



YC1121D

用户手册

Yichip Microelectronics

©2020

Revision History

Version	Date	Author	Description
1.0	2020-07-15	Qin.su	Initial version

目录

1	总体介绍	6
1.1	简介	6
	KEY FEATURES	6
1.2	系统架构	8
1.3	应用架构	9
2	存储空间	10
2.1	RAM 和寄存器地址空间	10
2.1.1	固定 RAM 配置	10
2.1.2	可编程空间 0-Retention	11
2.1.3	可编程空间 1-No Retention	11
2.1.4	硬件寄存器空间	11
2.1.5	CM0 核和 BT 核互访问性	11
2.1.5.1	CM0 访问 sRAM	11
2.1.5.2	BT 核访问 mRAM	11
2.1.5.3	寄存器侧区分 mRAM 和 sRAM	12
2.2	FLASH 说明和固件格式说明	12
2.2.1	芯片上电启动流程	12
2.2.1.1	BT 核上电启动流程	13
2.2.1.2	CM0 核上电启动流程	13
2.3	OTP 说明	14
3	中断系统	15
3.1	中断架构说明	15
3.2	自定义中断说明	15
4	通用输入输出 (GPIO)	17
4.1	GPIO 功能描述	17
4.1.1	通用 I/O	17
4.1.2	专用 I/O	17
4.1.3	GPIO 中断	18
4.1.4	外部唤醒	18
4.2	相关寄存器	18
4.2.1	GPIO 功能配置寄存器 CORE_GPIO_CONF	18
4.2.2	GPIO 功能对应寄存器值	19
4.2.3	GPIO 输入状态寄存器 CORE_GPIO_IN	20
4.2.4	GPIO 唤醒寄存器配置	20
4.2.4.1	GPIO 低有效唤醒寄存器 - CORE_GPIO_WAKEUP_LOW	20
4.2.4.2	GPIO 高有效唤醒寄存器 0- CORE_GPIO_WAKEUP_HIGH	21

4.2.4.3	GPIO 低有效唤醒写入寄存器.....	21
4.2.4.4	GPIO 高有效唤醒写入寄存器.....	22
5	CM0 主频配置-OR 时钟? ?	23
5.1	CM0 主频模式-TODO.....	23
5.1.1	<i>Crystal</i>	23
5.1.2	<i>DPLL</i>	23
5.1.3	<i>RC</i>	23
5.1.4	<i>LPOCLK</i>	23
5.2	相关寄存器.....	23
5.2.1	时钟源选择寄存器- <i>CORE_CLKSEL</i>	23
6	PWM 简介.....	25
6.1	相关寄存器.....	25
6.1.1	<i>PWM 时钟控制寄存器 - CORE_CLKOFF</i>	25
6.1.2	<i>PWM 控制寄存器- CORE_PWM0_REG</i>	25
7	看门狗 (WDT)	27
7.1	看门狗外设时钟.....	27
7.2	启用看门狗.....	27
7.3	看门狗触发.....	27
7.4	相关寄存器.....	27
7.4.1	<i>看门狗开关控制寄存器- CORE_CONFIG</i>	27
7.4.2	<i>看门狗清狗控制寄存器- CORE_WDT</i>	28
8	串行通信 (UART)	29
8.1	UART 外设时钟.....	29
8.2	DMA 支持.....	29
8.3	相关寄存器.....	29
8.3.1	<i>UART 时钟开关寄存器- CORE_CLKOFF</i>	29
8.3.2	<i>UART 时钟源选择寄存器- CORE_CONFIG</i>	29
8.3.3	<i>UART 寄存器</i>	30
8.3.3.1	UART 控制寄存器 0- <i>CORE_UART_CONTROL</i>	30
8.3.3.2	UART 控制寄存器 1- <i>CORE_UART_CONTROL1</i>	30
8.3.3.3	UART0 状态寄存器- <i>CORE_UART_STATE</i>	31
8.3.3.4	UART0 发送寄存器- <i>CORE_UART_TXDATA</i>	32
8.3.4	<i>UARTB 寄存器</i>	32
8.3.4.1	UARTB 控制寄存器 0- <i>CORE_UARTB_CONTROL0</i>	32
8.3.4.2	UARTB 控制寄存器 1- <i>CORE_UARTB_CONTROL1</i>	33
8.3.4.3	UARTB 状态寄存器- <i>CORE_UARTB_STATE</i>	33
8.3.4.4	UARTB 发送寄存器- <i>CORE_UARTB_TXDATA</i>	34
9	ADC (SAR_ADC) -TODO.....	35
9.1	功能描述.....	35
9.2	相关寄存器.....	35

9.2.1	ADC 控制寄存器 0- RFEN_ADC.....	35
9.2.2	ADC 控制寄存器 1- RF_ADC_GC.....	35
9.2.3	ADC 控制寄存器 2- RF_ADC_MODE.....	36
9.2.4	ADC 控制寄存器 3- RF_ADC_CH.....	36
9.2.5	ADC 控制寄存器 4- CORE_SUM_EN.....	36
9.2.6	ADC 数据寄存器 4- CORE_SUMDATA.....	37
9.2.7	ADC 校准值。.....	37

Confidential

1 总体介绍

1.1 简介

The YC1168/YC1155 is a very low power, high performance and highly integrated Bluetooth 5.0 BR/EDR/BLE , designed for operation over the 2400MHz to 2483.5Mhz ISM frequency band.

YC1168/YC1155 is manufactured using advanced 55nm CMOS low leakage process, which offers highest integration, lowest power consumption, lowest leakage current and reduced BOM cost while simplifying the overall system design. Rich peripherals including an 8-channel general purpose ADC, power-on-reset (POR), Arithmetic Accelerators, UART/SPI/I2C and 8 GPIOs(YC1155 is 17 GPIOs), which further reduce overall system cost and size.

YC1168/YC1155 operates with a power supply range from 1.8 to 5.5V and has very low power consumption in both Tx and Rx modes, enabling long lifetimes in battery-operated systems while maintaining excellent RF performance. The device can enter an ultra-low power sleep mode in which the registers and retention memory content are retained while low power Oscillator and sleep timer is ON.

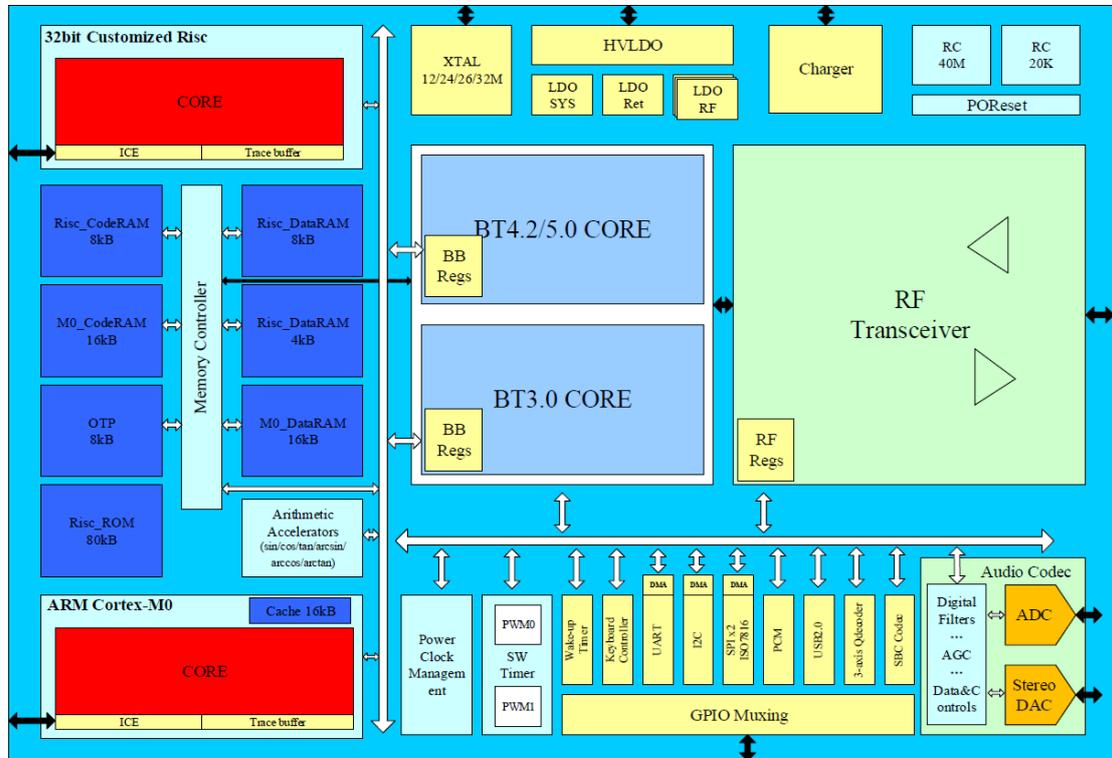
Key Features

- Bluetooth 5.0 Classical/BLE Proprietary double-mode RF SOC
- Charger/Bucker integrated
- Very Low Power Consumption
 - 10nA shut down mode (external interrupts)
 - 800nA sleep mode (32kHz RC OSC, sleep timer and register ON)
 - 2.1uA retention mode (32kHz RC OSC, sleep timer, 2k retention memory and register ON)
 - Rx peak current w/o DCDC
 - 16mA in BLE/2.4G mode
 - Tx peak current w/o DCDC @ -2dBm
 - 22mA in BLE/2.4G mode
 - Rx peak current with DCDC
 - 6.75mA in BLE/2.4G mode
 - Tx peak current with DCDC @ -2dBm
 - 16.5mA in BLE/2.4G mode
 - <25uA avg, 500ms sniff hold connection
- 2.4GHz Transceiver
- Oscillators
 - 16M/24M/32M XTAL supported (default 24M)
 - 40M RC oscillator
 - Low Jitter 20K RC oscillator
- Dual Core Digital Architecture
 - ARM Cortex-M0 Core for application
 - Data RAM 16kB+Cache 16kB
 - CPU clock speed up to 48Mhz
 - 32bit-Risc Core for link management
 - 80kB code ROM and 64Kbit OTP
 - 8kB patch RAM and 8kB data RAM
 - All RAMs can be set to retention mode
- Analog Peripherals
 - 8 channel ADC with 10 bit accuracy/3MSPS
- Digital Peripherals
 - Two-wire Master (I2C compatible), up to 400kbps; UART(RTS/CTS) with HCI-H5 protocol, up to 3.25Mbps; SPI Master, up to 24Mbps, internal QSPI connect **4MB Flash**
 - Individual QSPI can connect external PSRAM
 - AES256 HW encryption

-
- Single-end RFIO
 - -95dBm in BLE mode
 - support 250kbps, 1/2/3Mbps data rates
 - Tx Power up to +9dBm
 - Audio function
 - Mic PGA 0-18dB,3dB per step
 - 16-bits ADC
 - 2x16Bit DAC, Stereo
 - Audio SNR: ADC 88dB; DAC 92dB
 - LED drive capability
 - PWM
 - 20x8 key scan
 - USB2.0 fullspeed,4Eps, support host mode
 - SD Card Host Controller supported

Confidential

1.2 系统架构



Confidential

1.3 应用架构

YC1168/YC1155 是双核系统，分别由 32bit 的 RISC 核（后面简称为：**BT 核**）和 ARM Cortex-M0 组成（后面简称为：**CM0 核**）。两个核有各自的 RAM 空间，互相都可以访问对方的 RAM 区域，平时双核之间的数据通信主要通过 IPC 通道来完成。

根据各自内核的工作特点，BT 核和 CM0 核在具体应用设计中会承担不同的角色。

BT 核：

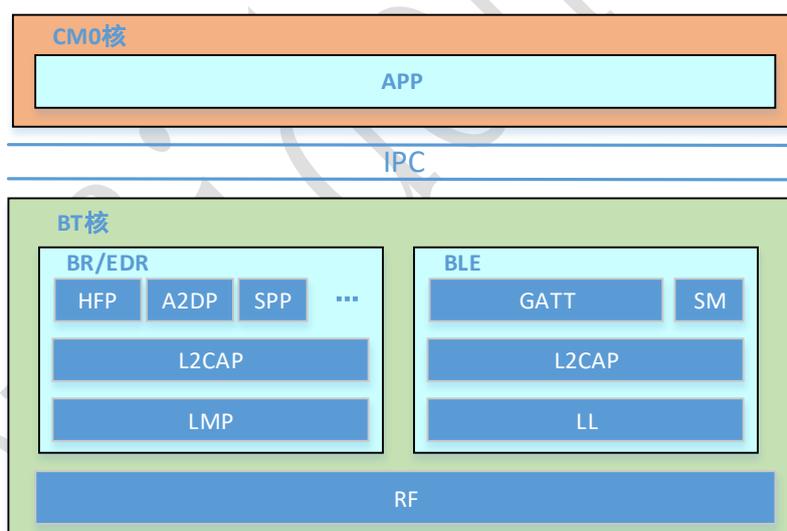
工作主频为 12M，开发语言是汇编语言，在 BT 核中实现了 BR/EDR(Classic) 和 BLE 的协议栈行为。

CM0 核：

工作主频为 24M/48M（可配），默认是 24M 主频，开发语言是 C 语言，CM0 核主要用于完成 APP 业务，如 GPIO 控制，I2C 行为等。

IPC 通道：

用于 BT 核和 CM0 核之间的业务通信。如进入可发现模式，发起连接，断开连接等业务。

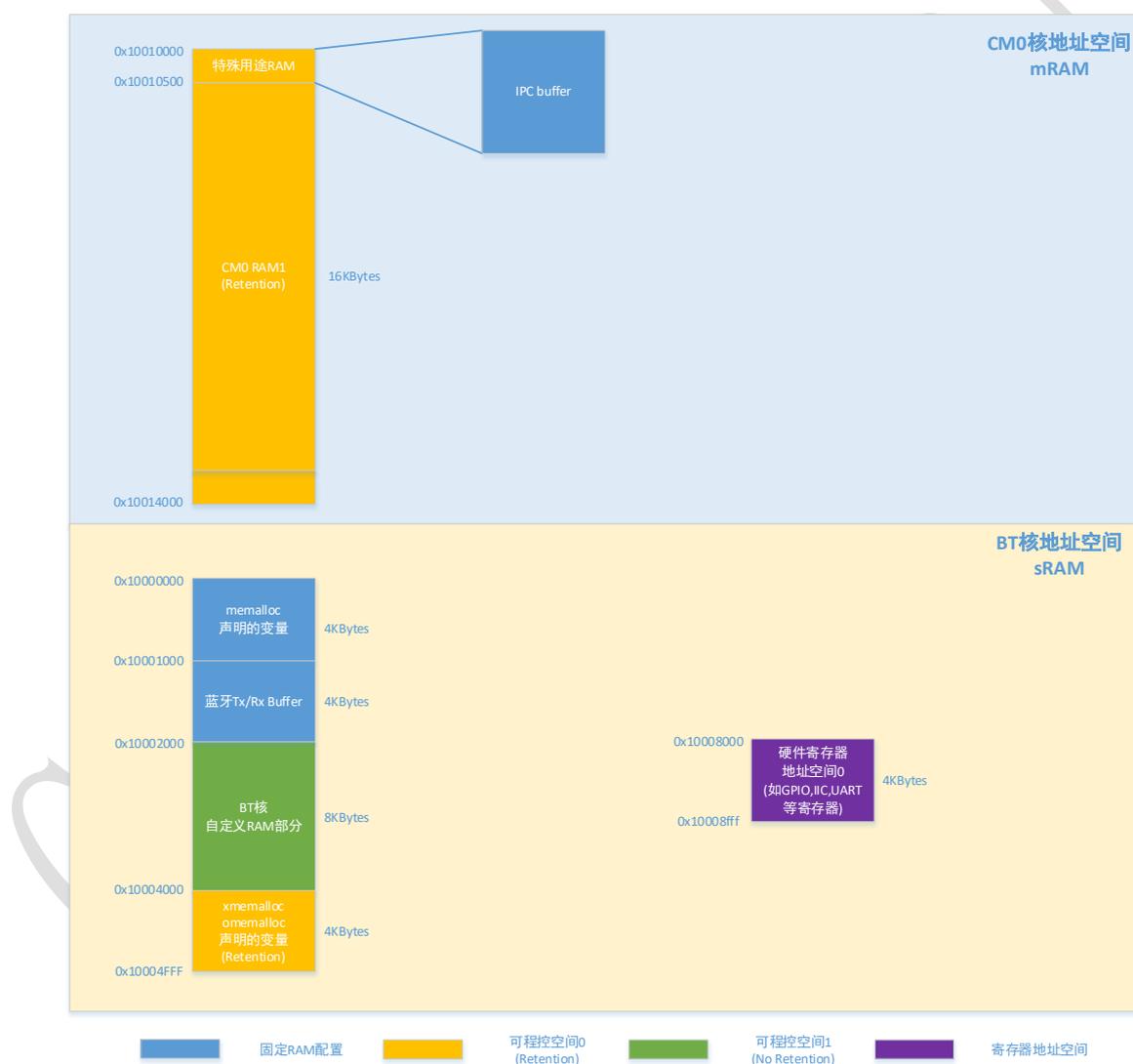


2 存储空间

2.1 RAM 和寄存器地址空间

BT 核和 CM0 都有各自的 RAM 空间，RAM 空间并不是各自独立的，两者之间可以互相访问。下图是 CM0 侧访问的全部 RAM 地址空间。

后续为方便，CM0 内存简称 mRAM，汇编内存简称 sRAM。



2.1.1 固定 RAM 配置

这部分 RAM 空间已被设置为专门的用途，应用开发过程中不建议修改和访问这个区域的 RAM。

2.1.2 可编程空间 0-Retention

这部分 RAM 可以被程序自由使用。这块区域是 Retention 的，在 LPM 模式下是保电不丢失的。

2.1.3 可编程空间 1-No Retention

这部分 RAM 可以被程序自由使用。这块区域是 No Retention 的，在 LPM 模式下是掉电的，所以在设计低功耗产品形态时，使用需要特别注意。

BT 核访问地址只取 0:16 位的地址空间，通过 Bit16 来区分是 CM0 的 RAM 还是 BT 核自身的 RAM。如 BT 核访问 CM0 核的 0x10011000 地址实际是访问 0x10000 地址；访问 BT 核的 0x10001000 地址实际是访问 0x1000 地址。

2.1.4 硬件寄存器空间

硬件寄存器访问的地址空间。通过配置寄存器可以控制 GPIO 等模块。具体硬件寄存器的说明见各个模块说明。

2.1.5 CM0 核和 BT 核互访问性

BT 核和 CM0 核的 RAM 空间可以互相访问。除了两个内核需要区分外，在很多硬件寄存器配置时，通常是设置 2 个字节的地址，所以还需要通过配置 bit16 来区分两边的地址空间。

2.1.5.1 CM0 访问 sRAM

从 CM0 核侧访问，BT 核的地址空间是 0x1000****的地址空间。CM0 核的地址空间是 0x1001****的地址空间。

CM0 可以直接访问 BT 核的 RAM，但是需要注意的是，CM0 核访问 BT 核地址空间时只能按照 Byte 操作，不可以访问连续两个地址。如要取 0x10000000 地址的数据，不能直接 `*(volatile short*)(0x10000000)` 操作，只能用 `*(volatile byte*)(0x10000000) | (*(volatile byte*)(0x10000000) << 8)` 操作。

需要注意的是，汇编核的工作主频是 12M，而 CM0 的工作主频最低是 24M，所以 CM0 访问 sRAM 速度会比较慢。

2.1.5.2 BT 核访问 mRAM

从 BT 核侧访问，BT 核的地址空间是 0x0****的地址空间。CM0 核的地址空

间是 0x1****的地址空间。

BT 核访问地址只取 0:16 位的地址空间,通过 Bit16 来区分是 CM0 的 RAM 还是 BT 核自身的 RAM。如 BT 核访问 CM0 核的 0x10011000 地址实际是访问 0x10000 地址;访问 BT 核的 0x10001000 地址实际是访问 0x1000 地址。

2.1.5.3 寄存器侧区分 mRAM 和 sRAM

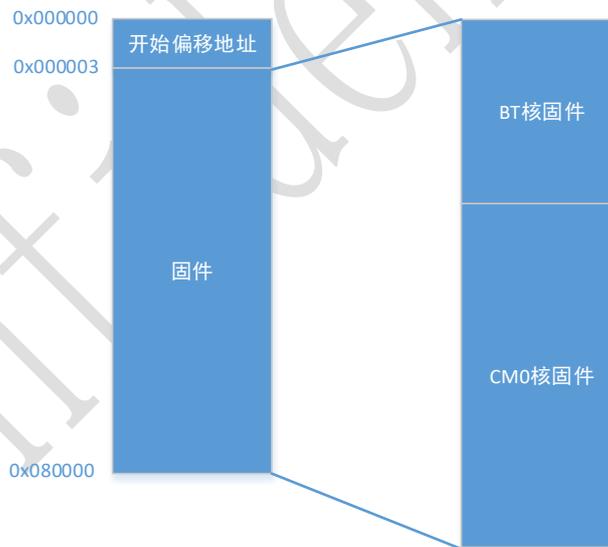
根据具体寄存器要求,设置相关的寄存器,一般有 bit16 设置,用于区分 mRAM 和 sRAM。

2.2 Flash 说明和固件格式说明

YC1168/YC1155 芯片是内置 4Mbits (512KBytes) Flash 的芯片,代码都存储在 Flash 中,通过 QSPI 访问。

由于芯片存在两个内核,对于 Bin 文件的格式有特别的需要。

Flash 中固件存储格式主要如下,前 3 个字节的开始偏移地址指向固件的开始地址(调整开始偏移地址可实现 OTA 等功能),固件分别包含 BT 核固件和 CM0 核固件。



2.2.1 芯片上电启动流程

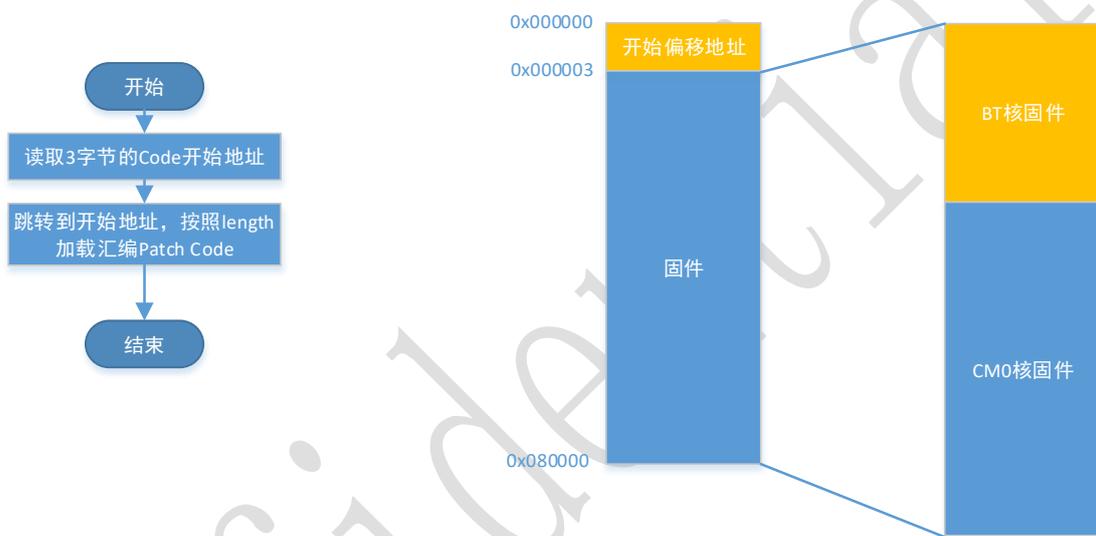
由于 1121D 有 2 个内核,CM0 核在 1121D 芯片中并不能单独工作,需要由汇编核启动。

此外预留在芯片中的 ROM Code 只有汇编代码,芯片上电后,汇编核就会开

始工作，除了完成芯片一些功能模块的初始化外，刚上电时最主要的一个工作就是加载 Patch Code，以便可以快速修改 Bug。

2.2.1.1 BT 核上电启动流程

汇编核上电后，通过 OTP 可以配置正确的 QSPI GPIO 口，这时芯片就可以正常访问 Flash 了，而后芯片先通过读取前 3 个字节的开始偏移地址，确定加载哪个 Code 区域的代码，之后分别加载代码段和数据段，如下图的黄色区域部分。其他部分的 Flash 内容并不在芯片上电加载工作中。



2.2.1.2 CM0 核上电启动流程

在 BT 核正常启动后，BT 核可以根据需要启动 CM0 核，以便完成更加丰富的功能需要。

从 Flash 结构可以发现，CM0 代码的开始地址段并没有直接给出来，CM0 的代码是拼接在 BT 核之后的，在实际场景中，CM0 的起始地址不一定连续拼接在 BT 核之后，这样导致一个问题，BT 核无法直观知道 CM0 代码的起始地址，不方便将 CM0 的代码信息配置到相关寄存器中。

由于 CM0 全是 Flash 代码，每次代码编译后，代码的起始地址都不相同。为解决该问题，在编译的时候，在拼接 Flash 文件时，需要将 CM0 代码的开始偏移地址写入到 BT 核数据区，这样 BT 核只需要读取该变量就可以获得 CM0 代码的开始位置。

当获取到该偏移地址后，就可以知道 CM0 代码在 Flash 中的位置，而后将相关信息写入到 XIP 寄存器中，就可以启动 CM0 核了。

2.3 OTP 说明

芯片目前共有 8Bytes 大小的 OTP，目前只用了 0x100 多个 byte 用于存储加载 FLASH 的 GPIO 配置和 ADC 校准值数据，。

Confidential

3 中断系统

3.1 中断架构说明

同 CM0 的中断框架，中断支持嵌套中断，和中断优先级，具体使用方法同 CM0，下面对程控中断源进行说明。可参考《ARM Cortex-M0 权威指南》相关章节。

3.2 自定义中断说明

除了 CM0 架构基本的 Reset/NMI/Hardfault/SVCall/PendSV/SysTick 等中断。本芯片根据需要设计了 26 个外部中断，具体中断使用方式详见各个章节的说明，中断定义如下：

中断号	中断入口	中断说明
IRQ0	USB_IRQHandler	USB 业务相关的中断入口
IRQ1	IIC_IRQHandler	IIC 业务相关的中断入口
IRQ2	QSPI_IRQHandler	QSPI 业务相关的中断入口
IRQ3	SPI_IRQHandler	SPI 业务相关的中断入口
IRQ4	UART_IRQHandler	UART 业务相关的中断入口
IRQ5	UARTB_IRQHandler	UARTB 业务相关的中断入口
IRQ6	ADC_IRQHandler	ADC 业务相关的中断入口
IRQ7	IIS_IRQHandler	IIS 业务相关的中断入口
IRQ8	BT_IRQHandler	BT 核事件上报中断入口
IRQ9	GPIO0_IRQHandler	GPIO0 中断入口
IRQ17	GPIO8_IRQHandler	GPIO8 中断入口
IRQ18	GPIO9_IRQHandler	GPIO9 中断入口
IRQ19	GPIO10_IRQHandler	GPIO10 中断入口
IRQ20	GPIO11_IRQHandler	GPIO11 中断入口
IRQ21	GPIO12_IRQHandler	GPIO12 中断入口
IRQ22	GPIO13_IRQHandler	GPIO13 中断入口
IRQ23	GPIO14_IRQHandler	GPIO14 中断入口
IRQ24	GPIO15_IRQHandler	GPIO15 中断入口
IRQ25	GPIO16_IRQHandler	GPIO16 中断入口
IRQ26	GPIO17_IRQHandler	GPIO17 中断入口
IRQ27	GPIO18_IRQHandler	GPIO18 中断入口
IRQ28	GPIO19_IRQHandler	GPIO19 中断入口
IRQ29	GPIO20_IRQHandler	GPIO20 中断入口
IRQ30	GPIO21_IRQHandler	GPIO21 中断入口

Confidential

4 通用输入输出（GPIO）

4.1 GPIO 功能描述

YC1168 芯片共有 8 个 GPIO，而 YC1155 共有 17 个 GPIO，每个 GPIO 口都可以复用为多种功能。

IO 口输入模式支持浮空输入、输入上拉以及输入下拉等多种输入模式。

4.1.1 通用 I/O

除 ADC 和硬件 Wakeup（该功能不同于 GPIO 唤醒，用于特殊模式）外，40 个 GPIO 口都可以配置为所支持的所有功能。

4.1.2 专用 I/O

如下表所示 YC1168 没有 EXEN 功能，YC1155 的 EXEN 功能只有 GPIO13 支持，ADC 和 Wakeup 只有选中的几个 IO 支持。

GPIOs	Function1	Function2
GPIO[0]		
GPIO[1]		
GPIO[2]		
GPIO[3]		
GPIO[4]		
GPIO[5]		
GPIO[6]		
GPIO[7]		
GPIO[8]		
GPIO[9]		
GPIO[10]		
GPIO[11]		
GPIO[12]		
GPIO[13]	EXEN	
GPIO[14]		
GPIO[15]		
GPIO[16]		
GPIO[17]		adc0
GPIO[18]		adc1

GPIO[19]		
GPIO[20]		adc2
GPIO[21]		adc3
GPIO[22]		adc4
GPIO[23]		adc5
GPIO[24]		adc6
GPIO[25]		adc7
GPIO[26]		
GPIO[27]		wakeup1
GPIO[28]		wakeup2
GPIO[29]		
GPIO[30]		
GPIO[31]		
GPIO[32]		
GPIO[33]		
GPIO[34]		
GPIO[35]		
GPIO[36]		
GPIO[37]		
GPIO[38]		
GPIO[39]		

4.1.3 GPIO 中断

GPIO 支持高电平中断。

4.1.4 外部唤醒

将 GPIO 配置为特定输入模式后。芯片所有 GPIO 管脚均支持超低功耗唤醒，GPIO 支持低电平唤醒或高电平唤醒。每个 GPIO 都有独立的 GPIO 低功耗唤醒使能位，由低功耗域下变量《mem_gpio_wakeup_low》与《mem_gpio_wakeup_high》来控制。BT 核在进入低功耗模式时会读取这两个变量，并保存到相关寄存器中。

4.2 相关寄存器

4.2.1 GPIO 功能配置寄存器 CORE_GPIO_CONF

功能：GPIO 配置寄存器

地址：0x10008080+0x00

长度：32Bytes

说明：第 n 个 GPIO 的配置寄存器地址为 0x10008080 +n

Address	Bit	功能	说明	R/W	复位值
0x10008080	[5:0]	GPIO0 IO 功能	赋值与功能对应关系见 IO 功能复用表	R/W	0
	[7:6]	GPIO0 IO 模式	00:FLOAT 01:PULL UP 10:PULL DOWN 11: ANALOG	R/W	0
	[255:8]	GPIO1-GPIO31	每个 IO 一个 Byte	R/W	0

4.2.2 GPIO 功能对应寄存器值

寄存器值	功能	寄存器值	功能
0	Input(float)	1	预留
2	QSPI_NCS	3	QSPI_SCK
4	QSPI_IO0	5	QSPI_IO1
6	QSPI_IO2	7	QSPI_IO3
8	UART0_TXD	9	UART0_RXD
10	UART0_RTS	11	UART0_CTS
12	UART1_TXD	13	UART1_RXD
14	UART1_RTS	15	UART1_CTS
16	PWM_OUT0	17	PWM_OUT1
18	PWM_OUT2	19	PWM_OUT3
20	PWM_OUT4	21	PWM_OUT5
22	PWM_OUT6	23	PWM_OUT7
24	I2S_DOUT	25	I2S_LRCKOUT
26	I2S_CLKOUT	27	预留
28	I2S_DIN	29	I2S_LRCKIN
30	I2S_CLKIN	31	SPID_MISO
32	SPID_NCS	33	SPID_SCK
34	SPID_MOSI	35	SPID_SDIO
36	SPID_NCSI	37	SPID_SCKI
38	QDEC_X0	39	QDEC_X1

40	QDEC_Y0	41	QDEC_Y1
42	QDEC_Z0	43	QDEC_Z1
44	IIC_SCL	45	IIC_SDA
46	DAC OUTP	47	DAC OUTN
48	预留	49	预留
50	预留	51	预留
52	预留	53	预留
54	预留	55	预留
56	预留	57	预留
58	预留	59	预留
60	JTAG_SW_CLOCK	61	JTAG_SW_DATA
62	GPIO_OUTPUT_LOW	63	GPIO_OUTPUT_HIGH
0x40	PULL UP	0x80	PUUL DOWN
0xC0	ANALOG		

4.2.3 GPIO 输入状态寄存器 CORE_GPIO_IN

功能：input 模式下读取 GPIO 状态寄存器

地址：0x1000831C

长度：4Bytes

说明：从第0bit 开始依次表示 GPIO0 到 GPIO31, 每个 bit 分别表示一个 GPIO 的状态。

Address	Bit	功能	说明	R/W	复位值
0x1000831C	[31:0]	GPIO0-GPIO31 输入状态位	0: 低电平 1: 高电平	R	0

4.2.4 GPIO 唤醒寄存器配置

GPIO 唤醒寄存器的操作通常都由 BT 核在进入低功耗模式时配置。CM0 只需要配置好 BT 核变量《mem_gpio_wakeup_low》和《mem_gpio_wakeup_high》即可。这两个变量长度为 5Bytes，每个 bit 代表对应唤醒功能是否开启。

4.2.4.1 GPIO 低有效唤醒寄存器- CORE_GPIO_WAKEUP_LOW

功能：读取当前 GPIO 低有效唤醒寄存器

地址：0x10008342

长度：4Bytes

说明：从第 0bit 开始依次表示 GPIO0 到 GPIO31，每个 bit 分别表示对应 GPIO 是否开启了低有效唤醒功能。在 BT 核完成该寄存器的读写管理。

Address	Bit	功能	说明	R/W	复位值
0x10008342	[31:0]	GPIO0-GPIO39 低有效唤醒	0: 未使能 1: 使能	R	0

4.2.4.2 GPIO 高有效唤醒寄存器 0- CORE_GPIO_WAKEUP_HIGH

功能：读取当前 GPIO 高有效唤醒寄存器

地址：0x10008346

长度：4Bytes

说明：从第 0bit 开始依次表示 GPIO0 到 GPIO31，每个 bit 分别表示对应 GPIO 是否开启了高有效唤醒功能。在 BT 核完成该寄存器的读写管理。

Address	Bit	功能	说明	R/W	复位值
0x10008347	[31:0]	GPIO0-GPIO31 高有效唤醒	0: 未使能 1: 使能	R	0

4.2.4.3 GPIO 低有效唤醒写入寄存器

功能：写入 GPIO 低有效唤醒写入寄存器

说明：由于 GPIO 唤醒配置需要在低功耗模式下生效，所以该寄存器写入行为需要特定操作。主要分为 2 步，先将需要写入的数值写入 0x804C；第二步设置特定 LPM 写入寄存器。

写入步骤：

Step1: 0x1000804C 写入 4Bytes 的所需配置 GPIO 低有效配置

Step2: 0x10008005 写入 0x40

Step3: 等待 2 LDO 翻转时间。CM0 核操作建议 1ms。

从第 0bit 开始依次表示 GPIO0 到 GPIO31，每个 bit 分别表示对应 GPIO 是否要开启低有效唤醒功能。在 BT 核完成该寄存器的读写管理。

Address	Bit	功能	说明	R/W	复位值
0x1000400C	[31:0]	GPIO0-GPIO31 低有效唤醒	0: 未使能 1: 使能	R/W	0

4.2.4.4 GPIO 高有效唤醒写入寄存器

功能：写入 GPIO 高有效唤醒写入寄存器

说明：由于 GPIO 唤醒配置需要在低功耗模式下生效，所以该寄存器写入行为需要特定操作。主要分为 2 步，先将需要写入的数值写入 0x804C；第二步设置特定 LPM 写入寄存器。

写入步骤：

Step1: 0x1000804C 写入 4Bytes 的所需配置 GPIO 高有效配置

Step2: 0x10008005 写入 0x80

Step3: 等待 2 LDO 翻转时间。CM0 核操作建议 1ms。

从第 0bit 开始依次表示 GPIO0 到 GPIO31，每个 bit 分别表示对应 GPIO 是否需要开启高有效唤醒功能。在 BT 核完成该寄存器的读写管理。

Address	Bit	功能	说明	R/W	复位值
0x1000400C	[31:0]	GPIO0-GPIO31 高有效唤醒	0: 未使能 1: 使能	R/W	0

Confidential

5 CM0 主频配置-OR 时钟？？

5.1 CM0 主频模式-TODO

为满足功耗和性能需要，CM0 的工作主频时钟源可以根据需要配置成多种时钟源，工作主频也可以在 24M 和 48M 切换，包括：

5.1.1 Crystal

晶体时钟源模式。

5.1.2 DPLL

DPLL 时钟源模式。

5.1.3 RC

RC 时钟模式，不使用。

5.1.4 LPOCLK

不使用。

5.2 相关寄存器

5.2.1 时钟源选择寄存器- CORE_CLKSEL

功能：时钟源选择寄存器

说明：一般配置系统时钟源为 crystal，分配系数为 1。

切换 48M 主频时，系统时钟源设置为 DPLL，分配系数为 1。

Address	Bit	功能	说明	R/W	复位值
0x10008042	[1:0]	系统时钟源	0:crystal 1: dp11 2:rc 3:lpoclk	R/W	0

	[4:2]	系统时钟分频系数	0-7	R/W	1
	[7:5]	其他用途	-	R/W	-

Confidential

6 PWM 简介

PWM: 就是脉冲宽度调制, 是一种对模拟信号电平进行数字编码的方法。一个周期包括高电平和低电平状态, 各状态的时间得根据占空比去设置。

芯片一共有 8 路 PWM 通道, 可以同时输出 8 路 PWM 波形。

6.1 相关寄存器

6.1.1 PWM 时钟控制寄存器 - CORE_CLKOFF

功能: 开启或关闭 PWM 模块工作时钟

说明: 要使用 PWM 模块时, 必须开启该模块的时钟。

Address	Bit	功能	说明	R/W	复位值
0x10008051	[4:0]	其他用途	-	R/W	0
	[5]	1: turn off PWM's clock	-	R/W	0
	[7:6]	其他用途	-	R/W	0

6.1.2 PWM 控制寄存器—CORE_PWM0_REG

Address	Bit	功能	说明	R/W	复位值
0x100080CD	[15:0]	PWM Channel0 高电平持续时间	-	R/W	0
0x100080CF	[15:0]	PWM Channel0 低电平持续时间	-	R/W	0
0x100080D1	[2:0]	PWM Channel0 时钟源分频系数	0-7:在系统主频 基础上分频, 如 系统主频为 24M, 分频系数 设置为 2, 这工 作频率为 12M	R/W	0
	[3]	PWM0 和 PWM1 输出同步	0:关闭 1:开启	R/W	0
	[4]	PWM0 先输出高 电平	0:先低电平 1:先高电平	R/W	0
	[5]	打开 PWM0	0:关闭 1:开启	R/W	0
0x100080D2	[279:0]	PWM1-PWM7 控 制寄存器	每个 PWM 有 5 个字节配置, 配 置参数如上所示	R/W	0

Confidential

7 看门狗（WDT）

由于 1168/1155 有 2 个 CPU 核，所以所设计的看门狗也有两个，每个核各自控制一个。当配置为 Reset 模式时，当一个核的看门狗触发时，芯片直接硬件重启。

7.1 看门狗外设时钟

看门狗外设时钟由 32K 时钟源提供，即看门狗外设计数器计数器工作在 32K 频率上。

7.2 启用看门狗

看门狗开启后可以根据需要关闭看门狗功能。

7.3 看门狗触发

这个看门狗只有系统复位模式，即计数满后，系统立即产生复位。只支持一秒内清狗，无法设置清狗时间，

7.4 相关寄存器

7.4.1 看门狗开关控制寄存器- CORE_CONFIG

功能：配置看门狗功能开关。

说明：1、分别控制两个开门狗的功能开关。

Address	Bit	功能	说明	R/W	复位值
0x10008043	[1:0]	其他用途	-	R/W	0
	[2]	CM0 核看门狗开关	0:关闭 1:开启	R/W	0
	[3]	BT 核看门狗开关	0:关闭 1:开启	R/W	0
	[7:4]	其他用途	-	R/W	0

7.4.2 看门狗清狗控制寄存器- CORE_WDT

功能：配置看门狗的清狗标志位。

说明：清狗动作，通过写入不同值来清除当前计数，以确保看门狗不被触发

Address	Bit	功能	说明	R/W	复位值
0x10008004	[0]	其他用途	-	R/W	0
	[1]	BT 看门狗清狗标志	-	R/W	0
	[2]	CM0 看门狗清狗标志	-	R/W	0

Confidential

8 串行通信 (UART)

通用异步收发器(UART)提供了一种灵活的方法与使用工业标准 NRZ 异步串行数据格式的部设备之间进行全双工数据交换。UART 利用波特率发生器提供宽范围的波特率选择。

芯片最大支持 2 个串口，串口波特率最大支持 2M 的速率。

8.1 UART 外设时钟

UART 外设时钟源可以配置为 Crystal 或者 DPLL 固定 48MHz，实际应用场景中一般配置为 DPLL，主频 48M 不会根据系统时钟的变化而改变。

8.2 DMA 支持

UART 外设使用 DMA 功能可以有的效减少系统 MIPS，提高数据传输效率。每个 UART 外设可以使用 2 个 DMA 通道，分别用来接收和发送数据。

8.3 相关寄存器

8.3.1 UART 时钟开关寄存器-CORE_CLKOFF

功能：开启或关闭 UART 模块工作时钟

说明：要使用 UART 模块时，必须开启该模块的时钟。

Address	Bit	功能	说明	R/W	复位值
0x10008051	[6:0]	其他用途	-	R/W	0
	[7]	开关 UART 模块时钟	0:开启 1:关闭	R/W	0

8.3.2 UART 时钟源选择寄存器-CORE_CONFIG

功能：配置 UART 工作时钟源

说明：使用 UART 模块前需要配置好 UART 模块的时钟源，建议选择 DPLL 时钟源。

Address	Bit	功能	说明	R/W	复位值
0x10008043	[0]	UART 时钟源选择	0:Crystal 1: DPLL 48M	R/W	0

	[7:1]	其他用途	-	R/W	0
--	-------	------	---	-----	---

8.3.3 UART 寄存器

8.3.3.1 UART 控制寄存器 0-CORE_UART_CONTROL

功能：配置 UART 参数 0。

说明：1、使用 UART 模块前需要配置好参数。

2、使用自动波特率时，需要通过状态寄存器读取波特率值。

Address	Bit	功能	说明	R/W	复位值
0x10008060	[0]	使能 UART	0:关闭 UART 1:使能 UART (初始化寄存器时先关闭再使能)	R/W	0
	[1]	设置奇偶校验	0:Parity_Even 1:Parity_Odd	R/W	0
	[2]	设置字长	0:8 bits 字长 1: 9 bits 字长	R/W	0
	[3]	设置停止位	0:一个停止位 1:两个停止位	R/W	0
	[4]	设置流控	0:无流控 1:使能流控	R/W	0
	[5]	设置智能卡	0:关闭智能卡模式 1:开启智能卡模式	R/W	0
	[6]	设置单线半双工模式	0:关闭 1:开启	R/W	0
	[7]	设置重置波特率标志	0:使用自动波特率 1:使用设定波特率	R/W	0

8.3.3.2 UART 控制寄存器 1- CORE_UART_CONTROL1

功能：配置 UART 参数 1。

说明：使用 UART 模块前需要配置好参数。

Address	Bit	功能	说明	R/W	复位值
0x10008052	[14:0]	设置波特率	写入所需的波特率	R/W	0

	[15]	设置 UART DMA MRAM 标记	0:DMA 使用 sRAM 1:DMA 使用 mRAM	R/W	0
0x10008054	[15:0]	设置 DMA RX 开始地址	设置 16 位的 UART DMA RX 开始地址	R/W	0
0x10008056	[15:0]	设置 DMA RX 结束地址	设置 16 位的 UART DMA RX 结束地址	R/W	0
0x10008058	[15:0]	设置 DMA RX 读地址	设置当前 16 位的 UART DMA RX 读地址	R/W	0
0x1000805A	[15:0]	设置 DMA TX 开始地址	设置 16 位的 UART DMA TX 开始地址	R/W	0
0x1000805C	[15:0]	设置 DMA TX 结束地址	设置 16 位的 UART DMA TX 结束地址	R/W	0
0x1000805E	[15:0]	设置 DMA TX 写地址	设置当前 16 位的 UART DMA TX 写地址	R/W	0

8.3.3.3 UART0 状态寄存器- CORE_UART_STATE

功能：UART0 状态控制寄存器。

说明：1、自动检测波特率较长，需要跨越多个寄存器读取。

2、UART 收发需要控制发送和接收 DMA 的读写指针。

Address	Bit	功能	说明	R/W	复位值
0x1000830C	[0]	TX FIFO FULL	发送 FIFO 是否已满	R	0
	[1]	TX FIFO empty	发送 FIFO 是否为空	R	0
	[2]	TX Busy	发送是否正在进行	R	0
	[3]	RX FIFO empty	接收 FIFO 是否为空	R	0
	[7:4]	自动检测波特率	获取自动检测到的波特率中的[3:0]	R	0
0x1000830D	[7:0]	自动检测波特率	获取自动检测到的波特率中的[12:4]	R	0
0x1000830E	[15:0]	TX DMA 剩余发送个数	获取当前 TX DMA 中剩余的发送数据总数	R	0
0x10008310	[15:0]	TX DMA 读指针	获取当前 TX DMA 的读指针	R	0
0x10008312	[15:0]	RX DMA 剩余读取个数	获取当前 RX DMA 中剩余的未读取数据总数	R	0
0x10008314	[15:0]	RX DMA 写指针	获取当前 RX DMA 的写指针	R	0

8.3.3.4 UART0 发送寄存器- CORE_UART_TXDATA

功能：UART 发送单个 Byte 的控制寄存器。

说明：发送效率较低，适用于部分特别用途场景。

Address	Bit	功能	说明	R/W	复位值
0x10008008	[7:0]	TX 单个 Byte	发送单个 Byte	W	0

8.3.4 UARTB 寄存器

8.3.4.1 UARTB 控制寄存器 0-CORE_UARTB_CONTROLO

功能：配置 UARTB 参数 0。

说明：1、使用 UARTB 模块前需要配置好参数。

2、使用自动波特率时，需要通过状态寄存器读取波特率值。

Address	Bit	功能	说明	R/W	复位值
0x10008071	[0]	使能 UART	0:关闭 UART 1:使能 UART (初始化寄存器时先关闭再使能)	R/W	0
	[1]	设置奇偶校验	0:Parity_Even 1:Parity_Odd	R/W	0
	[2]	设置字长	0:8 bits 字长 1: 9 bits 字长	R/W	0
	[3]	设置停止位	0:一个停止位 1:两个停止位	R/W	0
	[4]	设置流控	0:无流控 1:使能流控	R/W	0
	[5]	设置智能卡	0:关闭智能卡模式 1:开启智能卡模式	R/W	0
	[6]	设置单线半双工模式	0:关闭 1:开启	R/W	0
	[7]	设置重置波特率标志	0:使用自动波特率 1:使用设定波特率	R/W	0

8.3.4.2 UARTB 控制寄存器 1- CORE_UARTB_CONTROL1

功能：配置 UARTB 参数 1。

说明：使用 UARTB 模块前需要配置好参数。

Address	Bit	功能	说明	R/W	复位值
0x10008072	[14:0]	设置波特率	写入所需的波特率	R/W	0
	[15]	设置 UART DMA MRAM 标记	0:DMA 使用 sRAM 1:DMA 使用 mRAM	R/W	0
0x10008074	[15:0]	设置 DMA RX 开始地址	设置 16 位的 UART DMA RX 开始地址	R/W	0
0x10008076	[15:0]	设置 DMA RX 结束地址	设置 16 位的 UART DMA RX 结束地址	R/W	0
0x10008078	[15:0]	设置 DMA RX 读地址	设置当前 16 位的 UART DMA RX 读地址	R/W	0
0x1000807A	[15:0]	设置 DMA TX 开始地址	设置 16 位的 UART DMA TX 开始地址	R/W	0
0x1000807C	[15:0]	设置 DMA TX 结束地址	设置 16 位的 UART DMA TX 结束地址	R/W	0
0x1000807E	[15:0]	设置 DMA TX 写地址	设置当前 16 位的 UART DMA TX 写地址	R/W	0

8.3.4.3 UARTB 状态寄存器- CORE_UARTB_STATE

功能：UARTB 状态控制寄存器。

说明：1、自动检测波特率较长，需要跨越多个寄存器读取。

2、UART 收发需要控制发送和接收 DMA 的读写指针。

Address	Bit	功能	说明	R/W	复位值
0x10008354	[0]	TX FIFO FULL	发送 FIFO 是否已满	R	0
	[1]	TX FIFO empty	发送 FIFO 是否为空	R	0
	[2]	TX Busy	发送是否正在进行	R	0
	[3]	RX FIFO empty	接收 FIFO 是否为空	R	0
	[7:4]	自动检测波特率	获取自动检测到的波特率中的[3:0]	R	0
0x10008355	[7:0]	自动检测波特率	获取自动检测到的波特率	R	0

		特率	率中的[12:4]		
0x10008356	[15:0]	TX DMA 剩余发送个数	获取当前 TX DMA 中剩余的发送数据总数	R	0
0x10008358	[15:0]	TX DMA 读指针	获取当前 TX DMA 的读指针	R	0
0x1000835A	[15:0]	RX DMA 剩余读取个数	获取当前 RX DMA 中剩余的未读取数据总数	R	0
0x1000835C	[15:0]	RX DMA 写指针	获取当前 RX DMA 的写指针	R	0

8.3.4.4 UARTB 发送寄存器- CORE_UARTB_TXDATA

功能：UARTB 发送单个 Byte 的控制寄存器。

说明：发送效率较低，适用于部分特别用途场景。

Address	Bit	功能	说明	R/W	复位值
0x10008009	[7:0]	TX 单个 Byte	发送单个 Byte	W	0

9 ADC (SAR_ADC) -TODO

ADC 采样率为 600kHz,采样精度为 10 比特。

9.1 功能描述

ADC 有 8 个通道,最高采样率位 600kHz,最高采样精度为 10 比特,AD 的参考电压为 1V 和 0.5V,校准值写在 OTP 中,可测量电压范围为 0-1.4V。

9.2 相关寄存器

9.2.1 ADC 控制寄存器 0- RFEN_ADC

功能: ADC 使能寄存器

说明: 使能 ADC, 固定配置 0x7c

Address	Bit	功能	说明	R/W	复位值
0x10008906	[1:0]	其他用途	-	R/W	0
	[2]	clkpll_en_clk2sar adc	-	R/W	0
	[3]	misc_saradc_en	-	R/W	0
	[4]	misc_saradc_en_ biasgen	-	R/W	0
	[5]	misc_saradc_en_ constgm	-	R/W	0
	[6]	misc_saradc_en_ reg	-	R/W	
	[7]	其他用途	-	R/W	0

9.2.2 ADC 控制寄存器 1- RF_ADC_GC

功能: ADC 控制寄存器

说明: ADC 通道的选择, 固定配置 0xaa

Address	Bit	功能	说明	R/W	复位值
0x10008972	[1:0]		-	R/W	11
	[3:2]		-	R/W	10
	[5:4]		-	R/W	10

	[7:6]		-	R/W	10
--	-------	--	---	-----	----

9.2.3 ADC 控制寄存器 2- RF_ADC_MODE

功能：ADC 控制寄存器

说明：选择 ADC 通道

Address	Bit	功能	说明	R/W	复位值
0x10008971	[1:0]		-	R/W	11
	[3:2]		-	R/W	11
	[6:4]	测试模式： 0:GPIO 1:GPIO DIFF 2:HVIN 3:VINLPM	-	R/W	000
	[7]		-	R/W	1

9.2.4 ADC 控制寄存器 3-RF_ADC_CH

功能：ADC 控制寄存器

说明：选择 ADC 通道

Address	Bit	功能	说明	R/W	复位值
0x10008973	[2:0]	ADC 通道	000-GPIO17 001-GPIO18 010-GPIO20 011-GPIO21 100-GPIO22 101-GPIO23 110-GPIO24 111-GPIO25	R/W	000
	[5:3]		-	R/W	110
	[6]		-	R/W	0
	[7]		-	R/W	0

9.2.5 ADC 控制寄存器 4- CORE_SUM_EN

功能：ADC 控制寄存器

说明：使能 ADC

Address	Bit	功能	说明	R/W	复位值
---------	-----	----	----	-----	-----

0x10008063	[7]	enable ADC sum	-	R/W	0
------------	-----	----------------	---	-----	---

9.2.6 ADC 数据寄存器 4- CORE_SUMDATA

功能：ADC 数据寄存器

说明：读取 ADC 数据

Address	Bit	功能	说明	R/W	复位值
0x10008063	[15: 0]	adc data	-	R	0

9.2.7 ADC 校准值。

每颗芯片出厂前都会在 OTP 中烧录一串 adc 校准值，上电时，由 BT 读取该变量的值然后保存至变量《mem_0_5_adc_io_data》、《mem_3v_adc_hvin_data》、《mem_3v_adc_vinlpm_data》、《mem_1v_adc_io_data》、《mem_5v_adc_hvin_data》、《mem_3v3_adc_vinlpm_data》和《mem_otp_adc_flag》中，其中变量《mem_otp_adc_flag》的值为 0xaa55,如果不等于 0xaa55，则说明该校准值无效。

Confidential

Confidential