User Manual

GW3323

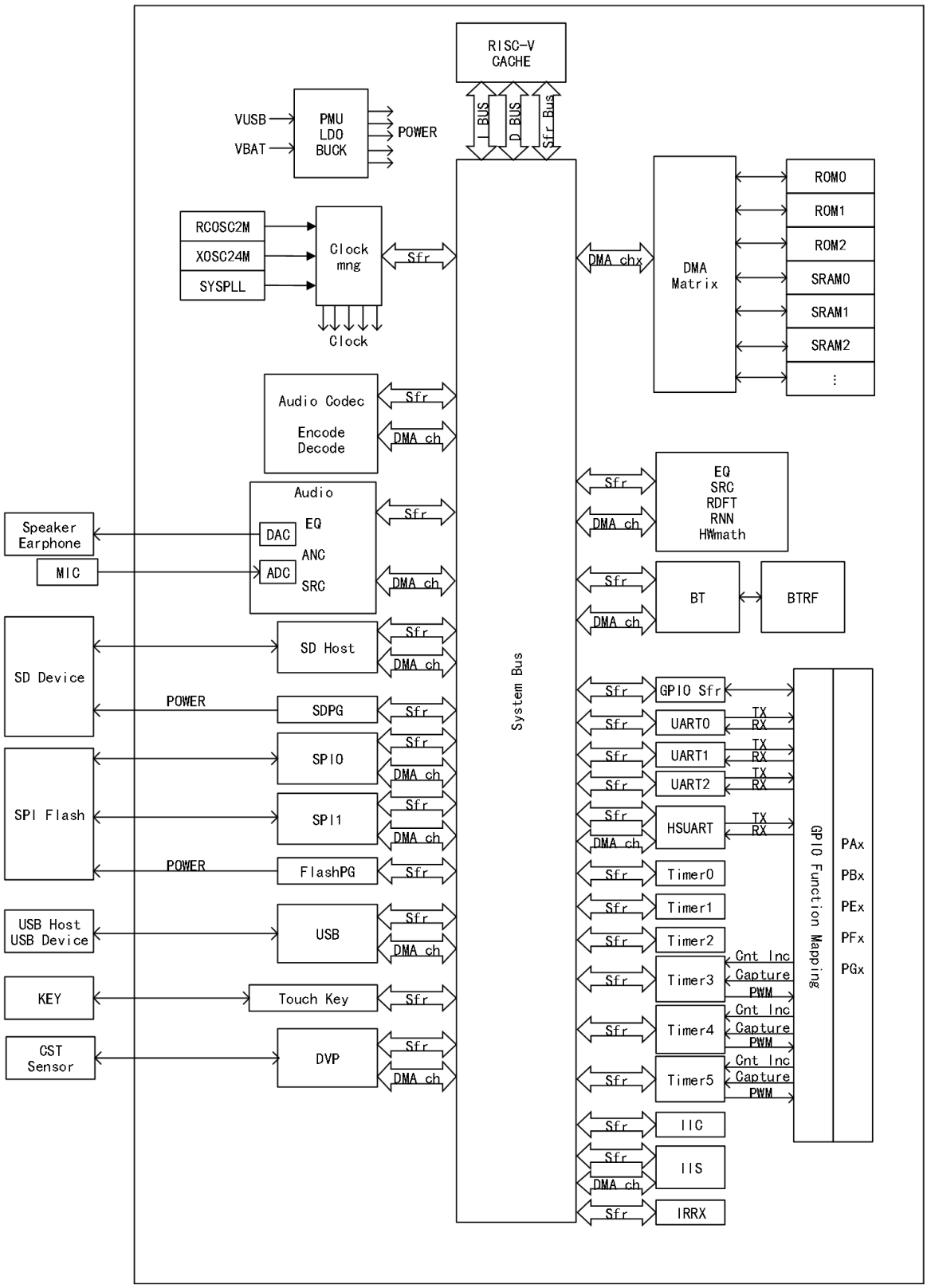
RISC-V based 32-bit MCU with Bluetooth

Version: V0.2

Contents

1 Block Diagram

Figure 1 GW3323 System Block Diagram

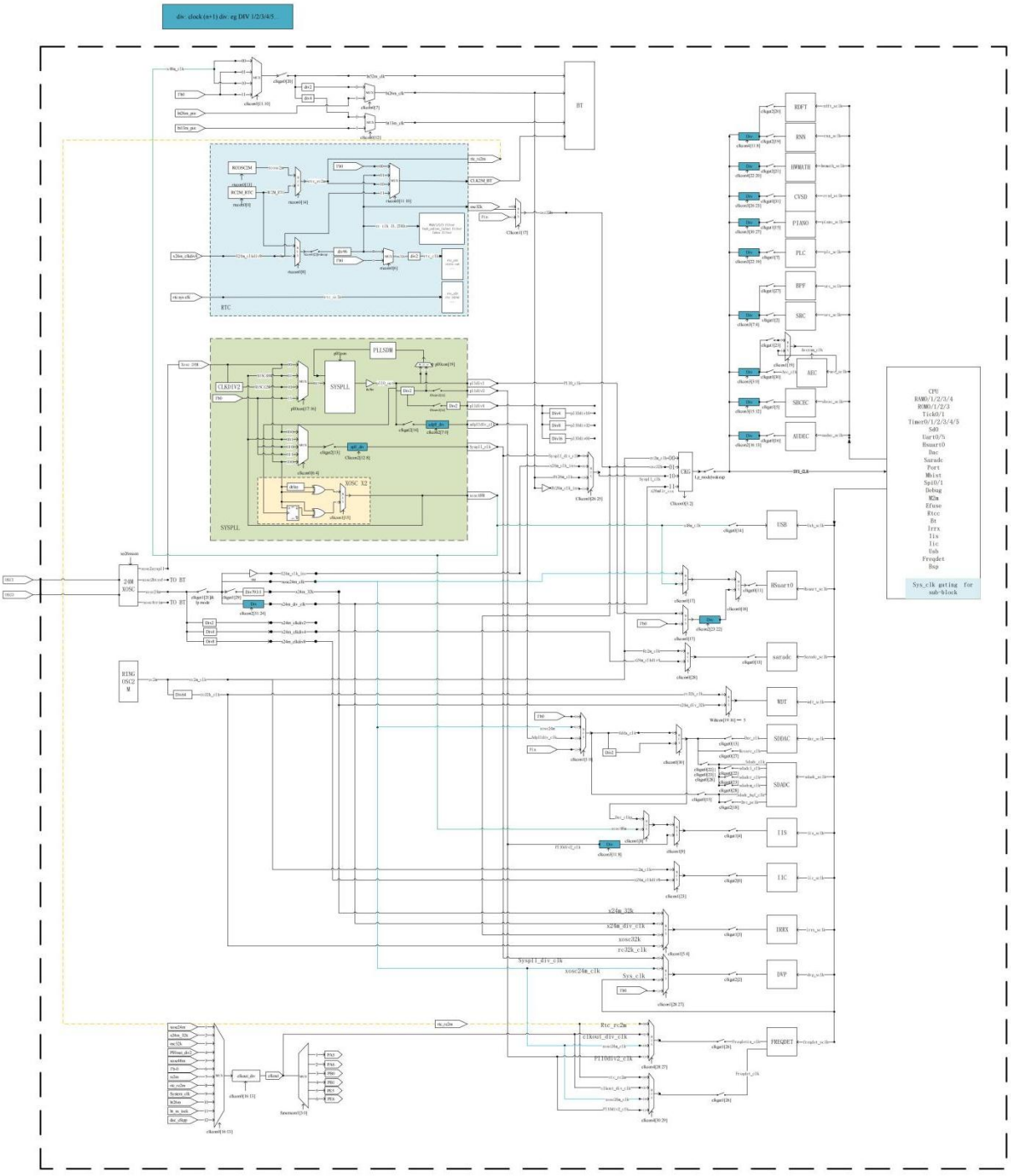


Note: Chx indicates the channel number x

2 System Management

2.1 System Clock

Figure 2 System Clock Tree



For details, see *GW3323_clock.pdf*.

The maximum frequency of this chip is 160MHz, which can be set by set_sys_clk().

Table 1 PLL clock error

PLL clock(Hz)	Value set by PLLCON0	Theoretical output clock	Actual output clock	Error%
2M	configure set_sys_clk()	2M	2M	0
24M	configure set_sys_clk()	24M	24M	0
48M	DI_LDO_SEL(0x2); DI_LDO2_SEL(0x3); DI_EN_XRES(0x0); DI_EN_NOTCH(0x0); DI_EN_TRIM(0x1); DI_EN_TEST_BUF(0x0); DI_CP_SEL(0x2); DI_CP_OFFSET(0x0); DI_LPF_SEL(0x3); DI_VCO_GAIN(0x4); DI_TRIM_VOL(0x4); DI_EN_DIV2(0x0); DI_EN_LDO(0x1); DI_EN_LDO2(0x1);	48	47.99	1%
60M		60	59.99	1%
120M		120	119.99	1%
147M		147	147	0
160M		160	160	0

The clock of all peripheral devices of the chip is divided by the system clock frequency, and the sampling rate can be known according to the clock tree and the baud rate register of each module.

2.2 Clock Register

Figure 3 Clock Register

CLOCK GATING BIT ENABLE									
CLKGAT0		CLKGAT1		CLKGAT2					
0	ROM0	0	-	0	IIC				
1	ROM1	1	-	1	-				
2	RAM0	2	SRC	2	DVP				
3	RAM1	3	IRRX	3	-				
4	RAM2	4	IIS	4	-				
5	RAM3	5	SBCEC	5	-				
6	RAM4	6	-	6	-				
7	-	7	PLC	7	-				
8	TMR0	8	TMR3	8	-				
9	SD0	9	TMR4	9	-				
10	UART0	10	TMR5	10	BDP				
11	HSUT0	11	UART2	11	PLLCLKDIV2				
12	DAC	12	SPI1	12	PLLCLKDIV4				
13	SARADC	13	-	13	SYSPLLDIV				
14	USB	14	-	14	ADDIV				
15	SDADC	15	PIANO	15	DAC				
16	AUDEC	16	TICK0	16	DAC_25				
17	PORT	17	-	17	DACDIV2SEL				
18	MBIST	18	-	18	DRC_PCLK				
19	SPI0	19	-	19	RNN				
20	BT	20	-	20	RDFT				
21	UART1	21	-	21	HWMATH				
22	SDADCL	22	-	22	-				
23	SDADCR	23	AECRAM	23	-				
24	TMR1	24	ROM2	24	-				
25	TMR2	25	ROM3	25	-				
26	RTCC	26	FRQEDET	26	-				
27	RECSRC	27	PBF	27	-				
28	-	28	DBG	28	-				
29	SPF	29	X26M	29	-				
30	AEC	30	M2MDMA	30	-				
31	CVSD	31	EFUSE	31	-				

Register 2-1 CLKCON0: Clock control register

Bit	Name	Mode	Default	Description
31	Sdadc_bqclkx2_sel	WR	0x0	sdadc clk select 0：addc_clk 1: addc_clkdiv2
30	XSOC_x2en_a	WR	0x0	XSOC double enable 0：disable 1：enable
29:28	Sarclk_sel	WR	0x0	Saradc clk select 0x0：rc2m 0x1：XSOC_div4
27:26	—	—	—	—
25	Tmrck_async_sel	WR	0x0	Timer increase clk asynchronization select 0：选择系统时钟同步的 timer increase clk 1：选择 timer increase clk
24:23	Tmrck_sel	WR	0x0	Timer increase clk select 0x0: osc32k 0x1: clkout 0x2: XSOC_div 0x3: rc2m_d0
22	Pllts_oe	WRW	0x0	Pllts clk output enable 0：output disable 1：output enable
21:19	Pllts_sel	WR	0x0	Pllts clk select 0x0: pll0_tsck 0x1: pll1_tsck 0x2: bt_sx_tsck 0x3: 0x0
18:17	Hutclk_sel	WR	0x0	Hsuart clk select Hutclk_sel[0]:0 select pll0_out/xosc ,1 select pll1_out/xosc52m Hutclk_sel[1]：0 select system clock (xosc/xosc_double)，1 select pll_div clk
16:13	Clkout_sel	WR	0x0	clock output select 0x1：xosc 0x2：XSOC_32k 0x3：osc32k 0x4：pll0div2_clk

Bit	Name	Mode	Default	Description
				0x5：xosc52m 0x6：pll1out 0x7：rc2m 0x8：rtc_rc2m 0x9：sys_clk 0xa：bt26m 0xb：bt_sx_tsck 0xc：dac_clk
12	Bt13m_sel	WR	0x0	Bt13m clk select 0：bt13m clk 1：PF0 input
11:10	Bt52m_sel	WR	0x0	Bt52m clk select： 0:xosc52m 1: invalid
9:8	-	-	-	-
7	Bt26m_sel	WR	0x0	Bt26m clock select： 0: select bt26m clk 1: invalid
6:4	Syspll_sel_a	WR	0x0	Sys pll select Syspll_sel[0]:0 select pll_clk,1 select xosc52m Syspll_sel[1]:0 select pll_clk/xosc52m Syspll_sel[2]:0 select pll_clk/xosc52m,1 select xosc26m
3:2	Sysck_sel	WR	0x0	Sys clk clock select 00：rc2m clk 01：osc32k clk 10：pll_div clk 11：XSOC_div clk
0	Rcosc_en_sw	WR	0x1	RCOSC clock software enable 0：RCOSC enable invalid 1：RCOSC enable valid

Register 2-2 CLKGAT0: Clock gate register

Bit	Name	Mode	Default	Description
31	CVSD	WR	0x1	cvsd clk enable bit 0: disable 1: enable

Bit	Name	Mode	Default	Description
30	AEC	WR	0x1	aec clk enable bit 0: disable 1: enable
29	SPF	WR	0x1	spf clk enable bit 0: disable 1: enable
28	SDADCM	WR	0x1	sdadcm clk enable bit 0: disable 1: enable
27	RECSRC	WR	0x1	recsrc clk enable bit 0: disable 1: enable
26	RTCC	WR	0x1	rtc clk enable bit 0: disable 1: enable
25	TMR2	WR	0x1	Timer2 clk enable bit 0: disable 1: enable
24	TMR1	WR	0x1	Timer1 clk enable bit 0: disable 1: enable
23	SDADCR	WR	0x1	saadcr clk enable bit 0: disable 1: enable
22	SDADCL	WR	0x1	Sdadcl clk enable bit 0: disable 1: enable
21	UART1	WR	0x1	Uart1 clk enable bit 0: disable 1: enable
20	BT	WR	0x1	bt clk enable bit 0: disable 1: enable
19	SPI0	WR	0x1	Spi0 clk enable bit 0: disable 1: enable

Bit	Name	Mode	Default	Description
18	MBIST	WR	0x1	mbist clk enable bit 0: disable 1: enable
17	PORT	WR	0x1	port clk enable bit 0: disable 1: enable
16	AUDEC	WR	0x1	audec clk enable bit 0: disable 1: enable
15	SDADC	WR	0x1	sdadc clk enable bit 0: disable 1: enable
14	USB	WR	0x1	usb clk enable bit 0: disable 1: enable
13	SARADC	WR	0x1	saradc clk enable bit 0: disable 1: enable
12	DAC	WR	0x1	dac clk enable bit 0: disable 1: enable
11	HSUT0	WR	0x1	hsuart0 clk enable bit 0: disable 1: enable
10	UART0	WR	0x1	uart0 clk enable bit 0: disable 1: enable
9	SD0	WR	0x1	sd0 clk enable bit 0: disable 1: enable
8	TMR0	WR	0x1	timer0 clk enable bit 0: disable 1: enable
7	—	WR	0x1	
6	RAM4	WR	0x1	Ram4 clk enable bit 0: disable

Bit	Name	Mode	Default	Description
				1: enable
5	RAM3	WR	0x1	Ram3 clk enable bit 0: disable 1: enable
4	RAM2	WR	0x1	Ram2 clk enable bit 0: disable 1: enable
3	RAM1	WR	0x1	Ram1 clk enable bit 0: disable 1: enable
2	RAM0	WR	0x1	Ram0 clk enable bit 0: disable 1: enable
1	ROM1	WR	0x1	Rom1 clk enable bit 0: disable 1: enable
0	ROM0	WR	0x1	Rom0 clk enable bit 0: disable 1: enable

Register 2-3 CLKCON1: Clock control register

Bit	Name	Mode	Default	Description
31	Dvp_pclk_inv_en	WR	0x0	Dvp clk Flip enable： 0x0:disable 0x1：enable
30:29	Dvp_delay_sel	WR	0x0	Dvp clk delay: 0:disable 1:enable
28:27	Dvp_outclk_sel	WR	0x0	Dvp clk select 0x0:syspll_div 0x1:XSOC 0x2:sys_clk
26:25	Splldiv_sel	WR	0x0	Sys pll div select： Syspll div_sel[0]：0 select syspll_div/bt26m,1 select XSOC_clk/bt26m negation Syspll div_sel[1]:0 select syspll_div/XSOC，1 select bt26m/bt26m negation

Bit	Name	Mode	Default	Description
24	Ft_clkpin_sel	WR	0x0	FT clk select 0x0:PF5 0x1:PA7
23	Iicclk_sel	WR	0x0	Iic clk select 0x0: rc2m 0x1: XSOC_div8
22	Pll0sdmsel_a	WR	0x0	-
21	XSOC_lpm_gen	WR	0x0	XSOC_lpm enable bit 0x0: disable 0x1: enable
20	—	—	—	—
19	Aecram_div1_sel	WR	0x0	Aecram div1 select 0x0:disable 0x1:enable
18	Usb6p5_sel	WR	0x0	-
17	K32_tscsel	WR	0x0	Osc32k select 0x0:osc32k 0x1:cp_pin
16	XSOC_dlysel_a	WR	0x0	XSOC clk delay select 0：disable 1：enable
15	X52m_insel_a	WR	0x0	Xo52m pre select 0：xo52m_pre0 1：xo52m_pre1
14	Uartck_sel	WR	0x0	Uart inc clk select 0：XSOC_div 1：XSOC_div4
13:12	Ttck_sel	WR	0x0	Tick inc clk select 0x0: XSOC_div 0x1: pll0div16 0x2: pll0div32 0x3: pll0div64
11:10	Usbpll_sel	WR	0x0	-
9:8	Iisclk_sel	WR	0x0	Iis clk select 0x0：dac_clk

Bit	Name	Mode	Default	Description
				0x1:xosc52m 0x2:dac_clk/xosc52m(depend on iisclk_sel[0]) 0x3:iis_div_clk
7:6	—	—	—	—
5:4	Irrxclksel	WR	0x0	Ir rx clk select 0x0:XSOC_32k 0x1:XSOC_div 0x2:osc32k 0x3:rc32k
3:2	—	—	—	—
1:0	Adda_clksel	WR	0x0	DAC clk select 0x3:adda_clk 0x2:XSOC 0x1:adpll_div_clk 0x0:0

Register 2-1 CLKGAT1: Clock control register

Bit	Name	Mode	Default	Description
31	EFUSE	WR	0x1	efuse clk enable bit 0: disable 1: enable
30	M2MDMA	WR	0x1	M2m clk enable bit 0: disable 1: enable
29	XSOC	WR	0x0	XSOC clk enable bit 0: disable 1: enable
28	DBG	WR	0x1	dbg clk enable bit 0: disable 1: enable
27	PBF	WR	0x1	pbf clk enable bit 0: disable 1: enable
26	FRQEDET	WR	0x1	freqdet clk enable bit 0: disable 1: enable

Bit	Name	Mode	Default	Description
25	ROM3	WR	0x1	Rom3 clk enable bit 0: disable 1: enable
24	ROM2	WR	0x1	Rom2 clk enable bit 0: disable 1: enable
23	AECRAM	WR	0x1	aecram clk enable bit 0: disable 1: enable
22:17	—	—	0x1	—
16	TICK0	WR	0x1	Tick0 clk enable bit 0: disable 1: enable
15	PIANO	WR	0x1	piano clk enable bit 0: disable 1: enable
14:13	—	—	0x1	—
12	SPI1	WR	0x1	Spi1 clk enable bit 0: disable 1: enable
11	UART2	WR	0x1	Uart2 clk enable bit 0: disable 1: enable
10	TMR5	WR	0x1	Timer5 clk enable bit 0: disable 1: enable
9	TMR4	WR	0x1	Timer4 clk enable bit 0: disable 1: enable
8	TMR3	WR	0x1	Timer3 clk enable bit 0: disable 1: enable
7	PLC	WR	0x1	plc clk enable bit 0: disable 1: enable

Bit	Name	Mode	Default	Description
6	—	—	0x1	—
5	SBCEC	WR	0x1	sbcec clk enable bit 0: disable 1: enable
4	IIS	WR	0x1	iis clk enable bit 0: disable 1: enable
3	IRRX	WR	0x1	Irrx clk enable bit 0: disable 1: enable
2	SRC	WR	0x1	src clk enable bit 0: disable 1: enable
1:0	—	—	0x1	—

Register 2-4 CLKCON2: Clock control register

Bit	Name	Mode	Default	Description
31:24	XSOC_div	WR	0x0	External crystal oscillator clock for divide
23:22	Hut_div	WR	0x0	HSUART divide
21:17	Clkout_div	WR	0x0	CLKOUT divide
16:13	Audec_div	WR	0x0	AUDECPLL divide
12:8	Syspll_div	WR	0x0	SYSPLL divide
7:4	Adpll_div	WR	0x0	ADPLL divide
3:0	Btpll_div	WR	0x0	BTPLL divide

Register 2-1 CLKGAT2: Clock gate register

Bit	Name	Mode	Default	Description
31:22	—	—	0x1	—
21	HWMATH	WR	0x1	hwmath clk enable bit 0: disable 1: enable
20	RDFT	WR	0x1	rdft clk enable bit 0: disable 1: enable

Bit	Name	Mode	Default	Description
19	RNN	WR	0x1	rnn clk enable bit 0: disable 1: enable
18	DRC_PCLK	WR	0x1	drc pclk enable bit 0: disable 1: enable
17	DACDIV2SEL	WR	0x0	Dacdiv2 clk enable bit 0: disable 1: enable
16	DAC_25	WR	0x0	Dac25 clk enable bit 0: disable 1: enable
15	DAC	WR	0x0	dac clk enable bit 0: disable 1: enable
14	ADDIV	WR	0x1	adddiv clk enable bit 0: disable 1: enable
13	SYSPLLDIV	WR	0x1	sysplldiv clk enable bit 0: disable 1: enable
12	PLLCLKDIV4	WR	0x1	Plldiv4 clk enable bit 0: disable 1: enable
11	PLLCLKDIV2	WR	0x1	Plldiv2 clk enable bit 0: disable 1: enable
10	BDP	WR	0x1	bsp clk enable bit 0: disable 1: enable
9:3	—	WR	0x1	Unused
2	DVP	WR	0x1	dvp clk enable bit 0: disable 1: enable
1	—	WR	0x1	Unused

Bit	Name	Mode	Default	Description
0	IIC	WR	0x1	iic clk enable bit 0: disable 1: enable

Register 2-5 CLKCON3: Clock control register

Bit	Name	Mode	Default	Description
26:23	piano_div	WR	0x0	PIANO divide
26:23	cvsd_div	WR	0x0	CVSD divide
22:19	plc_div	WR	0x0	PLC divide
18:16	usb_div	WR	0x0	USB divide
15:12	sbcec_div	WR	0x0	SBCEC divide
11:8	iis_div	WR	0x0	IIS divide
7:4	src_div	WR	0x0	SRC divide
3:0	aec_div	WR	0x0	AEC divide

Register 2-6 CLKCON4: Clock control register

Bit	Name	Mode	Default	Description
26:23	Dvp_out_div	WR	0x0	DVP_OUT divide
22:20	hwmath_div	WR	0x0	HWMATH divide
19:16	dvp_div	WR	0x0	DVP divide
11:8	rnn_div	WR	0x0	RNN divide
6:0	Btlp_div	WR	0x0	BTLP divide

3 Memory Access

Table 2 Memory Access

Number	Channel	SRAM0~2	SRAM3~4	Cache RAM	AEC RAM	AUDEC RAM	SBCENC RAM	RDFT RAM	RNN RAM
0	USB BT_RFTS	RW	-	-	RW	-	-	-	-
1	BT	RW	RW	-	-	-	-	-	-
2	AUBUF AUBUF1 ANCDAC DACDMAO	RW	RW	RW	RW	RW	RW	-	-
3	SDADC	W	W	W	W	W	W	-	-
4	SD SPI0 SPI1 IIS	RW	RW	RW	RW	RW	RW	RW	-
5	CVSD EQ PSRC PLC	RW	RW	-	RW	RW	RW	-	-
6	RDFT	RW	RW	-	-	-	-	-	-
7	AUDEC SBCENC	RW	RW	-	-	RW	-	-	-
8	GPDMA BSP	RW	RW	-	RW	RW	RW	-	-
9	HSUT DVP SPF	RW	RW	-	-	-	-	-	-
10	-	-	-	-	-	-	-	-	-
11	-	-	-	-	-	-	-	-	-
12	-	-	-	-	-	-	-	-	-
13	-	-	-	-	-	-	-	-	-
14	-	-	-	-	-	-	-	-	-
15	-	-	-	-	-	-	-	-	-

DMA supported peripherals:HSUART, SPI, SD, USB and IIS,SDADC,SDDAC.

Note:

（1） RDFT refers to Fourier transform.

（2） GPDMA refers to general-purpose DMA, used to transmit data between peripherals and/or memory through a linked list;

（3） BSP is Bit stream pickup;

（4） DVP, DVP interface, is one of standard protocols for cameras;

（5） SPF (abbreviation of SPDIF) is optical audio output;

4 Low Power Mode

GW3323 supports two low-power modes：

4.1 1. sleep mode,500uA

Sleep mode will auto gate system clock, close memory access, close RC2M, but some asyn clock should be disable by software.

Sleep mode wake up source as follow. After wakeup, software run continue or enter interruptif enable.

- BT wakeup
- port external interrupt wakeup(PA7,PB1,PB2,PB3,PB4,PB5,INT_FALL,INT_RISE)
- RTC 1s or alarm wakeup

4.1.1 Configuration of sleep mode

(1) Set sleep conditions

(2) Wait for sleep conditions to be met

(3) Save the status of the peripheral before entering sleep mode

(4) After the statuses of all IO ports are remembered, set all IO ports to input mode, adjust the system clock to 24MHz, and disable PLL

(5) Configure wake-up conditions

(6) Enter sleep mode

(7) Wait to be awakened

(8) Recover the IO port status and restore the system clock

(9) Disable wake-up conditions

4.2 Power-off mode (power off, 4uA)

The wake-up source of power-off mode is shown below. The chip is reset after wake-up.

- Wake-up through external interrupt level of ports (VUSB, PB0, PB1, PB2, PB5)
- RTC 1s or alarm wake-up

Note:

（1） To enter low-power mode when SARADC mode is turned on, it needs to be manually turned off.

（2） There is no power supply in the chip in power-off mode, so resetting the circuit connected to the Reset pin is invalid at this time.

（3） When the chip crashes in power-off mode, it needs to be powered off and restarted, or be designed as "Reset can control and Vbat decreases" in hardware circuit, achieving the same effect as power-off restart.

4.2.1 Configuration of power-down mode

(1) Set the conditions for power-off mode

(2) Wait for the power-down mode conditions to be met

(3) Set the buck circuit module

(4) Turn off all IO port configurations

(5) Turn off the clock

(6) Configure wake-up conditions

(7) Enter the sleep mode

(8) Wait to be awakened

(9) Reset after wake-up

5 Interrupts

5.1 Feature

The Bluetooth priority has been set in the library, from high to low: Bluetooth> timers and other hardware interrupts>threading.

The print function in the interrupt shall be the printk function, the global variable inside shall be declared volatile, and the interrupt function shall be placed in the isr area; i.e. add AT (.com_text.isr). Below is an example:

```
volatile u32 duty_n;
AT(.com_text.isr)
void timer3_isq(void)
{
    if (tmr_get_flag(TMR3, TMR_FLAG_UPDATE2) != RESET) {
        tmr_clear_flag(TMR3, TMR_FLAG_UPDATE2);
        gpio_toggle_bits(GPIOB_REG, GPIO_PIN_2);
        TMR3->duty1 = duty_n;
        printk("Tim3 done\r\n");
    }
}
```

5.2 interrupt vector table

Table 3 Exception vectors

Interrupt number	Address	Description
0	0x00	Reset
1	0x04	1.read Instruction error (invalid memory access) 2. Illegal instructions 3.LSUerror (invalid memory access) 4. Ebreak instruction
2	0x08	hardware breakpoint
3	0x0c	Expand interrupt instructions
4	0x10	Low priority interrupt
5	0x14	Watch point interrupt
6	0x18	-
7	0x1c	-
8	0x20~0x9c	High priority interrupt(see the following table)

Table 4 High priority interrupt vectors

Interrupt number	Address	Description

Interrupt number	Address	Description
0	0x20	Icache miss interrupt Dcache Miss interrupt
1	0x24	BT interrupt BLE interrupt BTDM interrupt
2	0x28	Software interrupt
3	0x2c	Timer0 interrupt
4	0x30	Timer1 interrupt
5	0x34	Timer2 interrupt
6	0x38	IR receiver interrupt
7	0x3c	USB interrupt
8	0x40	SD interrupt
9	0x44	Audio buffer 0 interrupt Audio buffer 1 interrupt
10	0x48	SDADC DMA interrupt PBF DMA(EQ) interrupt
11	0x4c	Audio codec interrupt SBC code interrupt AEC FFT interrupt
12	0x50	PLC interrupt CVSD interrupt
13	0x54	Piano interrupt SDADC sampling interrupt
14	0x58	UART0 interrupt UART1 interrupt UART2 interrupt
15	0x5c	HSUART interrupt DVP interrupt
16	0x60	Timer3 interrupt
17	0x64	Timer4 interrupt
18	0x68	Timer5 interrupt
19	0x6c	GPDMA interrupt
20	0x70	SPI0 interrupt SPI1 interrupt
21	0x74	UART0 key match interrupt UART1 key match interrupt UART2 key match interrupt
22	0x78	BT Match interrupt modem TX interrupt BT Match interrupt modem RX interrupt
23	0x7c	Frequency detection interrupt Touch key interrupt
24	0x80	DMA output interrupt RNN interrupt RDFT interrupt HWMATH interrupt

Interrupt number	Address	Description
25	0x84	SRC interrupt
26	0x88	Port interrupt
27	0x8c	IIS interrupt
28	0x90	SARADC interrupt
29	0x94	RTC and alarm interrupt LVD interrupt WDT interrupt
30	0x98	IIC interrupt BSP interrupt
31	0x9c	Tick0 interrupt Tick1 interrupt

Note:

（1） The related interrupts can be configured by calling sys_irq_init (int vector, int pr, isr_t isr), and this part is not open to customers. Vector is 0~31; pr is 0; isr is interrupt entry function.The CPU processes the same interrupt priority based on their order in the interrupt vector table.

（2） The interrupt priority of Bluetooth configured in system library is 1, and the interrupt priority of USB is 0.

（3） The same interrupt number can only have 1 interrupt function. For example, start uart0~2, and to make them all respond to serial port interrupts, the interrupt handling functions of these three serial ports shall be written in one function.

5.3 Interrupts Special Registers

Register 5-1 PICCON: Peripheral interrupt control Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused
16	GIEM	WR	1	Global interrupt enable mask bit 0: disable interrupt 1: enable interrupt
15:7	-	-	-	Unused
6:5	HPSDEN	WR	0x0	High priority shadow register select bit 00: high priority 01: high priority 2 10/11: high priority 3
4:2	-	-	-	Unused
1	LPINTEN	WR	0	Low priority interrupt enable bit 0: disable 1: enable
0	GIE	WR	0	Global interrupt enable bit 0: disable interrupt 0: disable interrupt

Register 5-2 PICCONSET: Peripheral interrupt control set Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused

Bit	Name	Mode	Default	Description
16	GIEM	W	0	Write 1 enable Global interrupt enable mask
15:8	-	-	-	Unused
7:3	-	-	-	Unused
2	HPINTEN	W	0	Write 1 enable High priority interrupt
1	LPINTEN	W	0	Write 1 enable Low priority interrupt
0	GIE	W	0	Write 1 enable Global interrupt

Register 5-3 PICCONCLR: Peripheral interrupt control clear Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused
16	GIEMDIS	W	0	Write 1 disable Global interrupt enable mask
15:8	-	-	-	Unused
7:3	-	-	-	Unused
2	HPINTDIS	W	0	Write 1 disable High priority interrupt
1	LPINTDIS	W	0	Write 1 disable Low priority interrupt
0	GIEDIS	W	0	Write 1 disable Global interrupt

Register 5-4 PICEN: Peripheral interrupt enable Register

Bit	Name	Mode	Default	Description
31:0	IntEN	WR	0x0	Interrupt 31 to 0 enable bit 0: disable 1: enable

Register 5-5 PICENSET: Peripheral interrupt enable set Register

Bit	Name	Mode	Default	Description
31:0	IntEN	W	0x0	Write 1 enable Interrupt 31 to 0

Register 5-6 PICENCLR: Peripheral interrupt enable clear Register

Bit	Name	Mode	Default	Description
31:0	IntDIS	W	0x0	Write 1 disable Interrupt 31 to 0

Register 5-7 PICPR: Peripheral high priority interrupt selection Register

Bit	Name	Mode	Default	Description
31:0	IntPR	WR	0x0	Interrupt 31 to 0 priority selection bit 0: low priority interrupt 1: high priority interrupt

Register 5-8 PICPR1: Peripheral high priority interrupt selection Register1

Bit	Name	Mode	Default	Description
31:0	IntPR1	WR	0x0	Interrupt 31 to 0 priority selection 1 bit; {PICPR1, PICPR} 00: low priority interrupt 01: high priority interrupt 10: high priority 2 interrupt 11: high priority 3 interrupt

Register 5-9 PICADR: Peripheral interrupt address Register

Bit	Name	Mode	Default	Description
31:8	BADR	WR	0x800	Interrupt entry address
7:0	-	-	0x0	

Register 5-10 PICPND: Peripheral interrupt pending Register

Bit	Name	Mode	Default	Description
31:3	IntPND[31:3]	R	0x0	Interrupt 31 to 3 pending bit 0: no interrupt pending 1: interrupt pending
2	SWIPND	WR	0	Software interrupt pending. Write 1 will clear software interrupt pending
1:0	IntPND[1:0]	R	0x0	Interrupt 1 to 0 pending bit 0: no interrupt pending 1: interrupt pending

6 WatchDog

6.1 User Guide

(1) configure WDT reset or interrupt

(2) Select WDT time out

(3) Clear WDT

6.2 WDT Special Function Registers

Register 6-1 WDTCON: WDT Control Register

Bit	Name	Mode	Default	Description
31	WDTPND	R	0	WDT time out pending 0: no pending 1: pending
30:28	-	-	-	Unused
27:24	TMRSEL_WR	W	0	WDT time select bit write enable When write 0xa, bit20~bit22 can be write to TMRSEL, other value will no affect
23	-	-	-	Unused
22:20	TMRSEL	R	0x4	WDT time select bit 000: 1ms 001: 256ms 010: 512ms 011: 1024ms 100: 2048ms 101: 4096ms 110: 8192ms 111: 16384ms
19:16	WDTCSEL_WR	W	0	WDT clock select When write 0xa, WDTCSEL =0, when write 0x5, WDTCSEL =1. other value will no affect
16	WDTCSEL	R	0	WDT clock select bit 0:RC32K 1:X32K from 26M divider
15:12	WDTIE_WR	W	0	WDT interrupt disable When write 0xa, WDTIE will disable, when write 0x5, WDTIE will enable. other value will no affect
12	WDTIE	R	0	WDT interrupt enable bit 0: Disable 1: Enable
11:8	WDTRSTEN_WR	W	0	WDT reset disable When write 0xa, WDTRSTEN will disable, other value will no affect
8	WDTRSTEN	WR	1	WDT reset enable bit 0: Disable 1: Enable
7:4	WDTEN_WR	W	0	WDT disable When write 0xa, WDTEN will disable, other value will no affect
4	WDTEN	WR	1	WDT enable bit 0: Disable 1: Enable

3:0	WDTCLR	W	0	WDT clear bit When write 0xa, WDT counter and WDTPND will be clear

7 GPIO Management

7.1 Features

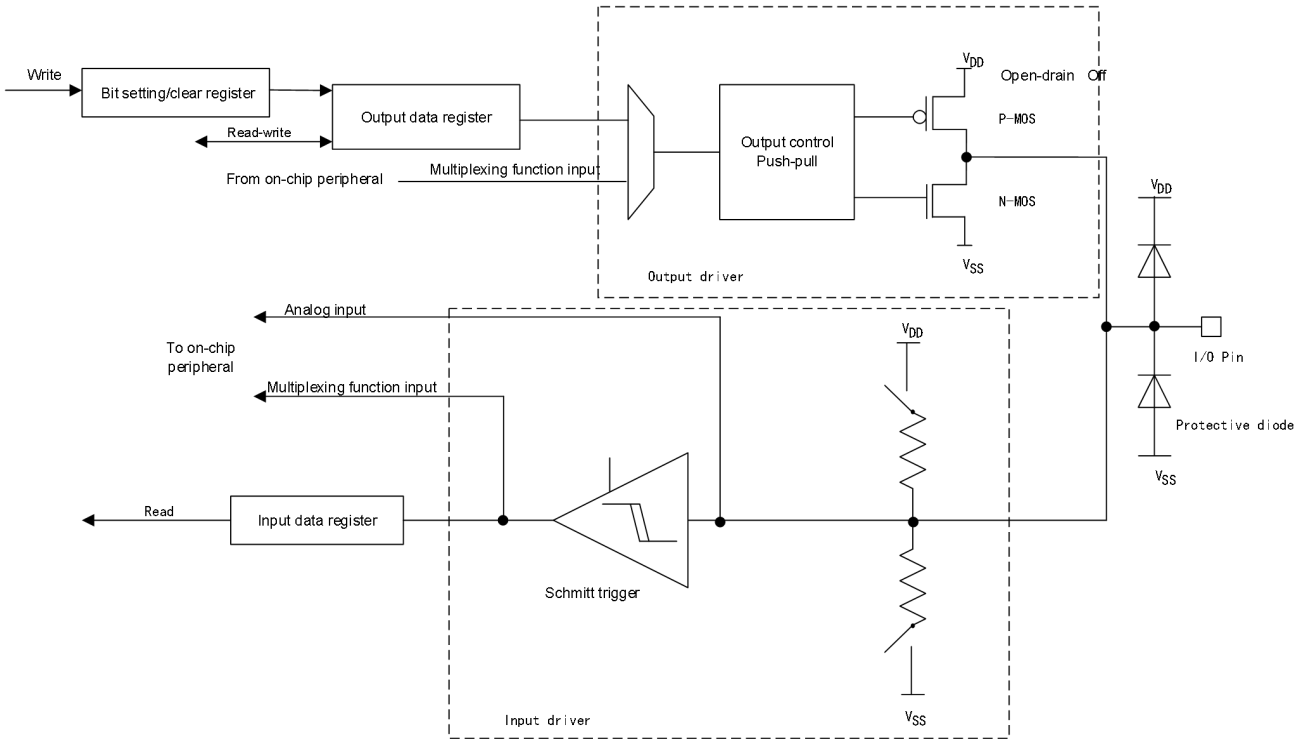
(1) Control GPIO input/output direction by using direction register;

(2) Internal pull-up/pull-down resistor by using pull-up/pull-down resistor control register;

(3) Select suitable output driving current capability;

7.2 GPIO internal block diagram

Figure 4 GPIO internal block diagram



7.3 GPIO general control register

Register 7-1 GPIOA: Port A data Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOA	WR	0x00	PAx data. Valid when PAx is used as GPIO 0: PAx is input low state when read and output low at PAx when write; 1:PAx is input high state when read and output high at PAx when write

Register 7-2 GPIOASET: Port A Set output data Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOASET	WO	X	Set PAx output data. Write 1 set output data. Write 0 affect nothing.

Register 7-3 GPIOACLR: Port A clear output data Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOACLR	WO	X	Clear PAx output data. Write 1 clear output data. Write 0 affect nothing.

Register 7-4 GPIOADIR: Port A direction Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOADIR	WR	0xFF	PAx direction control 0: Output 1: Input

Register 7-5 GPIOAPU: Port A pull-up Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOAPU	WR	0x0	PAx 10KΩ pull-up resister control. Valid when PAx is used as input 0: disable 1: enable

Register 7-6 GPIOAPD: Port A pull-down resister Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOAPD	WR	0x0	PAx 10KΩ pull-down resister control. Valid when PAx is used as input 0: disable 1: enable

Register 7-7 GPIOAPU200K: Port A pull-up resister Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOAPU	WR	0x0	PAx 200KΩ pull-up resister control. Valid when PAx is used as input 0: disable 1: enable

Register 7-8 GPIOAPD200K: Port A pull-down resister Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused

Bit	Name	Mode	Default	Description
7:0	GPIOAPD	WR	0x0	PAx 200KΩ pull-down resister control. Valid when PAx is used as input 0: disable 1: enable

Register 7-9 GPIOAPU300: Port A pull-up resister Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOAPU	WR	0x0	PAx 300Ω pull-up resister control. Valid when PAx is used as input 0: disable 1: enable

Register 7-10 GPIOAPD300: Port A pull-down resister Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOAPD	WR	0x0	PAx 300Ω pull-down resister control. Valid when PAx is used as input 0: disable 1: enable

Register 7-11 GPIOADE: Port A digital function enable register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOADE	WR	0xFF	PAx digital function enable 0: Port used as analog IO 1: Port used as digital IO

Register 7-12 GPIOAFEN: Port A function mapping enable register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOAFEN	WR	0xFF	PAx function mapping enable 0: Port used as GPIO 1: Port used as function IO

Register 7-13 GPIOADRV: Port A output driving select Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	GPIOADRV	WR	0x0	PAx output driving select 0: 8mA 1: 32mA

7.4 GPIO function mapping

Register 7-14 FUNCMCON0: Port function mapping control Register 0

Bit	Name	Mode	Default	Description
31:28	UT1RXMAP	WR	0x0	UART1 RX mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to TX pin by UT1TXMAP select 1111: Clear these bits Others is reserved
27:24	UT1TXMAP	WR	0x0	UART1 TX mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 1111: Clear these bits Others is reserved
23:20	-	-	-	-
19:16	-	-	-	-
15:12	UT0RXMAP	WR	0x0	UART0 RX mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 0101: map to G5 0110: map to G6 0111: map to TX pin by UT0TXMAP select 1111: Clear these bits Others is reserved
11:8	UT0TXMAP	WR	0x0	UART0 TX mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 0101: map to G5 0110: map to G6 0111: map to G7 1000: map to G8 1111: Clear these bits Others is reserved
7:4	SPI0MAP	WR	0x0	SPI0 mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 1111: Clear these bits Others is reserved
3:0	SD0MAP	WR	0x0	SD0 mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 0101: map to G5 0110: map to G6 1111: Clear these bits Others is reserved

Register 7-15 FUNCMCON1: Port function mapping control Register 1

Bit	Name	Mode	Default	Description
31:12	—	—	—	—
11:8	UT2RXMAP	WR	0x0	UART2 RX mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to TX pin by UT2TXMAP select 1111: Clear these bits Others is reserved
7:4	UT2TXMAP	WR	0x0	UART2 TX mapping 0000: no affect 0001: map to G1 0010: map to G2 1111: Clear these bits Others is reserved
3:0	—	—	—	—

Register 7-16 FUNCMCON2: Port function mapping control Register 2

Bit	Name	Mode	Default	Description
31:28	-	-	-	Unused
27:24	IICMAP	WR	0x0	IIC mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 0101: map to G5 0110: map to G6 0111: map to G7 1000: map to G8 1111: Clear these bits Others is reserved
23:20	IRMAP	WR	0x0	IR mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 0101: map to G5 0110: map to G6 0111: map to G7 1000: map to G8 1001: map to G9 1111: Clear these bits Others is reserved
19:16	TMR5MAP	WR	0x0	Timer5 PWM mapping 0000: no affect 0001: map to G1 1111: Clear these bits Others is reserved
15:12	TMR4MAP	WR	0x0	Timer4 PWM mapping 0000: no affect 0001: map to G1 1111: Clear these bits Others is reserved
11:8	TMR3MAP	WR	0x0	Timer3 PWM mapping 0000: no affect

Bit	Name	Mode	Default	Description
				0001: map to G1 1111: Clear these bits Others is reserved
7:4	TMR3CPTMAP	WR	0x0	Timer3 capture Pin mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 0101: map to G5 0110: map to G6 0111: map to G7 1111: Clear these bits Others is reserved
3:0	-	-	-	-

Register 7-17 FUNCMCON3: Port function mapping control Register 3

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:4	MPDMMAP	WR	0x0	MPDM interface mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 1111: Clear these bits Others is reserved
3:0	PDMMAP	WR	0x0	PDM interface mapping 0000: no affect 0001: map to G1 0010: map to G2 0011: map to G3 0100: map to G4 1111: Clear these bits Others is reserved

7.5 External Port interrupt wake up

Support eight wakeup source input, as the following table. Wakeup circuit 6 and wakeup circuit 7 is special for 32 port interrupts wake up.

Port interrupt source is:

Port_intsrc = {PG[4:0], PF[5:0], PE[7:0], PB[4:0], PA[7:0]};

//PG4 indicates port interruption 31 and PA0 indicates port interruption 0.

Note: When awakening along the rising or falling edge of PA0 to PA6,PB0, PE0 to PE7,PF0 to PF5, and PG0 to PG4 is selected, the PORTINTEN and PORTINTEDG registers need to be set.

Table 5 External Port interrupt wake up

Wakeup source	Wakeup circuit
PA7	Wakeup circuit 0
PB1	Wakeup circuit 1

PB2	Wakeup circuit 2
PB3	Wakeup circuit 3
PB4	Wakeup circuit 4
WKO(PB5)	Wakeup circuit 5
PORT_INT_FALL	Wakeup circuit 6
PORT_INT_RISE	Wakeup circuit 7

Register 7-18 WKUPCON: Wake up control Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused
16	WKIE	WR	0	Wake up interrupt enable 0: disable 1: enable
15:8	-	-	-	Unused
7:0	WKEN	WR	0x0	Wake up input 7~0 enable 0: disable 1: enable

Register 7-19 WKUPEDG: Wake up edge select Register

Bit	Name	Mode	Default	Description
31:24	-	-	-	Unused
23:16	WKPND	R	0x0	Wake up input 7~0 pending 0: no pending 1: wake up pending
15:8	-	-	-	Unused
7:0	WKEDG	WR	0x0	Wake up input 7~0 wakeup edge select 0: rising edge 1: falling edge

Register 7-20 WKUPCPND: Wake up clear pending Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
23:16	WKCPND	W	0x0	Wake up input 7~0 clear pending 0: no affect 1: clear wake up pending

Bit	Name	Mode	Default	Description
15:0	-	-	-	Unused

Register 7-21 PORTINTEN: Port interrupt enable Register

Bit	Name	Mode	Default	Description
31:0	PORTINTEN	WR	0x0	Port interrupt 0~31 enable bit 0: disable 1: enable

Register 7-22 PORTINTEDG: Port interrupt edge select Register

Bit	Name	Mode	Default	Description
31:0	PORTINTEDG	WR	0x0	Port interrupt 0~31 edge select bit 0: rise edge 1: fall edge

8 DMA

8.1 Feature

(1) DMA supports such peripherals as HSUART, SPI, SDIO, and USB.

(10) DMA cannot transmit data between registers or transfer data between two memory addresses.

(11) DMA-SPI only supports transmitting and does not support receiving

8.2 Functional configuration

Table 6 Functions

DMA module	DMA initialization function	DMA receiving function	DMA transmitting function
SDIO	-	sdio_read_data()	sdio_write_data()
HSUART	hsuart_init();	hsuart_dma_start()	hsuart_dma_start()
SPI	-	N/A	spi_set_dma_addr(); spi_set_dma_cnt()
USB	Packaged, not visible to users	Packaged, not visible to users	usb_bulk_send()

9 Timer

9.1 Features

(1) Timer0/1/2, only support 32bit timer function

(2) Timer3/4/5, can be configured to Timer-mode, Counter-mode, Capture-mode and PWM-mode

(3) When the main frequency of Timer is 160MHz, select the system clock and the timer frequency can be set to 160MHz.

(4) The official SDK with the system uses timer0 as tick, with a frequency of 1MHz.

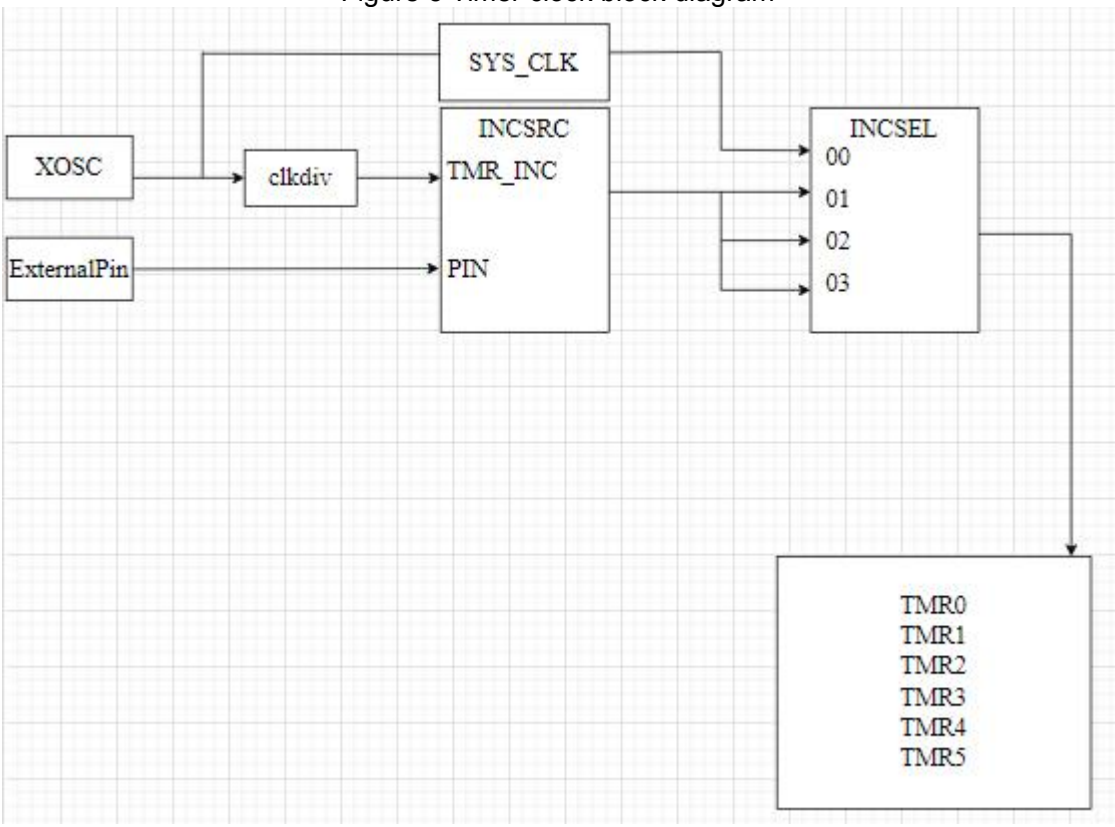
(5) The PWM of GW3323 timer is in edge-aligned mode (the pulse counter is in cycle count-up with an initial count value of 0), without center-aligned mode (the pulse counter is in bidirectional counting with an initial count value of 0), and without complementary output mode.

Table 6 T I/O ports corresponding to timer channels

Timer	Channel G1	Channel G2	Channel G3	Channel G4	Channel G5	Channel G6	Channel G7
Timer3-pwm0	PB0	PB3	PF0	PE0	-	-	-
Timer3-pwm1	PB1	PB4	PA3	PE4	-	-	-
Timer3-pwm2	PB2	PB5	PA4	-	-	-	-
Timer4-pwm0	PE5	PF1	-	-	-	-	-
Timer4-pwm1	PE6	PF2	-	-	-	-	-
Timer4-pwm2	PE7	PF3	-	-	-	-	-
Timer5-pwm0	PA5	PF4	-	-	-	-	-
Timer5-pwm1	PA6	PF5	-	-	-	-	-
Timer5-pwm2	PA7	-	-	-	-	-	-
Timer3-CPT(Capture Channel)	PA5	PA6	PB0	PB1	PE0	PE5	PE6
Timer4-CPT(Capture Channel)	PE7	-	-	-	-	-	-
Timer5-CPT(Capture Channel)	PF1	-	-	-	-	-	-

9.2 Timer clock select

Figure 5 Timer clock block diagram



9.3 Timer0/1/2 Special Function Registers

Register 9-1 TMR0CON/TMR1CON/TMR2CON: Timer0/1/2 Control Register

Bit	Name	Mode	Default	Description
31:10	-	-	-	Unused
9	TPND	WR	0	Timer overflow pending 0: not overflow 1: overflow
8	-	-	-	Unused
7	TIE	WR	0	Timer overflow interrupt enable 0: disable 1: enable
6	INCSRC	WR	0	Increase source select 0: select TMR_INC 1: select external PIN
5:4	-	-	-	Unused
3:2	INCSEL	WR	0x0	Increase clock selection 00: System Clock 01: Counter input rising 10: Counter input falling 11: Counter input edge
1	-	-	-	Unused

Bit	Name	Mode	Default	Description
0	TMREN	WR	0	Timer Enable Bit 0: Disable 1: Enable

Register 9-2 TMR0CPND/TMR1CPND/TMR2CPND: Timer0/1/2 clear pending Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
9	TPCLR	W	0	Timer overflow pending clear bit 0: inactive 1: clear pending
8:0	-	-	-	Unused

Register 9-3 TMR0CNT/TMR1CNT/TMR2CNT: Timer0/1/2 counter Register

Bit	Name	Mode	Default	Description
31:0	TMRCNT	WR	0x0	Timer counter. TMRCNT will increase when timer is enabled. It overflows when TMRCNT = TMRPR, TMRCNT will be clear to 0x0000 when overflow, and the interrupt flag will be set '1'.

Register 9-4 TMR0PR/TMR1PR/TMR2PR: Timer0/1/2 period Register

Bit	Name	Mode	Default	Description
31:0	TMRPR	WR	0xffffffff	Timer period = TMRPR + 1

9.4 Timer3/4/5 Special Function Registers

Register 9-5 TMR3CON/TMR4CON/TMR5CON: Timer3/4/5 Control Register

Bit	Name	Mode	Default	Description
31:18	-	-	-	Unused
17	CPND	WR	0	Timer capture pending 0: not capture 1: capture
16	TPND	WR	0	Timer overflow pending 0: not overflow 1: overflow
15:12	-	-	-	Unused
11	PWM2EN	WR	0	Timer pwm2 enable bit 0: disable 1: enable
10	PWM1EN	WR	0	Timer pwm1 enable bit 0: disable 1: enable
9	PWM0EN	WR	0	Timer pwm0 enable bit 0: disable 1: enable
8	CIE	WR	0	Timer capture interrupt enable 0: disable 1: enable

Bit	Name	Mode	Default	Description
7	TIE	WR	0	Timer overflow interrupt enable 0: disable 1: enable
6	INCSRC	WR	0	Increase source select 0: select TMR_INC 1: select external PIN
5:4	CPTEDSEL	WR	0x0	Timer Capture edge select 00: No Capture 01:Capture PIN rising edge 10: Capture PIN falling edge 11: Capture PIN edge
3:2	INCSEL	WR	0x0	Increase clock selection 00: System Clock 01: Counter input rising 10: Counter input falling 11: Counter input edge
1	CPTEN	WR	0	Timer capture Enable Bit 0: Disable 1: Enable
0	TMREN	WR	0	Timer Enable Bit 0: Disable 1: Enable

Register 9-6 TMR3CPND/TMR4CPND/TMR5CPND: Timer3/4/5 clear pending Register

Bit	Name	Mode	Default	Description
31:18	-	-	-	Unused
17	CPCLR	W	0	Capture pending clear bit 0: inactive 1: clear pending
16	TPCLR	W	0	Timer overflow pending clear bit 0: inactive 1: clear pending
15:0	-	-	-	Unused

Register 9-7 TMR3CNT/TMR4CNT/TMR5CNT: Timer3/4/5 counter Register

Bit	Name	Mode	Default	Description
31:0	TMRCNT	WR	0x0	Timer counter. TMRCNT will increase when timer is enabled. It overflows when TMRCNT = TMRPR, TMRCNT will be clear to 0x0000 when overflow, and the interrupt flag will be set ‘1’.

Register 9-8 TMR3PR/TMR4PR/TMR5PR: Timer3/4/5 period Register

Bit	Name	Mode	Default	Description
31:0	TMRPR	WR	0xffffffff	Timer period = TMRPR + 1

Register 9-9 TMR3CPT/TMR4CPT/TMR5CPT: Timer3/4/5 capture value Register

Bit	Name	Mode	Default	Description
31:0	TMRCPT	R	x	Timer capture value

Register 9-10 TMR3DUTY0/TMR4DUTY0/TMR5DUTY0: Timer3/4/5 pwm0 duty Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	TMRDUTY0	W	x	Timer pwm0 duty PWM0 low level length is TMRDUTY0+1 PWM 0 high level length is TMRPR-TMRDUTY0

Register 9-11 TMR3DUTY1/TMR4DUTY1/TMR5DUTY1: Timer3/4/5 pwm1 duty Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	TMRDUTY1	W	x	Timer pwm1 duty PWM1 low level length is TMRDUTY1+1 PWM1 high level length is TMRPR-TMRDUTY1

Register 9-12 TMR3DUTY2/TMR4DUTY2/TMR5DUTY2: Timer3/4/5 pwm2 duty Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	TMRDUTY2	W	x	Timer pwm2 duty PWM2 low level length is TMRDUTY2+1 PWM2 high level length is TMRPR-TMRDUTY2

10 RTC

10.1 Features

(1) Support 32bit Independent power supply real time counter

(2) Support alarm interrupt and second interrupt

10.2 Special Function Registers

Register 10-1 RTCCON: RTC Control Register

Bit	Name	Mode	Default	Description
31:23	-	-	-	Unused
22	INBOX	R	0	INBOX state 0: out of box 1: in box
21	VUSBOFF	R	0	VUSB off state 0: online 1: off state
20	VUSBONLINE	R	0	VUSB online state 0: not online 1: online
19	RTCWKP	R	0	RTC WK pin state 0: WK pin state is 0 1: WK pin state is 1
18	RTCWKSLPPND	R	0	RTC wakeup sleep pending 0: no pending 1: pending
17	ALMPND	R	0	RTC alarm pending 0: no pending 1: alarm pending
16:9	-	-	-	Unused
8	ALM_WKEN	WR	0	RTC alarm wakeup enable 0: disable 1: enable
7	RTC_WKSLPEN	WR	0	RTC wakeup sleep enable 0: disable 1: enable
6	VUSBRSTEN	WR	0	VUSB insert reset system enable 0: disable 1: enable
5	WKUPRSTEN	WR	0	RTC wake up power down mode reset system enable 0: disable 1: enable
4	ALMIE	WR	0	RTC alarm interrupt enable 0: disable 1: enable
3	RTC1SIE	WR	0	RTC 1S interrupt enable 0: disable 1: enable
2:1	BAUDSEL	WR	0x1	Increase clock selection 00: System Clock divide 4 01: System Clock divide 8 10: System Clock divide 16

Bit	Name	Mode	Default	Description
				11: System Clock divide 32
0	-	-	-	Unused

Register 10-2 RTCCPND: RTC clear pending Register

Bit	Name	Mode	Default	Description
31:19	-	-	-	Unused
18	CWKSLPPND	W	0	Write 1 will clear RTC wakeup sleep pending
17	CALMPND	W	0	Write 1 will clear RTC alarm pending
16:0	-	-	-	Unused

10.3 Independent Power RTC Registers

Register 10-3 RTCCNT: RTC counter Register

Bit	Name	Mode	Default	Description
31:0	RTCCNT	WR	0x0	32bit RTC counter

Register 10-4 RTCALM: RTC alarm Register

Bit	Name	Mode	Default	Description
31:0	RTCALM	WR	0xffffffff	32bit RTC alarm

Register 10-5 RTCCON0: RTC control Register 0

Bit	Name	Mode	Default	Description
31:18	-	-	-	-
17	TKSWRSTN	WR	0	Touch key reset 0: touch key reset 1: release touch key reset
16:15	VRTCSEL	WR	0x0	—
14	RCSEL	WR	0	RC select 0: RCOSC 1:RING RC
13	RCOSCEN	WR	0	—
12	VIOAONS	WR	0	—
11:10	CLK2MBTSSEL	WR	0x0	CLK2M to BT low power clock select bit 00: 0 01: RTC clock 32K 10: RTC 2M 11: XOSC 26M divide 8(3.25M)
9	CLK2MTKSSEL	WR	0	CLK2M in Touch Key power domain source select bit 0: RTC 2M 1: XOSC 26M divide 8(3.25M)

Bit	Name	Mode	Default	Description
8	CLK2MRTCSSEL	WR	0	CLK2M in RTC power domain source select bit 0: RTC 2M 1: XOSC 26M divide 8(3.25M)
7	PWRUP1ST	WR	1	RTC first power up flag 0: not first power up 1: first power up
6	EXT32KS	WR	0	External 32K select 0: use RTC internal 32K osc 1: use external 32K osc
5	-	-	-	-
4	TKITF_EN	WR	1	Touch key between core interface enable bit 0: disable 1: enable
3	SNIFF_EN	WR	0	Sniff mode disable VDDCORE_EN enable 0: disable 1: enable
2	CLK2M_EN	WR	0	CLK2M divide to RTC 32K clock source enable 0: disable 1: enable
1	X32KEN	WR	0	XOSC32K enable bit 0: disable 1: enable
0	RCEN	WR	0	RCOSC enable bit 0: disable 1: enable

Register 10-6 RTCCON1: RTC control Register 1

Bit	Name	Mode	Default	Description
7	VRTCEN	WR	0	VRTC enable bit, VRTC voltage forADC 0: disable 1: enable
6	WKPLVLS	WR	0	WK pin wakeup level select bit 0: low level wakeup 1: high level wakeup
5	WKPAEN	WR	0	WK pin analog enable bit, output WKO voltage forADC 0: disable 1: enable
4	WKPPUEN	WR	1	WK pin pull up enable bit 0: disable 1: enable
3:2	WKPPUS	WR	0x1	WK pin pull up select bit 00: 80K 01: 90K 10: 100K 11: 400K
1	WKPPD	WR	0	WK pin pull down 10K enable bit 0: disable 1: enable
0	WKPIE	WR	1	WK pin input enable bit 0: disable 1: enable

Register 10-7 RTCCON2: RTC control Register 2

Bit	Name	Mode	Default	Description
7	SELVDDPU	WR	1	SEL VDD pullup enable 0: disable 1: enable
6	32KSEL	WR	0	32K osc select bit 0: 32.768K 1: 32K
5:4	RSV	WR	0x0	Reserve,can't be changed default value.
3:2	RSV	WR	0x0	Reserve,can't be changed default value.
1:0	RSV	WR	0x0	Reserve,can't be changed default value.

Register 10-8 RTCCON3: RTC control Register 3

Bit	Name	Mode	Default	Description
15	-	-	-	Unused
14	TK_WKEN	WR	0	Touch key long press wakeup enable bit 0: disable 1: enable
13	-	-	-	Unused
12	INBOX_WKEN	WR	0	INBOX wake up enable bit 0: disable 1: enable
11	VSUB_WKEN	WR	0	VUSB wake up enable bit 0: disable 1: enable
10	WKP_WKEN	WR	0	WK pin wake up enable bit 0: disable 1: enable
9	RTC1S_WKEN	WR	0	RTC one second wakeup enable bit 0: disable 1: enable
8	ALM_WKEN	WR	0	RTC alarm wakeup enable bit 0: disable 1: enable
7	-	-	-	Unused
6	PDCOREEN	WR	1	Core power down enable bit 0: disable 1: enable
5	VCORESHTEN	WR	1	VDDCORE short enable bit 0: disable 1: enable
4	VDDXOEN	WR	1	VDDXO enable bit 0: disable 1: enable
3	VCOREAONEN	WR	1	VDDCORE AON enable bit 0: disable 1: enable

Bit	Name	Mode	Default	Description
2	VCOREEN	WR	1	VDDCORE enable bit 0: disable 1: enable
1	VIOEN	WR	1	VDDIO enable bit 0: disable 1: enable
0	BUCKEN	WR	0	BUCK enable bit 0: disable 1: enable

Register 10-9 RTCCON5: RTC control Register 5

Bit	Name	Mode	Default	Description
7	RSV	WR	0	Reserve,can't be changed default value.
6	RSV	WR	0	Reserve,can't be changed default value.
5:4	RSV	WR	0x0	Reserve,can't be changed default value.
3:2	RSV	WR	0x0	Reserve,can't be changed default value.
1	BUCKLPM	WR	0	BUCK low power mode enable 0: disable 1: enable
0	LDOM	WR	1	BUCK LDO mode select bit 0: buck mode 1: LDO mode

Register 10-10 RTCCON10: RTC control Register 10

Bit	Name	Mode	Default	Description
10	WKP10SC	WR	0	When write WK pin 10s pending clear 0: no affect 1: clear 10s pending When read: WK pin 10s pending 0: no 10s pending 1: 10s pending
9	WK3P	R	0	WK pin3 wake up pending 0: no pending 1: pending
8	WK2P	R	0	WK pin2 wake up pending 0: no pending 1: pending
7	WK1P	R	0	WK pin1 wake up pending 0: no pending 1: pending
6	TKP	R	0	TK wake up pending 0: no pending 1: pending
5	-	-	-	unused

Bit	Name	Mode	Default	Description
4	INBOXP	R	0	INBOX wake up pending 0: no pending 1: pending
3	VUSBP	R	0	VUSB wake up pending 0: no pending 1: pending
2	WKP	R	0	WK pin wake up pending 0: no pending 1: pending
1	RTC1SPC	WR	0	When write: RTC 1 second pending clear 0: no affect 1: clear 1s pending When read: RTC 1 second pending 0: no second pending 1: second pending
0	ALMPC	WR	0	When write: RTC alarm pending clear 0: no affect 1: clear alarm pending When read: Alarm pending 0: no alarm pending 1: alarm pending

Register 10-11 RTCCON11: RTC control Register 11

Bit	Name	Mode	Default	Description
10	RTCWKSLPEN	WR	0	RTC timer wakeup sleep enable 0: disable 1: enable
9:8	RTCWKSLPS	WR	0x0	RTC timer wakeup sleep time select 00: 110ms 01: 220ms 10: 440ms 11: 880ms
7	VIOCHG_SWEN	WR	0	VUSB to VDDIO LDO select control bit 0: VUSB to VDDIO LDO enable by pmu_normal & VIOCHG_EN 1: VUSB to VDDIO LDO enable by VIOCHG_SWEN & VIOCHG_EN
6	VUSBWKSEL	WR	0	VUSB wakeup select 0: VUSB insert filter wakeup 1: VUSB pull out filter wakeup
5	VUSBFILSEL	WR	0	VUSB off filter select 0: 840us 1: 12ms
4	WKOPRT	WR	0	WKO protect bit
3	LVDDETEN	WR	0	LVD detect enable after power up by wake up 0:disable 1:enable
2	WKPFEN	WR	1	WK pin filter enable bit 0:disable 1:enable

1:0	WKPFSEL	WR	0x0	WK pin filter select bit 00:8ms 01:32ms 10:128ms 11:512ms

Register 10-12 RTCCON12: RTC control Register 12

Bit	Name	Mode	Default	Description
7:4	-	-	-	Unused
3:0	WKP10SEN	WR	0xa	WK pin 10s reset enable 0xa: disable Others: enable

11 UART

11.1 Features

(1) UART is a serial port capable of asynchronous transmission.

(2) The UART can function in full duplex mode.

Table 7 I/O ports corresponding to serial channels

UART	Channel G1	Channel G2	Channel G3	Channel G4	Channel G5	Channel G6	Channel G7	Channel G8	Channel G9	Channel G11
TX0	PA7	PB2	PB3	PE7	PE0	PF1	PF5	VUSB	-	-
RX0	PA6	PB1	PB4	PE6	-	-	-	-	-	-
TX1	PA7	PA4	VUSB	-	-	-	-	-	-	-
RX1	PA6	PA3	-	-	-	-	-	-	-	-
TX2	-	PB2	VUSB	-	-	-	-	-	-	-
RX2	-	PB1	-	-	-	-	-	-	-	-
HSTTX	-	-	-	-	-	-	-	-	-	-
HSTRX	PA7	PB2	PB3	PE7	-	PA6	PB1	PB4	PE6	VUSB

11.2 User Guide

(1) Set IO in the correct direction.

(2) Configure UART0BAUD to choose sample rate

(3) Enable UART0 by setting

(4) Set TXIE or RXIE ‘to 1’ if needed

(5) write data to UART0DATA

(6) Wait for PND to change to ‘1’, or wait for interrupt

(7) Read received data from UART0DATA if needed.

11.2.1 UART Special Function Registers

Register 11-1 UARTCON: UART Control Register

Bit	Name	Mode	Default	Description
31:10	-	-	-	Unused
9	RXPND	R	0	RX pending 0: RX one byte not finish 1: RX one byte finish
8	TXPND	R	0	TX pending 0: TX one byte not finish 1: TX one byte finish
7	RXEN	WR	0	RX enable 0: RX disable 1: RX enable
6	ONELINE	WR	0	One-line mode 0: TX/RX separate 1: TX/RX one line

5	CLKSRC	WR	0	Clock source select 0: system clock 1: uart_inc
4	SB2EN	WR	0	Two Stop Bit enable 0: 1-bit Stop Bit 1: 2 bit Stop Bit
3	TXIE	WR	0	Transmit Interrupt Enable 0 = Transmit interrupt disable 1 = Transmit interrupt enable
2	RXIE	WR	0	Receive Interrupt Enable 0: Receiver interrupt disable 1: Receiver interrupt enable
1	BIT9EN	WR	0	BIT9 Enable Bit 0: Eight-bit mode 1: Nine-bit mode
0	UTEN	WR	0	UART Enable Bit 0: Disable UART module 1: Enable UART module

Register 11-2 UARTCPND: UART clear pending Register

Bit	Name	Mode	Default	Description
31:18	-	-	-	Unused
17	CRSTKEYPND	W	0	Reset Key match pending clear 0: N/A 1: Clear Reset key match Pending
16	CKEYPND	W	0	Key match pending clear 0: N/A 1: Clear key match Pending
15:10	-	-	-	Unused
9	CRXPND	W	0	RX pending clear 0: N/A 1: Clear RX Pending
8	CTXPND	W	0	TX pending clear 0: N/A 1: Clear TX Pending. Writing data to UTBUF will clear TXPND
7:0	-	-	-	Unused

Register 11-3 UARTBAUD: UART Baud Rate Register

Bit	Name	Mode	Default	Description
31:16	UARTRXBAUD	W	0	UART RX Baud Rate Baud Rate =Fsys clock / (UART0RXBAUD + 1)
15:0	UARTTXBAUD	W	0	UART TX Baud Rate Baud Rate =Fsys clock / (UART0TXBAUD + 1)

Register 11-4 UARTDATA: UART Data Register

Bit	Name	Mode	Default	Description
31:9	-	-	-	Unused
8	UARTBIT8	WR	x	UART Data bit 8

7:0	UARTDAT	WR	x	UART Data Write this register will load the data to transmitter buffer. Read this register will read the data from the receiver buffer..

12 HSUART

12.1 Features

(1) Support full duplex mode

(2) Support async clock between UART interface part and control part

(3) Support interrupt

(4) Support 8/9 bit data mod，but not data parity checking.

(5) Support 16 bit baud rate setting.

(6) UART TX：
- Support single byte TX
- Support DMA TX，minimum 1 Byte，maximum 1024 Byte
- Support DMA length configure by Byte
- Support DMA address configure by Byte
- Support 1/2 stop bit
- Support TXIE control
- Support TX pending

(7) UART RX：
- Support single byte RX
- Support DMA RX，minimum 1 Byte，maximum 1024 Byte
- Support loop buffer DMA mode
- Support DMA length configure by Byte
- Support DMA address configure by Byte
- Always 1 stop bit，and not check stop bit
- Only 8bit data mode at RX DMA
- Support RXIE control
- Support RX pending

12.2 User Guide

12.2.1 SYNC configure

(1) Configure RX 8/9 bit data mode by RXBITSEL_NS,HSUTTMREN_NS,HSUT0TMR.

(2) Configure TX TXBITSEL_NS,SPBITSEL_NS

(3) Configure HSUT0BAUD

(4) Configure RXEN_NS,TXEN_NS

(5) Setting UPRXCFG (HSUT0CON | =1 <<16;)

(6) Setting UPTXCFG (HSUT0CON | =1 <<17;)

12.2.2 TX 1 byte with buffer

(1) Configure GPIO in the correct direction.

(2) Configure HSUT0CON CLKSRC bit for high speed interface working source clock

(3) Configure HSUT0CON buffer mode ,8/9bit mode, 1/2 stop bit mode

(4) ConfigureHSUT0BAUD baud-rate

(5) Enable UTXEN by setting

(6) set UPTXCFG to sync setting to interface clock domain

(7) Configure TXIEto1’ if needed

(8) write data to HSUT0DATA to kick start TX

(9) Wait for TXPND to change to ‘1’, or wait for interrupt

(10) Clear PNDing

12.2.3 TX n byte with DMA

(1) Configure GPIO in the correct direction.

(2) Configure HSUT0CON CLKSRC bit for high speed interface working source clock

(3) Configure HSUT0CON DMA mode ,8bit mode, 1/2 stop bit mode

(4) ConfigureHSUT0BAUD baud-rate

(5) Configure TX DMA start address HSUT0TXADR

(6) Enable UTXEN by setting

(7) set UPTXCFG to sync setting to interface clock domain

(8) Configure TXIEto1’ if needed

(9) write data count to HSUT0TXCNTto kick start TX

(10) Wait for TXPND to change to ‘1’, or wait for interrupt

(11) Clear PNDing

12.2.4 RX 1 byte with buffer

(1) Configure GPIO in the correct direction.

(2) Configure HSUT0CON CLKSRC bit for high speed interface working source clock

(3) Configure HSUT0CON buffer mode ,8/9bit mode

(4) ConfigureHSUT0BAUD baud-rate

(5) Enable URXEN by setting

(6) Configure RXIEto1’ if needed

(7) set UPRXCFG to sync setting to interface clock domain

(8) Wait for RXPND to change to ‘1’, or wait for interrupt

(9) read data from HSUT0FIFO

(10) Clear PNDing for next RX data

note: if read RX data not in time, data will be covered by next RX done

12.2.5 RX n byte with DMA &loop buffer disable

(1) Configure GPIO in the correct direction.

(2) Configure HSUT0CON CLKSRC bit for high speed interface working source clock

(3) Configure HSUT0CON DMA mode ,8bit mode, not loopbuffer mode

(4) ConfigureHSUT0BAUD baud-rate

(5) Configure RX DMA start address HSUT0RXDADR

(6) Enable URXEN by setting

(7) set UPRXCFG to sync setting to interface clock domain

(8) Configure RXIEto1’ if needed

(9) write data count to HSUT0RXCNTto kick start RX

(10) Wait for RXPND to change to ‘1’, or wait for interrupt

(11) read RX data from SRAM or HSUT0RXFIFO

(12) Clear PNDing, and will clear HSUT0FIFOCNT at the same time

Reading from SRAM:

(1) get start address from HSUT0FIFOADR or program save variable

(2) get data count from HSUT0FIFOCNT or program save variable

(3) read N byte data from SRAM

(4) if not DMA loop buffer mode, clear RXPND will clear HSUT0FIFOCNT

(5) if DMA loop buffer mode, write N to SUBRXCNT（HSUT0CPND[16:0]）to dec HSUT0FIFOCNT

Reading from HSUT0FIFO

(1) get data count from HSUT0FIFOCNT

(2) set RXFIFO[8] to 1, and will auto clear RXFIFO[9]

(3) wait RXFIFO[9] change to 1

(4) get data from RXFIFO[7:0]

(5) loop n times from step 2 to 4.

12.2.6 DMA RX TIMER MODE:

DMA RX HSUT0TMR support for all DMA RX mode.

HSUT0TMR Counter enable：

- set function enable bit
- write data count to HSUT0RXCNT will pre-enable
- wait for HSUT0FIFOCNT ! = 0 and counter will be enable

HSUT0TMR Counter increase:

- TMRCNT increase 1 when RX BUS IDLE for 1 bit baud-rate time

HSUT0TMR clear：

- TMRCNT clear by RX BUS falling edge
- TMRCNT clear by DMA RX kick start
- TMRCNT clear by clear HSTMROV_PND

if HSTMROV_PND setting, HSUT0TMR disable until next DMA RX kick start

12.2.7 Application Note

12.2.7.1 For sync configuration from system domain to uart clock domain

- The follow configure must sync_manual
- SPBITSEL_NS, TXBITSEL_NS sync by setting UPTXCFG
- RXBITSEL_NS sync by setting UPRXCFG
- HSUT0BAUD sync by setting UPTXCFG or UPRXCFG

12.2.7.2 For DMA fail flag

If RX DMA to SRAM is blocked by other DMA channel for a long time, DMA error will happened. Next Rx data will cover the current data.

12.2.7.3 FLAG set and clear

Table 8 TXPND（ TX Hang-up）

Flag Bit	Value Read	Buffer Mode	DMA Mode	All Mode
TXPND（ TX Hang-up）	0	Write HSUT0DATA	Write HSUT0TXCNT	write 1 to HSUT0CPND[13]
	1	TX 1 byte finish	TX n byte finish	-------

Figure 6 DMA High Speed Serial Port Sending Function

```
} else if (rec_tra_sel == HSUT_TRANSMIT) {
    while(hsuart_get_flag(HSUART_FLAG_TX) == RESET);
    hsuart_clear_flag(HSUART_FLAG_TX);
    memcpy(SEND_BUF,(u8*)addr,len);
    HSUART->tx_cnt = 0;
    HSUART->tx_adr = SEND_BUF;
    HSUART->tx_cnt = (uint32_t)len;
```

Table 9 RXPND（ RX Hang-up）

Flag Bit	Read Value	Buffer Mode	DMA Mode	Notes
RXPND（ RX Hang-up）	-------	-------	-------	-------
	1	RX 1 byte done	RX and WRITE n byte to SRAM finish	HSUT0TMRCNT enable and TMROV set

Table 10 RXOV_set

Flag Bit	Write Value	Notes
RXOV_set	-------	-------
	0	write USHT0CPND[12] or write HSUT0RXCNT

Table 11 RXOVPND（RX Overflow Flag Bit）

Flag Bit	Read Value	Notes
RXOVPND （RX Overflow Flag Bit）	1	At DMA loopbuffer mode, RXFIFOCNT reach maximum（MAXFIFOCNT） and try to write data to SRAM
	0	URXEN disable、DMA mode disable、LOOPBUFFER mode disable、 write 1 to HSUT0CPND[11]

Table 12 RXFAIL

Flag Bit	Read Value	Notes
RXFAIL	1	DMA is writing data to SRAM, but next data had input to update DMA data
	0	URXEN disable、DMA mode disable、write 1 to HSUT0CPND[14]

Table 13 TMR_OV

Flag Bit	Read Value	Notes
TMR_OV	1	TMRCNT check UTRX BUS IDLE time larger then setting
	0	write 1 to HSUT0CPND[15] RXDMAEN or write to HSUT0RXCNT

12.2.7.4 FOR HSUT0DADR writing

Writing to HSUT0DASR , will also update these internal address :

FIFO read start address: fiford_adr

RXDMA start address: dmarx_wadr

12.2.7.5 FOR RXOVPND process

It is working at DMA loopbuffer mode.

After RXOVPND was set. It's recommend program process as flow：

1 If data overflow happen，RXPND and RXOVPND setting at the same time

2 Enter RXPND process first，get HSUT0FIFOCNT，and read data from RXFIFO

3 Enter RXOVPND，get HSUT0FIFOCNT，and read data from RXFIFO。

- HSUT0FIFOCNT ！= 0 If RXPND setting by received RXCNT byte data
- HSUT0FIFOCNT = 0 If RXPND setting by RXOVPND

4 Clear RXOVPND

- It will be also clear RXFIFOCNT and initial RXFIFO_ADR to current RXWR_ADR by HSUT0CPND[11].

12.3 HSUART Special Function Registers

Register 12-1 HSUT0CON: High speed UART Control Register

Bit	Name	Mode	Default	Description
31:18	-	-	-	Unused
17	UPTXCFG	W	0	HS TX interface setting update 0: N/A 1: update configure from sysclk to hsuartclk domain
16	UPRXCFG	W	0	HS RX interface setting update 0: N/A 1: update configure from sysclk to hsuartclk domain
15	TMROV	R	0	RX TIMER Overflow flag 0: RX timer not overflow 1:RX timer overflow
14	RXFAIL	R	0	RX Fail flag 0: RX DMA no error 1: RX DMA error
13	TXPND	R	1	TX pending 0: TX one byte/DMA n byte not finish 1: TX one byte/DMA n byte finish
12	RXPND	R	0	RX pending 0: RX one byte/DMA n byte not finish 1: RX one byte/DMA n byte finish
11	RXOVPND	R	0	RX overflow pending 0: RX DMA buffer not overflow 1: RX DMA buffer overflow
10	HSUTTMREN_NS	WR	0	HSUART DMA RX TIMER CNT enable 0: disable 1: enable
9	SPBITSEL_NS*	WR	0	TX Stop Bit select 0: 1 bit Stop Bit 1:2 bit Stop Bit
8	TXBITSEL_NS*	WR	0	TX Data bitselect 0:8-bit mode 1:9-bit mode
7	TXTRSMODE	WR	0	TX Transmit mode select 0: buffer mode 1:DMAmode
6	RXLPBUFEN	WR	0	RX DMA loop buffer mode enable bit 0: disable 1:enable
5	RXBITSEL_NS*	WR	0	RX Data bitselect 0:8-bit mode 1:9-bit mode
4	RXTRSMODE	WR	0	RX Transmit mode select 0: buffer mode 1:DMA mode
3	TXIE	WR	0	Transmit Interrupt Enable 0 = Transmit interrupt disable 1 = Transmit interrupt enable
2	RXIE	WR	0	Receive Interrupt Enable 0: Receiver interrupt disable 1:Receiver interrupt enable
1	UTXEN_NS	WR	0	UART TX Enable Bit 0: Disable TX UART module 1:Enable TX UART module
0	URXEN_NS	WR	0	UART RX Enable Bit 0: Disable UART RX module 1:Enable UART RX module

*_NS need sync to uart interface clock domain by UPTXCFG or UPRXCFG

Register 12-2 HSUT0CPND: HSUART clear pending Register

Bit	Name	Mode	Default	Description
31:16	SUBRXCNT*	W	0	Decrease RXFIFOCNT at RX DMA mode
15	CTMROV	W	0	RX timer overflow flagl clear 0: N/A 1:Clear flag
14	CRXFAIL	W	0	RX Fail clear 0: N/A 1: Clear RX fail flag
13	CTXPND	W	0	TX pending clear 0: N/A 1: Clear TX Pending
0	URXEN_NS	WR	0	UART RX Enable Bit 0: Disable UART RX module 1:Enable UART RX module
14	CRXFAIL	W	0	RX Fail clear 0: N/A 1: Clear RX fail flag
13	CTXPND	W	0	TX pending clear 0: N/A 1: Clear TX Pending
12	CRXPND	W	0	RX pending clear 0: N/A 1: Clear RX Pending. Writing data to UTBUF will clear TXPND
11	CRXOVPND	W	0	RX overflow pending clear 0: N/A 1:Clear RX overflowPending.
10:2	-	-	-	Unused
1	CUTTX	W	0	HS TX interface clear 0: N/A 1: clear to idle status at hsuartclk domain
0	CUTRX	W	0	HS RX interface clear 0: N/A 1: clear to idle status at hsuartclk domain

*SUBRXCNT：sub counter after 1 clock write to this register.

Register 12-3 HSUT0BAUD: HSUART Baud Rate Register

Bit	Name	Mode	Default	Description
31:16	HSUTRXBAUD	W	0	HSUART RX Baud Rate Baud Rate =source clock / (HSUTRXBAUD + 1)
15:0	HSUTTXBAUD	W	0	HSUART TX Baud Rate Baud Rate =source clock / (HSUTTXBAUD + 1)

Register 12-4 HSUT0DATA: HSUART Data Register

Bit	Name	Mode	Default	Description
31:9	-	-	-	Unused
8	HSUTBIT8	W	x	UART Data bit 8
7:0	HSUTDAT	W	x	UART Data Write this register will load the data to transmitter buffer.

Register 12-5 HSUT0TXCNT: HSUART TX counter Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	TXCNT	WR	0	HSUART TX DMA byte counter

Register 12-6 HSUT0TXADR: HSUART TX DMA start address Register

Bit	Name	Mode	Default	Description
31:0	HSUTTXADR	WR	0	HSUART TX DMA start address

Register 12-7 HSUT0RXCNT: HSUART RX counter Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	RXCNT	WR	0	HSUART RX DMA byte counter

Register 12-8 HSUT0FIFOCNT: HSUART RX FIFO counter Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	RXCNT	WR	0	HSUART RX DMA byte counter

Register 12-9 HSUT0RXUADR: HSUART RX up-Bound address Register

Bit	Name	Mode	Default	Description
31:0	RXUADR	WR	0	HSUART RX DMA buffer Up-limit address

Register 12-10 HSUT0RXDADR: HSUART RX Down-Bound address Register

Bit	Name	Mode	Default	Description
31:0	RXDADR	WR	0	HSUART RX DMA buffer Down-limit address

Register 12-11 HSUT0RXMCNT: HSUART RX Maximum counter Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	RXMAXCNT	WR	0	HSUART RX DMA byte maximum counter

Register 12-12 HSUT0FIFOADR: HSUART RX FIFO point address Register

Bit	Name	Mode	Default	Description
31:0	RXFIFOADR	R	0	HSUART RX DMA buffer Down-limit address

Register 12-13 HSUT0RXFIFO: HSUART RX Data FIFO Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
9	RXFIFO_DONE	R	0	RXFIFO read from SRAM done 0: read not finish 1; read done Auto clear by kick RXFIFO_RD bit
8	RXFIFO_RD	W	0	RX read pluse 0: N/A 1: read fifo kick start
7:0	RXFIFO	R	x	UART RX Data Read this register will read the data from the receiver SRAM buffer

Register 12-14: HSUT0FIFOADR HSUART RX Data FIFO ADDRESS Register

Bit	Name	Mode	Default	Description
31:0	HSUTFIFOADR	R	-	FIFO read address

Register 12-15: HSUT0TMR HSUART RX TIMER Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	HSUTTMR_NS	WR	0	HSUART DMA RX TIMER CNT

13 SPI

13.1 Features

(1) The chip has one SPI. The SPI is connected to the external I/O port by tying a cable and is available to users.

(2) Flash SPI is a 2-frequency division of the system clock. The SPI flash start address is 0x1000 0000.

(3) SPI can support different mode.

- general 3 wire mode, 1-bit clock in/out, 1-bit data output, 1-bit data input
- 2 wire mode, 1-bit clock in/out, 1-bit data output or input;
- 2 data bus mode, 1-bit clock in/out, 2-bit data output or input;
- 4 data bus mode, 1-bit clock in/out, 4-bit data output or input;

13.2 User Guide

SPI Normal 1bit-Mode Operation Flow:

(1) Set 3-wire mode or 2-wire mode and select the pin map

(2) Select RXSEL for Transmit or receive

(3) Configure clock frequency

(4) Select one of the four timing mode

(5) Enable SPI module by setting SPIEN ‘1’

(6) Set SPIIE ‘1’ if needed

(7) Write data to SPIBUF to kick-start the process

(8) Wait for SPIPND to change to ‘1’, or wait for interrupt

(9) Read received data from SPIBUF if needed

(10) Go to Step 8 to start another process if needed or turn off SPI0by clearing SPIIE and SPIEN

SPI Normal Multi-Bit-Mode Operation Flow:

(1) Set data bus width (bus4 or bus 2) and select the pin map

(2) Select RXSEL for Transmit or receive

(3) Configure clock frequency

(4) Select one of the four timing mode

(5) Enable SPI module by setting SPIEN ‘1’

(6) Set SPIIE ‘1’ if needed

(7) Write data to SPIBUF to kick-start the process

(8) If data bus width is 2 bit, write SPIBUF twice kick-start the transmission

(9) If data bus widths are 4 bit, write SPIBUF four times kick-start the transmission

(10) However, when receive data, only need write once to kick-start receive process

(11) Wait for SPIPND to change to ‘1’, or wait for interrupt

(12) Read received data from SPIBUF if needed

(13) Go to Step 8 to start another process if needed or turn off SPI by clearing SPIIE and SPIEN

SPI DMA Mode Operation Flow:

(1) Set IO in the correct direction and data width mode.

(2) Select RXSEL for DMA direction

(3) Configure clock frequency

(4) Select one of the four timing modes

(5) Enable SPI module by setting SPIEN to ‘1’

(6) Set SPIIE ‘1’ if needed

(7) configure SPI0DMAADR；

(8) Write data to SPI0_DMACNT to kick-start a DMA process

(9) Wait for SPIPND to change to ‘1’, or wait for interrupt

(10) Go to Step 8 to start another DMA process if needed or turn off SPI0 by clearing SPI0EN.

13.3 SPI Special Function Registers

Register 13-1 SPIxCON: SPI Control Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused
16	SPIPND	R	0	SPI pending 0: not finish SPI rx/tx 1: finish SPI rx/tx
15:14	-	-	-	Unused
13	HOLDENSW	WR	0	SPI software hold enable 0: disable 1: enable
12	HOLDENTX	WR	0	SPI hold enable when bt tx 0: disable 1: enable
11	HOLDENRX	WR	0	SPI hold enable when bt rx 0: disable 1: enable
10	SPIOSS	WR	0	SPI sample data is at the same clock edge with output data 0: SPI sample data is at the difference clock edge with output data 1: SPI sample data is at the same clock edge with output data
9	SPIMBEN	WR	0	SPI multiple bit bus enable bit 0: disable 1: enable
8	SPILF_EN	WR	0	SPI LFSR enable bit 0: disable 1: enable
7	SPIIE	WR	0	SPI interrupt enable 0: disable 1: enable
6	SMPS	WR	0	SPI output edge select bit, when SPIOSS = 0, sample data and output data is at different clock edge; when SPIOSS = 1, sample data and output data is at the same clock edge 0: output data at the falling edge; 1: output data at the rising edge;
5	CLKIDS	WR	0	SPI clock state when idle 0: clock stay at 0 1: clock stay at 1
4	RXSEL	WR	0	When in DMA mode or 2-wire mode, configure SPI Receive or Transmit select bit 0: transmit 1: receive
3:2	BUSMODE	WR	0x0	Data bus width select bit 00:3-wire mode; 1bit data in, 1bit data out 01:2-wire mode; 1bit data in/out 10: 2bit bidirectional data bus 11: 4bit bidirectional data bus
1	SPISM	WR	0	Slave mode select bit 0: master mode 1:slave mode
0	SPIEN	WR	0	SPI Enable bit 0: Disable 1: Enable

Register 13-2 SPI0BAUD: SPI Baud Rate Register

Bit	Name	Mode	Default	Description
31:16	-	-	-	Unused
15:0	SPI0BAUD	W	0	SPI Baud Rate Baud Rate =Fsys clock / (SPI_BAUD+1)

Register 13-3 SPI0CPND: SPI clear pending Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused
16	SPICPND	W	0	Write 1 will clear SPI pending
15:0	-	-	-	Unused

Register 13-4 SPI0BUF: SPI0 receive/send Data Register

Bit	Name	Mode	Default	Description
31:8	-	-	-	Unused
7:0	SPI0BUF	WR	x	SPI Data Write this register will load the data to transmitter buffer. Read this register will read the data from the receiver buffer..

Register 13-5 SPI0DMACNT: SPI0 DMA counter Register

Bit	Name	Mode	Default	Description
31:11	-	-	-	Unused
10:0	SPI0DMACNT	W	x	SPIDMA byte counter Write this register will kick start spi send/receive data Total number of bytes received / send is SPI0DMACNT

Register 13-6 SPI0DMAADR: SPI0 DMA address Register

Bit	Name	Mode	Default	Description
31:21	-	-	-	Unused
20:0	SPI0DMAADR	W	x	SPI DMA byte address

14 IIC

14.1 Features

(1) Support IIC one master

(2) Support asynchronous clock source from RC2M or XOSC26M

(3) Support out data maximum 4 Byte

(4) Support in data maximum 4 Byte

14.2 User Guide

IIS master mode Operation Flow:

- Configure IO mapping, SDA set pullup enable
- Configure IIC clock select from RC 2M or XOSC, set pre_div clock
- Configure IICCON0
- Configure IICCMDA for control byte and address byte
- Configure IICDATA for WRITE data
- Configure IICCON1
- Kick start
- Wait done flag or interrupt if need
- Clear DMA DONE flag and update IICCMDA or IICDATA
- Loop step 7

14.3 IIC Special Function Registers

Register 14-1 IICON0: IIC Control Register

Bit	Name	Mode	Default	Description
31	DONE	R	0	IIC DONE flag
30	ACKSTATUS	R	0	RX IIC slave ACK status 0: RX ACK 1: RX NAK
29	CLR_DONE	W	0	DONE flag clear 0: 1: clear
28	KS	W	0	Kick start 0: 1: Kick start
27	CLR_ALL	W	0	Clear All status 0: 1: clear
26:10	Rev.	WR	-	Unused
9:4	POSDIV	WR	0	IIC SCL pose div counter 0: div 1 1: div2 … N: div N+1
3:2	HOLDCNT	WR	0	SDA hold cnt when SCL failing 0: 1 cycle 1: 2 cycle …
1	INTEN	WR	0	IIC interrupt 0: disable 1: enable
0	IIC_EN	WR	0	IIC Enable Bit 0: Disable 1: Enable

Register 14-2 IICON1: IIC Control Register

Bit	Name	Mode	Default	Description
31:13	-	-	-	Unused
12	TXNAK_EN	WR	0	IIC TX NAK when read last data enable
11	STOP_EN	WR	0	IIC TX STOP enable
10	WDAT_EN	WR	0	IIC TX DATA enable
9	RDAT_EN	WR	0	IIC RX data enable
8	CTL1_EN	WR	0	IIC TX ctl 1 enable
7	START1_EN	WR	0	IIC TX start 1 enable
6	ADR1_EN	WR	0	IIC TX adr 1 enable

Bit	Name	Mode	Default	Description
5	ADR0_EN	WR	0	IIC TX adr 0 enable
4	CTL0_EN	WR	0	IIC TX ctl 0 enable
3	START0_EN	WR	0	IIC TX start 0 enable
2:0	DATA_CNT	WR	0	RX/TX data counter 0: 0 byte 1: 1 byte … N: N byte

IIC clock configuration :

IICCLK = source clk / (preclkdiv+1)

SCL = IICCLK / (posdiv+1)

Register 14-3 IICCMDA: IIC CMD/ADR Register

Bit	Name	Mode	Default	Description
31:24	CTL1	WR	0	Control 1 data
23:16	ADR1	WR	0	Address 1 data
15:8	ADR0	WR	0	Address 0 data
7:0	CTL0	WR	0	Control 0 data

Register 14-4 IICDATA: IIC DATA Register

Bit	Name	Mode	Default	Description
31:24	DATA3	WR	0	Data 3
23:16	DATA2	WR	0	Data 2
15:8	DATA1	WR	0	Data 1
7:0	DATA0	WR	0	Data 0

15 ADC

15.1 Features

This ADC has the following characteristics:

(1) Support 16 channel,10 bit sampling frequency

(2) The maximum sample rate is 78k/s; SARADC bit clock maximum is 1MHz

(3) ADC has internal 100K pull up resister

15.2 Channel select

Code for reference:

```
/*****************************************************************************
 * Module    : ADC Path Selection List
 *****************************************************************************/
#define ADCCH_PA5       0          //SARADC channel 0
#define ADCCH_PA6       1          //SARADC channel 1
#define ADCCH_PA7       2          //SARADC channel 2
#define ADCCH_PB1       3          //SARADC channel 3              WK2
#define ADCCH_PB2       4          //SARADC channel 4              WK3
#define ADCCH_PB3       5          //SARADC channel 5
#define ADCCH_PB4       6          //SARADC channel 6
#define ADCCH_PE5       7          //SARADC channel 7
#define ADCCH_PE6       8          //SARADC channel 8
#define ADCCH_PE7       9          //SARADC channel 9          ADS7844 ADC0
#define ADCCH_PF5       10          //SARADC channel 10
#define ADCCH_PB0       11          //SARADC channel 11            WK1
#define ADCCH_WKO       12          //SARADC channel 12            WKO/PB5
#define ADCCH_BGOP      13          //SARADC channel 13, Internal calibration voltage bandgap voltage
is 0.75V
#define ADCCH_VBAT      14          //SARADC channel 14
#define ADCCH_VUSB      15          //SARADC channel 15
```

15.3 User Guide

(1) Configure SADCBAUD

(2) Configure SADCST if need

(3) Enable SARADC

(4) Write SADCCH to enable channel to convert. Can enable more than one channel. Write SADCCH will kick start ADC convert.

(5) Wait for ADC_PND

15.4 ADC_CTL Special Function Registers

Register 15-1 SADCCON: SARADC Control Register

Bit	Name	Mode	Default	Description
31:20	-	-	-	Unused
19	ADCAEN	WR	0	Saradc auto enable analog enable bit 0: disable 1: enable
18	ADCANGIO	WR	0	Saradc auto enable analog IO enable bit 0: Disable 1: Enable
17	ADCIE	WR	0	Saradc interrupt enable bit 0: Disable 1: Enable
16	ADCEN	WR	0	Saradc enable bit 0: Disable 1: Enable
15	CH15PUEN	WR	0	Channel 15 internal pullup enable bit 0: Disable 1: Enable
14	CH14PUEN	WR	0	Channel 14 internal pullup enable bit 0: Disable 1: Enable
13	CH13PUEN	WR	0	Channel 13 internal pullup enable bit 0: Disable 1: Enable
12	CH12PUEN	WR	0	Channel 12 internal pullup enable bit 0: Disable 1: Enable
11	CH11PUEN	WR	0	Channel 11 internal pullup enable bit 0: Disable 1: Enable
10	CH10PUEN	WR	0	Channel 10 internal pullup enable bit 0: Disable 1: Enable
9	CH9PUEN	WR	0	Channel 9 internal pullup enable bit 0: Disable 1: Enable
8	CH8PUEN	WR	0	Channel 8 internal pullup enable bit 0: Disable 1: Enable
7	CH7PUEN	WR	0	Channel 7 internal pullup enable bit 0: Disable 1: Enable
6	CH6PUEN	WR	0	Channel 6 internal pullup enable bit 0: Disable 1: Enable
5	CH5PUEN	WR	0	Channel 5 internal pullup enable bit 0: Disable 1: Enable
4	CH4PUEN	WR	0	Channel 4 internal pullup enable bit 0: Disable 1: Enable

Bit	Name	Mode	Default	Description
3	CH3PUEN	WR	0	Channel 3 internal pullup enable bit 0: Disable 1: Enable
2	CH2PUEN	WR	0	Channel 2 internal pullup enable bit 0: Disable 1: Enable
1	CH1PUEN	WR	0	Channel 1 internal pullup enable bit 0: Disable 1: Enable
0	CH0PUEN	WR	0	Channel 0 internal pullup enable bit 0: Disable 1: Enable

Register 15-2 SADCCH: SARADC channel enable Register

Bit	Name	Mode	Default	Description
31:17	-	-	-	Unused
16	ADCPND	WR	0	Saradc finish pending 0:Not finish 1:Finish Write SARADCH register will clear this bit
15	CH15EN	WR	0	Channel 15 enable bit 0: Disable 1: Enable
14	CH14EN	WR	0	Channel 14 enable bit 0: Disable 1: Enable
13	CH13EN	WR	0	Channel 13 enable bit 0: Disable 1: Enable
12	CH12EN	WR	0	Channel 12 enable bit 0: Disable 1: Enable
11	CH11EN	WR	0	Channel 11 enable bit 0: Disable 1: Enable
10	CH10EN	WR	0	Channel 10 enable bit 0: Disable 1: Enable
9	CH9EN	WR	0	Channel 9 enable bit 0: Disable 1: Enable
8	CH8EN	WR	0	Channel 8 enable bit 0: Disable 1: Enable
7	CH7EN	WR	0	Channel 7 enable bit 0: Disable 1: Enable
6	CH6EN	WR	0	Channel 6 enable bit 0: Disable 1: Enable

Bit	Name	Mode	Default	Description
5	CH5EN	WR	0	Channel 5 enable bit 0: Disable 1: Enable
4	CH4EN	WR	0	Channel 4 enable bit 0: Disable 1: Enable
3	CH3EN	WR	0	Channel 3 enable bit 0: Disable 1: Enable
2	CH2EN	WR	0	Channel 2 enable bit 0: Disable 1: Enable
1	CH1EN	WR	0	Channel 1 enable bit 0: Disable 1: Enable
0	CH0EN	WR	0	Channel 0 enable bit 0: Disable 1: Enable

Register 15-3 SADCST: SAR ADC setup timing Register

Bit	Name	Mode	Default	Description
31:30	CH15ST	WO	0x0	Channel 15 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
29:28	CH14ST	WO	0x0	Channel 14 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
27:26	CH13ST	WO	0x0	Channel 13 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
25:24	CH12ST	WO	0x0	Channel 12 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
23:22	CH11ST	WO	0x0	Channel 11 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
21:20	CH10ST	WO	0x0	Channel 10 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
19:18	CH9ST	WO	0x0	Channel 9 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK

Bit	Name	Mode	Default	Description
				11: 8 SARADC_CLK
17:16	CH8ST	WO	0x0	Channel 8 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
15:14	CH7ST	WO	0x0	Channel 7 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
13:12	CH6ST	WO	0x0	Channel 6 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
11:10	CH5ST	WO	0x0	Channel 5 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
9:8	CH4ST	WO	0x0	Channel 4 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
7:6	CH3ST	WO	0x0	Channel 3 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
5:4	CH2ST	WO	0x0	Channel 2 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
3:2	CH1ST	WO	0x0	Channel 1 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK
1:0	CH0ST	WO	0x0	Channel 0 setup time 00:0 SARADC_CLK 01:2 SARADC_CLK 10: 4 SARADC_CLK 11: 8 SARADC_CLK

Register 15-4 SADCBAUD: SARADC Baud Rate Register

Bit	Name	Mode	Default	Description
31:10	-	-	-	Unused
9:0	SADCBAUD	WO	0x0	SARADC Baud Rate Baud Rate =Fadc clock / [2(SADCBAUD+1)].

Register 15-5 SADCDAT0~15: SARADC channel 0~15 Data Register

Bit	Name	Mode	Default	Description
31:10	-	-	-	Unused
9:0	SADCDAT	R	0x0	SARADC data, channel 0 to channel 15 register

16 DAC

16.1 Feature

Output the offset value of DC at up to 780KHz (1.3us) to change the output DAC level;

16.2 Control use

(1) my_dac_init() sets the output DAC level for starting DAC_R or DAC_L pin

(2) set_bia_voltage_reg_R() sets DAC_R level; set_bia_voltage_reg_L() sets DAC_L level

17 USB

17.1 Feature

The full speed of USB Devise is 12 Mbits/s, D + internal pull-up resistor of USB.

17.2 Control use

(1) usb_detect();//this function will detect whether the USV is connected or disconnected

(2) usb_bulk_send();//batch transmission of USB data, waiting to be read by the host

(3) usb_insert_callback();//callback function for engineering call after usb is inserted

(4) usb_ep_config();//functions for configuring input and output ports

18 Power Management

18.1 Charging process

After the vusb pin is connected to 5V voltage, the chip first judges whether the voltage connected to the vusb is high (the vusb voltage should be 0.2v higher than the battery voltage and be greater than 4.5v) (this condition can be configured whether to take effect). If the vusb voltage meets the charging conditions, enter the charging mode.

- Charging state when the battery voltage is below 2.2V: 0V charging function (the battery has protection board), and hardware solidification cannot be turned off, and the chip judges whether the lithium battery needs to be activated (the activation condition is the chip is powered on for the first time, and the battery voltage is below 2.2V). If the activation condition is met, first activate the battery with 120mA current, and when the battery voltage is above 2.2V, charge with the charging current configured by the software. The activation current cannot be configured only when the chip is powered on for the first time, and can be configured after power on. If the chip is not completely powered down again, the battery will be activated at the configured activation current in future charging.
- Charging state when the battery voltage is higher than 2.2V: After power on for the first time, when the battery voltage is above 2.2v, the chip will be charged in trickle mode (the default setting for the first power-on is 30mA, and it can be modified after startup). If it is not powered on for the first time and the chip has not been completely powered down (vbat is not below 1.5v), the chip will start charging in trickle mode. When the battery voltage is above 3V, the chip will be charged in constant-current mode, and the trickle charging current and constant charging current can be configured separately. Usually, the trickle charging current is smaller than the charging current in constant-current mode. When the battery node is close to 4.2v, enter the constant-voltage mode for charging, the maximum output voltage of the charging circuit is equal to the output voltage after the battery is unplugged, this voltage is calibrated to 4.2v±42mV, and no matter how long it is charged, the final charging voltage of the battery will not exceed this voltage.

In constant-voltage mode, the charging current will gradually decrease. When the charging current is less than the set charging cut-off current, the system judges that the charging ends. In addition, the system can configure a timer, and when the battery voltage reaches 4.17V, the timer can be started to stop charging by force when the counting time is reached.

18.2 Charging settings

(1) Constant charging current: 16 gears, 10~200mA

(2) Trickle charging current: 3 gears, 10~30mA

(3) Turn-off charging current: 8 gears, 2.5~35mA

(4) Turn-off charging voltage: 2 gears, 4.2V or 4.3V

(5) Trickle charging voltage: 2 gears, 2.9V or 3.0V

18.3 Charging control function

Bt_charge_off();//stop charging

Charge_start(u8 mode);//start charging, 1 represents constant current, while 0 represents trickle current

sadc_get_data(ADCCH_VBAT);//obtain sampling value of battery voltage

19 Bluetooth

19.1 Feature

(1) Core specifications: √ BR, √ VEDR, √ LE

(2) Power class: Class1(11dbm >= Pmax >=4dbm)

(3) BREDR modulation modes: √ GFSK, √ π/4-DQPSK, √ 8DPSK

(4) BREDR parameters: Image frequency (+2Mhz), Value n(3)

(5) LE modulation modes: √ IM PHY, √ 2M PHY, ×coded PHY, ×Stable Modulation Index(TX & RX)

(6) LE characteristics: ×AOA, ×AOD, ×CTE

(7) LE parameters: Imae frequency(+2Mhz), Valuen(3)

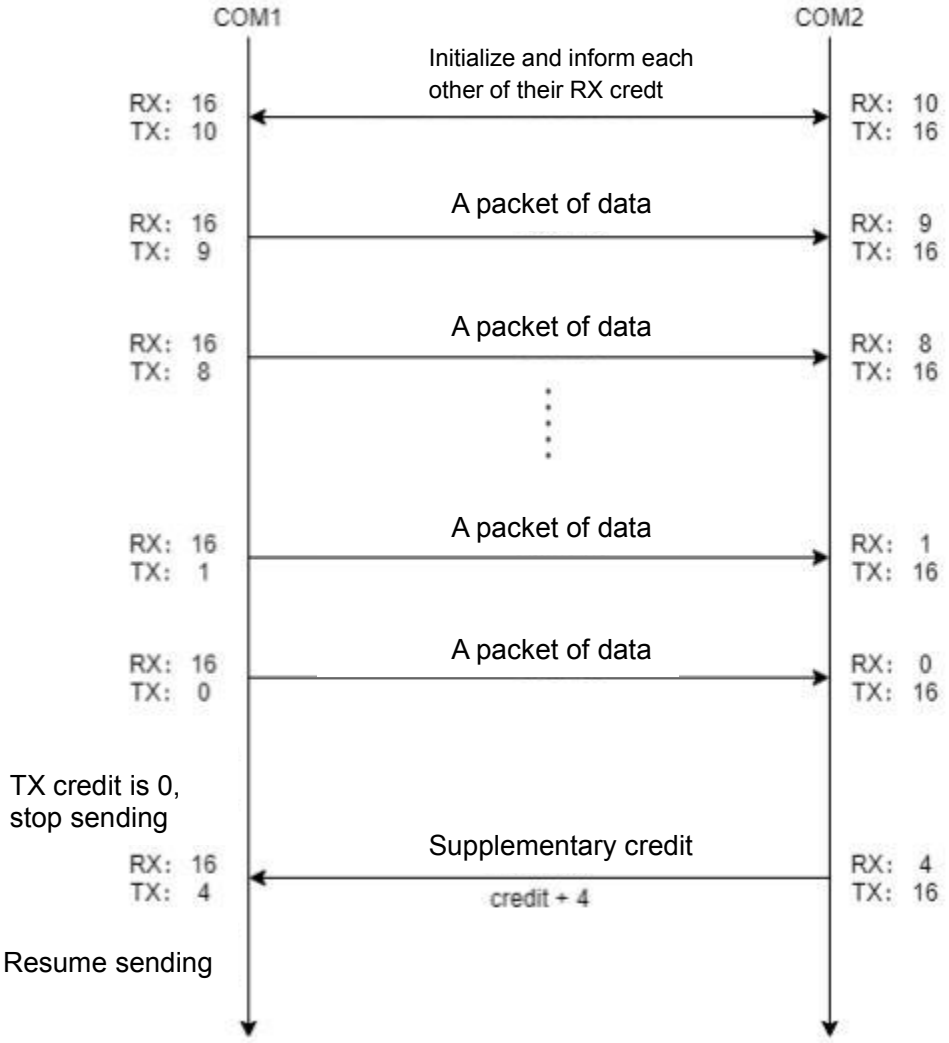
Note: Bluetooth data communication is reliable, including verification and encryption. Encryption is optional.

19.2 SPP protocol

19.2.1 SPP protocol is based on credit flow control mechanism

SPP protocol is an application layer protocol based on Bluetooth RFCOMM protocol, and can use the credit-based flow control mechanism. The general process of credit-based flow control mechanism is as follows.

Figure 7 Flow Control Mechanism of Credit



Both sides of the serial port will have the count of two credits, i.e. the credit of TX and RX respectively. During initialization, both sides of the serial port will inform the other side of the initial RX credit, and after obtaining the RX credit of the other side, record the value in their own TX credit.

After that, every time a data packet is transmitted, the TX credit will decrease by 1; every time a data packet is received (i.e. the spp_rx_callback function is set once), the RX credit will decrease by 1. When the TX credit decreases to 0, data cannot be transmitted and needs to wait for the other side to supplement the credit or disconnect; similarly, when the RX credit decreases to 0, the other side will stop transmitting data and needs to wait for itself to supplement the credit or disconnect.

In the communication process, in order to ensure data can be continuously received and transmitted, the credit needs to be continuously supplemented to the other side, and the amount of credits to be supplemented can be calculated and supplemented according to the space of the RX buffer. Then a control mechanism can be implemented to manage the flow based on the space of the RX buffer.

19.2.2 SPP use and development in SDK

19.2.2.1 Function enable

Open the macro of BT_SPP_RX_FLOW_CTRL_EN in the file of config.h.

19.2.2.2 Instructions for API

Two API are opened to inform the other side of SPP to supplement and update credit, i.e. void spp_set_rx_new_credit(uint8_t credit) and int spp_notify_rx_buffer_len(u16 len).

The former void spp_set_rx_new_credit(uint8_t credit) is to directly notify the Bluetooth underlying layer to inform the other side to update the credit.

The latter int spp_notify_rx_buffer_len(u16 len) is to divide the remaining available length of the incoming buffer by the MTU (maximum transmission unit, namely the maximum length of a packet of data) of the SPP to obtain the minimum number of data packets that can be received in the remaining buffer space, and then notify the Bluetooth underlying layer to inform the other side to update the credit.

If it is unclear how to better figure out the relationship between the remaining space of buff er and the credit, it is recommended to directly call the int spp_notify_rx_buffer_len(u16 len) interface, and use MTU to calculate, which can ensure that the buffer space will not overflow due to receiving of different data packet lengths each time. The MTU size of the SPP protocol is determined by the macro of SPP_MTU_SIZE.

The underlying layer will obtain the MTU size through the function of u16 get_spp_mtu_size(void).

spp_tx_pool is used by underlying layer, and the user cannot operate this data.

bt_updata_local_name(); update the SPP Bluetooth name

bt_setup();//will automatically write xcfg_cb.le_name into the Bluetooth chip, not needing to additionally call the bt_updata_local_name.

19.2.2.3 Update of RX buff er and credit

The user shall customize the RX buffer space of SPP and manage the buffer, so as to calculate the remaining available space of the buffer and the credit.

In the function u16 spp_rxpkt_init(void), some related content about initialization of RX buffer can be added; at the same time, the total length of the buffer needs to be returned, so that the initial credit can be set when the SPP connection is established.

The function of void bsp_spp_process(void) is continuously called in the main loop, data processing of rx can be placed in it, and every time the data is processed, spp_notify_rx_buffer_len will be called to update the remaining available length of the buffer. It is not recommended to directly perform data processing in the callback function void spp_rx_callback. The heavy burden of callback function will affect the operation of Bluetooth. spp_rx_callback () can receive a maximum of 511 bytes at a time.

When SPP flow control is enabled, the packet needs to be provided to the cell phone in spp_rx_callback (), i.e. calling spp_set_rx_new_credit(1). To prevent the phone from continuing to transmit data, do not provide the packet, record the number of owed packets, and when the phone is needed to continue to transmit data, make up for the owed packets.

If the return value of spp_notify_rx_buffer_len () is 0x75, it indicates that "flow control configuration is not enabled" and BT_SPP_RX_FLOW_CTRL_EN needs to be enabled; if the return value is 0x76, it indicates "RX credit error". RX credit is automatically subtracted by the Bluetooth underlying layer, and returns 0x76, which is the result obtained by dividing the incoming len parameter by MTU. If it is less than the current credit, the credit cannot increase

19.3 BLE protocol

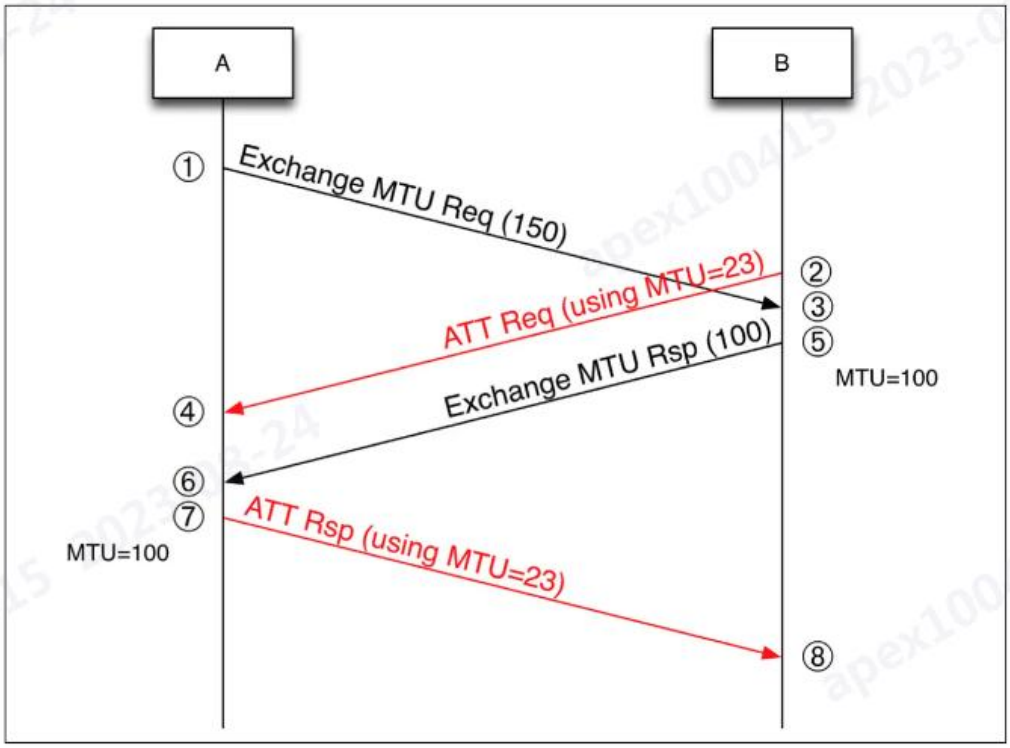
Table 15 BLE Service Characteristics

Description	UUID	Properties
Service	0xFF12	
WriteWithNoRsp Characteristic	0xFF15	Read/Write Without Response

Description	UUID	Properties
Notify Characteristic	0xFF14	Notify

19.3.1 Flow control mechanism of BLE protocol

MTU exchange is to set the maximum amount of data that can be exchanged in a PDU between the master and slave. Through MTU exchange and confirmation by both parties (Note that this MTU is not negotiable, but only informs the other party. Both parties will choose a smaller value as the future MTU after knowing the other party's limit. For example, if the master device sends an MTU request of 150 bytes, but the slave device responds with an MTU of 23 bytes, then both parties will use 23 bytes as the future MTU), the master and slave parties agree not to exceed this maximum data unit each time they perform data transmission, such as the exchange below:



gatt specifies that the default support of ATT_MTU cannot be less than 23. The specification has limited the attribute value to be only 512 bytes.

19.3.2 Use and Development of BLE in SDK

19.3.2.1 ble shortens the interval time between packets

ble_update_conn_param(12, 0, 400);//12*1.25ms, the interval for the mobile phone to transmit packets; the minimum value for the first parameter is 6, which should be a multiple of 6.

19.3.2.2 Ble sets the value of read value

void ble_init_att_for_handle(u16 handle, u8* buf, u16 len);

void ble_read_value(u8 *buf, u8 len)//ble reads the manufacturer name, battery level, and other Bluetooth services

{

ble_init_att_for_handle(gatts_Datas_Characteristic_base.handle, buf, len);

}

19.3.2.3 Bluetooth transmits the function return value

If ble_tx_notify returns 0, it indicates successful transmission; if the transmission is unsuccessful, return the error code, and actually not a single byte is transmitted. So the transmission fails, and the entire data needs to be transmitted again. Decimal 86 means "packet is empty, or the transmission length is greater than the buf size".

19.3.2.4 Clear the Bluetooth cache on mobile app

ble_service_change_indicate () i.e.: transmit ble_tx_indication (0x0100ffff) to "0x2a05" of "0x1801" service

19.3.2.5 Set the Bluetooth address

ble_get_local_bd_addr（）

19.3.2.6 Change the ble Bluetooth name

bt_setup();////will automatically write xcfg_cb.le_name into the Bluetooth chip.

19.4 FOTA upgrade

19.4.1 Use and development of FOTA in feature SDK

(1) The chip supports using BLE or SPP protocol to upgrade FOTA.

(2) FOTA adopts dual-backup upgrade, so it needs to ensure that FLASH has enough space to back up the program.

(3) The size of the program must be less than (FLASH_SIZE/2 -12) KB. For FLASH_ SIZE, please see the macro FLASH_SIZE in the file of config.h. For example, if FLASH is 1M, the maximum size of the program is (1024/2-12)=500KB. The customer needs to set the FLASH size according to the actual situation.

(4) The implementation codes related to FOTA are mainly in files of bsp_fot.c, bsp_fot.h, spp.c and app.c. Note that there are FOTA related codes in the callback functions of SPP and BLE, and the registration service of BLE. If FOTA function is needed, the content here shall not be modified during the development. Please refer to F*OT Protocol Document*

20 Revision history

Table 14 Document Revision History

Date	Revision	Change History
2023.5.10	0.1	New
2023.8.24	0.2	Add charging module, USB, BLE, clock register and other instructions; Change the description of other modules.

Statement

This document is formulated and published by Geehy Semiconductor Co., Ltd. (hereinafter referred to as “Geehy”). The contents in this document are protected by laws and regulations of trademark, copyright and software copyright. Geehy reserves the right to make corrections and modifications to this document at any time. Please read this document carefully before using Geehy products. Once you use the Geehy product, it means that you (hereinafter referred to as the “users”) have known and accepted all the contents of this document. Users shall use the Geehy product in accordance with relevant laws and regulations and the requirements of this document.

1. Ownership

This document can only be used in connection with the corresponding chip products or software products provided by Geehy. Without the prior permission of Geehy, no unit or individual may copy, transcribe, modify, edit or disseminate all or part of the contents of this document for any reason or in any form.

The “极海” or “Geehy” words or graphics with “®” or “™” in this document are trademarks of Geehy. Other product or service names displayed on Geehy products are the property of their respective owners.

2. No Intellectual Property License

Geehy owns all rights, ownership and intellectual property rights involved in this document.

Geehy shall not be deemed to grant the license or right of any intellectual property to users explicitly or implicitly due to the sale or distribution of Geehy products or this document.

If any third party’s products, services or intellectual property are involved in this document, it shall not be deemed that Geehy authorizes users to use the aforesaid third party’s products, services or intellectual property, unless otherwise agreed in sales order or sales contract.

3. Version Update

Users can obtain the latest document of the corresponding models when ordering Geehy products.

If the contents in this document are inconsistent with Geehy products, the agreement in thesales order or the sales contract shall prevail.

4. Information Reliability

The relevant data in this document are obtained from batch test by Geehy Laboratory or cooperative third-party testing organization. However, clerical errors in correction or errors caused by differences in testing environment may occur inevitably. Therefore, users should understand that Geehy does not bear any responsibility for such errors that may occur in this document. The relevant data in this document are only used to guide users as performance parameter reference and do not constitute Geehy’s guarantee for any product performance.

Users shall select appropriate Geehy products according to their own needs, and effectively verify and test the applicability of Geehy products to confirm that Geehy products meet their own needs, corresponding standards, safety or other reliability requirements. If loses are caused to users due to the user’s failure to fully verify and test Geehy products, Geehy will not bear any responsibility.

5. Legality

USERS SHALL ABIDE BY ALL APPLICABLE LOCAL LAWS AND REGULATIONS WHEN USING THIS DOCUMENT AND THE MATCHING GEEHY PRODUCTS. USERS SHALL UNDERSTAND THAT THE PRODUCTS MAY BE RESTRICTED BY THE EXPORT, RE-EXPORT OR OTHER LAWS OF THE COUNTIRIES OF THE PRODUCTS SUPPLIERS, GEEHY, GEEHY DISTRIBUTORS AND USERS. USERS (ON BEHALF OR ITSELF, SUBSIDIARIES AND AFFILIATED ENTERPRISES) SHALL AGREE AND PROMISE TO ABIDE BY ALL APPLICABLE LAWS AND REGULATIONS ON THE EXPORT AND RE-EXPORT OF GEEHY PRODUCTS AND/OR TECHNOLOGIES AND DIRECT PRODUCTS.

6. Disclaimer of Warranty

THIS DOCUMENT IS PROVIDED BY GEEHY "AS IS" AND THERE IS NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

GEEHY WILL BEAR NO RESPONSIBILITY FOR ANY DISPUTES ARISING FROM THE SUBSEQUENT DESIGN OR USE BY USERS.

7. Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL GEEHY OR ANY OTHER PARTY WHO PROVIDE THE DOCUMENT "AS IS", BE LIABLE FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, DIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE DOCUMENT (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY USERS OR THIRD PARTIES).

8. Scope of Application

The information in this document replaces the information provided in all previous versions of the document.