define port P40~P47 whether is N-ch open-drain output: =0:Normal mode; =1:N-ch open-drain mode;	R/W	00

10.2.14 PB port input enable register PIEB (0x30)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PIE77~PIE70	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
23:16	PIE67~PIE60	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
15:8	PIE57~PIE50	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
7:0	PIE47~PIE40	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF

10.2.15 PC port mode register PMC (input or output) (0x34)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PM117~PM110	=0 Output mode =1 Input mode	R/W	FF
23:16	PM107~PM100	=0 Output mode =1 Input mode	R/W	FF
15:8	PM97~PM90	=0 Output mode =1 Input mode	R/W	FF
7:0	PM87~PM80	=0 Output mode	R/W	FF

		=1 Input mode		

10.2.16 PC port data register PC (0x38)

Bit	Name	Description	R/W Sign	Reset Value
31:24	P117~P110	Define the data that the chip port needs to output. If the port is read in input mode, the pin level is read. If the port is read in output mode, the value of the output latch is read.	R/W	00
23:16	P107~P100	Define the data that the chip port needs to output. If the port is read in input mode, the pin level is read. If the port is read in output mode, the value of the output latch is read.	R/W	00
15:8	P97~P90	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00
7:0	P87~P80	Define data needed in the chip output port. If you read in the input mode, the pin level is read. If you read in the output mode, then read the output latch value	R/W	00

10.2.17 PC port reuse register PCC (0x3C) (Modified)

Bit	Name	Description	R/W Sign	Reset Value
31:29	Reserved	-----------	R	0
28	SPI_MUX	SPI_MUX defines whether P46 is used as SPI or P110. =0: Select P50/P51/P46/P47 as SPI0; =1: Select P110~P113 as SPI0;	R/W	00
27:24	PC113~PC110	PC113~PC110 define port multiplex configuration: =0:select IO port; =1: select SPI. P110 is selected as SPI0_SCSN P111 is selected as SPI0_SCLK P112 is selected as SPI0_MISO P113 is selected as SPI0_MOSI PC110 is only valid in slave mode of SPI0, do not set to 1 in master mode of SPI0.	R/W	00
23:20	PC107~PC104	PC107~PC104 define port multiplex configuration: =0:select IO port; =1: select SPI3. P104 is selected as SPI3_SCSN P105 is selected as SPI3_SCLK P106 is selected as SPI3_MISO P107 is selected as SPI3_MOSI PC104 is only valid in slave mode of SPI3, do not set to 1 in master mode of SPI3.	R/W	0
19:16	PC103~PC100	PC103~PC100 define port multiplexing configuration:	R/W	0

		=0: select as IO port; =1: select LCD.		
15:8	PC97~PC90	PC97~PC90 define port multiplex configuration: =0:select IO port; =1: select LCD.	R/W	00
7:0	PC87~PC80	PC87~PC80 define port multiplex configuration: =0:select IO port; =1: select LCD.	R/W	00

10.2.18 PC port pull-up/pull-down selection register PUC (0x40)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PU117~PU110	PU113~PU110 define port whether is internal pull-down: =0:No pull-up; =1: Internal pull-up.	R/W	00
23:16	PU107~PU100	PU107~PU100 define port whether is internal pull-down: =0:No pull-down; =1: Internal pull-down.	R/W	00
15:8	PU97~PU90	PU97~PU90 define port whether is internal pull-down: =0:No pull-down; =1: Internal pull-down.	R/W	00
7:0	PU87~PU80	PU87~PU80 define port whether is internal pull-down: =0:No pull-down; =1: Internal pull-down.	R/W	00

10.2.19 PC port input enable register PIEC (0x44)

Bit	Name	Description	R/W Sign	Reset Value
31:24	PIE117~PIE110	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
23:16	PIE107~PIE100	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
15:8	PIE97~PIE90	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF
7:0	PIE87~PIE80	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF

10.2.20 PC port input mode register PIMC (0x48)

Bit	Name	Description	R/W Sign	Reset Value

31:16	---	Reserved	R	0
15:12	PIL117~PIL114	Define port P117~P114 input buffer type: =0: CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1: TTL buffer, Vil=0.16VCC Vih=0.4VCC;	R/W	00
11:8	PID117~PID114	Define whether ports P117~P114 are N-ch open-drain outputs: =0: normal mode; =1: N-ch open drain mode;	R/W	00
7:4	PIL113~PIL110	Define port P113~P110 input buffer type: =0:CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1:TTL buffer, Vil=0.16VCC Vih=0.4VCC;	R/W	00
3:0	PID113~PID110	Define port P113~P110 whether is N-ch open-drain output: =0:Normal mode; =1:N-ch open-drain mode;	R/W	00

10.2.21 PD port mode register PMD (input or output) (0x50)

Bit	Name	Description	R/W Sign	Reset Value
31:24	---	Reserved	R	0
23:16	PM147~PM140	=0 Output mode =1 Input mode P143~P147 are not supported, and the configuration is invalid	R/W	FF
15:8	PM137~PM130	=0 Output mode =1 Input mode P133, P136, P137 are not supported, and the configuration is invalid	R/W	FF
7:0	PM127~PM120	=0 Output mode =1 Input mode	R/W	FF

10.2.22 PD port data register PD (0x54)

Bit	Name	Description	R/W Sign	Reset Value
31:24	---	Reserved	R	0
23:16	P147~P140	Define the data that the chip port needs to output. If the port is read in input mode, the pin level is read. If the port is read in output mode, the value of the output latch is read. P143~P147 are not supported, and the configuration is invalid	R/W	00
15:8	P137~P130	Define the data that the chip port needs to output. If the port is read in input mode, the pin level is read. If the port is read in output mode, the value of the output latch is read. P133, P136, P137 are not supported, and the configuration is invalid	R/W	00
7:0	P127~P120	Define the data that the chip port needs to output. If the port is read in input mode, the pin level is read. If the port is read in output mode, the value of the output latch is read.	R/W	00

10.2.23 PD port reuse register PCD (0x58)

Bit	Name	Description	R/W Sign	Reset Value
31:22	---	Reserved	R	00
21:20	PC142	Define the port P142 multiplex configuration: =00 : select IO port; =other : Reserved	R/W	0
19:18	PC141	Define the port P141 multiplex configuration: =00 : select IO port; =other : Reserved	R/W	0
17:16	PC140	Define the port P140 multiplex configuration: =00 : select IO port; =other : Reserved	R/W	0
15:8	PC137~PC130	PC137~PC130 define port multiplex configuration: =0 : select IO port; =1： Select LCD related pins or reserved P137~P136 reserved P135 corresponds to LCDVP2 P134 corresponds to LCDVP1 P133 reserved P132 corresponds to LCDVC P131 corresponds to LCDVB P130 corresponds to LCDVA	R/W	0x3F
7:0	PC127~PC120	PC127~PC120 define port multiplex configuration: =0 : select IO port; =1： select LCD P127~P120 corresponds to COM7~COM0	R/W	0xFF

10.2.24 PD port pull-up/pull-down selection register PUD (0x5C)

Bit	Name	Description	R/W Sign	Reset Value
31:24	---	Reserved	R	0
23:16	PU147~PU140	PU147~PU140 define port whether is internal pull-up: =0: No pull-up; =1: Internal pull-up. P143~P147 are not supported	R/W	00
15:8	PU137~PU130	PU137~PU130 define port whether is internal pull-down: =0: No pull-down; =1: Internal pull-down. P133/P136/P137 are not supported	R/W	00
7:0	PU127~PU120	PU127~PU120 define port whether is internal pull-down: =0: No pull-down; =1: Internal pull-down.	R/W	00

10.2.25 PD port input enable register PIED (0x84)

Bit	Name	Description	R/W Sign	Reset Value
31:24	---	Reserved	R	0
23:16	PIE147~PIE140	Input Enable: =1:No input enable; =0:Input enable; P143~P147 are not supported	R/W	FF
15:8	PIE137~PIE130	Input Enable: =1:No input enable; =0:Input enable; P133/P136/P137 are not supported	R/W	FF
7:0	PIE127~PIE120	Input Enable: =1:No input enable; =0:Input enable;	R/W	FF

10.2.26 PD port input mode register PIMD (0x88)

Bit	Name	Description	R/W Sign	Reset Value
31:16	---	Reserved	R	0
15:12	PIL147~PIL144	Define port P147~P144 input buffer type: =0: CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1: TTL buffer, Vil=0.16VCC Vih=0.4VCC; P144~P147 are not supported	R/W	00
11:8	PID147~PID144	Define whether ports P147~P144 are N-ch open-drain outputs: =0: normal mode; =1: N-ch open drain mode; P144~P147 are not supported	R/W	00
7:4	PIL143~PIL140	Define port P143~P140 input buffer type: =0:CMOS buffer, Vil=0.3VCC Vih=0.7VCC; =1:TTL buffer, Vil=0.16VCC Vih=0.4VCC; P143 are not supported	R/W	00
3:0	PID143~PID140	Define whether ports P143~P140 are N-ch open-drain outputs: =0: normal mode; =1: N-ch open drain mode; P143 are not supported	R/W	00

10.2.27 SEGCOM port reuse register PCE (0x60)

Bit	Name	Description	R/W Sign	Reset Value
31:4	---	Reserved	R	0
3:0	SEG3/COM7~SEG0/COM4	SEG3/COM7~SEG0/COM4 define port multiplex configuration: =0:select SEG;	R/W	00

		=1:select COM.		

10.2.28 PA port data set register PASET (0x64)

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:30	Reserved	-----------	R	0
29:24	P35~P30	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
23:16	P27~P20	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
15:8	P17~P10	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
7:5	---	Reserved	R	0
4:0	P04~P00	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00

Note: Read value is meaningless

10.2.29 PA port clear set register PACLR (0x68)

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:30	Reserved	-----------	R	00
29:24	P35~P30	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	0
23:16	P27~P20	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00
15:8	P17~P10	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00
7:5	---	Reserved	R	0
4:0	P04~P00	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00

Note: Read value is meaningless

10.2.30 PB port data set register PBSET (0x6C)

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:24	P77~P70	Set chip port status	R/W	00

		0:No effect 1:The port is set, output high level		
23:16	P67~P60	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
15:8	P57~P50	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
7:0	P47~P00	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00

Note: Read value is meaningless

10.2.31 PB port clear set register PBCLR (0x70)

Bit	Name	Description	R/W Sign	Reset Value
31:24	P77~P70	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	0
23:16	P67~P60	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00
15:8	P57~P50	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00
7:0	P47~P40	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00

Note: Read value is meaningless

10.2.32 PC port data set register PCSET (0x74)

Bit	Name	Description	R/W Sign	Reset Value
31:24	P117~P110	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
23:16	P107~P100	Set chip port status 0:No effect 1:The port is set, output high level	R/W	0
15:8	P97~P90	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00
7:0	P87~P80	Set chip port status 0:No effect	R/W	00

		1:The port is set, output high level		

Note: Read value is meaningless

10.2.33 PC port clear set register PCCLR (0x78)

Bit	Name	Description	R/W Sign	Reset Value
31:24	P117~P110	Clear chip port status 0:No effect 1:The port is cleared, output high level	R/W	00
23:16	P107~P100	Clear chip port status 0:No effect 1:The port is cleared, output high level	R/W	00
15:8	P97~P90	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00
7:0	P87~P80	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00

Note: Read value is meaningless

10.2.34 PD port data set register PDSET (0x7C)

Bit	Name	Description	R/W Sign	Reset Value
31:24	---	Reserved	R	00
23:16	P147~P140	Set chip port status 0:No effect 1:The port is set, output high level P143~P147 are not supported	R/W	00
15:8	P137~P130	Set chip port status 0:No effect 1:The port is set, output high level P133/P136/P137 are not supported	R/W	00
7:0	P127~P120	Set chip port status 0:No effect 1:The port is set, output high level	R/W	00

10.2.35 PD port clear set register PDCLR (0x80)

Bit	Name	Description	R/W Sign	Reset Value
31:24	Reserved	-----------	R	00
23:16	P147~P140	Clear chip port status 0:No effect 1:The port is cleared, output low level P143~P147 are not supported	R/W	00

15:8	P137~P130	Clear chip port status 0:No effect 1:The port is cleared, output low level P133/P136/P137 are not supported	R/W	00
7:0	P127~P120	Clear chip port status 0:No effect 1:The port is cleared, output low level	R/W	00

10.3 GPIO operation procedure

1、Configure the system control chapter module enable 1 register MOD1_EN bit 5 is 1 to turn on the GPIO module clock.

2、Configure the GPIO input and output mode.

3、Configure the GPIO port data register.

4、Configure the GPIO port multiplexing function. After selecting the GPIO multiplexing function, the GPIO input and output functions will follow the GPIO multiplexing configuration.

5、When the MCU is powered by 5v, and the peripheral I2C, SPI or other device operates at 3.3V, you can select GPIO that can be configured as an N-ch open-drain output and an input buffer type with TTL mode.

6、When using as an input IO port, configure the corresponding bit of the input enable register to 0 to enable the input enable. In low power mode, the IO port can be configured as input mode and the input enable can be turned off.

10.4 Pulse forwarding function IOCNT

The metering pulse forwarding function supports the metering pulse input from INT5/2/4/6/7 and output directly from QF/PF/SF1/SF2/SF after the IOCNT module.

10.4.1 Functional characteristics

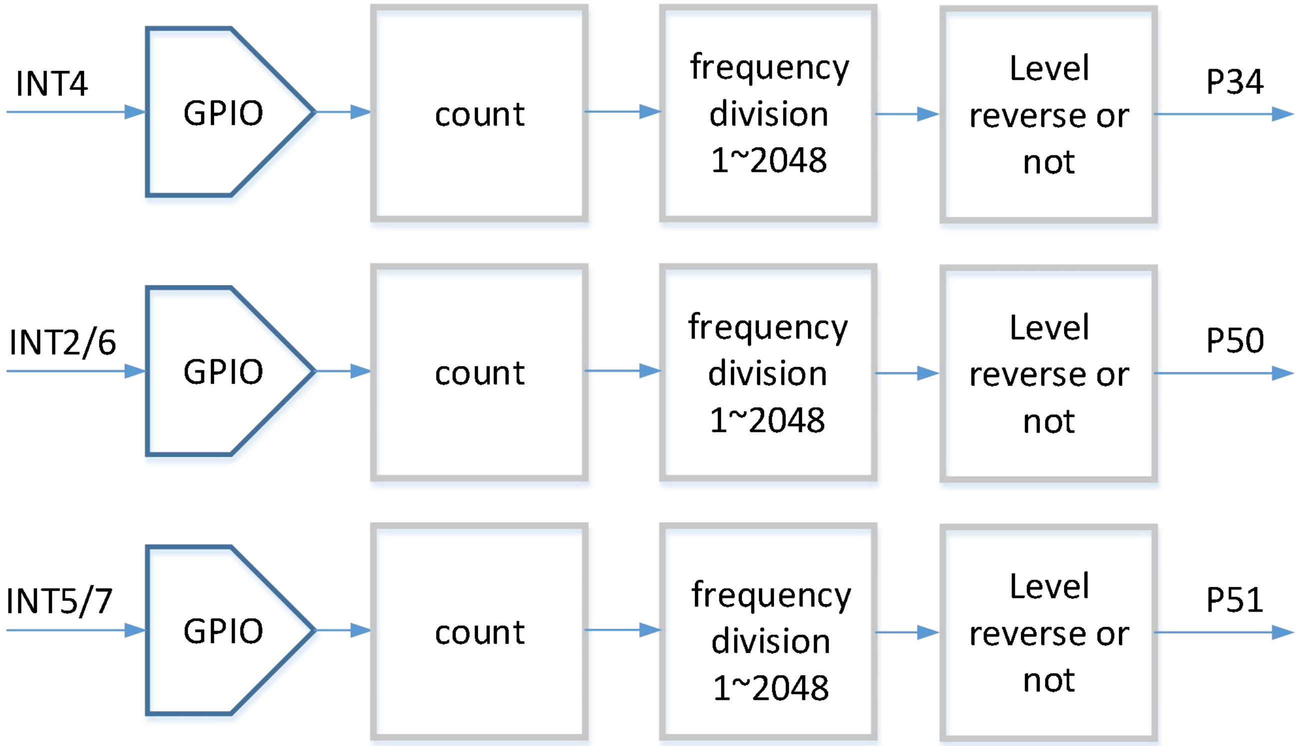
- Support input pulse direct forwarding output
- The input pulse level is reversed and then the output is forwarded
- Support input pulse 1~2048 frequency division and then forward output
- Input pulse counter Rising edge count and falling edge count can be configured
- Supports zero clearing after reading the input pulse meter value

10.4.2 Multiplex

- P50/PF is the output after INT2 input
- P50/SF1 is the output after INT6 input
- P51/QF is the output after INT5 input
- P51/SF2 is the output after INT7 input
- P34/SF is the output after INT4 input

10.4.3 Function description

The P34/P50/P51 channels are independent and can be configured with different frequency division ratios and level reverse output functions. The output pulse maintains a high level of 80ms, and if it is faster than 80ms, it maintains an equal duty output. Filter the input pulse for 32 system clock cycles. As shown in the figure:



10.4.4 Register description

10.4.4.1 P34 Pulse forwarding function set register P34_CFG（0x100）

Bit	Name	Description	R/W Sign	Reset Value
31	P34INT	=0: Keep the direction of P34 after frequency division unchanged, do not do reverse processing; =1: The signal after P34 frequency division is reversed.	R/W	00
30:20	P34CFG	Determine the pulse division ratio input to the P34 port (INT4) : =0: No frequency division =1: 2 frequency division … =7ff：2048 frequency division Frequency division ratio = P50CFG +1。 Maintain a high level for 80ms, and if it is faster than 80ms, maintain an equal duty; It is recommended to configure P50CFG as odd, which makes it easier to achieve equal duty.	R/W	00
19:0	P34CNT	Input pulse counter The counter is reset to zero by default after reading and can be configured as an accumulation type; The counting edge is configurable, with a default rising edge counting.	R	00

10.4.4.2 P50 Pulse forwarding function set register P50_CFG（0x104）

Bit	Name	Description	R/W Sign	Reset Value
31	P50INT	=0: Keep the direction of P50 after frequency division unchanged, do not do reverse processing; =1: The signal after P50 frequency division is reversed.	R/W	00
30:20	P50CFG	Determine the pulse division ratio input to the P50 port (INT2/6) : =0: No frequency division =1: 2 frequency division … =7ff：2048 frequency division Frequency division ratio = P50CFG +1。 Maintain a high level for 80ms, and if it is faster than 80ms, maintain an equal duty; It is recommended to configure P50CFG as odd, which makes it easier to achieve equal duty.	R/W	00
19:0	P50CNT	Input pulse counter The counter is reset to zero by default after reading and can be configured as an accumulation type; The counting edge is configurable, with a default rising edge counting.	R	00

10.4.4.3 P51 Pulse forwarding function set register P51_CFG（0x108）

Bit	Name	Description	R/W Sign	Reset Value
31	P51INT	=0: Keep the direction of P51 after frequency division unchanged, do not do reverse processing; =1: The signal after P51 frequency division is reversed.	R/W	00
30:20	P51CFG	Determine the pulse division ratio input to the P50 port (INT5/7) : =0: No frequency division =1: 2 frequency division … =7ff：2048 frequency division Frequency division ratio = P50CFG +1。 Maintain a high level for 80ms, and if it is faster than 80ms, maintain an equal duty; It is recommended to configure P50CFG as odd, which makes it easier to achieve equal duty.	R/W	00
19:0	P51CNT	Input pulse counter The counter is reset to zero by default after reading and can be configured as an accumulation type; The counting edge is configurable, with a default rising edge counting.	R	00

10.4.4.4 Pulse forwarding contr register IOCNT_CTL（0x10C）

Bit	Name	Description	R/W Sign	Reset Value
31:17	--	Reserved	R	0
16	CNTREG_CLR	IOCNT pulse counting type selection =0, Zeroing after reading =1, Cumulative type Default Zeroing after reading	R/W	0
15:11	--	Reserved	R	0
10	P51_MODE	P51 Pulse counting edge selection =0, Rising edge count =1, Falling edge count Default Rising edge count	R/W	0
9	P50_MODE	P50 Pulse counting edge selection =0, Rising edge count =1, Falling edge count Default Rising edge count	R/W	0
8	P34_MODE	P34 Pulse counting edge selection =0, Rising edge count =1, Falling edge count Default Rising edge count	R/W	0
7:3	--	Reserved	R	0
2	P51CNT_EN	P51 IOCNT enable configuration =0, Not enable =1, enable IOCNT	R/W	0
1	P50CNT_EN	P50 IOCNT enable configuration =0, Not enable =1, enable IOCNT	R/W	0
0	P34CNT_EN	P34 IOCNT enable configuration =0, Not enable =1, enable IOCNT	R/W	0

10.4.5 Operation steps

10.4.5.1 High leve pulse width effective

When the input signal is valid at a high level, it is recommended to connect an external pull-down resistor to the INTx hardware circuit, so that the INTx defaults to a low level input and uses rising edge counting. The steps are as follows:

1) Configure IOCNT_CTL register PXX_MODE, select rising edge count
2) Configure GPIO multiplex relationship
3) frequency division counting and level reversal can be configured
4) Configure pulse forwarding enable

10.4.5.2 Low level pulse width effective

When the input signal is valid at a low level, it is recommended to connect an external pull-up resistor to the INTx hardware circuit, so that the INTx defaults to a high level input and uses falling edge counting. The steps are as follows:

1) Configure IOCNT_CTL register PXX_MODE, select rising edge count
2) Configure GPIO multiplex relationship
3) frequency division counting and level reversal can be configured
4) Configure pulse forwarding enable

11 External Interrupt Controller

A built-in external interrupt controller (INTC) is used to process the interrupt request input from the chip pin, and can automatically wake up the CPU by interrupt when the CPU is hibernating.

11.1 Overview

External interrupt controller has the following features:

- Support 8 external interrupt mode settings: upper and lower edges and double edges can be set;
- Support for external interrupt status indication;
- Support for external interrupt trigger by software;
- Support for external interrupt status;
- Support for external interrupt mask;
- Supports external interrupt filtering. The filtering time in running mode is about 10 us.

11.2 Register Description

Module Register Base Address

Module Name	Physical Address	Address Mapping
INTC	0x40044000	0x40044000

INTC Module Register Address Offset

Register Name	Address Offset	Description
INTC_CTL	0x0	INTC Control Register
INTC_MODE	0x4	INTC Mode Register
INTC_MASK	0x8	INTC Mask Register
INTC_STA	0xc	INTC Status Register

11.2.1 INTC_CTL

INTC Control Register Address 0x40044000+0x0

Bit	Name	Description	R/W Sign	Reset Value
31:09	---	Reserved	R	0
8	IRQ_CTL	External interrupt merge function configuration： 0： 8 external interrupts INT0 ~ INT7 are independent and downward compatible. 1：8 external interrupts INT0 ~ INT7 share an interrupt signal EXT0, and determine which external interrupt is generated by querying the STA register interrupt flag.。	R/W	0
7:0	Enable	Enable signal, Enable[7:0] corresponds to the external interrupt request 7~0; Corresponding external pin is: P37~P30; 0: Close correspond external interrupt; 1: Enable correspond external interrupt;	R/W	0

11.2.2 INTC_MODE

INTC Mode Register Address 0x40044000+0x4

Bit	Name	Description	R/W Sign	Reset

				Value
31:16	---	Reserved	R	0
15:14	MODE7	External interrupt request 7 (INT7)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
13:12	MODE6	External interrupt request 6 (INT6)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
11:10	MODE5	External interrupt request 5 (INT5)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
9:8	MODE4	External interrupt request 4 (INT4)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
7:6	MODE3	External interrupt request 3 (INT3)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
5:4	MODE2	External interrupt request 2 (INT2)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
3:2	MODE1	External interrupt request 1 (INT1)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0
1:0	MODE0	External interrupt request 0(INT0)mode select 00:Rising edge 01:Falling edge 10:Double edge 11:Reserved	R/W	0

11.2.3 INTC_MASK

INTC Mask Register Address 0x40044000+0x8

Bit	Name	Description	R/W Sign	Reset Value
31:8	---	Reserved	R	0
7:0	MASK	MASK[7:0] correspond to the external interrupt request 7~0 0:Interrupt disabled 1:Interrupt enabled	R/W	0

11.2.4 INTC_STA

INTC Status Register Address 0x40044000+0xc

Bit	Name	Description	R/W Sign	Reset Value
31:08	---	Reserved	R	0
7:0	STA	STA[7:0] correspond to the external interrupt request 7~0 0:Interrupt event has not occurred 1:Interrupt event has occurred Note: Write 1 cleared	R/W	0

12 KBI

A built-in button interface controller is used to process the interrupt request input from the chip pin, and can automatically wake up the CPU by interrupt when the CPU sleeps.

12.1 Feature

Key interface controller has the following feature:

- Support for 8 keys, corresponding pin is KEY0~KEY7;
- Support for each key state check;
- Support each key input filter, filter time is 24ms;
- Support each key individually masked interrupts.

12.2 Register Description

Table 12- 1 KBI Register Base Address

Module Name	**Physical Address**	**Address Mapping**
KBI	0x40028000	0x40028000

Table 12- 2 KBI Register Address Offset

Register Name	**Address Offset**	**Description**
KBI_CTL	0x0	Control Register
KBI_SEL	0x4	SEL Register
KBI_DATA	0x8	Data Register
KBI_MASK	0xc	Mask Register

12.2.1 Control Register（0x0）

Table 12- 3 KBI Control Register KBI _CTL

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:8	---	Reserved	R	0
7:0	EN	Enable signal, EN [7:0] correspond to the KEY [7:0] Correspond to the external pins: P17/KEY7~P10/KEY0. 0: Close correspond KEY 1: Enable correspond KEY	R/W	0

12.2.2 SEL Register（0x4）

Table 12- 4 KBI SEL Register KBI _SEL

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:8	---	Reserved	R	0
7:0	SEL	SEL [7:0] correspond to the KEY [7:0] 0: Rising edge active 1: Falling edge active	R/W	0

12.2.3 Data Register（0x8）

Table 12- 5 KBI Data Register KBI _DATA

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:8	---	Reserved	R	0
7:0	DAT	DAT [7:0] correspond to the KEY [7:0]. Write 1 cleared 0: Key is not pressed 1: Key is pressed	R/W	0

12.2.4 Mask Register（0xC）

Table 12- 6 KBI Mask Register KBI _MASK

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:8	---	Reserved	R	0
7:0	MASK	MASK [7:0] correspond to the KEY [7:0] 0: Interrupt disabled 1: Interrupt enabled	R/W	0

12.3 KBI Operation Steps

1. Configure the KBI enable register KBI_EN in the System Control section, set the 8th bit and the corresponding KBI clock bit to 1, and turn on the clock.
2. Set the KBI control register KBI_CTL and enable the corresponding KBI.
3. KBI_SEL setting the KBI selection register, set the corresponding KBI as the rising edge or falling edge.
4. Enable the corresponding interrupt enable and enable the KBI interrupt NVIC_EnableIRQ (KBI_IRQn) on the configuration KBI_MASK.
5. Write a KBI interrupt service program:

```
void KBI_HANDLER(void)
                  {
                   if (KBI->DATA&0x01)
                   {

                   }
                   KBI->DATA = 0xff;
                  }
```

   All KBI interrupts are one ingress, and the interrupts generated by the KBI should be determined based on the KBI_DATA.
6. Complete

13 UART

SoC internally installed 6 UART interfaces for external asynchronous serial communication.

13.1 Overview

UART interface controller has the following feature:

- Six full-duplex UART interfaces;
- Internally installed baud rate generator, the baud rate is configured to support different;
- Data bits wide support 5/6/7/8bit;
- Stop bits can set 1 or 2bit;
- Optional 38kHz modulated IR,It also supports 32.768KHz;
- Supports automatic baud rate detection;
- Supports IR wake;
- Transmitter supports two-depth FIFO, receiver does not support FIFO;
- Supports Universal DMA functionality;
- Supports 300 baud rate@29MHz;
- Supports a maximum of 2048 baud rate@32.768KHz;
- All six UARTs support DMA;

13.2 Register Description

Table 13-1 UART Register Base Address

Module Name	**Physical Address**	**Address Mapping**
UART0	0x40000000	0x40000000
UART1	0x40004000	0x40004000
UART2	0x40008000	0x40008000
UART3	0x4000C000	0x4000C000
UART4	0x40018000	0x40018000
UART5	0x4001C000	0x4001C000

Table 13- 1 UART Register Address Offset

Register Name (X=0,1,2,3,4,5)	**Address Offset**	**Description**
UARTx_CTL	0x0	UART Control Register
UARTx_BAUD	0x4	UART Baud Rate Configuration Register
UARTx_STAT	0x8	UART Status Indication Register
UARTx_TXD	0xC	UART Transmit Data Register
UARTx_RXD	0x10	UART Receive Data Register
UARTx_FDIV	0x14	UART Baud Rate Fractional Configuration Register
UARTx _DMA_CTL	0x18	UART DMA Control Register
UARTx _DMA_TBADR	0x1c	UART DMA Transmit Starting Address Register
UARTx _DMA_RBAD	0x20	UART DMA Receive Starting Address Register
UARTx _DMA_TLEN	0x24	UART DMA Transmit Length

		Register
UARTx _DMA_RLEN	0x28	UART DMA Receive Length Register
UARTx _DMA_TADR	0x2c	UART Current Transmitting DMA Address Register
UARTx _DMA_RADR	0x30	UART Current Receiving DMA Address Register
UARTx _DMA_IE	0x34	UART DMA Interrupt Enable Register
UARTx _DMA_IF	0x38	UART DMA Interrupt Flag Register
UARTx _DMA_TO	0x3C	UART DMA Receive Timeout Configuration Register

Note: The control register and baud rate register cannot be modified during operation.

13.2.1 UART Control Register（0x00）

Table 13-2 UART Control Register UART_CTL

Bit	Name	Description	R/W Sign	Reset Value
31:15	---	Reserved	R	0
14	NEG	UART polarity selection: 0: Positive polarity, the default drive level is high, the polarity of the transmitted/received data remains unchanged. 1: Negative polarity, the default drive level is low, transmit/receive data polarity is reversed.	R/W	0
13	LMSB	LSB/MSB selection method 0: LSB transmission first 1: MSB transmission first Note: When PARS is selected as user-defined check, the check digit is regarded as the highest bit of data expansion. At this time, MSB is selected, and the first bit transmitted will be the check digit.	R/W	0
12	IRSEL	Infrared modulation polarity selection: 0: Positive polarity, that is, low-level modulation output, high level (default state) remains 1: negative polarity, that is, data is inverted, high-level modulation output, low-level hold Note: IRSEL only determines the level of the idle output (inactive level) and does not affect the level during the valid data period.	R/W	0
11	ILBE	Enable internal loop back 0: Internal loop back unable 1: Internal loop back enable, TXD and RXD shorted inside the module	R/W	0
10	IRE	Infrared modulation enable bit 0: Close infrared modulation output	R/W	0

		1: Open infrared modulation output, low-carrier modulation with 38k output data		
9:7	PARS	Select the parity bit 000: No parity 001: Odd parity 010: Even parity 011: Fixed zero parity 100: Fixed one parity Other: User-defined parity	R/W	0
6:5	DATLEN	Transmission bit data width 00:5-bit 01:6-bit 10:7-bit 11:8-bit	R/W	0
4	STOPS	Stop bit wide select 0:1-bit stop bit 1:2-bit stop bit	R/W	0
3	ERRIE	Error interrupt enable bit, the corresponding flag bit is the status indicator register bit5~bit2. 0: Disable interrupt 1: Enable interrupt	R/W	0
2	RXIE	Receive data interrupt enable bit, the corresponding flag bit is the status indicator register bit1. 0: Disable interrupt 1: Enable interrupt	R/W	0
1	TXIE	Transmit data interrupt enable bit, the corresponding flag bit is status indicator registers bit0. 0: Disable interrupt 1: Enable interrupt	R/W	0
0	EN	Module Enable 0: Disable 1: Enable	R/W	0

Note: The infrared modulation function is also supported when the system clock is 32.768 KHz.

13.2.2 UART Baud Rate Configuration Register（0x4）

Table 13- 3 UART Baud Rate Configuration Register UART_ BAUD

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:13	---	Reserved	R	0
12:0	CLKDIV	UARTx clock divide The formula of baud rate is: System Clock/[16*(CLKDIV+1)]	R/W	0

Note: When the system clock is 32.768 KHz, the maximum baud rate is 32768/16=2048bps according to the baud rate formula.

13.2.3 UART Status Indication Register（0x8）

Table 13- 4 UART Status Indication Register UART_STA

Bit	Name	Description	R/W Sign	Reset Value
31:8	---	Reserved	R	0
9	tx_fifo_full	Transmit FIFO is full: 0: Not full 1: Full	R	0
8	tx_fifo_empty	Transmit FIFO is empty: 0: Not empty 1: Empty	R	1
7	TB	Send state flag 0: Did not send 1: Sending data	R	0
6	RB	Receive status flag 0: Did not receive 1: Receiving data	R	0
5	DE	Data errors, write 1 cleared After the UART transmit FIFO is full, continue to write to the UART transmit register or write new transmit data during transmission. This bit will be set. 0: No error 1: Error	R/W	0
4	FE	Frame error, write one cleared The data received by the UART does not match the frame format flag. If the received stop bit is 0 instead of 1, the bit will be set. 0: No error 1: Error	R/W	0
3	OE	Overflow error, write one cleared The UART receive data register is not read in time and causes a receive overflow. This bit will be set. 0: No error 1: Error	R/W	0
2	PE	Parity error, write one cleared The data checksum error received by the UART, this bit will be set 0: No error 1: Error	R/W	0
1	TX	Send flag, write one cleared 0: Data not yet been sent or no data to be transmitted 1: Date has sent	R/W	0
0	RX	Receive flag, write one cleared 0: No receive data 1: Data has received	R/W	0

13.2.4 UART Transmit Data Register（0xC）

Table 13- 5 UART Transmit Data Register UART_ TXD

Bit	Name	Description	R/W Sign	Reset Value
31:9	---	Reserved	R	0
8	UP	User defined parity bit	R/W	0
7:0	TXDATA	Transmit data register	R/W	0

13.2.5 UART Receive Data Register（0x10）

Table 13- 6 UART Receive Data Register UART_ RXD

Bit	Name	Description	R/W Sign	Reset Value
31:9	---	Reserved	R	0
8	UP	Parity bit	R	0
7:0	RXDATA	Receive data register	R	0

13.2.6 UART Baud Rate Fractional Configuration Register（0x14）

Table 13- 8 UART Baud Rate Fractional Configuration Register UART x _FDIV

Bit	Name	Description	R/W Sign	Reset Value
31:14	---	Reserved	R	0
13:0	FDIV	Fractional division factor. The calculation formula is: $F = \left[\left(\frac{fi}{16 \times fo} - \left[\frac{fi}{16 \times fo}\right]\right) \times 2^{14} + 0.5\right]$ Where fi is the input clock (cpu current running clock), fo is the output clock, “ [ ] ” is the downward integer operator. For example, if the input clock is 1.8432MHz and the output clock is 9837Hz, then: $F = \left[\left(\frac{1843200}{16 \times 9837} - \left[\frac{1843200}{16 \times 9837}\right]\right) \times 2^{14} + 0.5\right]$ Find F=11647.	R/W	0

13.2.7 UART DMA Control Register（0x18）

Table 13- 9 UART DMA Control Register UART_DMA_CTL

Bit	Name	Description	R/W Sign	Reset Value
31:5	Reserved	Reserved	R	0
4	DMA_PARS	DMA Parity bit	R/W	0
3	RX_CYC_MODE	Receive circular mode enable	R/W	0
2	TX_CYC_MODE	Transmit circular mode enable	R/W	0
1	RX_DMA_EN	Receive DMA enable	R/W	0
0	TX_DMA_EN	Transmit DMA enable	R/W	0

13.2.8 UART DMA Transmit Starting Address Register（0x1C）

Table 13- 10 UART DMA Transmit Starting Address Register UART_DMA_TBADR

Bit	Name	Description	R/W Sign	Reset Value
31:17	Reserved	Reserved	R	0
16:0	DMA_TBADR	DMA transmit starting address（Byte address） The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

13.2.9 UART DMA Receive Starting Address Register（0x20）

Table 13- 11 UART DMA Receive Starting Address Register UART_DMA_RBADR

Bit	Name	Description	R/W Sign	Reset Value
31:17	Reserved	Reserved	R	0
16:0	DMA_RBADR	DMA receive starting address（Byte address） The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

13.2.10 UART DMA Transmit Length Register（0x24）

Table 13- 12 UART DMA Transmit Length Register UART_DMA_TLEN

Bit	Name	Description	R/W Sign	Reset Value
31:17	Reserved	Reserved	R	0
16:0	DMA_TLEN	DMA transmit length（Byte address）=（n）Byte The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

13.2.11 UART DMA Receive Length Register（0x28）

Table 13- 13 UART DMA Receive Length Register UART_DMA_RLEN

Bit	Name	Description	R/W Sign	Reset Value
31:17	Reserved	Reserved	R	0
16:0	DMA_RLEN	DMA receive length（Byte address）=（n）Byte The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

13.2.12 UART DMA Current Transmitting Address Register（0x2C）

Table 13- 14 UART DMA Current Transmitting Address Register UART_DMA_TADR

Bit	Name	Description	R/W Sign	Reset Value
31:17	Reserved	Reserved	R	0
16:0	DMA_TADR	Current transmitting DMA address（Byte address）	RO	0

13.2.13 UART DMA Current Receiving Address Register（0x30）

Table 13- 15 UART DMA Current Receiving Address Register UART_DMA_RADR

Bit	Name	Description	R/W Sign	Reset Value

31:17	Reserved	Reserved	R	0
16:0	DMA_RADR	Current receiving DMA address（Byte address）	RO	0

13.2.14 UART DMA Interrupt Enable Register（0x34）

Table 13-16 UART DMA Interrupt Enable Register UART_DMA_IE

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:6	Reserved	Reserved	R	0
5	RX_ERR_IE	Received data overlay interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
4	TX_ERR_IE	Transmit data error interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
3	RX_FIE	DMA receive all full interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
2	RX_HIE	DMA receive half full interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
1	TX_FIE	DMA transmit all empty interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
0	TX_HIE	DMA transmit half empty interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0

13.2.15 UART DMA Interrupt Flag Register（0x38）

Table 13-17 UART DMA Interrupt Flag Register UART_DMA_IF

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:7	Reserved	Reserved	R	0
6	RX_ DONE	UART module proprietary, If a receive timeout occurs， This bit will be set. write one cleared	R/WC	0
5	RX_ERR	Received data overlay flag, write one cleared	R/WC	0
4	--	Reserved	R	0
3	RX_FDONE	DMA receive all full interrupt flag, write one cleared	R/WC	0
2	RX_HDONE	DMA receive half full interrupt flag, write one cleared	R/WC	0
1	TX_FDONE	DMA transmit all empty interrupt flag, write one cleared	R/WC	0
0	TX_HDONE	DMA transmit half empty interrupt flag, write one cleared	R/WC	0

13.2.16 UART DMA Receive Timeout Configuration Register（0x3C）

Table 13-18 UART DMA Receive Timeout Configuration Register UART_DMA_TO

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**

31:4	Reserved	Reserved	R	0
3:0	TIMEOUT_CNT	UART Receive Timeout Configuration, When UART does not receive the start bit for a certain period of time, DMA automatically ends. n=n UART data bit width time. Invalid configuration for 0	RW	0

13.3 UART Data Receiving and Sending Procedure

1、 Enable the corresponding UART clock in the 0 register MOD0_EN in the system control chapter module enable module;

2、 Configure the baud rate configuration register UART x_ BAUD. For example, when the system clock is 3.6864MHz and the communication baud rate is 9600, the baud rate configuration register can be set to: UART0->BAUD = 3686400 / (9600*16)-1;

3、 Configure the communication control register UART x _CTL to select the data bit, stop bit, check mode and interrupt enable;

4、 Write 0x3f to clear the UART status indication register (UART x_ STA) status;

5、 Configure the UART interrupt enable, open the UART interrupt NVIC_ Enable IRQ (UART x_ IRQ n);

6、 Write an interrupt service routine, such as the UART0 interrupt service routine:

```
void UART0_HANDLER(void)
 {
        u32    status;
        u8     temp;
        status = UART0->STA;

        /* UART error irq */
        if((UART0->CTRL & 0x8) && (status & 0x3c))
        {
               /* Start adding user code. Do not edit comment generated here */
        }
       /* receive data complete irq */
        if((UART0->CTRL & 0x4) && (status & 0x1))
         {
        /* Start adding user code. Do not edit comment generated here */
        }
        /* transmit data complete irq */
        if((UART0->CTRL & 0x2) && (status & 0x2))
        {
               /* Start adding user code. Do not edit comment generated here */
        }
 }
```

The UART receive, transmit, and error interrupts are the same interrupt entry. The interrupt Enable bit and status flag opened by the control register should be used to determine which interrupt is active at this time;

7、 Process the received or sent data and complete it;

8、 Note: The UART port is full-duplex mode, which can transmit and receive at the same time. When RS485 half-duplex communication mode is used, when the RS485 chip is transmitting, there will be interference signal at the receiving end. In this case, it is recommended to turn off the receiving interrupt of the MCU when sending, and turn off the sending interrupt when receiving to eliminate interference.

14 ISO7816

SoC internal installed two ISO7816 channels, support for external two 7816 protocol interface device.

14.1 Overview

ISO7816 interface controller has the following features:

- Support standard ISO7816 protocol, working in master mode;
- Support card clock output, frequency can be set between 1 ~ 5MHz;
- Support 7816 various frequency division ratio setting;
- Support MSB first output logic low and logic high output LSB first data encoding;
- Support 1, 2 ETU width set the width of the error signal;
- Support 0~254 ETU width EGT configuration;
- Supports sending data transmission error retransmission mechanism, the number of retransmissions can be set between 0 to 7;
- 7816 card stack supports two interfaces (Esam and card): esam modules receive and transmit ports with a pin;
- Support card interface for receiving and sending separation;

14.2 Register Description

Table 14-1 ISO7816 Register Base Address

Module Name	Physical Address	Address Mapping
ISO7816	0x40038000	0x40038000

Table 14-2 ISO7816 Register Address Offset

Register Name	Address Offset	Description
ISO7816_CTL0	0x0	Control Register 0
ISO7816_CTL1	0x4	Control Register 1
ISO7816_CLK	0x8	Clock Configuration Register
ISO7816_BDDIV0	0xc	Baud Rate Configuration Register 0
ISO7816_BDDIV1	0x10	Baud Rate Configuration Register 1
ISO7816_STAT0	0x14	Status Indication Register 0
ISO7816_STAT1	0x18	Status Indication Register 1
ISO7816_DAT0	0x1c	Data Transmit Register 0
ISO7816_DAT1	0x20	Data Transmit Register 1

14.2.1 ISO 7816 Control Register 0（0x0）

Table 14-3 ISO7816 Control Register 0 ISO7816_CTL0

Bit	Name	Description	R/W Sign	Reset Value
31:28	---	Reserved	R	0
27	RX_GT0	Receive data GT select bit, when sending fixed 2etu 1: Receive data GT is 1etu 0: Receive data GT is 2etu	R/W	0
26	---	Reserved	R/W	0

25	---	Reserved	R/W	0
24:17	EGT0	EGT width selection value (0 to 255), that the extra guard time N, the default value N = 0. In the range of 0 to 254, N is used to calculate the delay between two consecutive data of a start edge: 12 etu + (Q × (N / f)). Formula, Q should take one of two values below: ——When T=15 does not exist in answer to reset, take F/D; ——When T=15 exit in answer to reset, take Fi/Di; N = 255 means that during transport protocol, a minimum of two consecutive characters start edge delay between the two directions of transmission are the same. The minimum delay value is: ——T=0,12etu ——T=1,11etu	R/W	0
16: 14	REP_CNT0	Automatic retransmission number control when data parity error occurs 000：0 time 001：1 time 010：2 times 011：3 times 100：4 times 101：5 times 110：6 times 111：7 times	R/W	011
13	RXPAR_ESEL0	Receive data parity error handling mode selection 1: Parity error, according to the T = 0 protocol post back error signal. Set RX_PAR_ERR flag, interrupt. 0: Parity error, do not send error signal, set RX_PAR_ERR flag, direct interrupt.	R/W	1
12:11	ERRWTH0	Error signal width select bit, it applies only to receive, and RXPAR_ESEL0 = 1 00:2 etu 01:1 etu 10:1.5 etu 11:2etu	R/W	01
10:8	PARSEL0	Parity bits select 000: No parity 001: Odd parity 010: Even parity 011: Fixed zero parity 100: Fixed one parity Other: Reserved	R/W	010
7	BGT_EN0	Data received BGT control bits transmitted 0: Close BGT function, data between transmission and reception don't insert BGT 1: Open BGT function, data between transmission and	R/W	0

		reception insert BGT (22etu)		
6	ERR_IRQ_EN0	Transmission error interrupt enable bit, data collision when transmitting data, the data is received and the received data frame format overrun error 0: Prohibit transmission error interrupt is generated 1: Enable transmission error interrupt is generated	R/W	0
5	RX_IRQ_EN0	Data receive interrupt enable bit, enables data is shifted from the shift register to the receive buffer register to generate an interrupt 0: Prohibit data reception interrupt is generated 1: Enable data reception interrupt is generated	R/W	0
4	TX_IRQ_EN0	Data transmit interrupt enable bit to enable the completion of the data from the transmit shift register to generate an interrupt 0: Prohibit sending data to generate an interrupt 1: Enable data transmission interrupt is generated	R/W	0
3	RX_EN0	Receive data enable 0: Prohibit data reception 1: Enable data reception	R/W	0
2	TX_EN0	Enable sending data 0: Prohibit data transmission 1: Enable data transmission	R/W	0
1	DIRSEL0	Data coding mode selection bit 0: LSB first pass being logical data encoding 1: MSB first pass negative logic data encoding (data negated)	R/W	0
0	EN0	ISO7816 The controller enable bit 0: Controller close 1: Controller open	R/W	0

14.2.2 ISO7816 Control Register 1（0x04）

Table 14-4 ISO7816 Control Register 1 ISO7816_CTL1

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31	CARD1_CHECK_EN	Card out Detect Flag, it is only active after the detection function of OLD is enable 1: Enable card out detection interrupt function 0: Disable card out detection interrupt function	R/W	0
30	OLD1_IRQ_EN	OLD detection interrupt function flag, it is only active after detection function of OLD is enable 1: Enable OLD detection interrupt function 0: Disable OLD detection interrupt function	R/W	0
29	OLD1_EN	OLD detection function flag 1: Enable OLD detection function 0: Disable OLD detection function	R/W	0

28	RX1_GT0	GT of received data choice bit, it is always 2etu when data is transmitted 1:GT of received data is 1etu 0:GT of received data is 2etu	R/W	0
27	---	Reserved	R/W	0
26	---	Reserved	R/W	0
25	IO1_EN	bidirectional data enable signal 1:78161_IO port is a bidirectional signal 0:78161_IO port is a one-way signal , output only, the data are input at 78161_I port	R/W	1
24:17	EGT1	EGT width selection (0~255), extra protection time N Default N=0. In the range of 0 to 254, N: before it was ready to receive the next character, the card need the delay(it sent by card or interface device) which start with onset of the first character: 12 etu +(Q×(N/f)) In the formula, Q should be one of two values: F/D, it is used to compute the value of etu. When T = 15 don't exist in the reset reply, Fi/Di, When T = 15 exist in the reset reply. N=255when the transmission protocol is effective, Minimize Delay between onset of two continuation character in either direction remains the same. Minimize Delay: When T=0,12etu When T=1,11etu	R/W	0
16: 14	REP_CNT1	Automatic retransmission number control when data parity error occurs 000:0time 001:1time 010:2times 011:3times 100:4times 101:5times 110:6times 111:7times	R/W	011
13	RXPAR_ESEL1	Receive data parity error handling mode selection 1: Parity check is wrong, According to T = 0 protocol, error signal will be post back, RX_PAR_ERR flag bit will be set and generate the interrupt. 0: Parity check is wrong, do not send error signal, RX_PAR_ERR flag bit will be set, generate the interrupt.	R/W	1
12:11	ERRWTH1	Width selection bit of the error signal	R/W	01

		00:2etu 01:1etu 10:1.5etu 11:2etu		
10:8	PARSEL1	Parity select bit 000: No parity 001: Odd 010: Even 011: Fixed to zero check 100: Fixed to one check Other: Reserved	R/W	010
7	BGT_EN1	BGT control bit between data transmission and reception 0: Disable BGT function, do not insert the BGT between data reception and transmission 1: Enable BGT function, insert the BGT between data reception to transmission	R/W	0
6	ERR_IRQ_EN1	Transmit error interrupt flag, Data conflicts when data is transmitting, Data overflow when data is receiving and the received data frame is error 0: Disable the interrupt when data transmission is error 1: Enable the interrupt when data transmission is error	R/W	0
5	RX_IRQ_EN1	Receive interrupt flag, data transfer from shift registers to receive buffer registers 0: Disable data reception interrupt to generate 1: Enable data reception interrupt to generate	R/W	0
4	TX_IRQ_EN1	Transmit interrupt flag, set by hardware after completion of a serial transfer from shift registers 0: Disable the interrupt between data transmission 1: Enable the interrupt between data transmission	R/W	0
3	RX_EN1	Receive data enable 0: Disable receive data 1: Enable receive data	R/W	0
2	TX_EN1	Transmit data enable 0: Disable transmit data 1: Enable transmit data	R/W	0
1	DIRSEL1	Data coding mode selection bit 0: LSB first pass that is positive logic data coding method 1: MSB first pass negative logic data coding method (negate values)	R/W	0
0	EN1	ISO7816 Controller flag	R/W	0

		0: Disable the controller 1: Enable the controller		

14.2.3 ISO7816 Clock Configuration Register（0x08）

Table 14- 5 ISO7816 Clock Configuration Register ISO7816_CLK

Bit	Name	Description	R/W Sign	Reset Value
31:4	---	Reserved	R	0
3	CLKO_EN	Card clock output enable bit 0: Disable card clock output 1: Enable card clock output	R/W	0
2:0	CLKDIV	ISO7816 Division factor of clock output (CLK_O) ISO7816 source clock of module gain from fsyspll of system clock 000: No frequency division; 001:2 frequency division; 010:4 frequency division; 011:8 frequency division; 100:16 frequency division 101:32 not support; 110: not support; 111:128 not support;	R/W	0

14.2.4 ISO7816 Baud Rate Configuration Register 0（0x0c）

Table 14- 6 ISO7816 Baud Rate Configuration Register 0 ISO7816_BDDIV0

Bit	Name	Description	R/W Sign	Reset Value
31:22	---	Reserved	R	0
21	FDS0_EN	Enable soft configuration coefficient of F/D 1: Baud rate coefficient will be determined by FDS0 which is written by software 0: Baud rate coefficient will be determined by FD0	R/W	0
20:8	FDS0	Baud rate coefficient which are configured by software, this bit can be written only when FDS0_EN is 1, In other cases, it is13’d372.	R/W	13’d372
7:0	FD0	8bit FI and DI is transmitted from answer to reset	R/W	8’h01

14.2.5 ISO7816 Baud Rate Configuration Register 1（0x10）

Table 14- 7 ISO7816 Baud Rate Configuration Register 1 ISO7816_BDDIV1

Bit	Name	Description	R/W Sign	Reset Value
31:22	---	Reserved	R	0
21	FDS0_EN	Enable soft configuration coefficient of F/D 1: Baud rate coefficient will be determined by FDS0 which is written by software 0: Baud rate coefficient will be determined by FD0	R/W	0
20:8	FDS0	Baud rate coefficient which are configured by software, this bit can be written only when FDS0_EN is 1, In other cases, it is13’d372.	R/W	13’d372
7:0	FD0	8bit FI and DI is transmitted from answer to reset	R/W	8’h01

14.2.6 ISO7816 State 0 Register（0x14）

Table 14- 8 ISO7816 State 0 Register ISO7816_STAT0

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:12	---	Reserved	R	0
11	FRAME_ERR0	Receive data frame format error interrupt flag this bit will be reset by writing '1' 1: Send the error frame format error of receiving data , it will generate the interrupt when transmission error interrupt is enable 0: Unsent the error frame format error of receiving data	R/W	0
10	BDDIV_R0	Baud rate matching instruction, the matching instructions between FI and DI, FD default is 8’h01, clock matching, it will be set as 1 when FD unmatched. 1: matched 0: unmatched	R	1
9	TX_FLAG0	Transmit data buffer empty flag. The automatic set after power-on reset, and it shows Buffer is empty and can be written. The flag will be automatically cleared after MCU is written. After shift data from transmit buffer registers to shift registers, this bit will be set as 1 1: Data transmit buffer is empty 0: The data transmit buffer has data to be transmitted	R	1
8	RX_FLAG0	Data buffer full flag,7816 Interface controller receives every bit data, hardware automatic clear settings, it shows Interface controller receives 1 bit data, Reading data receive buffer register will be reset. 1: 1byte data is received, data buffer is full 0: There is no data received, data buffer is empty	R	0
7	RXBUSY0	Reception data busy flag. Set by hardware, reset by software Hardware automatic clear settings 0: Receive data idle 1: RSR are receiving data, it will be set as 1 after start bit is received, it will automatic clear zero after stop bit is received	R	0
6	TXBUSY0	Transmit data busy flag. Set by hardware, reset by software Hardware automatic clear settings 0: Transmit data idle 1: TXSHF are sending data, it will be set as 1 after start bit is send, it will automatic clear zero after stop bit is send	R	0
5	TXPAR_ERRIF0	Send data parity error flag, there is still parity error after retry, then this bit is turned on. this bit will be reset by writing '1' 1: Parity error occurred when data is transmitted. 0: Parity error didn’t occur when data is transmitted.	R/W	0

4	RXPAR_ERRIF0	Receive data parity error flag bit, there is still parity error after retry, then this bit is turned on. this bit will be reset by writing '1' 1: Parity error occurred when receiving data is received 0: Parity error didn't occur when data is received	R/W	0
3	COL_IF0	Send data conflict error interrupt flag. Set by hardware, reset by software, this bit will be reset by writing '1' 0: No interrupt 1: Interrupt occurs	R/W	0
2	OVL_IF0	Receive data overflow flags. Set by hardware, reset by software this bit will be reset by writing '1' 0: No overflow 1: Interrupt occurs, the receive buffer register didn't be read, and received the new data. Overflow flag bit is enable	R/W	0
1	RXIF0	Transmit data interrupt flag bit. After shift data from shift registers to transmit buffer registers, it will be set as 1. Set by hardware, reset by software this bit will be reset by writing '1' 0: No interrupt 1: Interrupt occurs	R/W	0
0	TXIF0	Transmit data interrupt flag. After move data from send buffer registers to shift registers, it will be set as 1, set by hardware, reset by software, this bit will be reset by writing '1' 0: No interrupts 1: Interrupt occurs	R/W	0

14.2.7 ISO7816 State 1 Register（0x18）

Table 14-9 ISO7816 State 0 Register ISO7816_STAT1

Bit	Name	Description	R/W Sign	Reset Value
31:14	---	Reserved	R	0
13	CARD_OUT_FLAG	This bit is Effective, after CARD_CHECK_EN enabled. this bit will be reset by writing '1'. 1:Detected card was uprooted (the width of high level pulse of input port is more than 40mS) 0: Pulling out the card is not detected (the width of high level pulse of input port is not more than 40mS)	R	0
12	OLD_FLAG	After OLD_EN enabled, this bit is Effective, to match the received RA9105 signal of OLD interrupt flag bit, this bit will be reset by writing '1'. 1: OLD signal has been received. 0: OLD signal hasn't been received.	R/W	0
11	FRAME_ERR0	Receive data overflow flag, this bit will be reset by writing '1' 1: Send received data frame format error, Interrupt occurs	R/W	0

		when the transmit error interrupt is enabled 0: Unsent received data frame format error		
10	BDDIV_R1	Baud rate matching direction, FI and DI matching direction; FD defaults to 8'h01, clock matching, it will be set as 1 when FD unmatched. 1: matched 0: unmatched	R	1
9	TX_FLAG1	Send Buffer empty flag. It automatic setting after power-on reset, And it shows Buffer is empty, and it can be written. The flag will be automatically clean after MCU is written, After move data from send buffer registers to shift registers, this bit will be set as 1 1: Buffer is empty 0: There is data which are ready to send in the buffer	R	1
8	RX_FLAG1	The data reception complete flag, 7816 Interface receives every bit data, receiver channel generate the interrupt. Set by hardware, reading data receiving buffer register clean. Interface controller receives every bit data, Hardware automatic Clear Settings, it shows Interface controller receives 1 bit data, Read data receiving buffer register will be reset. 1:1byte data is received, receive data buffer is full 0: There is no data received, receive data buffer is empty	R	0
7	RXBUSY1	Receive data busy flag. Hardware set, reset by software Hardware automatic clear settings 0: Data reception go idle 1: RSR are receiving data, it will be set as 1 after start bit is received, it will automatic clear zero after stop bit is received	R	0
6	TXBUSY1	Transmit data busy flag. Hardware set, reset by software Hardware automatic Clear Settings 0: Data transmission go idle 1: TXSHF are sending data, it will be set as 1 after start bit is transmitted, it will automatic clear zero after stop bit is transmitted	R	0
5	TXPAR_ERRIF1	Transmit data parity error flag bit. Hardware set, reset by software this bit will be reset by writing '1' 1: Parity error occurred data is transmitted 0: Parity error didn't occur data is transmitted	R	0
4	RXPAR_ERRIF1	Receive data parity error flag bit. Hardware set, reset by software This bit will be reset by writing '1'. 1: Parity error occurred data is received 0: Parity error didn't occur data received	R/W	0

3	COL_IF1	Send data conflict error interrupt flag bit. set by hardware, reset by software this bit will be reset by writing '1' 0: No interrupts 1: Interrupt occurs	R/W	0
2	OVL_IF1	Receive data overflow flag bit. Set by hardware, reset by software this bit will be reset by writing '1' 0: There is no overflow 1: Interrupt occurs, the receive buffer register don't be read, and received the new data. Overflow flag bit is Enable	R/W	0
1	RXIF1	Receive data interrupts flags bit. After move data from shift registers to send buffer registers, it will be set as 1, set by hardware, reset by software this bit will be reset by writing '1' 0: No interrupts 1: Interrupt occurs	R/W	0
0	TXIF1	Sending data interrupts flag bit. After move data from send buffer registers to shift registers, it will be set as 1.Set by hardware, reset by software this bit will be reset by writing '1' 0: No interrupts 1: Interrupt occurs	R/W	0

14.2.8 ISO7816 Data Transmit Register 0（0x1C）

Table 14- 10 ISO7816 Data Transmit Register 0 ISO7816_DAT0

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
Reserved	--	--	R	0
8	DATA0[8]	It is PARITY bit of data frames,when parsel is in User-defined mode	R/W	0
7:0	DAT0	Data Register0	R/W	0

14.2.9 ISO7816 Data Transmit Register 1（0x20）

Table 14- 11 ISO7816 Data Transmit Register 1 ISO7816_DAT1

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
Reserved	--	--	R	0
8	DATA1[8]	It is PARITY bit of data frames,when parsel is in User-defined mode	R/W	0
7:0	DAT1	Data Register1	R/W	0

14.3 7816 And ESAM Communication Steps

1、 The 7816 communicates with the ESAM. There is no need to consider the isolation problem. The data IO can share one line. It is recommended to use the 7816 module 0.

2、 Configure the system control chapter module to enable the 0 register MOD0_EN, set the 13th position to 1, and turn on the 7816clock.

3、 The 7816control register is configured as ISO7816_CTL0. Using the national network ESAM, the register can be configured as 0x00000201.

4、 Clear the ISO7816 status register ISO7816_STAT0.

5、 Turn on the 7816bus clock. For example, when the system clock is 3.6864MHZ, ISO7816 -> CLK=0x09; at this time, the 7816module clock is 1.8432MHZ.

6、 Data can be read and written to the 7816 bus by interrupt mode or query status mode.

14.4 7816 And Card Communication Steps

1、 For the card table, the card needs to be isolated from the main power, and the 1-5MHZ clock is required to work normally. Most of the existing SOC uses a high-speed optical scheme to isolate the main power from the card. This solution has high cost and use high-speed optocouplers are used to isolate high-frequency clocks, which are less reliable at high and low temperatures. We provide a dedicated chip RN8501 for connection to the card, which uses two common optocouplers for data communication with the MCU.

2、 Configure the system control chapter module to enable the 0 register MOD0_EN, set the 13th position to 1, and turn on the 7816clock.

3、 The 7816control register is configured as ISO7816_CTL1. Because it is isolated from the card, it is separated from the transmission. ISO7816->CTRL1 can be configured as 0x60000201.

4、 Clear the ISO7816 status register ISO7816_STAT0.

5、 Turn on the 7816bus clock. For example, when the system clock is 3.6864MHZ, ISO7816 -> CLK=0x09; at this time, the 7816module clock is 1.8432MHZ.

6、 Can read and write data to the 7816 bus through interrupt mode or query status mode.

After using RN8501, the insertion and extraction detection of the card and the reset information reading of the card will be different from the separation scheme:

1、 Card insertion detection: Connect the detection pin of the card holder to the CHK of the RN8501. When the card is inserted into the card holder, the CHK pin is low level, and the RN8501 sends a low level signal of about 9MS through the 7816 port connected to the RN821x. When the OLD detection enable of ISO7816_CTL1 is turned on, an interrupt is generated and it is considered that there is a card inserted outside.

2、 Card pull-out detection: After the card of ISO7816_CTL1 is configured to pull out the detection enable position, the pullout of the card can be detected. Note: After the card operation is completed, the card pull-out detection interrupt can be opened.

3、The reset information of the card is read after the MCU and RN8501 communication handshake is completed.

4、More specific steps can be found in the RN8501 data sheet.

15 IIC Interface

Built-in I²C interface controller

15.1 Overview

The controller of I²C Interface has the following features:

- Master and slave mode were supported;
- 7-bit address were supported;
- Many frequency division ratio settings were supported
- 100kbps and fast mode(400kbps) were supported;

15.2 Register Descriptions

Table 15- 1 I²C Register Base Address

Module Name	Physical Address	Mapping Address
I²C	0x40024000	0x40024000

Table 15- 2 I²C Register Offset Address

Register	Address Offset	Descriptions
I²C_CTL	0x0	Control Register
I²C_CLK	0x4	Clock Configuration Register
I²C_STAT	0x8	Status Indication Register
I²C_ADDR	0xC	Slave Device Address Register
I²C_DATA	0x10	Transmit And Receive Data Register

15.2.1 Control Register（0x0）

Table 15- 3 Control Register I²C_CTL

Bit	Name	Description	R/W Sign	Reset Value
31:6	---	Reserved	R	0
5	MODE	MASTER/SLAVE 1: MASTER 0: SLAVE	R/W	0
4	ACK	ACK sending enable 1: After received the signal of the ninth SCL, generate ACK 0: After received the signal of the ninth SCL, don't generate ACK	R/W	0
3	IRQE	I²C interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
2:1	BUSCON	A bus control bit, start command is effective when bus is idle or host is posted. Start command is effective when host is posted	R/W	0

		When timing of the stop or start is detected, command bit will be clear 00:no action 01: Producing the time for START 10: Producing the time for STOP 11: Reserved		
0	EN	Module Enable 1: Enable I^2C 0: Disable I^2C	R/W	0

15.2.2 Clock Configuration Register（0x4）

Table 15- 4 Clock Configuration Register I^2C_CLK

<table>
<tr><th>Bit</th><th>Name</th><th>Description</th><th>R/W Sign</th><th>Reset Value</th></tr>
<tr><td>31:3</td><td>---</td><td>Reserved</td><td>R</td><td>0</td></tr>
<tr><td>2:0</td><td>CLKDIV</td><td>I^2C clock separate frequency parameters selection bit:
I^2C calculating formula for communication clock rate: SCL=APBCLK/m, there into m produced by CLKDIV, shown in the table below. It generated High-speed mode or Normal-mode communication clock according to different System frequency and Separate frequency parameters.
If configuration option is not in the table below, it defaults to divide-by-ten.
<table>
<tr><td>System frequency</td><td colspan="2">Separate frequency parameters /CLKDIV(m)</td></tr>
<tr><td></td><td>High-speed mode</td><td>Normal-mode</td></tr>
<tr><td>1.8432MHz</td><td>Not support</td><td>010（20）</td></tr>
<tr><td>3.6864MHz</td><td>Not support</td><td>011（38）</td></tr>
<tr><td>7.3728MHz</td><td>010（20）</td><td>100（76）</td></tr>
<tr><td>14.7456MHz</td><td>011（38）</td><td>101（154）</td></tr>
<tr><td>29.4912MHz</td><td>100（76）</td><td>110（306）</td></tr>
</table>
</td><td>R/W</td><td>001</td></tr>
</table>

When the main frequency clock frequency is 1.8432 / 3.6864 / 7.3728 / 14.7456 / 29.4912 MHz, the frequency division coefficient and interface clock error are as follows:

System frequency Mhz	Fast mode (400K)	Division factor	Interface clock （KHz）	Error	Normal mode	Division factor	Interface clock （KHz）	Error
1.8432	Not support	/	/	/	010（20）	20	92.160	-8%
3.6864	Not support	/	/	/	011（38）	38	97.010	-3%
7.3728	010（20）	20	368.640	-8%	100（76）	76	97.010	-3%
14.7456	011（38）	38	388.042	-3%	101（154）	154	95.751	-4%

29.4912	100（76）	76	388.042	-3%	110（306）	306	96.376	-4%

I^2C Clock frequency detailed configuration table：

CLKD IV	Division factor **m**	**System frequency APB_CLK（MHz）/I2C clock(KHz)**				
		1.8432	**3.6864**	**7.3728**	**14.7456**	**29.4912**
000	6	307.200	614.400	1228.800	2457.600	4915.200
001	10	184.320	368.640	737.280	1474.560	2949.120
010	20	92.160	184.320	368.640	737.280	1474.560
011	38	48.505	97.010	194.021	388.042	776.084
100	76	24.252	48.505	97.010	194.021	388.042
101	154	11.968	23.937	47.875	95.751	191.501
110	306	6.023	12.047	24.094	48.188	96.376
111	10	184.320	368.640	737.280	1474.560	2949.120

SCL=APBCLK/m

15.2.3 Status Indication Register（0x8）

Table 15- 5 Status Indication Register I^2C_STAT

Bit	**Name**	**Description**	**R/W Sign**	**Reset Value**
31:9	---	Reserved	R	0
8	DIR	Direction of Reading or writing flag 1: Write. 0: Read.	R	0
7	MATCH	Address matching, when timing of the stop or start is detected, command bit will be clear 0: Address mismatch 1: Address matching	R	0
6	BUSY	Traffic Status flag 0: IIC is idle 1: IIC is busy	R	0
5	COL	Sending conflict interrupt. this bit will be reset by writing '1' Sending data register is not empty or When receiving data, user write new data to data register. Trigger sending interrupt flag 0: No trigger send conflict interrupt 1: Trigger send conflict interrupt	R/W	0
4	OVERF	Receive overflow interrupt flag. this bit will be reset by writing '1' When receiving data, a new data is received before previous data don't be took away, Trigger overflow interrupt flag 0: No trigger overflow interrupt 1: Trigger overflow interrupt	R/W	0

3	TXEMPT	Sending data register is empty error flag. this bit will be reset by writing '1' In slave mode, the host asked slave to send data, when send buffer is empty, trigger send data is empty error interrupt flag 0: No trigger send data is empty error interrupt 1: Trigger send data is empty error interrupt	R/W	0
2	TRANC	Transfer complete interrupt flag. this bit will be reset by writing '1' When sending data, send buffer is empty, or when receiving data, receive buffer is full. Trigger transfer complete interrupt flag 0: Transfer has not been completed 1: Transfer has been completed	R/W	0
1	RX_NACK	Received NACK interrupt flag. this bit will be reset by writing '1' 1: Received NACK 0: No NACK was received	R/W	0
0	STPD	STOP time sequence inspection interrupt flag. this bit will be reset by writing '1' When timing of the start is detected or module is power off, this bit will be clear 0: No STOP timing detected 1: Detect STOP timing	R/W	0

15.2.4 Slave Device Address Register（0xC）

Table 15- 6 Slave Device Address Register I^{2}C_ADDR

Bit	Name	Description	R/W Sign	Reset Value
31:8	---	Reserved	R	0
7:1	SADR	Device address, it can't be written during the transport address. In host mode, it is slave device address; In slave mode, this address is used to compare with address which host send.	R/W	0
0	RW	Direction of Reading or writing of host flag 0: write 1: read	R/W	0

15.2.5 Transmit And Receive Data Register（0x10）

Table 15- 7 Transmit and Receive Data Register I^{2}C_DATA

Bit	Name	Description	R/W Sign	Reset Value
31:8	---	Reserved	R	0
7:0	TRDAT	Send/receive data	R/W	0

16 SPI Interface

The chip integrates 4 SPI M/S interfaces with ordinary DMA, supports SPI full-duplex mode, is used to communicate with external SPI interface devices, can be programmed to achieve master mode and slave mode work.

16.1 Overview

The controller of SPI Interface has the following features:

- 4 full-duplex SPI interfaces;
- master / slave mode were supported;
- Setting the clock polarity and phase were supported;
- Send and receive independent double buffer were supported;
- Transfer mode of LSB and MSB can be 8bit or 16 bit or 32bit;
- 256 kinds of baud rate can be set, up to 3.6864MHz(fcpu/8);
- Complete interrupt of data transmission is supported;
- Conflict interruption of data transmission were supported;
- Error interruption of SCSN mode were supported;
- DMA function were supported

16.2 I/O pin multiplexer and mapping

- ✓ SPI0：P50/P51/pP46/P47、P110~P113，cannot be reused as SPI0 at the same time
- ✓ SPI1：P40~P43、P52~P55，not reusable at the same time
- ✓ SPI2：P40~P43
- ✓ SPI3：P104~P107

16.3 Descriptions of Functions

SPI Interface meets the SPI HOST protocol standard, SPI clock mode is depending on the setting of CPOL(Clock Polarity) and CPHA(Clock Phase) parameter:CPOL shows front edge seat of clock is positive edge or negative edge, CPHA shows front edge seat of clock is data sampling or data setup.

The detailed working modes are listed in the table below：

Table 16- 1 SPI clock mode

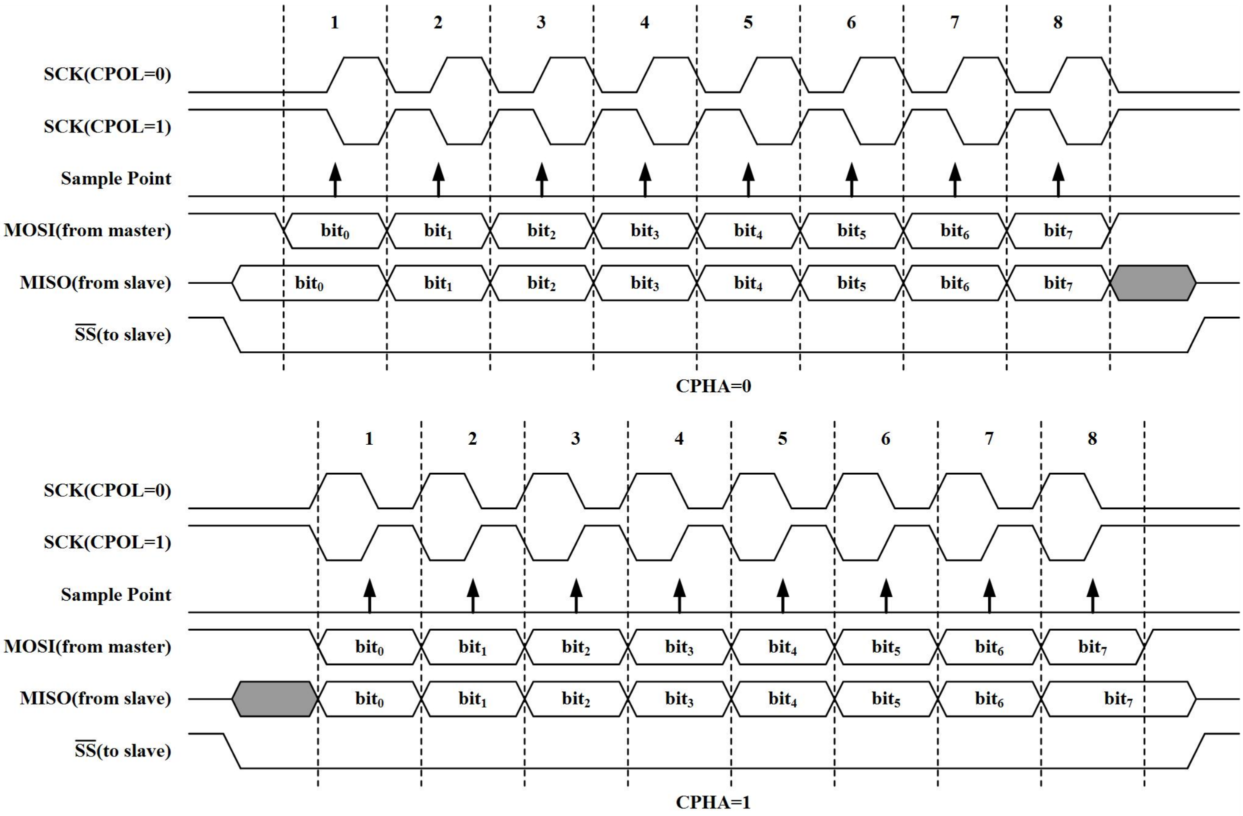
SPI MODE	**CPOL/CPHA**	Front edge seat	Rear edge
0	**0/0**	positive edge, data sampling	negative edge, data setup
1	**0/1**	positive edge, data setup	negative edge, data sampling
2	**1/0**	negative edge, data sampling	positive edge, data setup
3	**1/1**	negative edge, data setup	positive edge, data sampling

The width of the data transmission can be 8bit or 16 bit or 32bit, SPI clock source from the system clock, generate the communication clock After a divider coefficient.。

Support data transmission interrupt conflicts, data reception overflow interrupt, transmission end interrupt and SS mode error interrupt other four interrupts.。

Send data conflicts, when data is transmitting (txbusy1), then there is a write command, then TXCOLIF set as 1. If COL_IRQ_EN=1, interrupt occurs, meanwhile this send command wouldn't be response, the data which is sending will transferred over.

Figure 16-1 How the SPI clock works



Receive data overflow: Before the data over which to receive don't enter into shifting register,RXDATA register is not read, data overflow will generate ,RXCOLIF will be set as 1,if COL_IRQ_EN=1,interrupt occurs, meanwhile new receiving data will be save into receive data register,the old data will be overwritten.

End Of transmission interrupt: End Of transmission (sck_end), if TR_IRQ_EN=1, interrupt occurs, meanwhile, TRIF will be set as 1.

SCSN mode error interrupt: In the slave mode, SCSN must be input, When data is transmitted, SCSN become high, SCSN mode error flag will be set as 1;In the host mode, only enable the host model SCSN error detection(SCSN_EN=1),meanwhile SCSN is low, SCSN mode error flag will be set as 1. When SCSN mode error flag is 1, transmission is terminated, SPI module is reset, if ERR_IRQ_EN=1, interrupt occurs.

Note: After the control register is configured, when it is in the host mode, the SPI read/write operation will be started only when data is written to the data transmission register.

16.4 Register Descriptions

Table 16- 2 SPI register address

modules	physical address	mapping address
SPI0	0x40020000	0x40020000
SPI1	0x40050000	0x40050000
SPI2	0x40054000	0x40054000
SPI3	0x40058000	0x40058000

Table 16- 3 SPI register offset address

Register（X=0,1,2,3）	Address Offset	Descriptions

SPIx_CTL	0x0	Control Register
SPIx_STAT	0x4	Status Register
SPIx_TXDATA	0x8	Transmit Data Register
SPIx_RXDATA	0xC	Receive Data Register
SPIx_DMA_CTL	0x14	SPIX DMA Control Register
SPIx_DMA_TBADR	0x18	SPIX DMA Transmit Starting Address Register
SPIx_DMA_RBADR	0x1c	SPIX DMA Receive Starting Address Register
SPIx_DMA_TLEN	0x20	SPIX DMA Transmit Length Register
SPIx_DMA_RLEN	0x24	SPIX DMA Receive Length Register
SPIx_DMA_TADR	0x28	SPIX Current Transmitting DMA Address Register
SPIx_DMA_RADR	0x2c	SPIX Current Receiving DMA Address Register
SPIx_DMA_IE	0x30	SPIX DMA Interrupt Enable Register
SPIx_DMA_FLG	0x34	SPIX DMA Interrupt Flag Register

16.4.1 SPI Control Register（0x0）

Table 16- 4 SPI Control Register SPI_CTL

Bit	Name	Descriptions	R/W	Reset Value
31:27	---	Reserved	R	0
26	SLV_TX_ADV	SPI slave high-speed mode enable bit 0: SPI slave low-speed mode, fcpu/8, downward compatible 1: SPI slave high-speed mode, Slave rates up tofcpu/4 Note: Slave high-speed mode, regardless of the path delay, the slave rate can be up to fcpu/2	R/W	0
25	SCSN_POS_IRQEN	SCSN_POS interrupt enable 0: Disable interrupt 1: Enable interrupt	RW	0
24	SCSN_NEG_IRQEN	SCSN_NEG interrupt enable 0: Disable interrupt 1: Enable interrupt	RW	0
23	TXEMPT_IRQEN	TXEMPT interrupt enable 0: Disable interrupt 1: Enable interrupt	RW	0
22	TX_DFLT_EN	Send data content when BUF is empty 1: Send SPI_TXDFLT register value 0: Send the value of the last transmission	RW	0

		SPI_TXDATA		
21	---	Reserved bit, default is 0, don't write 1	R/W	0
20	---	Reserved bit, default is 0, don't write 1	R/W	0
19:12	CLKDIV	SCK clock division factor SCK frequency = system clock frequency/(2*(CLKDIV + 1))	R/W	0
11:10	WIDTH	Data width selection 0:8bit 1:16bit 2:32bit 3: Reserved, 8bit	R/W	0
9	SCSN_EN	SCSN model error detection enable, Works only in the master mode 0: Disable SCSN model error detection in the master mode, SCSN is GPIO 1: Enable SCSN model error detection in the master mode, SCSN as the input of SPI	R/W	0
8	CPHA	Clock phase selection 0: Former edge sampling data 1: Former edge data is established	R/W	0
7	CPOL	Clock polarity selection 0: “SCK” will be set as low, when idle 1: “SCK” will be set as high, when idle	R/W	0
6	LMSB	LSB/MSB selection 0: MSB is transferred first 1: LSB is transferred first	R/W	0
5	TXCOL_IRQ_EN	Data conflict interrupt enable 0: Disable write conflict interrupt 1: Enable write conflict interrupt	R/W	0
4	RXCOL_IRQ_EN	Data conflict interrupt enable 0: Disable read conflict interrupt 1: Enable read conflict interrupt	R/W	0
3	ERR_IRQ_EN	SCSN mode error interrupt enable 0: Disable mode error interrupt 1: Enable mode error interrupt	R/W	0
2	TR_IRQ_EN	Data transmission interrupt enable 0: Disable transmit data interrupt 1: Enable transmit data interrupt	R/W	0
1	MAST/SLAV	MASTER/SLAVE 1: MASTER 0: SLAVE	R/W	1
0	EN	Enable 0: Disable SPI Interface 1: Enable SPI Interface	R/W	0

16.4.2 SPI Status Register（0x4）

Table 16- 5 SPI Status Register SPI_STAT

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:8	---	Reserved	R	0
7	SCSN_POS	Flags for CSN pull-up events when acting as a slave 0: No CSN pull-up event occurred 1: A CSN pull-up event occurred	R/W	0
6	SCSN_NEG	Flags for CSN pull-down events when acting as a slave 0: No CSN pull-down event occurred 1: A CSN pull-down event occurred	R/W	0
5	TXEMPT	Send data null conflict flag bit. When BUF is empty, a write data null conflict occurs when the SPI bus needs to send data 0： No data sent null conflict interrupt 1： Generate sent data null conflict interrupt	R/W	0
4	TXBUSY	Sending data busy state flag. 0: Sending data is idle, bus can write command of SPITX register 1: Data is sending, bus can’ t write command of SPITX register	R	0
3	TXCOLIF	Write conflict flag. this bit will be reset by writing '1' When data is sending (TXBUSY is 1), user write new sending data to SPI, then new sending data will be lost. Write conflict flag will be set as 1. 0: No write data conflict interrupt 1: Generate the interrupt of write data conflict	R/W	0
2	RXCOLIF	Receiving data overflow flag. this bit will be reset by writing '1'. When receiving data in a row, if user don’t read RXDATA register, generate the receiving data overflow 0: No receiving data overflow interrupt 1: Generatethe receiving data overflow interrupt	R/W	0
1	ERRIF	SCSN mode conflict interrupt flag bit: when SPI is in host mode, only when SCSN_EN is 1, meanwhile“SCSN”is low, this bit will be set as 1; when SPI is in slave mode, “SCSN” as slave input, When the data was transmitted, if “SCSN” is high, this bit will be set as 1; if ERR_IRQ_EN=1,interrupt occurs, if generate the mode conflict, SPI module will be reset. this bit will be reset by writing '1' 0: No mode conflict interrupt 1: Generatethe mode conflict interrupt	R/W	0
0	TRIF	Sending data interrupt flag bit, When the data transfer is finished, this bit will be set as 1, if TR_IRQ_EN=1,interrupt	R/W	0

		occurs, this bit will be reset by writing '1' 0: No sending data interrupt 1: Generatethe sending data interrupt, send data register is empty.		

16.4.3 SPI Transmit Data Register（0x8）

Table 16-6 SPI Transmit Data Register SPI_TXDATA

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:0	TXDATA	Transmit data register	R/W	0

16.4.4 SPI Receive Data Register（0xC）

Table 16-7 SPI Receive Data Register SPI_RXDATA

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:0	RXDATA	Receive data register	R	0

16.4.5 SPI Default Transmit Data Register（0x10）

Table 16-8 SPI Default Transmit Data Register SPI_TXDFLT

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:0	TXDFLT	Default transmit data register	R/W	0

16.4.6 SPI DMA Control Register（0x14）

Table 16-9 SPI DMA Control Register SPI_DMA_CTL

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:4	Reserved	Reserved	R	0
3	RX_CYC_MODE	Receive circular mode enable	R/W	0
2	TX_CYC_MODE	Transmit circular mode enable	R/W	0
1	RX_DMA_EN	Receive DMA enable	R/W	0
0	TX_DMA_EN	Transmit DMA enable	R/W	0

16.4.7 SPI DMA Transmit Starting Address Register（0x18）

Table 16-10 SPI DMA Transmit Starting Address Register SPI_DMA_TBADR

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:17	Reserved	Reserved	R	0
16:0	DMA_TBADR	DMA transmit starting address（Byte address） The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

16.4.8 SPI DMA Receive Starting Address Register（0x1C）

Table 16- 11 SPI DMA Receive Starting Address Register SPI_DMA_RBADR

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:17	Reserved	Reserved	R	0
16:0	DMA_RBADR	DMA receive starting address（Byte address） The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

16.4.9 SPI DMA Transmit Length Register（0x20）

Table 16- 12 SPI DMA Transmit Length Register SPI_DMA_TLEN

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:17	Reserved	Reserved	R	0
16:0	DMA_TLEN	DMA transmit length（Byte address）=（n）Byte The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

16.4.10 SPI DMA Receive Length Register（0x24）

Table 16- 13 SPI DMA Receive Length Register SPI_DMA_RLEN

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:17	Reserved	Reserved	R	0
16:0	DMA_RLEN	DMA receive length（Byte address）=（n）Byte The lowest two bits must be configured according to the specific interface data bit width configuration	R/W	0

16.4.11 SPI DMA Current Transmitting Address Register（0x28）

Table 16- 14 SPI DMA Current Transmitting Address Register SPI_DMA_TADR

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:17	Reserved	Reserved	R	0
16:0	DMA_TADR	Current transmitting DMA address（Byte address）	RO	0

16.4.12 SPI DMA Current Receiving Address Register（0x2C）

Table 16- 15 SPI DMA Current Receiving Address Register SPI_DMA_RADR

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:17	Reserved	Reserved	R	0
16:0	DMA_RADR	Current receiving DMA address（Byte address）	RO	0

16.4.13 SPI DMA Interrupt Enable Register（0x30）

Table 16- 16 SPI DMA Interrupt Enable Register SPI_DMA_IE

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:6	Reserved	Reserved	R	0
5	RX_ERR_IE	Received data overlay interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
4	TX_ERR_IE	Transmit data error interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
3	RX_FIE	DMA receive all full interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
2	RX_HIE	DMA receive half full interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
1	TX_FIE	DMA transmit all empty interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0
0	TX_HIE	DMA transmit half empty interrupt enable 0: Disable interrupt 1: Enable interrupt	R/W	0

16.4.14 SPI DMA Interrupt Flag Register（0x34）

Table 16- 17 SPI DMA Interrupt Flag Register SPI_DMA_IF

Bit	**Name**	**Descriptions**	**R/W**	**Reset Value**
31:6	Reserved	Reserved	R	0
5	RX_ERR	Received data overlay flag, write one cleared	R/WC	0
4	--	Reserved	R	0
3	RX_FDONE	DMA receive all full interrupt flag, write one cleared	R/WC	0
2	RX_HDONE	DMA receive half full interrupt flag, write one cleared	R/WC	0
1	TX_FDONE	DMA transmit all empty interrupt flag, write one cleared	R/WC	0
0	TX_HDONE	DMA transmit half empty interrupt flag, write one cleared	R/WC	0

17 Energy Integration Unit D2F

17.1 Overview

The chip integrates 12 custom energy integration units, with power signals D2FP0~D2FP11 as inputs. The electric constant Hfconst3 can be set for integration based on the input power and electric constant Hfconst3. The electric energy is stored in the D2FE00~D2FE11 registers, where D2FP0~D2FP2 support pulse output, classified as D2F_ OUT0~D2F_ OUT2, as a check pulse output, supports electrical energy pulse interruption. When the output pulse period is greater than or equal to 160ms, the high level of the check pulse is fixed at 80ms, and the low level changes according to the cycle value. It will output an equal duty pulse when the output pulse period is less than 160ms.

Base address: 0x4005C000

17.2 Features

- 12 channels D2F
- Eletrical constant configurable
- Power is 28 bits signed number, input range is ±2^27
- Supports pluse interrupt
- Supports output 3 channels check pluse: D2F_OUT0~D2F_OUT2
- The mode of energy register can configure as accumulated or cleared after read type

17.3 Register Description

Module Name	**Physical Address**	**Address Mapping**
D2F	0x4005C000	0x4005C000

Register Name	**Address Offset**	**Description**
HFconst3	0x0	D2F high frequency electrical constant register
IE	0x4	D2F interrupt enable register
IF	0x8	D2F interrupt flag register
CFG	0xC	D2F configuration register
D2FP00~D2FP11	0x10~0x3C	D2F power input register
D2FE00~D2FE11	0x40~0x6C	D2F custom energy register

Above registers do not support bitband operation.

17.3.1 D2F high frequency electrical constan register HFconst3 (0x00)

Bit	Name	Description	R/W sign	Reset value
31:16	Reserved	Reserved	R	0
15:0	HFCONST3	12 custom energy register D2FE00-D2FE11 and 3 custom CF electrical constan of pluse CF0-CF2. High frequengy constant register is 16 unsigned number.	R/W	0

HFCONST3 is used for the eletrical constant of 12 custom energy registers D2FE00-D2FE11 and 3 custom CF pulses (CF0-CF2). The high-frequency pulse constant register is a 16 bit unsigned number.

17.3.2 D2F interrupt enable register IE(0x04)

This register is the D2F interrupt enable register. When this event occurs, the corresponding D2FIF is set, and if the corresponding interrupt allow bit is configured to 1, a D2F interrupt (interrupt number: 4) is generated.

Bit	Name	Description	R/W sign	Reset value
31:12	Reserved	Reserved	R	0
11	D2F11IE	Energy integral unit D2F11 interrupt enable	R/W	0
10	D2F10IE	Energy integral unit D2F10 interrupt enable	R/W	0
9	D2F9IE	Energy integral unit D2F9 interrupt enable	R/W	0
8	D2F8IE	Energy integral unit D2F8 interrupt enable	R/W	0
7	D2F7IE	Energy integral unit D2F7 interrupt enable	R/W	0
6	D2F6IE	Energy integral unit D2F6 interrupt enable	R/W	0
5	D2F5IE	Energy integral unit D2F5 interrupt enable	R/W	0
4	D2F4IE	Energy integral unit D2F4 interrupt enable	R/W	0
3	D2F3IE	Energy integral unit D2F3 interrupt enable	R/W	0
2	D2F2IE	Energy integral unit D2F2 interrupt enable	R/W	0
1	D2F1IE	Energy integral unit D2F1 interrupt enable	R/W	0
0	D2F0IE	Energy integral unit D2F0 interrupt enable	R/W	0

17.3.3 D2F interrupt flag register IF (0x08)

This register is the D2F event status register, and when the D2FE energy register is added with 1, the corresponding flag is set to 1. When the corresponding interrupt allowed bit is set to 1, state position 1 will cause CPU EMU interrupt 1 (interrupt number: 4) to be generated.

Bit	Name	Description	R/W sign	Reset value
31:12	Reserved	Reserved	R	0
11	D2F11IF	Energy integral unit D2F11 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
10	D2F10IF	Energy integral unit D2F10 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
9	D2F9IF	Energy integral unit D2F9 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
8	D2F8IF	Energy integral unit D2F8 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
7	D2F7IF	Energy integral unit D2F7 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
6	D2F6IF	Energy integral unit D2F6 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
5	D2F5IF	Energy integral unit D2F5 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0

4	D2F4IF	Energy integral unit D2F4 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
3	D2F3IF	Energy integral unit D2F3 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
2	D2F2IF	Energy integral unit D2F2 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
1	D2F1IF	Energy integral unit D2F1 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0
0	D2F0IF	Energy integral unit D2F0 energy pluse flag. Write 1 and cleared.If the corresponding bit is D2FIE=1, cleared and clear the interrupt at the same time.	R/W	0

17.3.4 D2F configuration register CFG (0x0C)

Bit	Name	Description	R/W sign	Reset value
31:2	Reserved	Reserved	R	0
1	D2F_DISABLE	D2F module enable =0, enable D2F module =1, resets all counters and number arithmetic units without resetting register values. The default value is 1.When starting the integration, the bit should be configurated to 0 first, and then keep the bit at 0, and write the D2FP register directly to start the integration immediately.	R/W	1
0	ERegCAR	Energy register type select. ＝0, cleared after read. ＝1, cumulative type. The default value is 0。	R/W	0

17.3.5 D2F power input register D2FP (0x10~0x3C)

Offset address	10H	14H	18H	1CH	20H	24H	28H
Register	D2FP00	D2FP01	D2FP02	D2FP03	D2FP04	D2FP05	D2FP06
Default	0x0	0x0	0x0	0x0	0x0	0x0	0x0
Offset address	2CH	30H	34H	38H	3CH		
Register	D2FP07	D2FP08	D2FP09	D2FP10	D2FP11		
Default	0x0	0x0	0x0	0x0	0x0		

The power input register D2FP00-11 can use binary complement format, with 28 signed bits, where the highest bit is the sign bit.

When the power value is written to the D2FP register, the power will be integrated according to the pulse constant Hfconst3, and the integrated electric energy is stored in the D2FE00~D2FE11 register, in which D2FE00, D2FE01, and D2FE02 can output the check pulse, and bet on the D2F_OUT0, D2F_OUT1, and D2F_OUT2 respectively, and the relevant multiplexing configuration is shown in the GPIO chapter.

17.3.6 D2F custom energy register D2FE (0x40~0x6C)

Offset address	40H	44H	48H	4CH	50H	54H	58H
Register	D2FE00	D2FE01	D2FE02	D2FE03	D2FE04	D2FE05	D2FE06
Default	0x0	0x0	0x0	0x0	0x0	0x0	0x0
Offset address	5CH	60H	64H	68H	6CH		
Register	D2FE07	D2FE08	D2FE09	D2FE10	D2FE11		
Default	0x0	0x0	0x0	0x0	0x0		

The custom energy register is the 24-bit register D2FE [23:0], and the custom energy register can be selected as an cumulative type or a cleared after read type.

18 Ecure Password Accelerator SEA

SEA (Security Encryption Accelerator) is a module that provides hardware acceleration for symmetric encryption algorithms (AES) and public-key cryptography algorithms (ECDSA, ECDH, etc.). It includes AES hardware acceleration unit, PKA public key algorithm accelerator (ECC hardware acceleration unit, RSA hardware acceleration unit), hash algorithm accelerator, 128-bit finite field multiplication unit, and true random number generator.

The firmware RS-SEA based on this module has passed the FIPS140-3 algorithm certification, and the list of certified algorithms is as follows, please refer to the NIST official website for details:*Cryptographic Algorithm Validation Program | CSRC (nist.gov)*。

Module	Item	Sepecific content	Compliant standards
AES	ECB	Key Length-128、192、256	SP800-38A
	CBC	Key Length-128、192、256	SP800-38A
	CTR	Key Length-128、192、256	SP800-38A
	CFB128	Key Length-128、192、256	SP800-38A
	OFB	Key Length-128、192、256	SP800-38A
	GMAC	Key Length-128、192、256	SP800-38D
	GCM	Key Length-128、192、256	SP800-38D
DRBG	CTR	Mode:AES-128、AES-192、AES-256	SP800-90A
HASH	SHA-1	Message Length:0~65536 increment 8	FIPS 180-4
	SHA-224	Message Length:0~65536 increment 8	FIPS 180-4
	SHA-256	Message Length:0~65536 increment 8	FIPS 180-4
	SHA-384	Message Length:0~65536 increment 8	FIPS 180-4
	SHA-512	Message Length:0~65536 increment 8	FIPS 180-4
ECDSA	KeyGen	P-192、P-224、P-256、P-384、P-521	FIPS 186-4
	KeyVer	P-192、P-224、P-256、P-384、P-521	FIPS 186-4
	SigGen	P-192、P-224、P-256、P-384、P-521	FIPS 186-4
	SigVer	P-192、P-224、P-256、P-384、P-521	FIPS 186-4

18.1 AES hardware acceleration unit

18.1.1 Features

AES hardware acceleration unit supports customers to encrypt/decrypt data packets with symmetric AES algorithm, and its main functions are as follows:

- 128bit/192bit/256bit key length
- KeyExp key expension
- ECB, CBC, CTR, CFB128, OFB, GCM six type data flow processing modes
- Multiplication in GF(2^128) domain, which collaborate with software can accelerate the identity authentication (GMAC) process in GCM algorithms

18.1.2 Time consumption information

Hardware accelerative unit in the module	**Time consumption (Cycles)**
128 bits key expension	126
192 bits key expension	141
256 bits key expension	162

128 bits encryption/decryption unit	55
192 bits encryption/decryption unit	63
256 bits encryption/decryption unit	71
128 bits finite field multiplication unit	128

Note: Time consumption refers to the number of AHB clock cycles consumed from the start of the operation to the end of the operation.

18.2 ECC hardware accelerative unit

18.2.1 Features

The ECC hardware acceleration unit accelerates Elliptic Curve (EC) operations, which can significantly improve the implementation efficiency of the Elliptic Curve-based encryption and decryption protocol. The elliptic curve protocols that can be completed by this module with software include: NIST (P), SEC (P), SEC (K), Brainpool, etc., and the signature authentication algorithms that can be completed include: ECDSA (EC Digital Signature Algorithm), ECDH (EC Deffie-Hellman) and their variants.

The main functions:

- Supports modulo addition and modulo subtraction operations up to 521bit word length
- Supports pre-calculation of Montgomery parameters up to 521 bits
- Supports Montgomery modulo multiplication and modulo inverse operations up to 521 bits
- It supports the conversion of coordinates from integer field to Montgomery field before point addition and doubling operations
- Support word length from 160bit~521bit multiplier, point plus two basic point operations
- Supports the domain conversion of point operations to end coordinates from Montgomery field to integer field

Note: Renergy provides customers with algorithm library functions based on ECC hardware acceleration, which is convenient for customers to quickly realize ECDSA and other algorithm applications.

18.2.2 Time consumption information

Hardware accelerative unit in the module	**Time consumption (Cycles)**
J-parameter calculation	64
H-parameter calculation	2021~24960
Montgomery modulo multiplication	2151~25670
Montgomery modulo inverse	6330~108764
Modulo addition	21~50
Modulo subtraction	21~50
Point addition	7610~123982
Multiplying points	7317~116492
Algorithm	**Time consumption (Reference)**
ECDSA_NIST-P256	72ms~78ms(29MHz clock)
ECDSA_Brainpoolp512r1	350ms~359ms(29MHz clock)

Note: Time consumption refers to the range of AHB clock cycles from the start of the operation to the end of the calculation.

18.3 RSA hardware accelerative unit

18.3.1 Features

The RSA hardware acceleration unit provides hardware acceleration for RSA encryption and decryption operations, which can significantly improve the implementation efficiency of RSA encryption and decryption protocol, and this module supports RSA encryption and decryption with word lengths from 32 bits to 576 bits.

Main functions:

- Support the RSA encryption and decryption protocol based on Euler function φ(N)=(P-1)(Q-1)
- Supports modulo exponentiation of word lengths from 32 bits to 576 bits
- Supports modulo multiplication of word lengths from 32 bits to 2048 bits

18.3.2 Time consumption information

Hardware accelerative unit in the module	**Time consumption (Cycles)**
J-parameter calculation	64
H-parameter calculation	2021~24960
Modulo multiplication	2366~28043
Modulo exponentiation	2675~1010246

Note: Time consumption refers to the range of AHB clock cycles from the start of the operation to the end of the calculation.

18.4 HASH hardware accelerative unit

18.4.1 Features

According to the number of hash bits, the secure hashing algorithm is divided into: 160-bit SHA-1, 224-bit SHA-224, 256-bit SHA-256, 384-bit SHA-384, 512-bit SHA-512, etc.

The main functions of the hash algorithm accelerator are as follows:

- Supports SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 secure hash algorithm acceleration, and the actual execution of grouping and data transfer is completed by software
- Supports CPU status query and interrupt mode

18.4.2 Time consumption information

Hardware accelerative unit in the module	**Time consumption (Cycles)**
SHA-1	101
SHA-224	85
SHA-256	85
SHA-384	101
SHA-512	101

Note: Time consumption refers to the range of AHB clock cycles from the start of the operation to the end of the calculation

18.5 Truly random number generator TRNG

18.5.1 Featrues

Truly random number generators use Fibonacci loops to vibrate as an entropy source to generate n-scale random bit streams. Features are as follows:

- Supports up to 29MHz system frequency and outputs random number bit streams
- It can output a random bit stream at the frequency of the current system main frequency/N(1~65536)

- Supports random number randomness detection and error reporting function, and supports error correction algorithm
- Supports LFSR pseudo-random algorithm
- Supports address interleaving randomness enhancement algorithm is supported
- Supports interrupt mode to output 128-bit random number each time

18.5.2 Time consumption information

Generate data types	**Time consumption(Cycles)**
The first set of 128 bits data after the module is enabled	209
128 bits data of other sets	128

Note: Time consumption, the number of AHB clock cycles consumed to start the operation to 128 bits (Cycles).

18.6 Software configuration process

Please refer to the relevant chapters of "Renergy Library Function" about the firmware RS-SEA v1.0.0 of this module, and the use process and application scheme are referred to "Renergy Application Note".

19 Simple Timer SIMP_TC

19.1 Overview

Four 16-bit system beat counters are integrated into the module and have two operating modes:

- Single operation mode
- Cycle operation mode

19.2 Function description

_SIMP_TC module is a simple 16-bit system beat counter, which uses the system clock running by the CPU to count, and when the count value reaches the preset target value, an overflow flag is generated;

The counter supports two different modes: single mode and cycle, after the single mode starts counting, the counter generates an overflow flag after the counting value reaches the preset target value, and stops counting; In cyclic mode, after the counting is started, the counter generates an overflow flag after the counting value reaches the preset target value, and continues counting from 0.

The counter starts counting after the software enables the enable bit, and stops counting after the software disables the counting enable bit.

19.3 Register deacription

SIMP _TC registers' base address

Name	Physical address	Mapping address
SIMP_TC	0x40060000	0x40060000

SIMP_TC registers' offset address

Name	Offset address	Description
CTRL1	0x0	SIMP _TC1 control register
LOAD1	0x4	SIMP_TC1 target count value register
VAL1	0x8	SIMP_TC1 current count value register
CTRL2	0xc	SIMP _TC2 control register
LOAD2	0x10	SIMP_TC2 target count value register
VAL2	0x14	SIMP_TC2 current count value register
CTRL3	0x18	SIMP _TC3 control register
LOAD3	0x1c	SIMP_TC3 target count value register
VAL3	0x20	SIMP_TC3 current count value register
CTRL4	0x24	SIMP _TC4 control register

LOAD4	0x28	SIMP_TC4 target count value register
VAL4	0x2c	SIMP_TC4 current count value register

19.3.1 Control register CTRL1 (0x0)

Bit	Name	Description	R/W sign	Reset value
31:4	---	Reserved	R	0
3	IRQEN1	Interrupt enable bit 0: disable interrupt 1:enable interrupt	R/W	0
2	MODE1	Count mode bit 0: single count 1: cycle count	R/W	0
1	OV1	Count overflow flags 0: counter is not overflowing 1: counter is overflowing Note: write 1 and cleared	R/W	0
0	EN1	Count enable bit 0: count stop 1: count start	R/W	0

19.3.2 Target count value register LOAD1 (0x4)

Bit	Name	Description	R/W sign	Reset value
15:0	LOAD1	Target count value: LOAD + 1	R/W	0

19.3.3 Current count value register VAL1 (0x8)

Bit	Name	Description	R/W sign	Reset value
15:0	VAL1	Current counter's value	R	0

19.3.4 Control register CTRL2 (0xC)

Bit	Name	Description	R/W sign	Reset value
31:4	---	Reserved	R	0
3	IRQEN2	Interrupt enable bit 0: disable interrupt 1:enable interrupt	R/W	0
2	MODE2	Count mode bit 0: single count 1: cycle count	R/W	0
1	OV2	Count overflow flags 0: counter is not overflowing 1: counter is overflowing Note: write 1 and cleared	R/W	0
0	EN2	Count enable bit	R/W	0

		0: count stop 1: count start		

19.3.5 Target count value register LOAD2 (0x10)

Bit	Name	Description	R/W sing	Reset value
15:0	LOAD2	Target count value:LOAD + 1	R/W	0

19.3.6 Current count value register VAL2 (0x14)

Bit	Name	Description	R/W sign	Reset value
15:0	VAL2	Current counter's value	R	0

19.3.7 Control register CTRL3 (0x18)

Bit	Name	Description	R/W sign	Reset value
31:4	---	Reserved	R	0
3	IRQEN3	Interrupt enable bit 0: disable interrupt 1:enable interrupt	R/W	0
2	MODE3	Count mode bit 0: single count 1: cycle count	R/W	0
1	OV3	Count overflow flags 0: counter is not overflowing 1: counter is overflowing Note: write 1 and cleared	R/W	0
0	EN3	Count enable bit 0: count stop 1: count start	R/W	0

19.3.8 Target count value register LOAD3 (0x1C)

Bit	Name	Description	R/W sign	Reset value
15:0	LOAD3	Target count value:LOAD + 1	R/W	0

19.3.9 Current count value registerVAL3 (0x20)

Bit	Name	Description	R/W sign	Reset value
15:0	VAL3	Current counter's value	R	0

19.3.10 Control register CTRL4 (0x24)

Bit	Name	Description	R/W sign	Reset value
31:4	---	Reserved	R	0
3	IRQEN4	Interrupt enable bit 0: disable interrupt 1:enable interrupt	R/W	0
2	MODE4	Count mode bit	R/W	0

		0: single count 1: cycle count		
1	OV4	Count overflow flags 0: counter is not overflowing 1: counter is overflowing Note: write 1 and cleared	R/W	0
0	EN4	Count enable bit 0: count stop 1: count start	R/W	0

19.3.11 Target count value register LOAD4 (0x28)

Bit	Name	Description	R/W sign	Reset value
15:0	LOAD4	Target count value :LOAD + 1	R/W	0

19.3.12 Current count value register VAL4 (0x2C)

Bit	Name	Description	R/W sign	Reset value
15:0	VAL4	Current counter’s value	R	0

20 Memory transfer module M2M

The Mem2Mem module uses the DMA mode to automatically transfer data from the specified source address to the destination address without the CPU participating. The minimum data unit that can be configured for import is 1 Word, and the source data and target data can be configured in significant bits.

20.1 Features

- The source and destination addresses are configurable, and the addresses are aligned by Word
- The source data length is configurable, and the minimum unit is Word
- Input data significant bits are configurable
- Outputput data significant bits are configurable
- Each Word output data to be stored to the destination address in reverse byte order can be configurated
- The output data can be stored to the destination address in reverse order according to the Word address
- A flag will be generated when the data transfer is complete, and an interrupt can be configured to occur

20.2 Description

20.2.1 Configurable significant bit width of input and output data

The input and output data configuration here refers to how many bytes are valid in each Word data.

The effective bit width of the input and output data can be independently configured through the registers for ivld [3:0] and ovld [3:0], and each Word data can be configured to be valid for 1~4Bytes.

If the byte of the input data is invalid, the byte corresponding to each byte in the source data will be discarded during data migration.

If the byte of the output data is invalid, indicating that the SRAM of each Word in the destination address does not store valid data.

- When the significant bit widths of input and output data are set to 4 bytes, data can be directly migrated to SRAM
- When the significant bit width of the input data is less than 4 bytes, the invalid data in the input source data can be discarded
- When the significant bit width of the output data is less than 4 bytes, the function of inserting dummy bytes into the target data can be realized

Other functions can also be implemented through the combination of output outputs, see the "**Example Description**" section for details.

20.2.2 Configurable dummy value when output address is invaild

This can be done through register configuration, which populates the configured register value when the bit of the output address is invalid. By default, no write operations are performed on invalid addresses.

20.2.3 Reverse order output data

Configure registers M2M_MODE. ORV_EN=1 enables the output of Word data in reverse order. For example, if the byte width of the output Word data is 3 bytes, the three bytes of data (A0, A1, and A2) of the source data are retrieved each time, and then de-ordered (A2, A1, A0) and stored in SRAM.

20.2.4 Configurable source/target address and input data width

The source data address and the destination data address are configured through registers, and the configuration aligned by Word is supported. The length of the input data can be configured via registers, and the minimum scale is Word. The output data length is adaptive based on the input and output length and the significant bits of the input and output data.

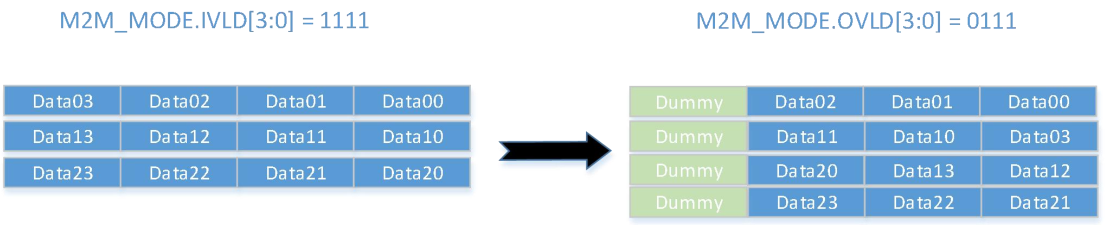
20.2.5 Data arranged in reverse order by address

The configuration register is M2M_MODE. ADDR_RVEN=1 enables the output data to be placed in reverse order of Word addresses. The address reverse order feature only supports alignment by Word. Each time you get one Word data (A3, A2, A1, A0) from the source data, write (A3, A2, A1, A0) to the destination address, and then minus 4 from the destination address to store the next Word with the source data. And so on until the last Word is saved.

Note: When using reverse order of addresses, it is necessary to ensure that the M2M_MODE. IVLD = 4’b1111, M2M_MODE. OVLD = 4’b1111, M2M_SADDR [1:0] = 0, M2M_DADDR[1:0] = 0; M2M_ILEN = 4*n (n is an integer greater than 0).

20.3 Example description

20.3.1 The input data is valid for 4Bytes and the output is valid for 3Bytes



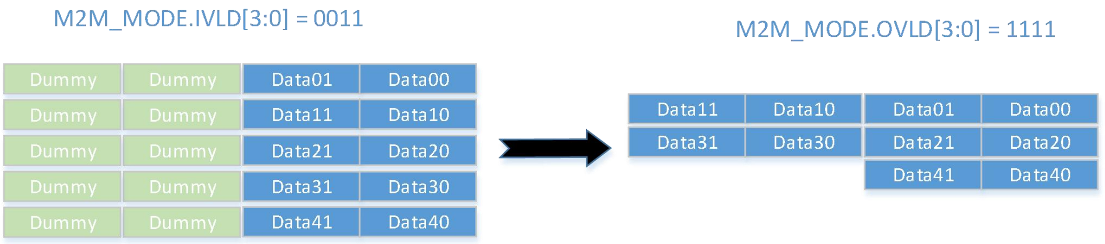
M2M_MODE.IVLD[3:0] = 1111

Data03	Data02	Data01	Data00
Data13	Data12	Data11	Data10
Data23	Data22	Data21	Data20

M2M_MODE.OVLD[3:0] = 0111

Dummy	Data02	Data01	Data00
Dummy	Data11	Data10	Data03
Dummy	Data20	Data13	Data12
Dummy	Data23	Data22	Data21

20.3.2 The input data is valid for 2Bytes, and the output is valid for 4Bytes



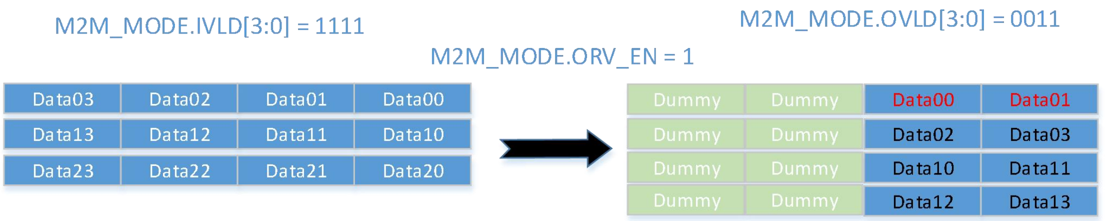
M2M_MODE.IVLD[3:0] = 0011

Dummy	Dummy	Data01	Data00
Dummy	Dummy	Data11	Data10
Dummy	Dummy	Data21	Data20
Dummy	Dummy	Data31	Data30
Dummy	Dummy	Data41	Data40

M2M_MODE.OVLD[3:0] = 1111

Data11	Data10	Data01	Data00
Data31	Data30	Data21	Data20
		Data41	Data40

20.3.3 The input data is stored in reverse order of bytes (4 bytes of input is valid, 2 bytes of output is valid)



M2M_MODE.IVLD[3:0] = 1111

M2M_MODE.ORV_EN = 1

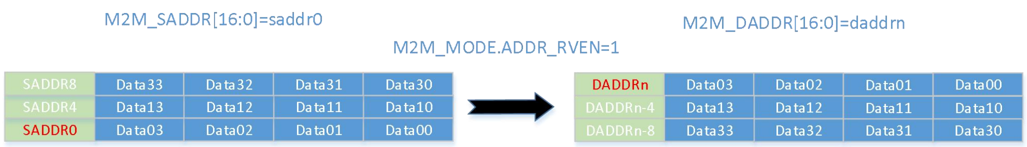
Data03	Data02	Data01	Data00
Data13	Data12	Data11	Data10
Data23	Data22	Data21	Data20

M2M_MODE.OVLD[3:0] = 0011

Dummy	Dummy	Data00	Data01
Dummy	Dummy	Data02	Data03
Dummy	Dummy	Data10	Data11
Dummy	Dummy	Data12	Data13

From the figure above, the output address is discharged in reverse order from the input address; In the source address, data00 is placed at the lowest address of a Word, and in the destination address, data00 is placed at the highest address of a Word.

20.3.4 Data arranged in reverse order by address



M2M_SADDR[16:0]=saddr0

M2M_MODE.ADDR_RVEN=1

SADDR8	Data33	Data32	Data31	Data30
SADDR4	Data13	Data12	Data11	Data10
SADDR0	Data03	Data02	Data01	Data00

M2M_DADDR[16:0]=daddrn

DADDRn	Data03	Data02	Data01	Data00
DADDRn-4	Data13	Data12	Data11	Data10
DADDRn-8	Data33	Data32	Data31	Data30

From the figure above, the first Word data of the source data (data03, data02, data01, data00) is written to the destination address daddrn, and then the destination address is minus 4 to store the Word of the next source data. And so on until the last Word is saved.

20.4 Transfer speed

The fastest way to complete a Word data transfer is within 6 cycles. The actual speed still matches the configured M2M_ MODE Competition related to IVLD, OVLD, and SRAM.

20.5 Register description

Name	Physical	Mapping address
M2M	0x4006_8000	0x4006_8000

M2M module register’s offset address

Name	Offset address	R/W	Reset value	Description
M2M_MODE	0x00	R/W	0xFF	M2M mode configuration register
M2M_CTL	0x04	R/W	0x0	M2M control register
M2M_DUMMY	0x08	R/W	0x0	M2M invalid address DUMMY value configuration register
M2M_SADDR	0x0C	R/W	0x0	M2M source data address configuration register
M2M_DADDR	0x10	R/W	0x0	M2M target data address configuration register
M2M_ILEN	0x14	R/W	0x0	M2M source data lenthconfiguration register
M2M_IE	0x18	R/W	0x0	M2M interrupt enable register
M2M_IF	0x1C	R/W	0x0	M2M interrupt flag register

The above registers do not support bitband operations.

20.5.1 M2M_MODE (0x0)

M2M mode configuration register, address: 0x4006_8000, default value: 0xFF.

Bit	Name	Description	R/W	Reset value
31:11	--	Reserved	RO	0
10	ADDR_RVEN	The data is arranged in reverse order by Word address =0: Not reverse =1: In reverse order by Word address	RW	0
9	DUMMY_EN	=0: Invalid byte locations in the target address are not written or read =1: Invalid byte locations in the target address is written to the dummy value	RW	0
8	ORV_EN	The output Word data is placed in reverse order of Bytes =0: Not reverse =1: Reverse	RW	0
7:4	OVLD	Output Word data valid Byte configuration =0:Byte invaild =1:Byte vaild If the corresponding byte is 0, the byte address in the destination address is invalid or M2M_MODE. DUMMY_EN is written when the DUMMY value is valid.	RW	0xf
3:0	IVLD	Input Word data valid Byte configuration =0:Byte invaild	RW	0xf

		=1:Byte vaild The 4-bit configuration corresponds to each byte in each Word data by byte, and the high level indicates that the corresponding byte is valid.		

20.5.2 M2M_CTL (0x4)

M2M control register, address: 0x4006_8004, default value: 0x0.

Bit	Name	Description	R/W	Reset value
31:1	--	Reserved	RO	0
0	M2M_EN	=0: No opeartion =1: Start data transfer. After the specified length data is migrated, it is automatically cleared. When the bit is 1, write 0 to terminate the current transfer process.	RW	0

20.5.3 M2M_DUMMY (0x8)

M2M invalid address DUMMY value configuration register, address: 0x4006_8008, default value: 0x0.

Bit	Name	Description	R/W	Reset value
31:0	DUMMY	When M2M_MODE. When DUMMY_EN=1, this register is valid. The 32bit value is measured by Byte and M2M_MODE. OVLD configures a one-to-one correspondence between bytes of invalid addresses.	RW	0

20.5.4 M2M_SADDR (0xC)

M2M source data address configuration register, address: 0x4006_800C, default value: 0x0;

Bit	Name	Description	R/W	Reset value
31:17	--	Reserved	RO	0
16:0	SADDR	The source data address needs to be aligned by Word. The address here is the SRAM offset address, and the source data address + source data length cannot exceed the SRAM address range.	RW	0

20.5.5 M2M_DADDR (0x10)

M2M target data address configuration register, address: 0x4006_8010, default value: 0x0.

Bit	Name	Description	R/W	Reset value
31:17	--	Reserved	RO	0
16:0	DADDR	The source data address needs to be aligned by Word. The address here is the SRAM offset address, and the source data address + source data length cannot exceed the SRAM address range.	RW	0

20.5.6 M2M_ILEN (0x14)

M2M source data lenth configuration register, address: 0x4006_8014, default value0 x0.

Bit	Name	Description	R/W	Reset value

31:17	--	Reserved	RO	0
16:0	ILEN	Input data lenth configuration, aligned by Word. Data length cannot exceed the SRAM address range. Note: Enabling M2M at ILEN=0 is not allowed.	RW	0

20.5.7 M2M_IE (0x18)

M2M interrupt enable register, address: 0x4006_8018, default value: 0x0；

Bit	Name	Description	R/W	Reset value
31:1	--	Reserved	RO	0
0	DONE_IE	Interruption enabled after the transfer is completed =0: disable =1: enable interrupt	RW	0

20.5.8 M2M_IF (0x1C)

M2M interrupt flag register, address: 0x4006_801C, default value: 0x0

Bit	Name	Description	R/W	Reset value
31:1	--	Reserved	RO	0
0	DONE	Transfer completed flag, write 1 and cleared =0: transfer incomplete =1: transfer complete Everytime configurates M2M_CTL.M2M_EN=1, thisb it cleared automatically.	RW1C	0

20.6 Software configuration process

1) Configure source data start address M2M_SADDR
2) Configure target data start address M2M_DADDR
3) Configure data lenth M2M_ILEN
4) Configurw mode register M2M_MODE
5) Configure interrupt enable M2M_IE
6) Configure transfer start M2M_CTL
7) Wait for interrupt or flag. Read data from target address,and clear flag

21 DSP Core

This module is a DSP module for data signal processing, including operations such as FFT computation in floating-point format, arithmetic units for conversion between integers and floating-point numbers, basic arithmetic units for floating-point operations, Cordic algorithm for computing sine, cosine, and root-mean-square, IIR acceleration operation, FIR acceleration operation, linear interpolation operation, and Lagrange interpolation operation.

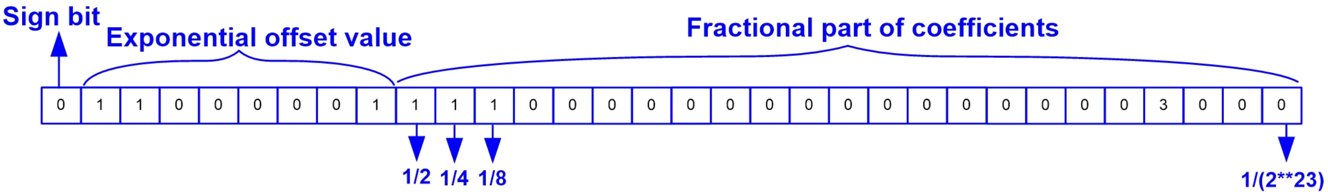
The floating-point numbers referred to here are all single-precision floating-point numbers, with 23 bits of effective digits after the decimal point (in binary).

21.1 Features

- Support conversion between integers and floating-point numbers
- Support floating-point addition, subtraction, multiplication, and division.
- Support single butterfly operation (for complex numbers) and continuous butterfly operation with DMA.
- Support the entire process of radix-2 FFT, with point numbers of 64, 128, 256, 512, and 1024.
- Support automatic data movement operation for bit-reversal, with point numbers of 4, 8, 16, 32, 64, 128, 256, 512, and 1024.
- Support sine and cosine calculation.
- Support root-mean-square calculation.
- Support arctangent calculation.
- Support single IIR calculation and IIR calculation with DMA.
- Support FIR filtering operation.
- Support linear interpolation.
- Support Lagrange interpolation.

21.2 Basic Principles of Calculation Engine

21.2.1 Single-Precision Floating-Point Numbers



The figure above shows the representation method of floating-point numbers. The numeric value of a floating-point number is expressed as $\pm 2^n(1+\mathrm{f})$, where n represents the exponent value and f represents the fractional part.

The sign bit represents the positive or negative value of the floating-point number, with 0 representing a positive integer and 1 representing a negative number.

The exponent part is represented using a biased notation, with the highest bit of the 8-bit "exponent bias value" representing the exponent sign and the low 7 bits representing the actual exponent value. For example, the exponent value of 1 represented in biased notation is 1+127=128; the exponent value of -10 represented in biased notation is -10+127=117.

The fractional part of coefficient represents the numbers after the decimal point.

21.2.2 Special Value

Type	Sign Bit(1bit)	Exponent Bias Value (8bit)	Fractional Part of Coefficient (23bit)

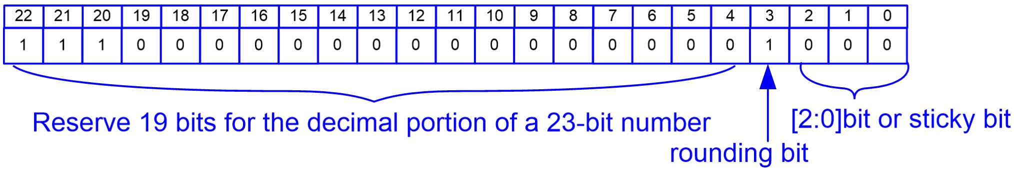
0	0/1	0	0
Denormal	0/1	0	Non-zero
Infinity	0/1	255	0
NaN	0/1	255	Non-zero
Normal	0/1	1~254	Any

The table above lists some special values in floating-point number representation. When performing floating-point arithmetic, it is necessary to check whether the input and output data are special values, perform special processing, and provide flags accordingly.

21.2.3 Floating-Point Rounding

When the bit width of the coefficient decimal part in floating-point arithmetic is greater than the actual representable bit width, rounding is needed to process the data.

First, the following concepts are defined:



22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

For truncating m-bit data to n bits as shown in the figure above:

- rounding bit:Starting from the leftmost highest bit, it is the (n+1)th bit;
- Stick bit: Starting from the leftmost highest bit, all bits starting from the (n+2)th lowest bit.

21.2.4 IEEE 754 Standard Rounding Modes

The IEEE 754 standard specifies four rounding modes:

1. Nearest: Round to nearest, rounding bit=1 and non-zero bits in the sticky bit cause rounding up.
2. +∞ : For positive numbers, rounding bit=1 or non-zero bits in the sticky bit cause rounding up, and no rounding is done for other numbers.
3. -∞ : For negative numbers, rounding bit=1 or non-zero bits in the sticky bit cause rounding up, and no rounding is done for other numbers.
4. Zero: Rounding bit and sticky bit are directly discarded.

21.2.5 Rounding modes of this chip

This chip supports six rounding modes that can be selected based on the configuration:

1. When the configuration is rnd=000, it rounds to the nearest value.
2. When the configuration is rnd=001, it rounds to zero.
3. When the configuration is rnd=010, it rounds up towards positive infinity.
4. When the configuration is rnd=011, it rounds up towards negative infinity.
5. When the configuration is rnd=100, it rounds up.
6. When the configuration is rnd=101, it rounds to nearest and ties away from zero.

21.2.6 Integer to Floating-Point Conversion

Below is a brief explanation of the idea and principle behind integer to floating-point conversion:

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

1. Sign bit:
   a) The highest bit is the sign bit for the floating-point number: fp_sign = bit(31);
   b) If the original data is negative, it needs to be converted to two's complement before the next step.
2. Exponent:
   a) Starting from the highest bit, look for the first "1" and record its position as n. The exponent value is n (27 in the table above).
   b) According to the encoding rules, the exponent is represented as fp_exp=n+127;
3. Mantissa:
   a) If n>23, take 24 bits starting from the (n-1)th bit, and round the remaining bits in the original data to determine the lowest bit of m. m is the mantissa.
   b) If n=23, take 24 bits starting from the (n-1)th bit directly as the mantissa.
   c) If n<23, take all the remaining bits starting from the (n-1)th bit, padding zeros to the low bits, as the mantissa.

21.2.7 Converting Floating-Point Numbers to Integers

1. Check whether the original data is a special number (0, NaN, ±∞);
2. Determine whether the conversion result overflows based on the exponent part e of the original data. If the exponent part e > 32+127, the conversion result overflows. Otherwise, proceed to the next step;
3. Take out the 23-bit decimal part m, add a leading 1 to the highest bit to form a 24-bit decimal m;
4. Shift the decimal m left by e bits, pad the highest (32-e) bits with 0s and the lowest e bits with 0s, resulting in 56 bits of data;
5. Right shift the result by 24 bits to obtain the final result.

21.2.8 Floating-Point Multiplication

1. Multiplication of special data:
   a) NaN multiplied by any number is NaN.
   b) 0 multiplied by infinity is NaN.
   c) 0 multiplied by any other number is 0.
2. Multiplication of normal data:
   a) The sign bit of the result is obtained by XOR-ing the sign bits of the two operands.
   b) The exponent part of the result is obtained by adding the exponent parts of the two operands and subtracting 127($e = e_0 + e_1 - 127$).
   c) The mantissa part of the result is obtained by multiplying the two mantissa parts (expanded to 24 bits each), resulting in a 48-bit data m. The result is checked for overflow. If overflow occurs, the exponent e is increased by 1. The exponent e is then checked for overflow. In either case, m is shifted right by 3 bits. If no overflow occurs, m is shifted right by 2 bits. The high 23 bits of m are taken as the mantissa of the result.
3. Overflow processing supports the 4 overflow handling modes specified in IEEE754, which can be configured through registers.

21.2.9 Floating-Point Addition

1. Checking for Special Numbers: If either operand is NaN, the result is NaN.
2. Normal Numbers:
   a) The sign bit of the result is determined by the sign bit of the operand with the larger absolute value.
   b) The exponents of the operands are compared. The difference in exponents, F, is calculated by subtracting the smaller exponent e_1 from the larger exponent e_0.
   c) The mantissa of the operand with the smaller exponent is shifted right by F bits. Then, the mantissas of both operands are expanded by one bit and added together.
   d) The exponent of the result is equal to e_0 if there is no overflow. If overflow occurs, the exponent is increased by 1, $\mathrm{e} = e_0 + 1$.
   e) The mantissa of the result is obtained by adding the mantissas of the operands. The higher 2 bits of the resulting mantissa are discarded. If overflow occurs, the result is shifted right by 1 bit (equivalent to dividing by 2). The resulting mantissa is the final result.

21.2.10 Butterfly Operation

$$X'_1(k) = X_1(k) + W_N^k X_2(k)$$
$$X'_2(k) = X_1(k) - W_N^k X_2(k)$$

Where $X_1(k)$、$X_2(k)$ are inputs, $X'_1(k)$、$X'_2(k)$ are the outputs of the butterfly operation.

If the original data is complex, then we have:

$$X_1(\mathrm{k}) = X_{1r}(k) + jX_{1i}(k)$$
$$X_2(\mathrm{k}) = X_{2r}(k) + jX_{2i}(k)$$
$$W_N^k = W_{Nr}^k + jW_{Ni}^k$$
$$X'_1(k) = X'_{1r}(k) + jX'_{1i}$$

Therefore, we can obtain:

$$X'_1(k) = X_1(k) + W_N^k X_2(k)$$
$$= (X_{1r}(k) + jX_{1i}(k)) + \left(W_{Nr}^k + jW_{Ni}^k\right) * (X_{2r}(k) + jX_{2i}(k))$$

Extracting the real and imaginary parts, we obtain:

$$X'_{1r} = X_{1r} + W_{Nr}^k X_{2r} + W_{Ni}^k X_{2i}$$
$$X'_{1i} = X_{1i} + W_{Nr}^k X_{2i} - W_{Ni}^k X_{2r}$$

Similarly,

$$X'_{2r} = X_{1r} - W_{Nr}^k X_{2r} - W_{Ni}^k X_{2i}$$
$$X'_{2i} = X_{1i} - W_{Nr}^k X_{2i} + W_{Ni}^k X_{2r}$$

21.2.11 IIR Filter

IIR (infinite impulse response) filter is the most common type of linear digital filter. Its output at a given time depends on both its input and previously computed input values.

IIR filter is recursive because the difference equation involves feedback. As a result, its stability time is longer than that of FIR filters, and the impulse response may have infinite width. Therefore, it is important to consider the stability of the filter.

The transfer function of IIR filter in z-domain is as follows:

$$H(z)=\frac{\sum_{k=0}^{M} b_k z^{-k}}{a_0+\sum_{k=1}^{N} a_k z^{-k}}=\frac{Y(z)}{X(z)}$$

The time-domain calculation formula (direct form I) of IIR filter is as follows:

$$y[n]=\sum_{k=0}^{M} b_k x[n-k]-\sum_{k=1}^{N} a_k y[n-k]$$

For a second-order IIR filter, where both M and N are 2 in the above equation, it is commonly designed with a_0 feedback coefficient of 1. Therefore, the above equation can be simplified as:

$$y[n]=b_0\times x(n)+b_1\times x(n-1)+b_2\times x(n-2)-a_1\times y(n-1)-a_2\times y(n-2)$$

Where x(n) is the current input data, x(n-1) is the previous input data (one unit delayed in time), and x(n-2) is the input data from two units ago (two units delayed in time); y(n) is the current output data, y(n-1) is the previous output data (one unit delayed in time), and y(n-2) is the output data from two units ago (two units delayed in time).

21.2.12 FIR Filters

FIR (Finite Impulse Response) filters are also known as finite-length unit-impulse response filters and are a type of non-recursive filter. FIR filters with constant coefficients are a type of LTI (linear and time-invariant) digital filter. The finite impulse response signifies that there is no feedback in the filter. The relationship between the output of an FIR filter with a length of N and an input time series x[n] is given by a finite convolution sum, which has the following form:

$$y[n]=\sum_{k=0}^{N} a[k]*x[n-k]$$

In the above equation, a[k] represents the filter coefficients, and x[n-k] denotes a delayed version of x[n] by k cycles.

Figure of Direct-Form FIR Filter:

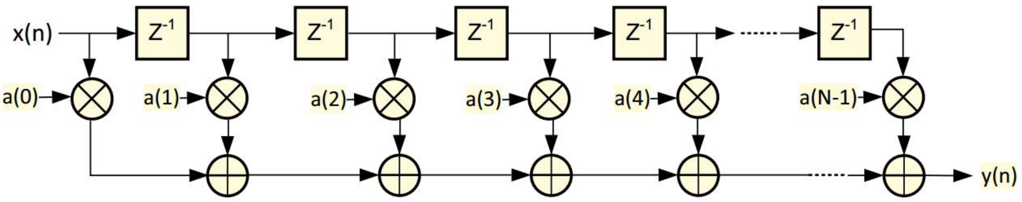


Figure 1-1: **Conventional Tapped Delay Line FIR Filter Representation**

The above equation represents an N-1 order FIR filter with N taps (coefficients). This filter is composed of N multipliers and N-1 accumulators. The input signal is time-varying, and the output of the FIR filter is the sum of the weighted (coefficients) inputs at each moment in time.

For example, a 4th order FIR filter has 5 coefficients. When implemented using a serial method, 5 multipliers and 4 adders are required for one calculation:

$$y[n] = a0*x[n] + a1*x[n-1] + a2*x[n-2]+ a3*x[n-3]+ a4*x[n-4]$$

21.2.13 Linear Interpolation

Given n points: $(1, y_1), \ldots, (n, y_n)$, find y(i) for x as the i-th output data abscissa.

Formula: ,[x]=x (round down)

Calculate the output data point by point, with the interpolation position increasing by step in each loop:

```
for i=1:1:out_len
    n=floor(t);
    y[i]= x[n]+ (t-n) * (x[n+1]-x[n]);
    t=t+step;
End
```

21.2.14 Lagrange Interpolation

Given n points: $(1, y_1), \ldots, (n, y_n)$, find y(i) for x as the i-th output data abscissa.

3-point Lagrange interpolation, which utilizes a quadratic function for each small interval.

Formula:

$$cur = round(x);$$

$$y(i) = \frac{x - cur}{2} * \left[(x - (cur + 1)) * y_{cur-1} + (x - (cur - 1) * y_{cur+1} \right] - (x - (cur - 1)) * (x - (cur + 1)) * y_{cur}$$

Calculate the output data point by point, with the interpolation position increasing by step in each loop:

```
for i=1:1:out_len
    n=floor(t);
    y[i]=(t-n)*0.5*((t-n-1)*x[n-1]+(t-n+1)*x[n+1]) - (t-n+1)*(t-n-1)*x[n]
    t=t+step;
End
```

21.3 Arithmetic instruction

21.3.1 Integer to Floating Point (int2fp/int2fp_dma)

21.3.1.1 Single Data Conversion

Integer to Floating Point

This mode converts 32-bit (or 24-bit) signed integers input from the register into 32-bit single-precision floating-point format, and outputs to the register which can be read by the CPU.

Normalization processing can be applied to the converted floating-point number by dividing it by 2^n, where n can be configured from 0 to 32. If it is configured as 0, it means no normalization.

By default, the input data register is set up to receive a 32-bit signed integer as input, but for special applications, if the input integer is a 24-bit signed number, MAC_CTL0[28] needs to be configured as 1 and this mode needs to be selected.

Software configuration process:

1) Set MAC_CTL0[0] =1 to select the integer-to-floating-point single-point mode;
2) Set MAC_CTL0[16:14] to select the floating-point rounding mode;
3) Configure MAC_CTL0[21:17] to select the normalization level;
4) For 24-bit input data, set MAC_CTL[0] = 1;
5) Write the raw integer to be converted to register MAC_IN0;
6) Read register MAC_OUT0 to obtain the converted floating-point value.

Once the software has configured the mode, conversion is carried out by writing data to the MAC_IN0 register. The calculation result is then saved to the MAC_OUT0 register, so conversion results can be obtained simply by reading the result register after setting the original integer value.

Note that if the software requires continuous conversion, the above mode configuration needs to be executed only once. Before switching modes, each time just write the value to MAC_IN0, and conversion will be executed automatically.

21.3.1.2 Continuous Conversion of Multiple Data

This mode supports converting a whole section of integers in SRAM into floating-point numbers and writing the results back to SRAM. It should be noted that the raw data stored in SRAM are signed integers (either 32-bit signed integers or 24-bit signed integers). For special applications where a 24-bit integer takes up 4 bytes, with the valid data stored in the low 3-byte position and the highest byte being invalid, MAC_CTL [0] needs to be configured as 1.

The target address can be configured to be the same as the source data address, which will overwrite the original data after the conversion, saving SRAM space. Alternatively, if the target address is configured to be different from the source data address, the converted data will be written to another area in SRAM.

Software configuration process:

1) Configure register MAC_CTL0[1]=1 to select the integer-to-floating-point DMA mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode for the floating-point numbers;
3) Configure MAC_CTL0[21:17] to select the normalization level;
4) Configure register DMA_SRBADR to select the source data DMA start address;
5) Configure register DMA_TRBADR to select the target data DMA start address;
6) Configure register DMA_LEN to select the DMA length;
7) Set MAC_CTL1[0]=1 to initiate the conversion;
8) Wait for the flag bit MAC_FLG[3] to be set to 1, and the CPU can obtain the converted results from SRAM;

After converting the specified length of data, a completion flag will be generated. If the interrupt enable signal is configured, a completion interrupt will be generated. The software needs to clear the flag bit in the interrupt.

21.3.2 Floating-Point to Integer Conversion (fp2int/fp2int_dma)

21.3.2.1 Single Data Conversion

Floating to Integer Point

This mode converts a 32-bit floating-point number input from a register to a 32-bit signed integer format, outputs it to a register, and can be read by the CPU. It supports scaling the floating-point number by 2^n before conversion. Here, n can be configured from 0 to 32, and when it is configured as 0, it means not to scale.

Software configuration process:

1) Configure register MAC_CTL0 [2] =1 to select floating-point to integer single point mode.
2) Configure register MAC_CTL0[16:14] to select the rounding mode for the floating-point number
3) If you need to scale the floating-point number before conversion, configure MAC_CTL0[26:24]=n
4) Write the original floating-point number to be converted to register MAC_IN0.
5) Read register MAC_OUT0 to obtain the converted integer value.

After the software configures the mode, the conversion will start as soon as the MAC_IN0 register is configured. The result of the conversion is saved to register MAC_OUT0. Therefore, once the original value is configured, the converted result can be obtained by reading the result register directly.

Note that if the software needs to perform multiple consecutive conversions, the above mode configuration only needs to be executed once. Before switching modes, as long as MAC_IN0 register is written, the conversion will be executed.

21.3.2.2 Continuous Conversion of Multiple Data

This mode supports converting a whole section of floating-point numbers in SRAM into integers, with the result being written back into SRAM. The target address can be configured to be the same as the source data address, which will overwrite the original data after the conversion to save SRAM space. Otherwise, configuring the target address to be different from the source data address will write the converted data into another area of SRAM.

Software configuration process:

1) Configure register MAC_CTL0[3]=1 to select the floating-point to integer DMA mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode of the floating-point number;
3) If it is necessary to amplify the floating-point number before conversion, configure MAC_CTL0[26:24]=n;
4) Configure register DMA_SRBADR to select the source data DMA starting address;
5) Configure register DMA_TRBADR to select the target data DMA starting address;
6) Configure register DMA_LEN to select the DMA length;
7) Configure register MAC_CTL1[0]=1 to start the conversion;
8) Wait for the flag bit MAC_FLG [4], and the CPU can obtain the converted result from SRAM.

After converting the specified length of data, a completion flag will be generated, and if the interrupt enable signal is configured, a completion interrupt will be generated. The software needs to clear the flag bit in the interrupt.

21.3.3 Floating-Point Multiplication (fp_mult)

Floating-Point Multiplication

This mode performs a floating-point multiplication on the data in input registers MAC_IN0 and MAC_IN1. Four modes are supported for input and output data formats:

1) Input integer, output integer
2) Input integer, output floating-point number
3) Input floating-point number, output integer
4) Input floating-point number, output floating-point number

If the input is an integer, the hardware will first convert the integer to a floating-point format before performing the floating-point multiplication. The operation produces a floating-point format product. If the output format is chosen as integer, the hardware will convert the floating-point format product to integer format.

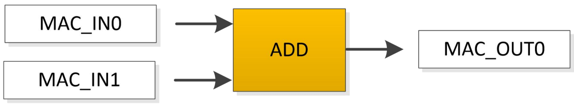
Software configuration process:

1) Configure register MAC_CTL0[4]=1 to select the floating-point multiplication mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode of the floating-point number;
3) Configure register MAC_CTL0[13:12] to select the input and output data mode;
4) Configure registers MAC_IN0 and MAC_IN1 (multiplicand and multiplier);
5) Wait for the flag bit MAC_FLG[9]=1, indicating that multiplication is complete;
6) Read register MAC_OUT0 to obtain the product.

In this mode, each time the MAC_IN1 register is configured, the multiplication operation is automatically started. Therefore, the software should first configure the MAC_IN0 register and then configure the MAC_IN1 register. Additionally, each time the MAC_IN1 register is configured, the flag bit of the previous operation is automatically cleared until the current calculation is complete, and the flag bit is set again.

Since input and output data formats are different, data conversion is necessary, so the time required for each floating-point multiplication operation depends on the selected mode.

21.3.4 Floating-Point Addition (fp_add)



Floating-Point Addition

This mode performs a floating-point addition on the data in input registers MAC_IN0 and MAC_IN1. The data format of both input registers must be in floating-point format, and the output result is also in floating-point format.

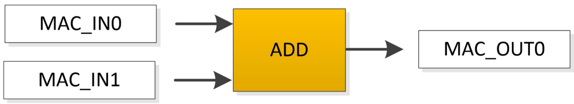
Software configuration process:

1) Configure register MAC_CTL0[5]=1 to select the floating-point addition mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode of the floating-point number;
3) Configure registers MAC_IN0 and MAC_IN1 (addend and augend);

4) Read register MAC_OUT0 to obtain the sum.

In this mode, every time the MAC_IN1 register is configured, the addition operation is automatically started. Therefore, the software should first configure the MAC_IN0 register and then configure the MAC_IN1 register.

21.3.5 Floating-Point Subtraction (fp_sub)



Floating-Point Subtraction

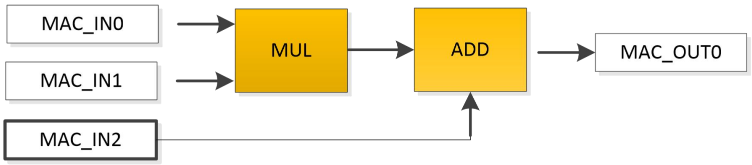
This mode performs a floating-point subtraction on the data in input registers MAC_IN0 and MAC_IN1. The data format of both input registers must be in floating-point format, and the output result is also in floating-point format.

Software configuration process:

1) Configure register MAC_CTL0[6]=1 to select the floating-point subtraction mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode of the floating-point number;
3) Configure registers MAC_IN0 and MAC_IN1 (minuend and subtrahend);
4) Read register MAC_OUT0 to obtain the difference.

In this mode, every time the MAC_IN1 register is configured, the subtraction operation is automatically started. Therefore, the software should first configure the MAC_IN0 register and then configure the MAC_IN1 register.

21.3.6 Floating-point Multiply-Addition Operation (fp_mlad)



Floating-point Multiply-Addition Operation

This mode performs a floating-point multiplication on the data in input registers MAC_IN0 and MAC_IN1, followed by addition with MAC_IN2. The resulting output is stored in MAC_OUT0. The input data format for the input registers must be in floating-point format, and the output result is also in floating-point format.

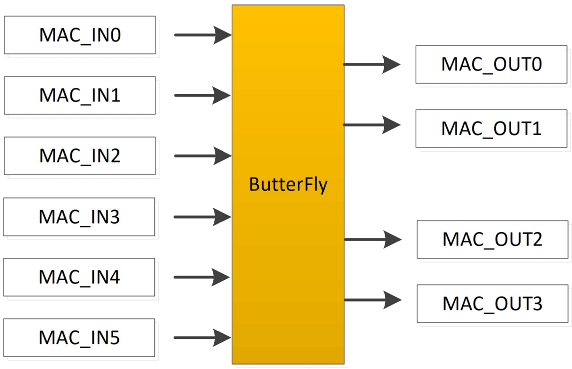
Software configuration process:

1) Configure register MAC_CTL0[7]=1 to select floating-point multiply-add mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode for floating-point numbers;
3) Configure registers MAC_IN0, MAC_IN1, and MAC_IN2 (multiplier, multiplicand, and addend);
4) Read register MAC_OUT0 to obtain the result of the multiply-add operation.

In this mode, each time the MAC_IN2 register is configured, the multiply-add operation is automatically started. Therefore, the software should first configure the MAC_IN0 and MAC_IN1 registers before configuring the MAC_IN2 register.

21.3.7 Floating-point Butterfly Operation (Single) (btfy/btfy_dma)

21.3.7.1 Single Data Conversion



Butterfly Operation

This mode performs a single butterfly operation on two original data inputs and one parameter input in the input registers and stores the result in a register for the CPU to read. In this mode, the original data inputs and the parameter input must be in floating-point format.

$$X'_{1r} = X_{1r} + W_{Nr}^{k}X_{2r} + W_{Ni}^{k}X_{2i}$$
$$X'_{1i} = X_{1i} + W_{Nr}^{k}X_{2i} - W_{Ni}^{k}X_{2r}$$
$$X'_{2r} = X_{1r} - W_{Nr}^{k}X_{2r} - W_{Ni}^{k}X_{2i}$$
$$X'_{2i} = X_{1i} - W_{Nr}^{k}X_{2i} + W_{Ni}^{k}X_{2r}$$

One butterfly operation is calculated using the formula: X_{1r}、 X_{1i} and X_{2r}、 X_{2i} are the real and imaginary parts of the two original data inputs,W_{Nr}^{k}、 $W_{Ni}^{k}$are the real and imaginary parts of the parameter input.

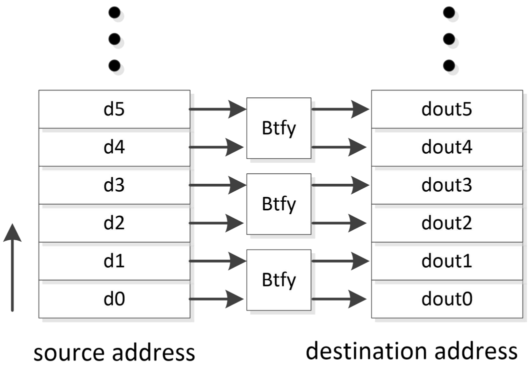
Software configuration process:

1) Configure register MAC_CTL0[8]=1 to select butterfly single mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode for floating-point numbers;
3) Configure registers MAC_IN0/1/2/3/4/5 to input data;
    - MAC_IN0=X_{1r}
    - MAC_IN1=X_{1i}
    - MAC_IN2=X_{2r}
    - MAC_IN3=X_{2i}
    - MAC_IN4=W_{Nr}^{k}
    - MAC_IN5=W_{Ni}^{k}
4) Configure register MAC_CTL1[0]=1 to start the butterfly operation;
5) Wait for the flag bit MAC_FLG [5] to be set and read the results from MAC_OUT0/1/2/3.
    - MAC_OUT0=X'_{1r}
    - MAC_OUT1=X'_{1i}
    - MAC_OUT2=X'_{2r}
    - MAC_OUT3=X'_{2i}

If interrupt enable signal is configured, an interrupt is generated after the single butterfly operation. Each time the register is configured to start the butterfly operation, the flag bit is automatically cleared.

21.3.7.2 Multiple Data Continuous Conversion

This mode calculates the butterfly operation on multiple sets of data in SRAM.



Two adjacent numbers in the same address are subject to a butterfly operation, and the result is written back to SRAM. As shown in the figure below, from a low address to a high address, two data are taken from the source data area for calculation and then written into the destination SRAM after each butterfly operation.

This mode supports having the source DMA address and the target DMA address be the same to overwrite the source data, which saves SRAM.

Software configuration process:

1) Configure register MAC_CTL0[9]=1 to select butterfly DMA mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode for floating-point numbers;
3) Configure register DMA_SRBADR to select the starting address of the source data's real part for DMA transfer;
4) Configure register DMA_SIBADR to select the starting address of the source data's imaginary part for DMA transfer;
5) Configure register DMA_PRBADR to select the starting address of the parameter table's real part for DMA transfer;
6) Configure register DMA_PIBADR to select the starting address of the parameter table's imaginary part for DMA transfer;
7) Configure register DMA_TRBADR to select the starting address of the target data's real part for DMA transfer;
8) Configure register DMA_TIBADR to select the starting address of the target data's imaginary part for DMA transfer;
9) Configure register DMA_LEN to select the number of data points (a power of 2);
10) Configure register MAC_CTL1[0]=1 to start the calculation;
11) Wait for the flag bit MAC_FLG [6] and retrieve the calculation result from SRAM.

When all the data has been calculated, a flag bit is generated. If interrupt enable is configured, an interrupt is generated. The software needs to clear the flag bit.

21.3.7.3 FFT (fp_fft)

In this mode, the hardware automatically performs a radix-2 FFT calculation on the specified data (in floating-point format) in SRAM, and stores the result back into SRAM while generating a completion flag. The data length can be configured to 64, 128, 256, 512, or 1024.

Since FFT operations require a parameter table, to speed up the FFT process, for 128/64-point FFTs, the hardware has built-in a parameter table allowing for the direct table lookup instead of accessing SRAM every time.

This mode supports having the source data and target data addresses be the same, allowing for data overwriting and reducing SRAM usage.

Software configuration process:

1) Configure register MAC_CTL0[10]=1 to select FFT mode;
2) Configure register MAC_CTL0[16:14] to select the rounding mode for floating-point numbers;
3) If using the internal parameter table, configure MAC_CTL0[27]=1; otherwise, the table is stored in SRAM and configure the DMA_PRBADR and DMA_PIBADR registers to select the starting addresses for the table's real part and imaginary part, respectively;
4) Configure the DMA_SRBADR register to select the starting address for the source data's real part for DMA transfer; configure the DMA_SIBADR to select the starting address for the source data's imaginary part for DMA transfer;
5) Configure the DMA_TRBADR register to select the starting address for the target data's real part for DMA transfer; configure the DMA_TIBADR register to select the starting address for the target data's imaginary part for DMA transfer;
6) Configure the DMA_LEN register to select the data length;
7) Wait for the completion flag bit MAC_FLG [7] to become 1, indicating that the FFT is complete.

21.3.7.4 Bit-Reverse Moving of Data (bitrev_move)

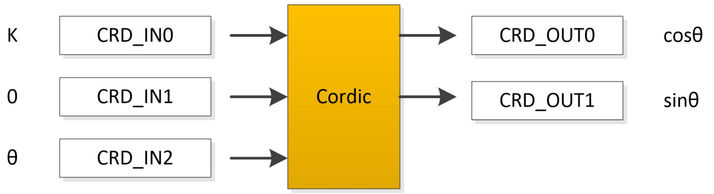
In this mode, the hardware automatically reorders a section of raw data in SRAM according to the bit-reverse algorithm and writes it to a specified location in SRAM.

The data length supports 4, 8, 16, 32, 64, 128, 256, 512, or 1024, and each data occupies 4 bytes in SRAM, stored aligned to the word in SRAM.

Software configuration process:

1) Configure register MAC_CTL0[11]=1 to select bit-reverse mode;
2) Configure the DMA_SRBADR register to select the starting address for the source data for DMA transfer;
3) Configure the DMA_TRBADR register to select the starting address for the target data for DMA transfer;
4) Configure the DMA_LEN register to select the data length;
5) Wait for the completion flag bit MAC_FLG [8] to become 1, indicating completion.

21.3.8 Sine-Cosine Calculation (sin_cos)



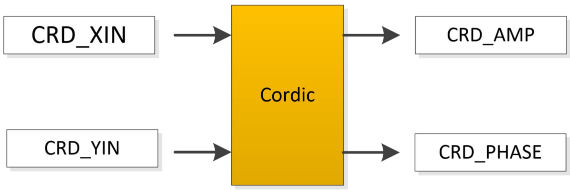
Sine-Cosine Calculation

This mode calculates the sin and cos functions given an input angle. The hardware uses the cordic rotation mode and configures $X_0 = \frac{1}{P} = K = 0.60725$，$Y_0$=0，$Z_0 = \theta$ to output the result$[\cos\theta, \sin\theta, 0]$.

Software configuration process:

1) Configure the CRD_ANGLE register for the input angle (register value = radian*(2^32)/(2*pi));
2) Configure CRD_CTL[1]=1 to start the operation;
3) Wait for the completion flag bit CRD_FLG [1], and read the CRD_COSINE/CRD_SINE registers to obtain the result.

21.3.9 Root-Mean-Square and Arctangent Calculation (fp_sqrt/fp_atan)



Root-Mean-Square

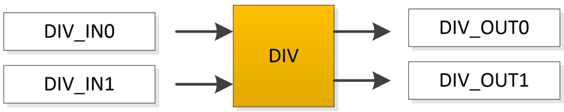
This mode calculates the root-mean-square of the input data CRD_XIN and CRD_YIN. The hardware uses the cordic vector mode, configures X_0=X， Y_0=Y， $Z_0 = 0$ to output $\left[P\sqrt{X_i^2 + Y_i^2}，\ 0，\ \arctan\left(\frac{Y_i}{X_i}\right)\right]$.

Software configuration process:

1) Configure the CRD_XIN register for the value of X;
2) Configure the CRD_YIN register for the value of Y;
3) Configure CRD_CTL[0]=1 to start the operation;
4) Wait for the completion flag bit CRD_FLG [0], and read the CRD_AMP/CRD_PHASE registers to obtain the result. CRD_AMP= $\sqrt{X_0^2 + Y_0^2}/4*P$， $CRD_{PHASE} = \arctan\left(\frac{Y_0}{X_0}\right) *2\hat{}31/pi$

If interrupt enable is configured, an interrupt is generated after the computation is completed.

21.3.10 Floating-Point Divider (fp_div)



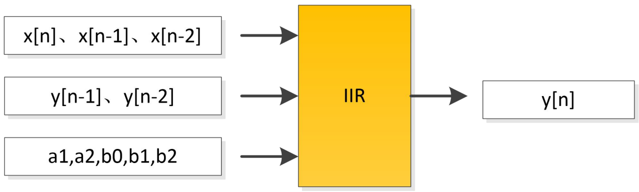
Division

This mode performs a floating-point division operation, taking in two floating-point numbers and outputting the quotient and remainder.

Software configuration process:

1) Configure the dividend into the DIV_IN0 register;
2) Configure the divisor into the DIV_IN1 register;
3) Configure MAC_CTL1[2]=1 to start the division operation;
4) Wait for the completion flag bit MAC_FLG[4], and read the DIV_OUT0 register to obtain the quotient.

21.3.11 IIR Filter



IIR Calculation

This mode performs an IIR filter calculation. Given floating-point inputs a_1、a_2、b_0、b_1、b_2、y[n-2]，y[n-1]，x[n]，x[n-1]，x[n-2], the IIR operation is executed and the output y[n] is produced.

21.3.11.1 Single Data Conversion

1) Configure MAC_CTL0[4:0]=13 (IIR_ONCE_MODE) to select IIR single mode;
2) Configure MAC_CTL0[9:7] to select the rounding mode for floating-point numbers;
3) Configure MAC_IN01/2/3/4/5, DIV_IN0/1, and MAC_OUT0/1 registers to input data.

(Note: MAC_IN0 must be configured last. After MAC_IN0 is written, hardware will automatically start the IIR operation)

- MAC_IN1=x[n-1]
- MAC_IN2=x[n-2]
- MAC_IN3=b_0
- MAC_IN4=b_1
- MAC_IN5=b_2
- DIV_IN0=a_1
- DIV_IN1=a_2
- MAC_OUT0=y[n]
- MAC_OUT1=y[n-1]
- MAC_IN0= x[n]

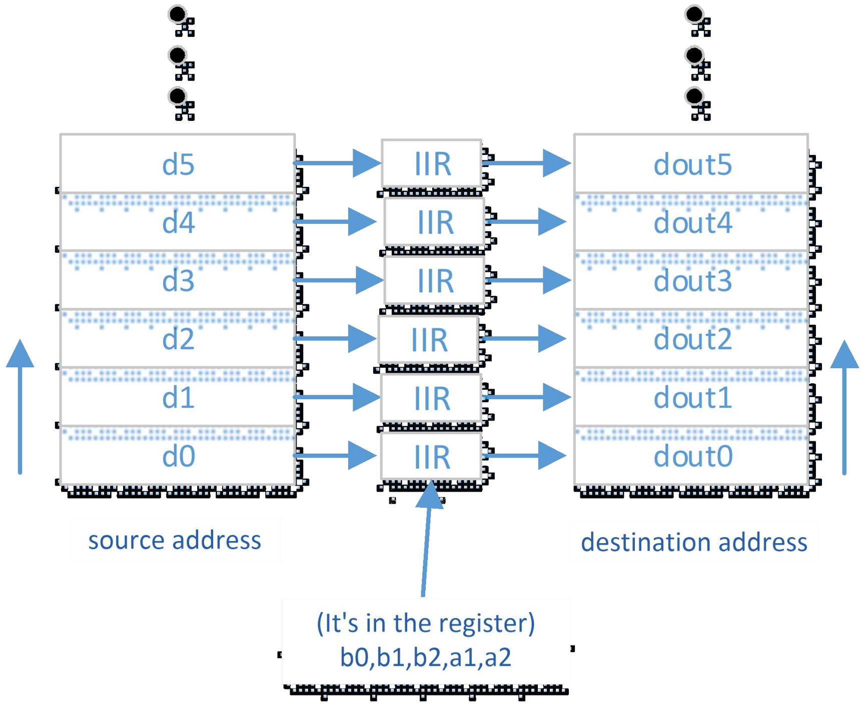
4) Wait for the flag bit MAC_FLG[10] (IIR_ONCE_DONE) to become set, and read the result from MAC_OUT0.
   - MAC_OUT0=y[n]

If interrupt enable is configured (FFT_IE [10].IIR_ONCE_IE=1), an interrupt is generated after the IIR operation. The software needs to clear the flag bit.

(Note: IIR uses DIV_IN0/1 to configure the parameters a1 and a2. DIV_IN0/1 cannot be changed during the IIR operation. Therefore, IIR and division cannot be used simultaneously.)

21.3.11.2 Multi-Data Continuous Conversion

This mode performs IIR operation on multiple sets of data in SRAM. Three adjacent numbers are taken from the source data area and subjected to IIR calculation.



The resulting output is then written back into the target SRAM area. As shown in the diagram, the data is taken from the source area from low to high addresses.

Software configuration process:

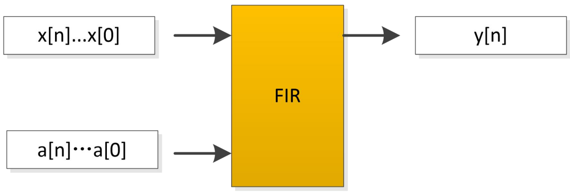
1) Configure MAC_CTL0[4:0]=14 to select IIR DMA mode;
2) Configure MAC_CTL0[9:7] to select the rounding mode for floating-point numbers;
3) Configure MAC_IN1/2/3/4/5, DIV_IN0/1, MAC_OUT0/1 to specify the input data;
   - MAC_IN1=x[n-1]
   - MAC_IN2=x[n-2]
   - MAC_IN3= b_0
   - MAC_IN4= b_1
   - MAC_IN5= b_2
   - DIV_IN0= a_1
   - DIV_IN1= a_2
   - MAC_OUT0=y[n]
   - MAC_OUT1=y[n-1]
4) Configure DMA_SRBADR as the starting address for data x[n], DMA_TRBADR as the target address for data y[n], DMA_LEN as the data length, and DMA reads SRAM stepping as MAC_CTL1[3:1];
5) Configure MAC_CTL2 [0]=1 (DMA_EN) to start the operation (It is recommended to clear the bit first before

writing).

6) Wait for the IIR_DMA_DONE flag bit MAC_FLG[11] to be set, and read the target SRAM address to obtain y[n].

If interrupt enable signal is configured (FFT_IE [11].IIR_DMA_IE=1), an interrupt is generated after the IIR DMA operation. The flag bit needs to be cleared by software.

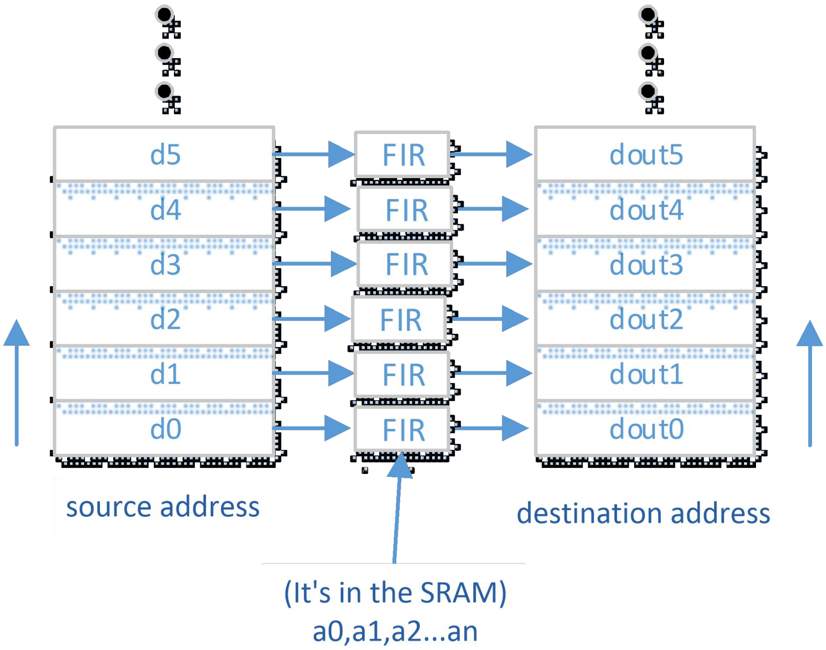
21.3.12 FIR Filter

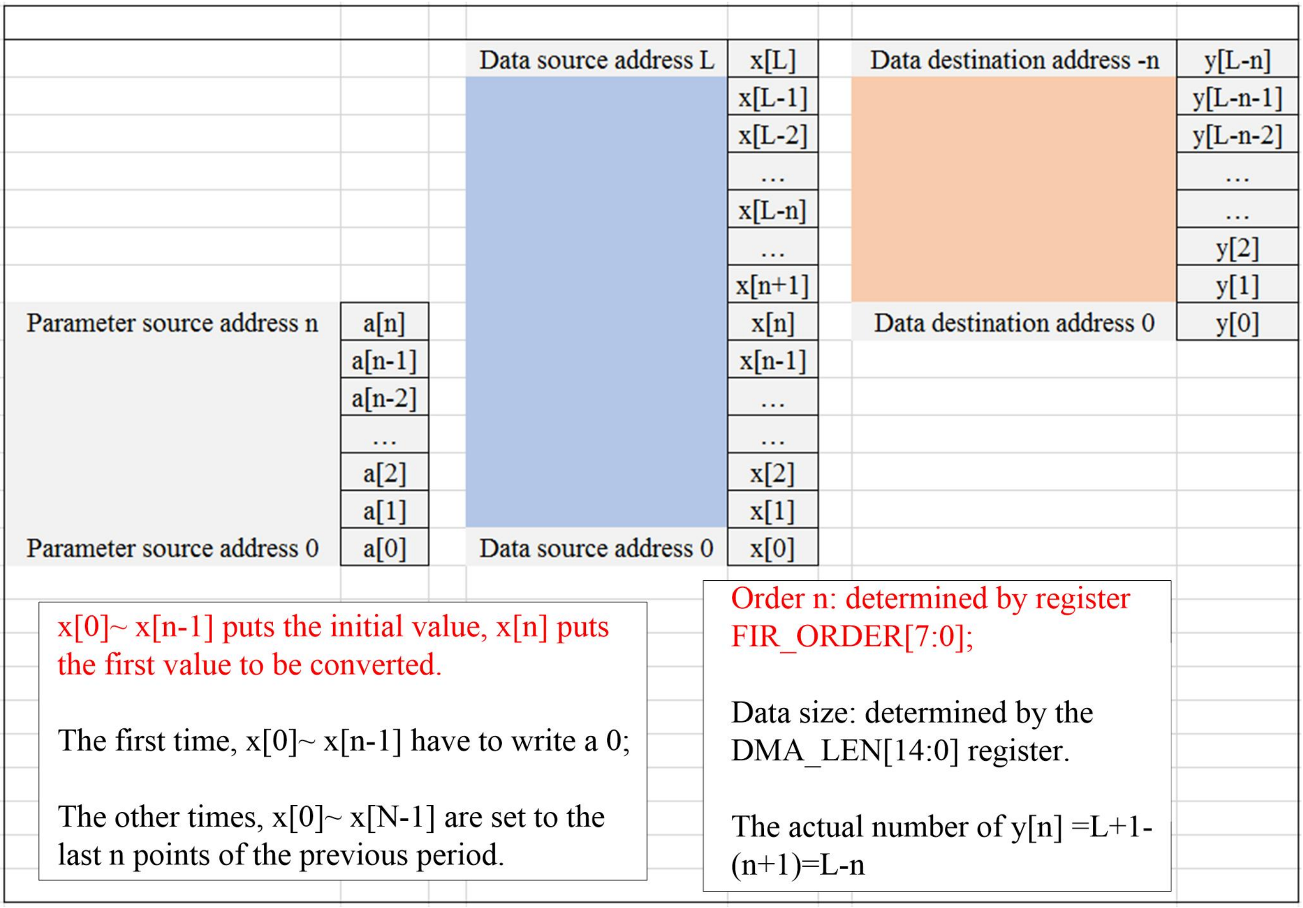


FIR Calculation

This mode performs FIR filtering, using the input floating-point data x[n]…x[0] and parameters a[n]…a[0] to perform FIR operation and produce output y[n].

This mode performs FIR operation on multiple sets of data in SRAM. N adjacent numbers are taken from the source data area and subjected to FIR calculation. The resulting output is then written back into the target SRAM area.





x[0]~ x[n-1] puts the initial value, x[n] puts the first value to be converted.

The first time, x[0]~ x[n-1] have to write a 0;

The other times, x[0]~ x[N-1] are set to the last n points of the previous period.

Order n: determined by register FIR_ORDER[7:0];

Data size: determined by the DMA_LEN[14:0] register.

The actual number of y[n] =L+1-(n+1)=L-n

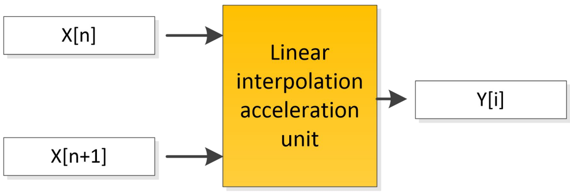
As shown in the diagram, the data is taken from the source area from low to high addresses.

Software configuration process:

1) Configure MAC_CTL0[4:0]=15 to select FIR DMA mode;
2) Configure MAC_CTL0[9:7] to select the rounding mode for floating-point numbers;
3) Configure the FIR order, DMA_SRBADR as the starting address for data x[n], DMA_PRBADR as the starting address for parameter a, DMA_TRBADR as the target address for data y[n], DMA_LEN as the data length, and DMA reads SRAM stepping as MAC_CTL1[3:1];
4) Configure MAC_CTL2 [0] =1 (DMA_EN) to start the operation (It is recommended to clear the bit first before writing).
5) Wait for the FIR_DMA_DONE flag bit MAC_FLG[11] to be set, and read the target SRAM address to obtain y[n].

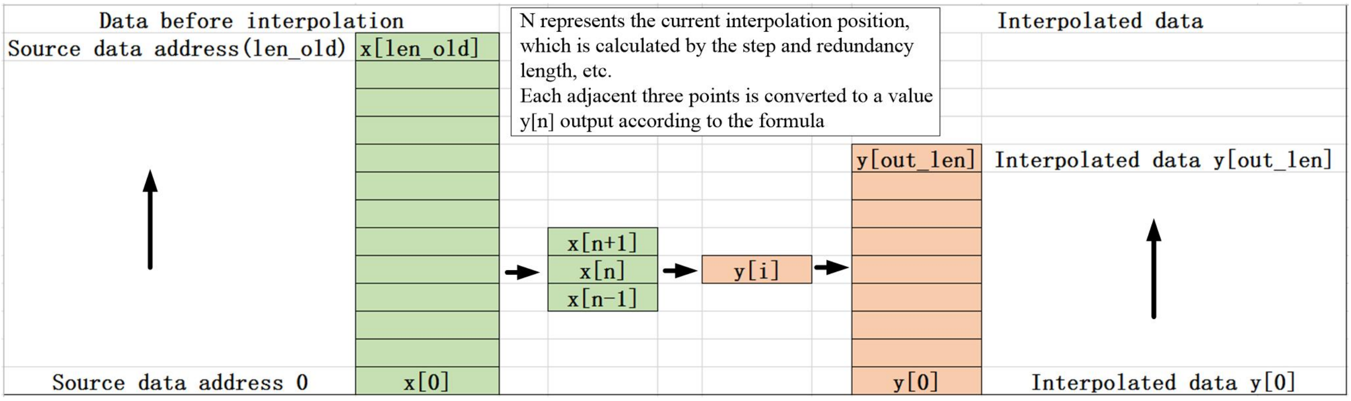
If interrupt enable signal is configured (FFT_IE [11].FIR_DMA_IE=1), an interrupt is generated after the FIR DMA operation. The flag bit needs to be cleared by software. (Note: dma_len cannot be less than fir_order, dma_len >=fir_order)

21.3.13 Linear Interpolation



This mode implements a linear interpolation acceleration algorithm. Every two adjacent points produce an output y[i] value based on a formula.

This mode performs linear interpolation operation on multiple sets of data in SRAM. Two adjacent numbers are taken from the source data area and subjected to linear interpolation calculation. The resulting output is then written back into the target SRAM area.



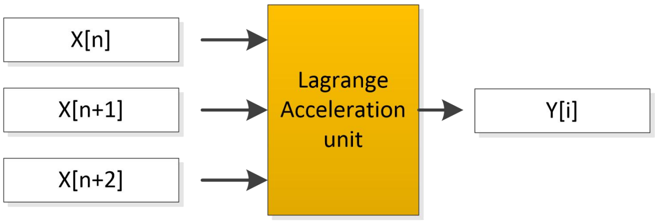
As shown in the above figure, from low to high address, every three numbers are extracted from the source data area, the linear interpolation calculation is performed, and then the result is written to the target SRAM.

Software configuration process:

1) Software calculates various parameters, including the actual available input data length 'len', the length of data after interpolation 'out_len', and the current interpolation position 't';
2) Configure MAC_CTL0[4:0]=16 (LINE_INTP) to select linear interpolation mode;
3) Configure MAC_CTL0[9:7] to select the rounding mode for floating-point numbers;
4) Configure DMA_SRBADR as the starting address for data x[n], DMA_TRBADR as the target address for data y[i];
5) Configure MAC_IN5[31:0] (the last data from the previous cycle);
6) Configure MAC_CTL1[0] to select whether to use the last data from the previous cycle;
7) Configure INTP_LEN[12:0] to select the actual available input data length;
8) Configure DMA_LEN[10:0] to select the length of data after interpolation;
9) Configure INTP_LOC[31:0] to select the current interpolation position;
10) Configure INTP_STEP[31:0] to select the step;
11) Configure MAC_CTL2 [0]=1 (DMA_EN) to start the operation (It is recommended to clear the bit first before writing).
12) Wait for the LINE_INTP_DONE flag bit FFT_FLG[13] to be set, and read the target SRAM address to obtain y[n].
13) Read INTP_LOC [31:0] and set t=INTP_LOC [31:0]. Software calculates the redundant length 'red_len' as len-t+step after this interpolation operation is complete.
14) Depending on the specific situation, software can choose whether to save the last data. If necessary, write the value to MAC_IN5 [31:0] (the last data from the previous cycle).

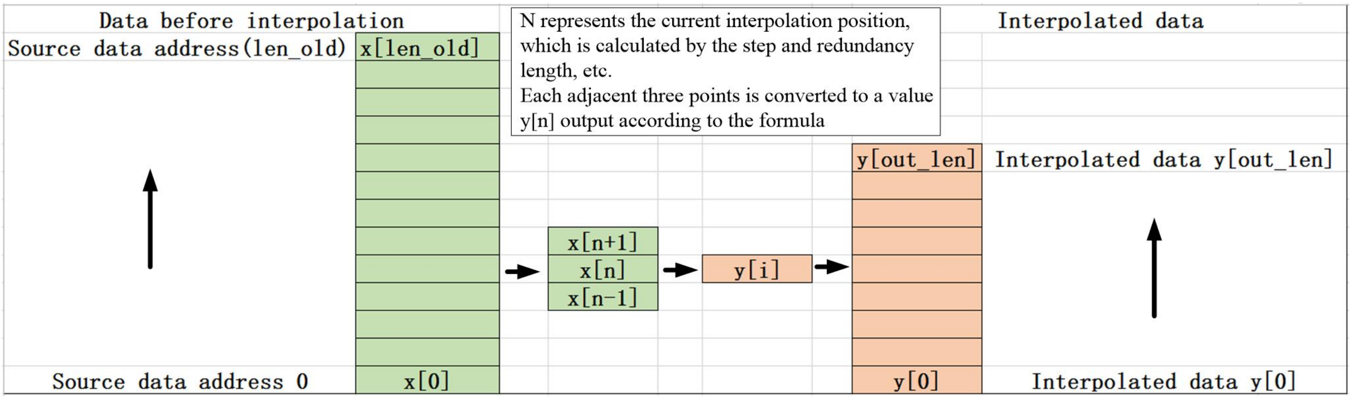
If interrupt enable signal is configured (FFT_IE [13].LINE_INTP_IE=1), an interrupt is generated after the linear interpolation operation. The flag bit needs to be cleared by software.

21.3.14 Lagrange Interpolation



This mode implements a Lagrange interpolation acceleration algorithm. Every three adjacent points produce an output y[i] value based on a formula.

This mode performs Lagrange interpolation operation on multiple sets of data in SRAM.



Three adjacent numbers are taken from the source data area and subjected to Lagrange interpolation calculation. The resulting output is then written back into the target SRAM area. As shown in the diagram, the data is taken from the source area from low to high addresses.

Software configuration process:

1) Software calculates various parameters, including the actual available input data length ‘len’, the length of data after interpolation ‘out_len’, and the current interpolation position ‘t’;
2) Configure MAC_CTL0[4:0]=17 (LAGR_INTP) to select Lagrange interpolation mode;
3) Configure MAC_CTL0[9:7] to select the rounding mode for floating-point numbers;
4) Configure DMA_SRBADR as the starting address for data x[n], DMA_TRBADR as the target address for data y[i];
5) Configure MAC_IN5[31:0] (the last data from the previous cycle);
6) Configure MAC_CTL1[0] to select whether to use the last data from the previous cycle;
7) Configure INTP_LEN[12:0] to select the actual available input data length;
8) Configure DMA_LEN[9:0] to select the length of data after interpolation;
9) Configure INTP_LOC[31:0] to select the current interpolation position;
10) Configure INTP_STEP[31:0] to select the step;
11) Configure MAC_CTL2 [0] =1 (DMA_EN) to start the operation (It is recommended to clear the bit first before writing).
12) Wait for the LAGRINTP_DONE flag bit MAC_FLG[14] to be set, and read the target SRAM address to

obtain y[n];

13) Read INTP_LOC [31:0] and set t=INTP_LOC [31:0]. Software calculates the redundant length ‘red_len’ as len-t+step after this interpolation operation is complete.
14) Depending on the specific situation, software can choose whether to save the last data. If necessary, write the value to MAC_IN5 [31:0] (the last data from the previous cycle).

If interrupt enable signal is configured (FFT_IE [14].LAGR_INTP_IE=1), an interrupt is generated after the Lagrange interpolation operation. The flag bit needs to be cleared by software.

21.4 Implementation Details

21.4.1 Integer to Floating-Point Conversion

Since the input integer width is 32 bits and the resulting floating-point number has a very small range of values for the exponent, there will be no overflow. Moreover, when normalizing, the highest value supported is 32, so no overflow processing is required.

Normalization involves subtracting n from the exponent and leaving the fractional part unchanged.

21.4.2 Floating-Point to Integer Conversion

Before converting, the floating-point number needs to be magnified to make the integer part large enough to ensure more effective bits.

Magnifying by 2^n involves adding n to the exponent and leaving the fractional part unchanged.

21.4.3 FFT Explanation

Software only needs to store the original data in SRAM in advance and configure the DMA starting address, target address, and data length. Hardware automatically performs computation by fetching data and produces the result of the butterfly. If the data length is 64, there will be 32 butterfly operations per stage (64/2=32), and a total of 6 stages in $\log_2 64 = 6$. Therefore, hardware automatically computes 32*6 butterfly operations and saves the result to the specified location in SRAM, generating a completion flag or interrupt.

For each butterfly operation, the hardware reads the SRAM four times to obtain the original data (the imaginary part of the original input data is zero in the first round), writes to the SRAM four times to save the butterfly operation result, and if the parameter table is stored in SRAM, reads the SRAM twice to obtain the parameters. Therefore, the number of SRAM cycles required to complete data of a specified length is:

$$k_0 = \frac{\mathrm{n}}{2} * log_2 n * (4+4) - \frac{\mathrm{n}}{2} * 2, \text{ Hardware parameter table hardening.}$$

$$k_0 = \frac{\mathrm{n}}{2} * log_2 n * (4+4+2) - \frac{\mathrm{n}}{2} * 2, \text{ Parameter table stored in SRAM.}$$

Data points	Read/write SRAM cycles	
	Hardware table	SRAM table
64	1472	1856
128	3456	4352

256	7936	9984
512	17920	22528
1024	39936	50176

Number of cycles required to read/write SRAM to complete the FFT

A single butterfly operation can be completed in as little as 10 clock cycles (if the parameter table is hardware-solidified) or 12 clock cycles (if the parameter table is stored in SRAM) when performing SRAM read/write and butterfly computation.

Clock cycle	1	2	3	4	5	6	7	8	9	10	11	12
Hardware behavior	R	R	R	R	R	R						
		H	H	H	H	H	H					
								Mul	Mul	Mul	MUl	
								Add	Add	Add	Add	
								Add	Add	Add	Add	
									R	R	R	R

So the total time is:

$k_0 = \frac{n}{2} * log_2 n * 10$，Hardware parameter table hardening.

$k_0 = \frac{n}{2} * log_2 n * 12$，Parameter table stored in SRAM.

Data points	FFT clock cycles		FFT time (us) 16M clock	
	Hardware table	SRAM table	Hardware table	SRAM table
64	1920	2304	120	144
128	4480	5376	280	336
256	10240	12288	640	768
512	23040	27648	1440	1728
1024	51200	61440	3200	3840

To achieve the fastest completion time for FFT, add the time to configure DMA and other mode registers to the calculation time shown in the table. The time in the table is based on the assumption that the SRAM is idle and accessed only by the FFT module. In reality, there may be other modes competing for access to the SRAM, which could result in an extended FFT time.

21.4.4 Explanation of Cordic

Arc tangent parameter table, ranging from arctan(1), 45 degrees, arctan(1/2), all the way up to arctan(1/2^n). All parameters need to be scaled proportionally, taking arctan (1) =2^n as the standard, and scaling the others by the same factor a. The scaling factor is a fixed-point operation that introduces error.

During each iteration process, x and y undergo shift processing (floor approximation), which also introduces errors. If rounded instead of truncated, the error will be reduced.

The number of iterations is determined by the parameter values after scaling. If the parameter value is 0, the iteration process ends.

21.4.4.1 Vector mode

Confirmation of bit width:

1. Firstly, the bit width of the input x and y is 32-bit.
2. In this mode, the input z=0, so the scaling factor of arctan is independent of the input and can be chosen arbitrarily. Only the desired output angle accuracy determines the scaling factor n of arctan.
3. Determine the number of iterations based on the scaling factor of arctan.
4. The final output angle accuracy is related to the scaling factor n of arctan. The output angle result must be divided by n to obtain the actual angle (since this angle is accumulated, z(i+1)=z(i)-arctan(1/2^i); both sides are multiplied by n). It is also possible to output the result by selecting the high bits (which is more accurate and produces shorter output results).

In the vector mode, the input x and y can be equally scaled, resulting in higher accuracy. Therefore, for small x and y, better accuracy can be achieved by scaling them before feeding into the CORDIC operation.

Firstly, pre-processing is required to move all angles to the first quadrant. If it is in the second quadrant, the initial angle z=90 degrees after moving to the first quadrant. If it is in the third quadrant, the initial angle z=180 degrees. If it is in the fourth quadrant, the initial angle z=270 degrees.

```
if(x<0){
    x=-x;
    y=-y;
    z=z-(1~3)pi/2*n;    // n is the scaling factor
}
```

21.4.4.2 Rotation mode

Confirmation of bit width:

1 The module input is in radians with a range of -π to π. Assuming the input radians are represented by n bits, the scaling factor for the input radians is p0=(2^n)/2pi.

2 According to the formula, z(i+1)=z(i)-arctan(1/2^i), the scaling factor of atan is the same as the scaling factor of the input angle z when doing addition and subtraction operations. atan_t(n)=p0*atan(n), and other parameter values are scaled by the same factor p0.

3 Normally, atan_t(1)=p0*atan(n)=2^(n-3), so the bit width of the parameter table is n-3. If the bit width of the parameter table is set longer to reduce the error of fixed-point arithmetic, the input data only needs to be right-shifted to achieve the same scaling factor when performing addition and subtraction operations. For example, if the bit width of atan during fixed-point arithmetic is n-3+5, the amplification before the operation is::

```
x_in = {x_in,5'h0};
y_in ={y_in,5'h0};
z_in ={z_in,5'h0}；
```

After iteration, select the high bits for output.

4 The output result after iteration is also scaled up by p0. The scaling factor of the output can be adjusted according to the bit width of the output data.

Implementation of rotation mode: Input X_0=k, Y_0=0, Z_0=θ. The input angle range is -pi~pi. Firstly, preprocess the angle and flip it to the range of 0: π/2. Then iterate according to the formula to obtain the result. The result is then multiplied by the gain factor K. The bit width of the input x and y is related to the number of iterations, as each iteration shifts 1 bit.

5

Implementation process of rotation mode:

Input $X_0 = k$，$Y_0=0$，$Z_0 = \theta$；The input angle range is -pi~pi.

1. Firstly, preprocess the angle and flip it to the 0: π/2 range.
2. Then iterate according to the formula to get the result.
3. And finally multiply the gain factor K.
4. The bit width of input x and y is related to the number of iterations because each bit is shifted 1 bit.

21.5 Register

21.5.1 Register List

Base Addres	**0x40058000**				
Offset Address	**Register Name**	**R/W**	**Byte Length**	**Reset Value**	**Description**
0x0	MAC_CTL0	R/W	4	0x0	MAC Control Register 0
0x4	MAC_CTL1	R/W	4	0x0	MAC Control Register 1
0x8	MAC_CTL2	R/W	4	0x0	MAC Control Register 2
0xC	MAC_IN0	R/W	4	0x0	MAC Input Register 0
0x10	MAC_IN1	R/W	4	0x0	MAC Input Register 1
0x14	MAC_IN2	R/W	4	0x0	MAC Input Register 2
0x18	MAC_IN3	R/W	4	0x0	MAC Input Register 3
0x1C	MAC_IN4	R/W	4	0x0	MAC Input Register 4
0x20	MAC_IN5	R/W	4	0x0	MAC Input Register 5
0x24	MAC_OUT0	R/W	4	0x0	MAC Output Register 0
0x28	MAC_OUT1	R/W	4	0x0	MAC Output Register 1
0x2C	MAC_OUT2	R/W	4	0x0	MAC Output Register 2
0x30	MAC_OUT3	R/W	4	0x0	MAC Output Register 3
0x34	DIV_IN0	R/W	4	0x0	Division Unit Input 0
0x38	DIV_IN1	R/W	4	0x0	Division Unit Input 1
0x3C	DIV_OUT0	R/W	4	0x0	Division Unit Output
0x40	DMA_SRBADR	R/W	4	0x0	DMA source data Base address of FFT real part
0x44	DMA_SIBADR	R/W	4	0x0	DMA source data Base address of FFT imaginary part

0x48	DMA_PRBADR	R/W	4	0x0	DMA source data Base address of the twiddle factors table real part
0x4C	DMA_PIBADR	R/W	4	0x0	DMA source data Base address of the twiddle factors table imaginary part
0x50	DMA_TRBADR	R/W	4	0x0	DMA target data Base address of FFT real part
0x54	DMA_TIBADR	R/W	4	0x0	DMA target data Base address of FFT imaginary part
0x58	DMA_LEN	R/W	2	0x0	DMA length configuration
0x5C	DSP_IE	R/W	2	0x0	FFT Interrupt Enable Register
0x60	DSP_FLG	R/W	2	0x0	FFT Flag Register
0x64	ALU_STA0	R/W	4	0x0	ALU Status Register 0
0x68	ALU_STA1	R/W	2	0x0	ALU Status Register 1
0x6C	CRD_CTL	WO	1	0x0	Cordic ControlRegiste
0x70	CRD_XIN	R/W	4	0x0	Cordic Vector mode data x input
0x74	CRD_YIN	R/W	4	0x0	Cordic Vector mode data y input
0x78	CRD_AMP	R	4	0x0	Cordic Vector mode Amplitude Output
0x7C	CRD_PHASE	R	4	0x0	Cordic Vector mode Phase Output
0x80	CRD_ANGLE	R	4	0x0	Cordic Rotation mode Angle Input
0x84	CRD_COS	R	4	0x0	Cordic Rotation mode Cosine Output
0x88	CRD_SIN	R	4	0x0	Cordic Rotation mode Sine Output
0x8C	CRD_IE	R/W	1	0x0	Cordic Interrupt Enable
0x90	CRD_FLG	R/W	1	0x0	Cordic Completion Flag Register
0x94	INTP_LEN	R/W	2	0x0	Interpolation Input Data Length Register
0x98	INTP_LOC	R/W	4	0x0	Current Interpolation Position Register
0x9C	INTP_STEP	R/W	4	0x0	Interpolation Algorithm Step Register

21.5.2 MAC_CTL0 (0x0)

Offset Address: 00H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:23	Reserved	--	R	0
22:21	I2F_PRE_EN	Integer to floating point preprocessing.	R/W	0

		If it is set to 1, it means that in every 4 bytes, the highest byte data is invalid, and the valid data is only 3 bytes. If it is set to 0, it means that 4 bytes are valid.		
20	FFT_TB_EN	FFT uses a fixed parameter table saved by the chip, only 64 points and 128 points are valid. =0: Do not use the fixed parameter table saved by the chip =1: Use chip-saved fixed parameter table	R/W	0
19:15	F2I_MUL	Before converting a floating-point number to an integer, multiply the floating-point number by a factor. 0 The factor is equal to 1 n The factor is equal to 2^n	R/W	0
14:10	I2F_DIV	After converting the integer to a floating point number, divide the floating point number by a factor. 0 The factor is equal to 1 n The factor is equal to 2^n	R/W	0
9:7	ROUND_MODE	Configure rounding mode during floating point arithmetic. When rnd=000, the rounding mode is rounding to nearest. When rnd=001, the rounding mode is rounding toward zero. When rnd=010, the rounding mode is rounding to positive infinity. When rnd=011, the rounding mode is rounding to negative infiinity. When rnd=100, the rounding mode is rounding to nearst up. When rnd=101, the rounding mode is rounding away from zero,. Other: Reserved.	R/W	0
6	MUL_OUT_FM	Floating-point multiplication, output data format: =0: floating-point =1: integer	R/W	0
5	MUL_IN_FM	Floating point multiplication, input data format: =0: floating-point =1: integer	R/W	0
4:0	MODE_SEL	5’d0: All modes are disabled 5’d1: Integer to floating-point conversion, single enable 5’d2: Integer to floating-point conversion, DMA enable 5’d3: Floating-point to integer conversion, single enable 5’d4: Floating-point to integer conversion, DMA enable 5’d5: Floating-point multiplication, mode enable 5’d6: Floating-point addition, mode enable	R/W	0

		5’d7: Floating-point subtraction, mode enable 5’d8: Floating-point multiply-accumulate operation, mode enable (y = ax + b) 5’d9: Butterfly operation, single enable 5’d10: Butterfly operation, DMA enable 5’d11: FFT mode enable 5’d12: Bit Reverse mode enable 5’d13: Second-order IIR single operation, mode enable 5’d14: Second-order IIR DMA operation, mode enable 5’d15: N-order FIR DMA operation, mode enable 5’d16: Linear interpolation operation, mode enable 5’d17: Lagrange interpolation operation, mode enable 5’d18~31: All modes are disabled.		

21.5.3 MAC_CTL1 (0x04)

Offset Address: 04H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:12	Reserved	--	R	0
11:4	FIR_ORDER	FIR filter order configuration: 0~3: 3rd order; 4: 4th order; 5: 5th order; … 254: 254th order; 255: 255th order;	R/W	0
3:1	DMA_STEP	Step size for DMA read from SRAM, only effective for IIR, FIR DMA mode: 0:Step size 1; 1:Step size 2; ... 7: Step size 8;	R/W	0
0	INTP_LAST_EN	Interpolation mode uses the last data from the previous cycle, enable signal	R/W	0

21.5.4 MAC_CTL2 (0x08)

Offset Address: 08H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:3	Reserved	---	R	0
2	DIV_KICK	The division operation unit starts the enable signal	WO	0
1	BTFY_ONCE_KICK	Butterfly operation unit one-shot modetart signal Active at high level	WO	0
0	DMA_EN	DMA enable signal configuration bit: All DMA-related operations are started by configuring this bit to 1, and this bit will be automatically cleared to 0 after DMA	R/W	0

		completes the operations.		

Note: 1.Bit1 and Bit2 are write-only bits.

Refer to chapter 21.3 for the operation instructions and specific configurations that can be implemented using the MAC unit.

21.5.5 MAC_IN0 (0x0C)

Offset Address: 08H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:0	MAC_IN0	Multiplication and addition unit data input port 0	R/W	0

21.5.6 MAC_IN1 (0x10)

Offset Address: 0CH Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:0	MAC_IN1	Multiplication and addition unit data input port 1	R/W	0

21.5.7 MAC_IN2 (0x14)

Offset Address: 10H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:0	MAC_IN2	Multiplication and addition unit data input port 2	R/W	0

21.5.8 MAC_IN3 (0x18)

Offset Address: 14H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:0	MAC_IN3	Multiplication and addition unit data input port 3	R/W	0

21.5.9 MAC_IN4 (0x1C)

Offset Address: 18H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:0	MAC_IN4	Multiplication and addition unit data input port 4	R/W	0

21.5.10 MAC_IN5 (0x20)

Offset Address: 1CH Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:0	MAC_IN5	Multiplication and addition unit data input port 5	R/W	0

21.5.11 MAC_OUT0 (0x24)

Offset Address: 20H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	MAC_OUT0	Multiplication and addition unit data output port 0	R/W	0

21.5.12 MAC_OUT1 (0x28)

Offset Address: 24H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	MAC_OUT1	Multiplication and addition unit data output port 1	R/W	0

21.5.13 MAC_OUT2 (0x2C)

Offset Address: 28H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	MAC_OUT2	Multiplication and addition unit data output port 2	R/W	0

21.5.14 MAC_OUT3 (0x30)

Offset Address: 2CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	MAC_OUT3	Multiplication and addition unit data output port 3	R/W	0

21.5.15 DIV_IN0 (0x34)

Offset Address: 30H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	DIV_IN0	Division unit data input port (dividend)	R/W	0

21.5.16 DIV_IN1 (0x38)

Offset Address: 34H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	DIV_IN1	Division unit data input port (divisor)	R/W	0

21.5.17 DIV_OUT0 (0x3C)

Offset Address: 38H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	DIV_OUT0	Division unit output port (quotient)	R	0

21.5.18 DMA_SRBADR (0x40)

Offset Address: 3CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value

31:16	Reserved	--	R	0
15:0	DMA_SRBADR	DMA source data Base address: (Word address) FFT/BTFY_DMA mode: DMA source data Base address of FFT real part. I2F_DMA/F2I_DMA/BIT_REV mode: DMA source data Base address	R/W	0

21.5.19 DMA_SIBADR (0x44)

Offset Address: 40H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:16	Reserved	--	R	0
15:0	DMA_SIBADR	DMA source data Base address: (Word address) FFT/BTFY_DMA mode: DMA source data Base address of FFT imaginary part.	R/W	0

21.5.20 DMA_PRBADR (0x48)

Offset Address: 44H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:16	Reserved	--	R	0
15:0	DMA_PRBADR	DMA source data Base address: (Word address) DMA source data Base address of the twiddle factors table real part.	R/W	0

21.5.21 DMA_PIBADR (0x4C)

Offset Address: 48H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:16	Reserved	––	R	0
15:0	DMA_PIBADR	DMA source data Base address: (Word address) DMA source data Base address of the twiddle factors table imaginary part.	R/W	0

21.5.22 DMA_TRBADR (0x50)

Offset Address: 4CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:16	Reserved	––	R	0
15:0	DMA_TRBADR	DMA target data Base address: (Word address) FFT/BTFY_DMA mode: DMA target data Base address of FFT real part. I2F_DMA/F2I_DMA/BIT_REV mode: DMA target data Base address.	R/W	0

21.5.23 DMA_TIBADR (0x54)

Offset Address: 50H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:16	Reserved	--	R	0
15:0	DMA_TIBADR	DMA target data Base address: (Word address) FFT/BTFY_DMA mode: DMA target data Base address of FFT imaginary part.	R/W	0

21.5.24 DMA_LEN (0x58)

Offset Address: 54H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:10	Reserved	--	R	0
9:0	DMA_LEN	When the number of points is configured as n, length=(n+1)Word I2F_DMA/F2I_DMA/BTFY_DMA mode: Support any configuration. Bit-reverse mode: Only supports configuring the number of points as 4, 8, 16, 32, 64, 128, 256, 512 or 1024. FFT mode: Only supports configuring the number of points as 64, 128, 256, 512 or 1024.	R/W	0

21.5.25 DSP_IE (0x5C)

Offset Address: 5CH Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:15	Reserved	--	R	0
14	LAGR_DMA_IE	Lagrange Interpolation Completion Interrupt enable	R/W	0
13	LINE_DMA_IE	Linear Interpolation Completion Interrupt enable	R/W	0
12	FIR_DMA_IE	FIR DMA Calculation Completion Interrupt enable	R/W	0
11	IIR_DMA_IE	IIR DMA Calculation Completion Interrupt enable	R/W	0
10	IIR_ONCE_IE	Single IIR Operation Completion Interrupt enable	R/W	0
9	MULT_IE	Floating point multiplication interrupt enable	R/W	0
8	BITREV_IE	Bit-reverse mode interrupt enable	R/W	0
7	FFT_IE	FFT Mode Interrupt Enable	R/W	0
6	BTFY_DMA_IE	Butterfly DMA Interrupt Enable	R/W	0
5	BTFY_ONCE_IE	One time Butterfly Interrupt Enable	R/W	0
4	F2I_DMA_IE	Floating point to integer DMA interrupt enable	R/W	0
3	I2F_DMA_IE	Integer to floating point DMA interrupt enable	R/W	0
2	DIV_IE	Division unit interrupt enable	R/W	0
1	DMA_IE	DMA interrupt enable	R/W	0

0	MAC_IE	Multiplication and addition unit	R/W	0

Note: The above IE control bits, =1 enable interrupt, =0 disable interrupt.

21.5.26 DSP_FLG (0x60)

Offset Address: 60H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:15	Reserved	--	R	0
14	LAGR_DMA_DONE	Lagrange Interpolation Completion Interrupt Flag Write 1 to Clear 0	R/WC	0
13	LINE_DMA_DONE	Linear Interpolation Completion Interrupt Flag Write 1 to Clear 0	R/WC	0
12	FIR_DMA_DONE	FIR DMA Calculation Completion Interrupt Flag Write 1 to Clear 0	R/WC	0
11	IIR_DMA_DONE	IIR DMA Calculation Completion Interrupt Flag Write 1 to Clear 0	R/WC	0
10	IIR_ONCE_DONE	Single IIR Operation Completion Interrupt Flag Write 1 to Clear 0	R/WC	0
9	MULT_DONE	Floating-point multiplication complete interrupt flag Write 1 to clear 0	R/WC	0
8	BITREV_DONE	Bit-reverse mode interrupt flag Write 1 to clear 0	R/WC	0
7	FFT_DONE	FFT mode interrupt flag Write 1 to clear 0	R/WC	0
6	BTFY_DMA_DONE	Butterfly DMA Interrupt Flag Write 1 to clear 0	R/WC	0
5	BTFY_ONCE_DONE	Butterfly one time mode Interrupt Flag Write 1 to clear 0	R/WC	0
4	F2I_DMA_DONE	Floating point to integer DMA interrupt flag Write 1 to clear 0	R/WC	0
3	I2F_DMA_DONE	Integer to floating point DMA interrupt flag Write 1 to clear 0	R/WC	0
2	DIV_DONE	Division Unit Interrupt Flag Write 1 to clear 0	R/WC	0
1	DMA_DONE	DMA complete signal interrupt flag Write 1 to clear 0	R/WC	0
0	MAC_DONE	Multiplication and addition unit calculation (MUL or BTFY_ONCE) completed Write 1 to clear 0	R/WC	0

21.5.27 ALU_STA0 (0x64)

Offset Address: 60H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:24	F2I_STATUS	Floating point to integer unit operation status flag	R	1

		The definition is the same as 'Multiplication and addition unit 0-operation status flag.		
23:16	I2F_STATUS	Integer to floating point unit operation status flag The definition is the same as 'Multiplication and addition unit 0-operation status flag.	R	1
15:8	ADDSUB1_STATUS	Multiplication and addition unit 1-operation status flag The definition is the same as ‘Multiplication and addition unit 0-operation status flag’	R	1
7:0	ADDSUB0_STATUS	Multiplication and addition unit 0-operation status flag bit0: Integer or floating point output is zero. bit1: floating point output is infinity. bit2: Floating point operation is invalid. It is also set to 1 when one of the inputs is NaN. bit3: floating point number less than minimum normalized number. bit4: floating point number greater than maximum normalized number. bit5: Integer or floating-point output is not equal to an infinitely precise result. bit6: The size of the rounded integer result is larger than the largest representable two integers with the same sign. bit7: Reserved.	R	1

21.5.28 ALU_STA1 (0x68)

Offset Address: 64H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:16	Reserved	--	R	0
15:8	DIV_STATUS	The operation state of the division unit. The definition is the same as ‘Multiplication and addition unit 0-operation status flag’	R	x
7:0	MUL_STATUS	Multiplication unit operation status. The definition is the same as ‘Multiplication and addition unit 0-operation status flag’.	R	1

21.5.29 CRD_CTL (0x6C)

Offset Address: 68H Reset Value: 0x0

Bit	**Name**	**Description**	**R/W**	**Reset Value**
31:1	Reserved	--	R	0
1	CRD_ROT_KICK	cordic rotation mode start signal	WO	0
0	CRD_VEC_KICK	cordic vector mode enable signal	WO	0

Note: This register has no read function, it is a write-only register.

21.5.30 CRD_XIN (0x70)

Offset Address: 6CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	X_IN	Cordic Vector mode data x input	R/W	0

21.5.31 CRD_YIN (0x74)

Offset Address: 70H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	Y_IN	Cordic Vector mode data y input	R/W	0

21.5.32 CRD_AMP (0x78)

Offset Address: 74H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	AMP_OUT	Cordic Vector mode Amplitude Output $\mathrm{AMP\_OUT} = \sqrt{X_0^2 + Y_0^2/4*P}$	R/	0

21.5.33 CRD_PHASE (0x7C)

Offset Address: 78H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	PHASE_OUT	Cordic Vector mode Phase Output $\mathrm{PHASE\_OUT} = \arctan\left(\frac{Y_0}{X_0}\right)*2\text{^}31/pi$	R	0

21.5.34 CRD_ANGLE (0x80)

Offset Address: 7CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	ANGLE_IN	Cordic rotation mode input in radians in the range [-pi:pi] Enter -1*2^31 for -pi, enter 1*2^31 for pi ANGLE_IN =radian*2^31/pi	R/W	0

21.5.35 CRD_COSINE (0x84)

Offset Address: 80H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	COSINE_OUT	Cordic Vector mode Cosine Output COSINE_OUT=cos(angle)*(2^(31-2))/(2*pi) angle=radian*180/pi	R	0

21.5.36 CRD_SINE (0x88)

Offset Address: 84H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	SINE_OUT	Cordic Vector mode Sine Output SINE_OUT=sin(angle)*(2^(31-2))/(2*pi)	R	0

21.5.37 CRD_IE (0x8C)

Offset Address: 88H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:1	Reserved	--	R	0
0	CRD_ROT_IE	Cordic Rotation Mode Completion Interrupt Enable Bit	R/W	0
0	CRD_VEC_IE	Cordic Vector Mode Completion Interrupt Enable Bit	R/W	0

21.5.38 CRD_FLG (0x90)

Offset Address: 8CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:1	Reserved	--	R	0
1	CRD_ROT_DONE	Cordic Rotation Mode Completion Flag Write 1 to clear 0	R/W	0
0	CRD_VEC_DONE	Cordic Vector Mode Completion Flag Write 1 to clear 0	R/W	0

21.5.39 INTP_LEN (0x94)

Offset Address: 94H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:13	Reserved	--	R	0
12:0	INTP_LEN	The actual available input data length for the interpolation algorithm， i.e., the number of x[n]. Points configuration n, length = (n+1) Word. (Integer type)	R/W	0

21.5.40 INTP_LOC (0x98)

Offset Address: 98H Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	INTP_LOC	Current Interpolation Position (floating-point) for Interpolation Algorithm	R/W	0

21.5.41 INTP_STEP (0x9C)

Offset Address: 9CH Reset Value: 0x0

Bit	Name	Description	R/W	Reset Value
31:0	INTP_STEP	Step Size (floating-point) for Interpolation Algorithm	R/W	0

21.6 Software Operation Process

Please refer to the routine provided by Renergy Micro-Technologies Co., Ltd. for specific implementation. The following instructions are for learning reference only.

21.6.1 Full Program Description:

1. The ADC samples the data integer d0, which is stored in SRAM with a bit width in 24-bit binary complement format.
2. Data preprocessing: Normalization of ADC data (binary complement format) to signed floating-point format d1. The range of representation is -1 to 1.
3. FFT operation: Input to the FFT operation on all converted numbers d1, in complex format, where the real part is d1 and the imaginary part is 0. The FFT operation results in an FFT output. The output is a complex number, the real part Re and the imaginary part Im are floating point numbers.
4. Calculation of harmonic content:

   ```
   repeat(6){
     $Y_0 = \sqrt{Re0 * Re0 + Im0 * Im0}$,
     for(n=1;n<42;n++){
     $Y_n = \sqrt{Ren * Ren + Imn * Imn}$,
     Yin=k*Yn/Y0,
     }
   }
   ```

   Calculate the 41st harmonic content.

   Both current and voltage need to be done separately for harmonic content Yu and Yi.
5. Phase angle calculations and harmonic power::

   ```
   repeat(3){
     for(n=1;n<42;n++){
       Aun=atan(Ren/Imn),// For voltage data
       Ain=atan*(Ren/Imn),// For current data
       A=((Au-Ai)+A0*n)*180/pi,
       PFn=cos(A*pi/180)
     }
   }
   ```

   Calculate the phase angle 40 times without the fundamental frequency. Requires an accuracy of 0.01 degrees.
6. Calculation of harmonic power: Phn=FSA*Yun*Yin*PFn. The 41st harmonic is calculated and then accumulated, where the FSA is obtained externally.

21.6.2 Operation process:

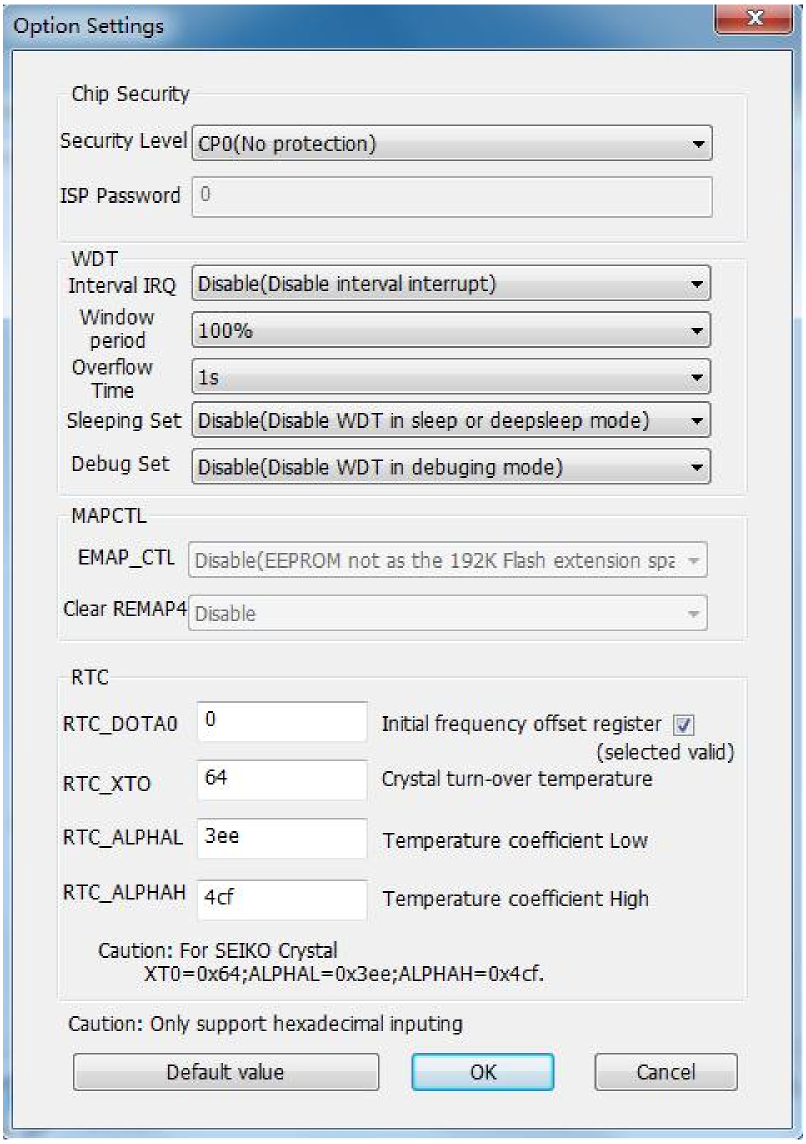
1. Cache the ADC sampled data sdata0 to SRAM, with each point having 24-bit data.
2. Use i2f_dma mode to convert sdata1 in SRAM to the floating-point format and normalize it to obtain sdata2. Note that only the third byte of each word address in SRAM is valid at this point. Therefore, it is necessary to configure MAC_CTL0[28]=1, and ignore the highest byte..
3. Use bitrev_mode mode to do bit-reverse to transform sdata2 into sdata3.
4. Use the FFT calculation module to perform FFT calculation on sdata3 to obtain the result (the software needs to write the FFT calculation table to SRAM in advance).
5. Use the hardware CORDIC module, division module to calculate harmonic content and harmonic power.

22 Option Byte

It built in an area of option byte, when the chip is reset, it will automatically configure option byte and perform specific function. Option byte includes

Protection of the chip, WDT, EMAP and RTC setting.

Programming of option byte can be setting by the programming tool from Renergy (MINIPRO programming unit or ISP programming tools), Taking the MINIPRO programmer as an example, the option bytes can be set by opening the programming option dialog box, as shown in the following figure (see the 《MINIPRO instruction manual of programming unit》 for detailed operation methods).。



Option Settings

Chip Security

Security Level: CP0(No protection)

ISP Password: 0

WDT

Interval IRQ: Disable(Disable interval interrupt)

Window period: 100%

Overflow Time: 1s

Sleeping Set: Disable(Disable WDT in sleep or deepsleep mode)

Debug Set: Disable(Disable WDT in debuging mode)

MAPCTL

EMAP_CTL: Disable(EEPROM not as the 192K Flash extension spa

Clear REMAP4: Disable

RTC

RTC_DOTA0: 0 Initial frequency offset register ☑ (selected valid)

RTC_XTO: 64 Crystal turn-over temperature

RTC_ALPHAL: 3ee Temperature coefficient Low

RTC_ALPHAH: 4cf Temperature coefficient High

Caution: For SEIKO Crystal XT0=0x64;ALPHAL=0x3ee;ALPHAH=0x4cf.

Caution: Only support hexadecimal inputing

22.1 Chip Protection Settings

Protect function of option byte can protect built-in Flash, user can protect the chip by protection lever setting and ISP password setting. The following protection levels are provided:

Protection level	Name	Description
0	CP0	Without any protection (no password is required for ISP access)

1	CP1	SWD Interface can access chip, password is required for ISP access
2	CP2	Disable access chip by SWD Interface, password is required for ISP access
3	CP3	Disable access chip by SWD and ISP Interface(ISP only provides the function of erasing the whole FLASH (under this protection level, the erasing operation will make the chip 's protection level CP0))

22.2 WDT Setting

Option byte provide interval interruption of WDT, window open cycle, overflow time, CPU sleep setting, CPU debug setting，See detailed meaning on the section of WDT. As shown in the table below:

Name	**Description**	**default**
Interval interrupt	0：Disable (Disable interrupt of intervals) 1：Enable (When reaches 75% of the spills, interval interrupt occurs)	0
Window open cycle	0：25% 1：50% 2：75% 3：100% During the window open, write 0xBB to WDTE register, watchdog reset and count again; During the window close, write 0xBB to WDTE register, The internal reset signal generate.	3
Overflow time	0：16ms 1：32ms 2：128ms 3：512ms 4：1s 5：2s 6：4s 7：8s	4
CPU sleep setting	0: Disable (When CPU is sleep or deepsleep, WDT is off) 1: Enable (When CPU is sleep or deepsleep, WDT is on)	0
CPU debug setting	0：Disable（When CPU in a state of commissioning, WDT is off） 1：Enable（When CPU in a state of commissioning, WDT is on） Notes: The CPU in the debugging state means that the user stops Cortex M0 through the debugging interface (the PC pointer stops counting).	0

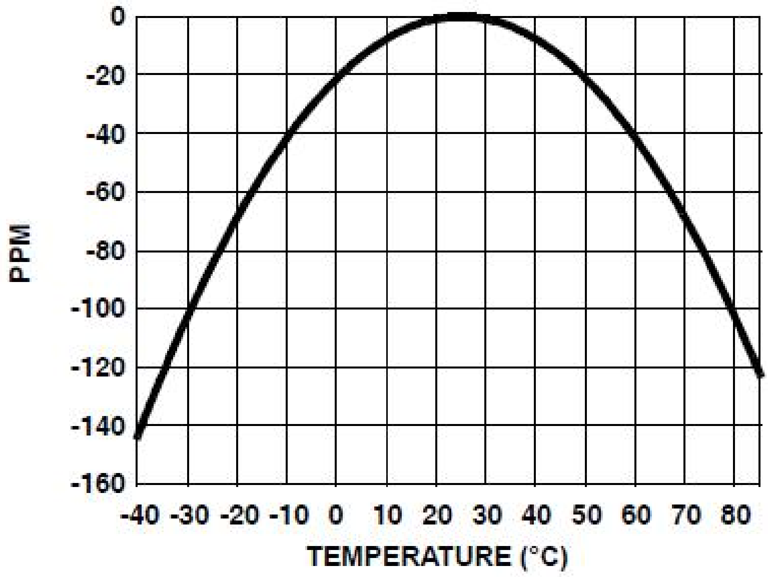
22.3 RTC Setting

The RTC of the RN8XXXX has a built-in automatic temperature compensation function that automatically compensates the temperature of the 32k crystal to provide an accurate second pulse output in the range of-25 °C to 70 °C.

Among them, the temperature frequency curve of the crystal as shown below, this is a quadratic curve with a vertex of 25 degrees. (f=f0-alpha*(T-T0)), T0 is 25 degrees). However, The alpha of quadratic curve is different between the high temperature section (25 ℃ ～85 ℃) and low-temperature (-25 ℃ ～70 ℃), so the option bytes provide separately the parameter RTC_ALPHAL and RTC_ALPHAH ,Each of them is filled with round (alpha *

32768) ,which means rounding operation.

If the choice is high consistency crystals (VT-200-F) provided by Seiko, then ALPHAL = 0x3ee, ALPHAH = 0x4cf.



23 Programming Support

SoC support the built-in programming of the internal EEPROM and FLASH.

It is recommended that customers call the Renergy library function to implement the IAP function; using the Renergy programmer to complete the ISP function.

23.1 Overview

SoC programming system has the following features:

- Built in the FLASH content protection mechanisms
- Supports ISP programming mode;
- Support IAP programming mode;
- Support downloads recording mode by SWD;
- support volume production recording mode;

23.2 Flash Protection Mechanisms

Flash protection is to allow users to enable different levels of security to restrict the access to the on-chip Flash and ISP. Protection mechanisms protect the different level of protection in the following table. Users can set chip protection class by set the “option byte”.

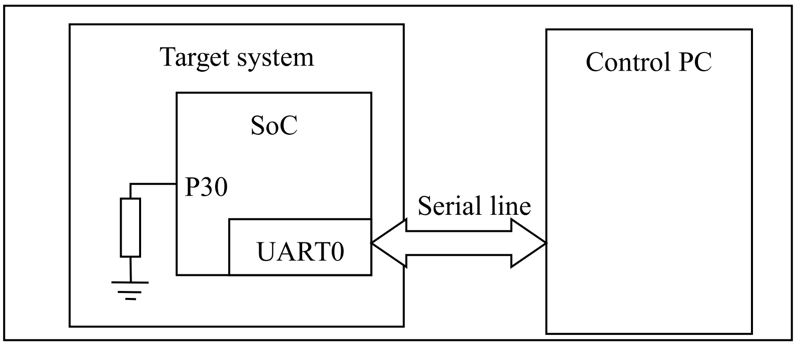
Table 23- 1 The protection level of SoC

Protect level	Name	description
0	CP0	Without any protection (no password is required for ISP access)
1	CP1	SWD Interface can access chip, password is required for ISP access
2	CP2	Disable access chip by SWD Interface, password is required for ISP access
3	CP3	Disable access chip by SWD and ISP Interface (ISP only have the function of the whole erase FLASH (Erase operation will reduce the protection level to CP0 in CP3)

23.3 In System Programming (ISP)

The user can pull P30 signal to low, and reset the SoC, let SoC into the ISP mode. ISP mode of connection diagram as shown in figure 22-1.

Figure 22-1 ISP hardware configuration diagram



ISP major process:

1. According to the connection diagram configuration and connect the target system and control host;
2. Reset the target system;
3. Control host configuration of serial port for a start bit, 8 data bits, 1 stop bit， the baud rate is not more than 115200bps;
4. Control the host to send “e”;
5. The target system response “Synchnonized /r/n”;
6. Control the host to send “Synchnonized /r/n”;
7. The target system response “7373(1843)/r/n”;(If the current system frequency is 7.3728M,send 7373;If the current system frequency is1.8432M,send 1843)
8. Control host can perform the corresponding ISP commands according to need;

23.3.1 ISP Communications Protocol

All command of ISP send in the form of an ASCII text. Text use (/r) or (/n) as end mark.All ISP response is < CR > < LF > the end of the ASCII string sent. The data were sent and received in the original format (not converted to ASCII).

- Command format

command parameter0 parameter1 ... parametern/r/n
{DATA}

- Response format

Return code/r/n
Response 0/r/n
Response 1/r/n
...
Response /r/n
{DATA}

- Data format

After starting the orders of WM and RM, the data transmission of ISP will start up. The data transmits in a unit of line, and the maximum of 32 bit data in a line is 16(If the number of the data less than 16, the actual number will be sent); Each transmission completes 1 Block (1 Block contains a maximum of 32 rows (less than 32 rows, the corresponding number of rows)) data, and sends a check row (the negative complement of the cumulative checksum of the Block data (calculated in word))。

When chip received a complete block of data, the data will be verified. If the check is passed, send an "OK / r / n" command; If the validation is error or the data packets is illegal, then sent a "RS / r / n" command. If the programmer receives “RS/r/n” command, you need to re-send the Block data.

When the row data is 0x7e, it means 0x7d, 0x5e sent; when the row data is 0x7d, it means 0x7d, 0x5d sent

Data line format: (B behalf transmit data Byte, hexadecimal)

Table 23- 2 ISP data transmission format

The first row	**1**	**2**	**3**	**4**	**5**	**6**	**……**	**64**	**65**	**The last row**
0x7e	Num	B0	B1	B2	B3	B4	……	B62	B63	0x7e

Check line format: (ASCII code. S represent the cumulative checksum SUM)

Table 23- 3 ISP data validation format

The first row	**1**	**2**	**3**	**4**	**5**	**The last row**
0x7e	0xff	S0	S1	S2	S3	0x7e

23.3.2 The Used Resources

- RAM resources used

ISP uses the RAM within the range 0x10001000 to 0x10002800 on chip, Stack is located at the top of RAM. Flash can use the RAM within the scope 0x10000000-0x10001000 (4KB) for programming.

23.3.3 ISP Command

Each ISP command supports specific status code. When receive the undefined command,command processing program sends the return code INVALID_COMMAND。

Command and return code are ASCII format. Only when receives the ISP command execution is completed, the ISP command processor will send CMD SUCCESS, this time the host can send a new ISP commands.

ISP command can be divided into three types:

1. Normal command: Only under the CP0, or CP1, CP2 and password is right, it can be access
2. UN command Under the CP0, CP1, CP2 level of protection (password does not provide), it can be access
3. In any case, FC, AL command can access

Table 23- 4 ISP command

Command	Instruction	Nature
Baud rate setting	BS <Baud rate> <stop bit>	Normal command
Echo	RD < switch settings>	Normal command
Write memory	WM <Address> <Byte size> <Mode>	Normal command
Read memory (including flash, sram)	RM <Address> <Byte size> <Mode>	Normal command
Flash page erase	FP <Page address>	Normal command

Flash block erase	FS <Block address>	Normal command
Flash chip erase	FC	
Flash block blank check	FQ <Block address>	Normal command
Memory compare	MC <Address1> <Address2> <Byte size>	Normal command
Running	GO <Address>	Normal command
Unlock	UN <password>	
Access to confidential level	AL	
Enable PFPM	PM <switch settings>	Normal command
Reset by software	RS	Normal command
Enable NVM(FLASH)	NV <NVM option>	Normal command

- Baud rate setting

Table 23- 5 ISP command

Command	BS <Baud rate> <stop bit>
Input	Baud rate:9600 or 19200 or 38400 or 57600 or 115200 Stop bit:1 or 2
Return code	CMD_SUCCESS or INVALID_BAUD_RATE or INVALID_STOP_BIT or INVALID_PARAM
Annotation	Change ISP communication serial frame format, including baud rate and stop bit. Serial port start bit is 1, data bit is 8. New frame format is effective After return CMD SUCCESSS.
Example	"BS 9600 2" Serial port baud rate will be set as 9600bps, two stop bits.

- Echo

Table 23- 6 ISP command

Command	RD <switch settings>
Input	Switch setting: 0 (off) or 1 (on)
Return code	CMD_SUCCESS or INVALID_PARAM
Annotation	Command and data echo. Default is on. When echo on, SoC send command and data which to receive back to host.
Example	"RD 0" echo off.

- Write memory

Table 23- 7 ISP command

Command	WM <Address> <Byte size> <mode>
Input	Address: address to start, it should be 32 bits; Byte size: the number of bytes, must be in multiples of four; mode:0 as a serial port,1 as parallel
Return code	CMD_SUCCESS or FM_MODE_ERROR or ADDR_NOT_ALIGN or COUNT_ERROR or COUNT_ERROR or ADDR_NOT_MAPPED or INVALID_PARAM
Annotation	Write data to SRAM
Example	"WM 268436224 4 0"

	“78” “56” “34” “12” Write 0x12345678 to address 0x10000300 via a serial port

- Read memory

Table 23- 8 ISP command

Command	RM <Address> <Byte size> <Mode>
Input	Address: address to read, it should be 32 bits; Byte size: To compare the number of bytes, must be in multiples of four; mode:0 as a serial port,1 as parallel
Return code	CMD_SUCCESS or FM_MODE_ERROR or ADDR_NOT_ALIGN or COUNT_ERROR or COUNT_ERROR or ADDR_NOT_MAPPED or INVALID_PARAM
Annotation	Read the content of SARM of SoC
Example	“RM 268436224 4 0” Read the content which SRAM address is 0x10000300 via a serial port

- Flash page erase

Table 23- 9 ISP command

Command	FP <Page address>(FPGA version is 0 to 3071)
Input	Page address: Optional between 0 to 1535
Return code	CMD_SUCCESS or INVALID_PAGE or INVALID_PARAM
Annotation	Erase the content of the specify page of Flash of SoC
Example	“FP 0” Erase the content of the page 0

- Flash block erase

Table 23- 10 ISP command

Command	FS <Block address>
Input	Block address: Optional between 0 to 47
Return code	CMD_SUCCESS or INVALID_SECTOR or INVALID_PARAM
Annotation	Erase the content of the specify block of EEPROM of SoC
Example	“FS 0” Erase the content of the block 0

- Flash chip erase

Table 23- 11 ISP command

Command	FC
Input	NC
Return code	CMD_SUCCESS or INVALID_PARAM
Annotation	Erase all the content of Flash of SOC.
Example	“FC” erase all the content of Flash.

- Flash block blank check

Table 23- 12 ISP command

Command	FQ <Block address>
Input	Block address: Optional between 0 to 47
Return code	CMD_SUCCESS or INVALID_SECTOR or INVALID_PARAM
Annotation	Check if the content of the specify block of EEPROM is empty (Unprogrammed after erase it)
Example	"FQ 1" Check if the content of the 1 block is empty

- Flash Programming

Table 23- 13 ISP command

Command	FW <FLASH address> <RAM address> <Byte size>
Input	FLASH address: target address of FLASH to write RAM address: the SRAM address of source buffer Byte size: the number of bytes written(If Byte size is different from number of bytes of Flash page, the rest of this Flash will be set as 0
Return code	CMD_SUCCESS or COUNT_ERROR or SRC_ADDR_NOT_ALIGN or SRC_ADDR_NOT_MAPPED or DST_ADDR_NOT_ALIGN or DST_ADDR_NOT_MAPPED or INVALID_PARAM
Annotation	It is used to program FLASH.
Example	"FW 0 268436224 128" copy 128 bytes data which SRAM address is 0x10000300 to FLASH address 0.

- Memory compare

Table 23- 14 ISP command

Command	MC <Address1> <Address2> <byte size>
Input	Address1(DST): Starting address of to compare the memory region 1, it should be with the word alignment; Address2(SRC): Starting address of to compare the memory region 2, it should be with the word alignment; Byte size: To compare the number of bytes, must be in multiples of four;
Return code	CMD_SUCCESS or COUNT_ERROR or SRC_ADDR_NOT_ALIGN or SRC_ADDR_NOT_MAPPED or DST_ADDR_NOT_ALIGN or DST_ADDR_NOT_MAPPED or COMPARE_ERROR or INVALID_PARAM
Annotation	This command is used to compare the content of the two regions of memory.
Example	"MC 268436224 268436224 4" Compare 4 bytes data which SRAM address is 0x10000300 with 4 bytes data which SRAM address is 0x10000300

- Running

Table 23- 15 ISP command

Command	GO <Address>
Input	Address: Address of Code execution start Flash or RAM. This address must be Thumb address
Return code	CMD_SUCCESS or ADDR_NOT_THUMB or ADDR_NOT_MAPPED orINVALID_PARAM
Annotation	This command is used to execute the program in RAM or Flash.Once successfully execute the command, it could no longer return to the ISP command handler.

Example	“GO 5” Jump to address 0 x00000004 to execute

- Unlocked

Table 23- 16 ISP command

Command	UN
Input	Password:32bit Hexadecimal number
Return code	CMD_SUCCESS or INVALID_PASS or INVALID_PARAM
Annotation	This command is used to unlock ISP.
Example	“UN 567” Enter the password 567 to unlock the ISP

- Access to confidential level

Table 23- 17 ISP command

Command	AL
Input	NC
Return code	CMD_SUCCESS or INVALID_PARAM
Annotation	This command is used to access to confidential level of ISP
Example	“AL” will return confidential level of SoC

- Reset by software

Table 23- 18 ISP command

Command	RS
Input	NC
Return code	CMD_SUCCESS or INVALID_PARAM
Annotation	This command enable reset by software
Example	“RS” will enable reset by software

- Enable NVM

Table 23- 19 ISP command

Command	NV <NVM option>
Input	NVM option:0(Flash)or 1(EEPROM)
Return code	CMD_SUCCESS or INVALID_PARAM
Annotation	This command boots the flash code
Example	“NV 0” Enable Flash write, programming.

23.3.4 ISP Return Code

Table 23- 20 ISP command

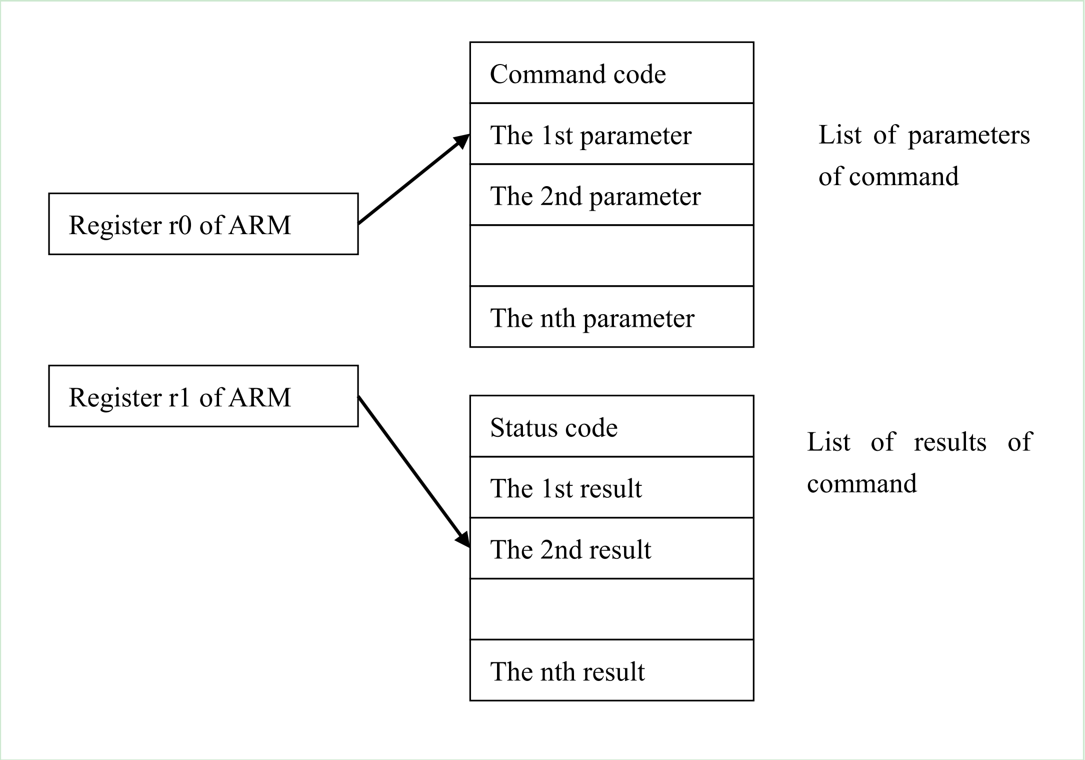
Return Code （ ASCII code）	Mark	Description
0	CMD_SUCCESS	Successfully executed command. Only after successfully

		executed command, ISP will send this code
1	INVALID_COMMAND	Invalid command
2	INVALID_PARAM	Invalid parameter (ASCII of parameter is not 0-9)
3	INVALID_BAUD_RATE	Invalid baud rate
4	INVALID_STOP_BIT	Invalid stop bit
5	ADDR_NOT_ALIGN	Address is not for boundary with byte
6	COUNT_ERROR	Byte count is not in multiples of four
7	ADDR_NOT_MAPPED	Address have space of crossing the line
8	INVALID_SECTOR/INVALID_PAGE	Invalid SECTOR_NUM or PAGE_NUM
9	SECTOR _NOT_BLANK	SECTOR is not empty
10	SRC_ADDR_NOT_ALIGN	Source address is not for boundary with byte
11	SRC_ADDR_NOT_MAPPED	Source address have space of crossing the line
12	DST_ADDR_NOT_ALIGN	Destination address is not for boundary with byte
13	DST_ADDR_NOT_MAPPED	Destination address have space of crossing the line
14	COMPARE_ERROR	Contrast Error
15	FM_MODE_ERROR	Memory model error
16	ADDR_NOT_THUMB	The address is not Thumb command
17	INVALID_PASS	Wrong password

23.4 In-Application Programming (IAP)

About In-Application Programming, we should call programs of IAP by word pointer in the register r0, the word pointer point to RAM that contains the command code and parameters. Result of IAP command return to results table which register r1 is pointing to. User can give the same value to pointer in the register r0 and r1, so we can reuse the command table to hold the result. Parameter table should be big enough to save all the result. About parameter passing , please see table 21-7.The number of parameters and the results depend on IAP command. “Flash programming”, the maximum number of for command parameters, the number of result is 1. Command processor send status code (INVALID_COMMAND) after receive an undefined command. Program of IAP is Thumb code, address is 0x1800_1c01.

Figure 23-2 IAP parameter passing



23.4.1 IAP Command

IAP command	Command code	Instructions
Flash page erase	0x50	See the section of ISP
Flash block erase	0x51	See the section of ISP
Flash chip erase	0x52	See the section of ISP
Flash block blank check	0x53	See the section of ISP
Flash programme	0x58	See the section of ISP
Enable NVM	0x5a	See the section of ISP
Analog reset by software	0x5b	See the section of ISP

23.4.2 IAP Usage

IAP is use as follows:

- Online Updater (update FLASH)

When online upgrade, it needs to erase or write Flash. It will last about 4 milli second to erase Flash, Increase processing delays about the interrupt.

An implementation method of the IAP:

When the user needs to online upgrade, he needs to add a procedures section of IAP upgrade with software. This program is used to receive programs or data from remote host by the communication port (example UART). By IAP Interface SOC supplied. These programs or data is written to FLASH in the SOC.

23.5 Mass Production Platform

Renergy provides a variety of programming methods, Specific elaborate can refer to “Renergy programming platform instructions”.

24 Package size and soldering conditions

24.1 Package size

24.1.1 LQFP128L

LQFP128L (1414×1.4)		14.00×14.00×1.40	e=0.40

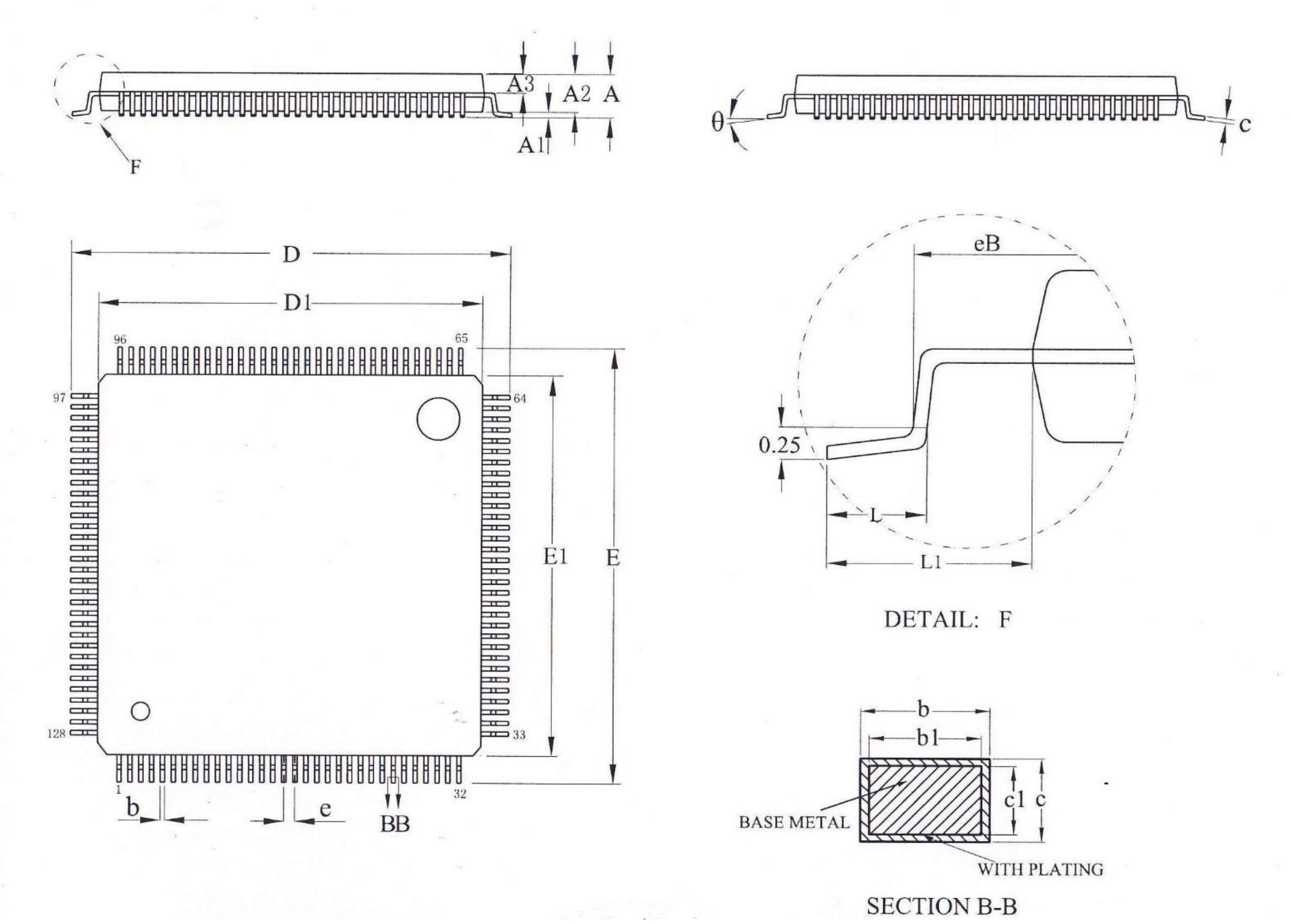
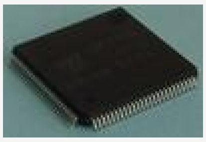


Figure 24-1 LQFP128L Package size drawing

SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	---	---	1.60
A1	0.05	---	0.20
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.14	---	0.22
b1	0.13	0.16	0.19
c	0.13	---	0.17

c1	0.12	0.13	0.14
D	15.80	16.00	16.20
D1	13.90	14.00	14.10
E	15.80	16.00	16.20
E1	13.90	14.00	14.10
eB	15.05	---	15.35
e	0.40BSC		
L	0.45	---	0.75
L1	1.00BSC		
θ	0	-----	7º

24.1.2 LQFP100L

LQFP100L (1414×1.4)		14.00×14.00×1.40	e=0.50

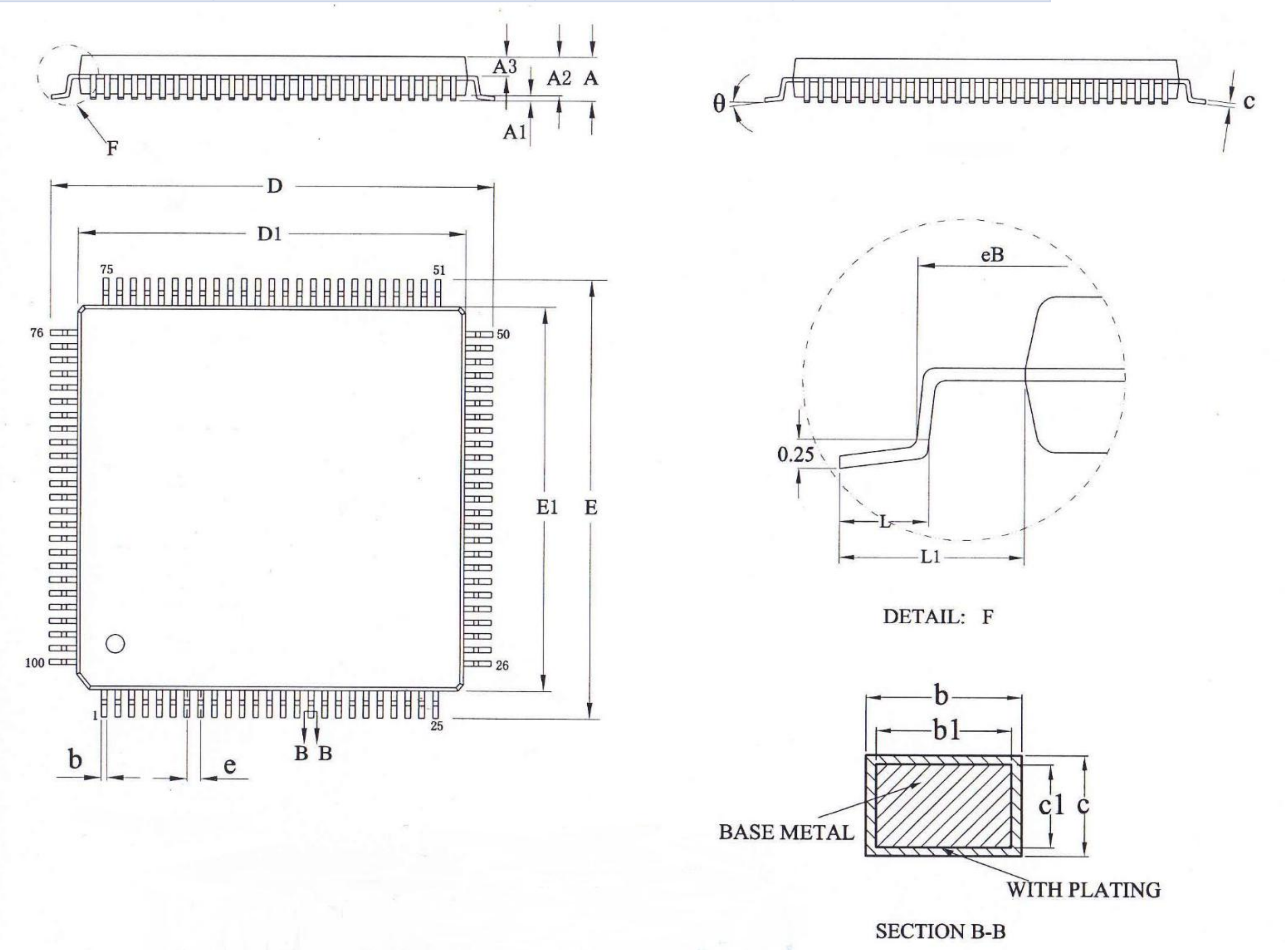


Figure 24-2 LQFP100L Package size drawing

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	15.80	16.00	16.20
D1	13.90	14.00	14.10
E	15.80	16.00	16.20
E1	13.90	14.00	14.10
eB	15.05	---	15.35
e	0.50BSC		
L	0.45	---	0.75
L1	1.00BSC		
θ	0	-----	7º

24.1.3 LQFP64L

LQFP64L (0707×1.4)		7.00×7.00×1.40	e=0. 40

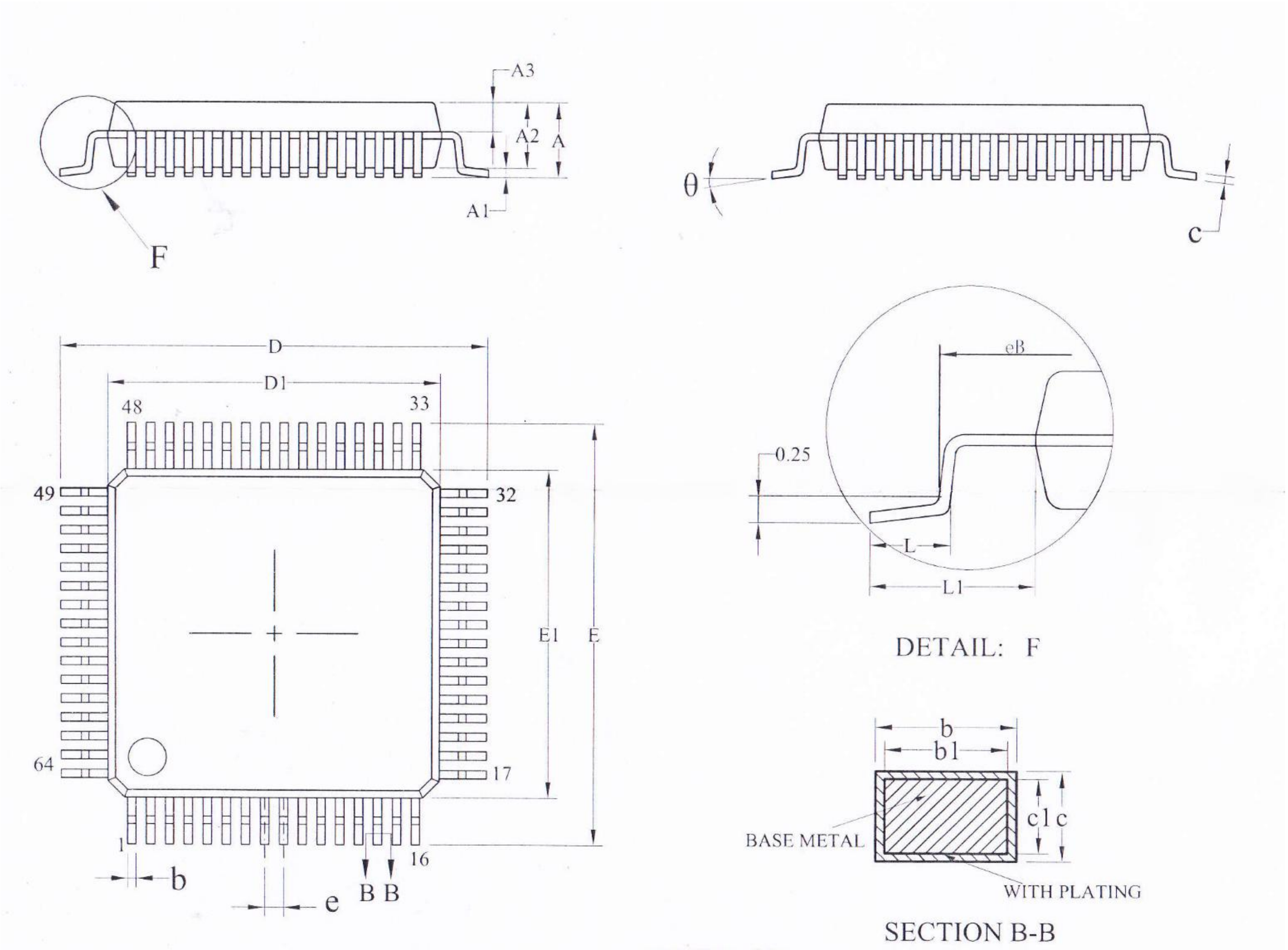
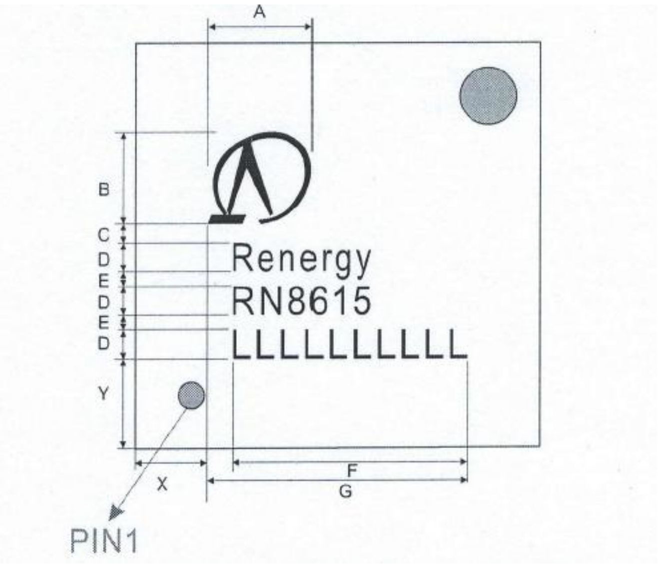


Figure 24-3 LQFP64L Package size drawing

SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	---	---	1.60
A1	0.05	---	0.20
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.16	---	0.24
b1	0.15	0.18	0.21
c	0.13	---	0.17
c1	0.12	0.13	0.14
D	8.80	9.00	9.20
D1	6.90	7.00	7.10
E	8.80	9.00	9.20
E1	6.90	7.00	7.10
eB	8.10	---	8.25
e	0.40BSC		
L	0.40	---	0.65

L1	1.00BSC		
θ	0	-----	7°

The product appearance diagram is as follows, Take the RN8615 as an example, other products are similar:



The first line is Renergy Logo；

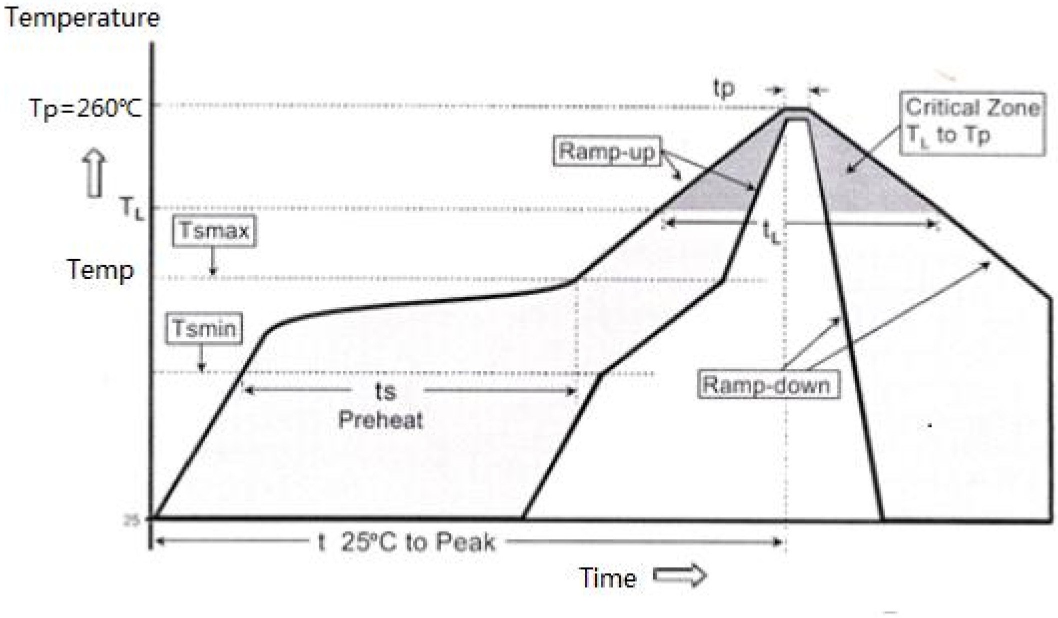
The second line Renergy is English abbreviation；

The third line RN8xxx is Renergy product model；

The fourth line is Product lot number;

The dot in the bottom left corner is PIN1 flag.

24.2 Reflow oven temperature setting conditions



Temperature setting curve of reflow oven

Distribution map feature	Value
Holding temperature TL Peak temperature Tp Average tilt rate of rise(TL to Tp)	217℃ 260℃ Max 3℃/Second
Warm up Minimum temperature（Ts min） Maximum temperature（Ts max） Time（min-max）（ts）	 150℃ 200℃ 60-180 second
Ts max - TL tilt rate of rise (Ts max to TL)	Max 3℃/Second