



# **V94XX(A)**

## **数据手册**

Specifications are subject to change without notice.

© 2022 Vango Technologies, Inc.

This document contains information that is proprietary to Vango Technologies, Inc.  
Unauthorized reproduction of this information in whole or in part is strictly prohibited.

V94XX(A)是一款高性能、低功耗的单相电能计量 SoC 芯片，集成了 Cortex-M0 核，模拟前端、电能计量模块，最多 256KB FLASH (V94XXA 具有 128KB flash)，32KB SRAM (V94XXA 具有 16KB SRAM)，UART、SPI、I<sup>2</sup>C 接口，LCD，WDT 和 RTC。V94XX(A)系列支持多种低功耗工作模式。

V94XX(A)内置一个电能计量模块，支持多种模式的全波和基波能量计量，并支持监测多种电网事件。同时，波形数据可通过 Px 以 SPI 协议传输出去，或者通过波形缓存存在本地。

## 特点

- 工作电压：2.2V~3.6V  
其中 ADC 部分和计量部分支持的工作电压范围：2.6V~3.6V
- 工作电流：
  - 正常模式不带计量：1.633mA@6.5536MHz
  - 正常模式带计量：4.233mA@6.5536MHz
  - IDLE 模式：0.379mA@6.5536MHz
  - 浅睡眠模式(LCD 开启)：9.3μA
  - 浅睡眠模式(LCD 关闭)：3.1μA
  - 深睡眠模式：2.9 μA
- 封装：
  - LQFP100(V9400(A))
  - LQFP80(V9430)
  - LQFP80(V9431(A))
  - LQFP64(V9410(A))
  - LQFP48(V9420A)
- 工作温度：-40~+85°C
- 储存温度：-55~+150°C
- MCU
  - 32 位 Cortex-M0 核，最高速度高达 26.2144 MHz
  - 单指令周期乘法
  - 标准两线 SWD 调试接口
  - V94XX 内置 256KB FLASH 存储器，V94XXA 内置 128KB FLASH 存储器。FLASH 具有写保护和加密功能，支持 ISP 和 IAP
  - V94XX 内置 32K 字节的 SRAM，V94XXA 内置 16K 字节的 SRAM。SRAM 带奇偶校验，在浅睡眠模式下可数据保留
  - 256 字节的 SRAM，在深睡眠模式下也有数据保留
  - 可实时监测异常，包括 FLASH 校验和错误，SRAM 奇偶校验错误，内存地址错误和内存对齐错误
- 外设
  - 最多 5 个 UART 控制器，具有奇偶校验功能
  - 每个串口发送通道可以配置带红外载波以进行红外传输
  - 最多 1 个 ISO7816 控制器
  - 最多 1 个 SPI 控制器（主/从机制）
  - 最多 1 个 I<sup>2</sup>C 控制器（主/从机制）

- 4 个 32 位定时器
- 4 个 16 位 PWM 定时器
- 4 通道 DMA 控制器
- 128/192/256 位 AES 编解码器
- ECC 加速算法引擎
- 双帧缓存 LCD 控制器
  - ✓ 4COM/6COM/8COM
  - ✓ 1/3 或 1/4 偏置
  - ✓ 支持多类型的扫描频率
  - ✓ LCD 电压：默认输出电压是 3.3V 左右，2.7~3.6V 可调节，调节步长是 0.06V
- 周期可编程的看门狗定时器
- 支持多唤醒源
- 最多 75 个 GPIO
- 最多 14 个 GPIO 外部中断唤醒口
- 模拟控制器
  - 16 位 ADC，具有 10 Ksps 及最多 9 个外部输入
  - ADC 支持手动采样模式或自动采样模式
  - 2 路比较器，单端输入或差分输入
  - 内置 32KHz 及 6.5 MHz RC 时钟
  - 内置 2 个 PLL
  - 支持外部 32.768 KHz 晶体或 6.5536 MHz 晶体（可选）
  - 内置 32.768 KHz 和 6.5536 MHz 晶体的监测电路
  - 每个时钟都可选做为系统时钟
  - 支持数字时钟分频器，最高可达 1/256
  - 支持可编程的低电压监测
  - 支持 DVCC1 的上电复位
  - 支持电压监测
- RTC
  - 支持 RTC 1 ppm 级别的手动校正
- 电能计量
  - 3 路独立的过采样  $\Sigma/\Delta$  ADC：1 路电压；1 路 A 通道电流；另外 1 路 B 通道电流
  - 计量精度高：
    - 满足 IEC 62053-21:2020/ IEC 62053-22:2020 和 IEC 62053-23:2020 的要求
    - 5000:1 动态范围内，全波/基波有功能量计量误差小于 0.1%
    - 5000:1 动态范围内，全波/基波无功能量计量误差小于 0.2%
    - 5000:1 动态范围内，全波/基波电压/电流有效值误差小于 0.5%
  - 提供各种测量数据：
    - 电压/电流信号直流分量
    - 全波/基波电压/电流有效值瞬时值/平均值
    - 全波/基波有功/无功功率瞬时值/平均值

- 全波 10 或 12 周波有效值
- 全波视在功率瞬时值/平均值
- 正向/反向能量，有功/无功/视在功率/电流有效值/常数/基波通道可选
- 频率和相位
- 支持直流信号测量
- 支持软件校表
- 支持小信号加速校表
- 支持过流、过压、欠流、欠压、电压骤升、电压骤降检测
- 支持波形缓存和波形输出
- 电流输入：支持锰铜、CT、霍尔及 TMR

# 产品系列

外设	V9400(A)	V9430	V9431(A)	V9410(A)	V9420A
ADC 输入通道	9	8	8	5	4
TinyADC 输入通道	2	2	2	1	2
UART	5	3	5	5	5
UART32K	2	2	2	2	2
ISO7816	1	0	1	1	1
SPI	1	1	1	1	1
I <sup>2</sup> C	1	1	1	1	1
比较器	2	2	2	1	1
GPIO	75	59	63	48	32
外部中断 IO	14	11	14	10	8
32 位定时器	4	4	4	4	4
16 位 PWM 定时器	4	4	4	4	4
PWM_Out	4	4	4	4	4
DMA channel	4	4	4	4	4
LCD	65x4, 63x6, 61x8	51x4, 49x6, 47x8	55x4, 53x6, 51x8	41x4, 39x6, 37x8	25x4, 23x6
FLASH	256KB(94XX) 128KB(94XXA)	256KB	256KB(94XX) 128KB(94XXA)	256KB(94XX) 128KB(94XXA)	128KB
SRAM	32KB(94XX) 16KB(94XXA)	32KB	32KB(94XX) 16KB(94XXA)	32KB(94XX) 16KB(94XXA)	16KB
电能计量通道	3	3	3	3	3
封装	100-LQFP	80-LQFP	80-LQFP	64-LQFP	48-LQFP

# 声明

杭州万高科技股份有限公司保留对本手册所涉及的产品及相关的技术信息进行补正或更新的权利。使用本手册时，请您从我们的销售渠道或登录公司网站 [http: //www.vangotech.com](http://www.vangotech.com) 获取最新信息。

# 目录

特点 .....	2
产品系列 .....	5
声明 .....	6
目录 .....	7
第 1 章 电气特性 .....	17
1.1 绝对最大额定值 .....	17
1.2 正常工作电压 .....	17
1.3 驱动特性 .....	17
1.4 功耗 .....	17
1.5 复位和电源控制模块特性 .....	19
1.6 GPIO 特性 .....	19
1.7 通用 ADC 特性 .....	20
1.8 模拟比较器特性 .....	20
1.9 时钟和锁相环特性 .....	21
1.10 FLASH 和 SRAM 特性 .....	21
1.11 晶振的 ESR 特性 .....	22
1.12 电能计量相关特性 .....	22
1.13 时钟输出稳定时间及唤醒时间 .....	23
1.14 通用 ADC 转换时间 .....	24
1.15 TinyADC 转换时间 .....	25
1.16 芯片可靠性分析结果 .....	25
第 2 章 引脚分布图 .....	26
2.1 V9400(A)引脚分布图 .....	26
2.2 V9430 引脚分布图 .....	27
2.3 V9431(A)引脚分布图 .....	28
2.4 V9410(A)引脚分布图 .....	29
2.5 V9420A 引脚分布图 .....	30
2.6 V94XX(A)管脚描述 .....	31
第 3 章 功能框图 .....	41
3.1 系统功能框图 .....	41
3.2 电源功能框图 .....	42
3.3 时钟功能框图 .....	43
第 4 章 内存映射 .....	44
4.1 寄存器地址 .....	45
4.1.1 PMU 寄存器地址 .....	45
4.1.2 ANA 寄存器地址 .....	45
4.1.3 RTC 寄存器地址 .....	46
4.1.4 FLASH 寄存器地址 .....	47
4.1.5 GPIO 寄存器地址 .....	48
4.1.6 DMA 寄存器地址 .....	49
4.1.7 UART 寄存器地址 .....	50
4.1.8 UART32K 寄存器地址 .....	51
4.1.9 ISO7816 寄存器地址 .....	51
4.1.10 TIMER/PWM 寄存器地址 .....	51
4.1.11 LCD 寄存器 .....	53
4.1.12 SPI 寄存器地址 .....	54
4.1.13 I <sup>2</sup> C 寄存器地址 .....	54
4.1.14 MISC 寄存器地址 .....	55
4.1.15 CRYPT 寄存器地址 .....	55
4.1.16 Info 参数寄存器 .....	55
第 5 章 电源系统 .....	62

5.1	寄存器地址 .....	63
5.2	寄存器定义 .....	63
5.2.1	ANA_REG5 寄存器 .....	63
5.2.2	ANA_REG6 寄存器 .....	63
5.2.3	ANA_REG7 寄存器 .....	63
5.2.4	ANA_REG8 寄存器 .....	64
5.2.5	ANA_REG9 寄存器 .....	64
5.2.6	ANA_REGA 寄存器 .....	64
5.2.7	ANA_REGF 寄存器 .....	64
5.2.8	ANA_CTRL 寄存器 .....	64
5.2.9	ANA_CMPOUT 寄存器 .....	65
5.2.10	ANA_INSTS 寄存器 .....	65
5.2.11	ANA_INTEN 寄存器 .....	66
5.3	DVCC .....	67
5.4	电源监测 .....	67
5.4.1	VDCIN 电源检测 .....	67
5.4.2	VDD 电源监测 .....	68
第 6 章	工作模式 .....	69
6.1	简介 .....	69
6.2	特点 .....	69
6.3	功能框图 .....	70
6.4	寄存器地址 .....	70
6.5	寄存器定义 .....	71
6.5.1	PMU_DSLEEPEN 寄存器 .....	71
6.5.2	PMU_DSLEEPPASS 寄存器 .....	71
6.5.3	PMU_CONTROL 寄存器 .....	71
6.5.4	PMU_STS 寄存器 .....	72
6.5.5	ANA_CTRL 寄存器 .....	72
6.6	复位 .....	73
6.6.1	片外复位 .....	74
6.6.2	看门狗复位 .....	75
6.6.3	上电复位 .....	75
6.6.4	M0 软复位 .....	75
6.6.5	从浅睡眠/深睡眠/IDLE 下唤醒 .....	75
6.7	工作模式的转换 .....	78
6.8	应用笔记 .....	80
6.8.1	外部 IO 唤醒 .....	80
6.8.2	进入深睡眠模式的流程 .....	81
6.8.3	进入浅睡眠的流程 .....	82
6.8.4	进入 IDLE 的流程 .....	83
第 7 章	时钟系统 .....	84
7.1	简介 .....	84
7.2	特点 .....	84
7.3	功能框图 .....	85
7.4	寄存器地址 .....	85
7.5	寄存器定义 .....	86
7.5.1	ANA_REG3 寄存器 .....	86
7.5.2	ANA_REG4 寄存器 .....	86
7.5.3	ANA_REG9 寄存器 .....	87
7.5.4	ANA_REGB 寄存器 .....	87
7.5.5	ANA_REGC 寄存器 .....	87
7.5.6	ANA_CMPOUT 寄存器 .....	88
7.5.7	PMU_CONTROL 寄存器 .....	88
7.5.8	PMU_STS 寄存器 .....	88
7.5.9	MISC2_CLKSEL 寄存器 .....	89
7.5.10	MISC2_CLKDIVH 寄存器 .....	89
7.5.11	MISC2_CLKDIVP 寄存器 .....	89
7.5.12	MISC2_HCLKEN 寄存器 .....	90



7.5.13	MISC2_PCLKEN 寄存器 .....	90
第 8 章	模拟控制器 .....	92
8.1	简介 .....	92
8.2	特点 .....	92
8.3	功能框图 .....	92
8.4	寄存器地址 .....	92
8.5	寄存器定义 .....	93
8.5.1	ANA_REGx 寄存器 .....	93
8.5.2	ANA_REG0 寄存器 .....	95
8.5.3	ANA_REG1 寄存器 .....	95
8.5.4	ANA_REG2 寄存器 .....	96
8.5.5	ANA_REG3 寄存器 .....	96
8.5.6	ANA_REG4 寄存器 .....	97
8.5.7	ANA_REG5 寄存器 .....	97
8.5.8	ANA_REG6 寄存器 .....	97
8.5.9	ANA_REG7 寄存器 .....	97
8.5.10	ANA_REG8 寄存器 .....	98
8.5.11	ANA_REG9 寄存器 .....	98
8.5.12	ANA_REGA 寄存器 .....	99
8.5.13	ANA_REGB 寄存器 .....	99
8.5.14	ANA_REGC 寄存器 .....	99
8.5.15	ANA_REGD 寄存器 .....	100
8.5.16	ANA_REGE 寄存器 .....	100
8.5.17	ANA_REGF 寄存器 .....	100
8.5.18	ANA_CTRL 寄存器 .....	100
8.5.19	ANA_CMPOUT 寄存器 .....	102
8.5.20	ANA_INSTS 寄存器 .....	102
8.5.21	ANA_INTEN 寄存器 .....	104
8.5.22	ANA_ADCCTRL 寄存器 .....	105
8.5.23	ANA_ADCDATAx 寄存器 .....	107
8.5.24	ANA_CMPCNTx 寄存器 .....	107
8.5.25	ANA_MISC 寄存器 .....	107
第 9 章	通用 ADC .....	108
9.1	简介 .....	108
9.2	特点 .....	108
9.3	ADC 功能框图 .....	108
9.4	寄存器地址 .....	108
9.5	寄存器定义 .....	109
9.5.1	ANA_REG1 寄存器 .....	109
9.5.2	ANA_REG3 寄存器 .....	109
9.5.3	ANA_INSTS 寄存器 .....	110
9.5.4	ANA_INTEN 寄存器 .....	110
9.5.5	ANA_ADCCTRL 寄存器 .....	110
9.5.6	ANA_ADCDATAx 寄存器 .....	113
9.6	电池电压测量和外部电压测量 .....	113
9.7	温度测量 .....	114
第 10 章	比较器控制器 .....	116
10.1	简介 .....	116
10.2	特点 .....	116
10.3	功能框图 .....	116
10.4	寄存器地址 .....	117
10.5	寄存器定义 .....	117
10.5.1	ANA_REG2 寄存器 .....	117
10.5.2	ANA_REG3 寄存器 .....	117
10.5.3	ANA_REG5 寄存器 .....	118
10.5.4	ANA_CTRL 寄存器 .....	118
10.5.5	ANA_CMPOUT 寄存器 .....	119

10.5.6	ANA_INSTS 寄存器 .....	119
10.5.7	ANA_INTEN 寄存器 .....	119
10.5.8	ANA_CMPCNTx 寄存器 .....	120
第 11 章	TinyADC 控制器 .....	121
11.1	简介 .....	121
11.2	特点 .....	121
11.3	功能框图 .....	121
11.4	寄存器地址 .....	121
11.5	寄存器定义 .....	122
11.5.1	ANA_REGF 寄存器 .....	122
11.5.2	ANA_CMPOUT 寄存器 .....	122
11.5.3	ANA_INSTS 寄存器 .....	122
11.5.4	ANA_INTEN 寄存器 .....	123
11.5.5	ANA_MISC 寄存器 .....	123
第 12 章	看门狗 .....	124
12.1	简介 .....	124
12.2	特点 .....	124
12.3	看门狗在不同模式下的状态 .....	124
12.4	寄存器地址 .....	124
12.5	寄存器定义 .....	124
12.5.1	PMU_WDTPASS 寄存器 .....	124
12.5.2	PMU_WDTEN 寄存器 .....	125
12.5.3	PMU_WDTCLR 寄存器 .....	125
第 13 章	RTC 控制器 .....	126
13.1	简介 .....	126
13.2	特点 .....	126
13.3	功能框图 .....	126
13.4	寄存器读写 .....	126
13.4.1	RTC 寄存器写操作 .....	126
13.4.2	RTC 寄存器读操作 .....	127
13.5	寄存器地址 .....	127
13.6	寄存器定义 .....	128
13.6.1	RTC_SEC/MIN/DAY/WEEK/MONTH/YEAR 寄存器 .....	128
13.6.2	RTC_WKUSEC 寄存器 .....	129
13.6.3	RTC_WKUMIN 寄存器 .....	129
13.6.4	RTC_WKUHOURL 寄存器 .....	129
13.6.5	RTC_WKUCNT 寄存器 .....	130
13.6.6	RTC_CAL 寄存器 .....	130
13.6.7	RTC_DIV 寄存器 .....	130
13.6.8	RTC_CTL 寄存器 .....	131
13.6.9	RTC_PWD 寄存器 .....	131
13.6.10	RTC_CE 寄存器 .....	131
13.6.11	RTC_LOAD 寄存器 .....	132
13.6.12	RTC_INTSTS 寄存器 .....	132
13.6.13	RTC_INTEN 寄存器 .....	132
13.6.14	RTC_PSCA 寄存器 .....	133
13.6.15	RTC_ACTI 寄存器 .....	133
13.6.16	RTC_ACF200 寄存器 .....	133
13.6.17	RTC_ACP0 寄存器 .....	134
13.6.18	RTC_ACP1 寄存器 .....	134
13.6.19	RTC_ACP2 寄存器 .....	134
13.6.20	RTC_ACP3 寄存器 .....	134
13.6.21	RTC_ACP4 寄存器 .....	134
13.6.22	RTC_ACP5 寄存器 .....	135
13.6.23	RTC_ACP6 寄存器 .....	135
13.6.24	RTC_ACP7 寄存器 .....	135
13.6.25	RTC_ACKx 寄存器 .....	135

13.6.26	RTC_ACKTEMP 寄存器 .....	135
13.6.27	RTC_WKUCNTR 寄存器 .....	136
13.7	Info 参数寄存器(与 RTC 温度补偿有关) .....	136
13.8	应用笔记 .....	138
13.8.1	温度校正 .....	138
13.8.2	频率误差校正 .....	138
13.8.3	RTC 手动温度补偿 .....	139
第 14 章	FLASH 控制器 .....	140
14.1	简介 .....	140
14.2	特点 .....	140
14.3	功能框图 .....	140
14.4	寄存器地址 .....	141
14.5	寄存器定义 .....	141
14.5.1	FLASH_PASS 寄存器 .....	141
14.5.2	FLASH_CTRL 寄存器 .....	142
14.5.3	FLASH_PGADDR 寄存器 .....	142
14.5.4	FLASH_PGDATA 寄存器 .....	142
14.5.5	FLASH_PGBx 寄存器 .....	143
14.5.6	FLASH_PGHwX 寄存器 .....	143
14.5.7	FLASH_SERASE 寄存器 .....	143
14.5.8	FLASH_CERASE 寄存器 .....	143
14.5.9	FLASH_DSTB 寄存器 .....	144
14.5.10	FLASH_INT 寄存器 .....	144
14.5.11	FLASH_CSSADDR 寄存器 .....	144
14.5.12	FLASH_CSEADDR 寄存器 .....	144
14.5.13	FLASH_CSVALUE 寄存器 .....	144
14.5.14	FLASH_CSCVALUE 寄存器 .....	145
14.5.15	FLASH_STS 寄存器 .....	145
14.5.16	MISC2_FLASHWC 寄存器 .....	145
第 15 章	GPIO 控制器 .....	146
15.1	简介 .....	146
15.2	特点 .....	146
15.3	功能框图 .....	146
15.4	寄存器地址 .....	146
15.5	寄存器定义 .....	148
15.5.1	PMU_IOAOEN 寄存器 .....	148
15.5.2	PMU_IOAIE 寄存器 .....	148
15.5.3	PMU_IOADAT 寄存器 .....	148
15.5.4	PMU_IOAATT 寄存器 .....	148
15.5.5	PMU_IOAWKUEN 寄存器 .....	149
15.5.6	PMU_IOASTS 寄存器 .....	149
15.5.7	PMU_IOAINTSTS 寄存器 .....	150
15.5.8	PMU_IOASEL 寄存器 .....	150
15.5.9	PMU_IOANODEG 寄存器 .....	150
15.5.10	IOX_OEN 寄存器 .....	150
15.5.11	IOX_IE 寄存器 .....	151
15.5.12	IOX_DAT 寄存器 .....	151
15.5.13	IOX_ATT 寄存器 .....	151
15.5.14	IOX_STS 寄存器 .....	151
15.5.15	IOB_SEL 寄存器 .....	152
15.5.16	IOE_SEL 寄存器 .....	152
15.5.17	IO_MISC 寄存器 .....	152
15.6	特殊功能 IO .....	153
15.6.1	特殊功能 IOA .....	153
15.6.2	特殊功能 IOB .....	153
15.6.3	特殊功能 IOC .....	154
15.6.4	特殊功能 IOD .....	154
15.6.5	特殊功能 IOE .....	155

15.6.6	特殊功能 IOF .....	155
第 16 章	DMA 控制器.....	156
16.1	简介 .....	156
16.2	特点 .....	156
16.3	功能框图.....	156
16.4	寄存器地址 .....	156
16.5	寄存器定义 .....	157
16.5.1	DMA_IE 寄存器 .....	157
16.5.2	DMA_STS 寄存器.....	158
16.5.3	DMA_CxCTL 寄存器 .....	159
16.5.4	DMA_CxSRC 寄存器.....	161
16.5.5	DMA_CxDST 寄存器.....	161
16.5.6	DMA_CxLEN 寄存器 .....	161
16.5.7	DMA_AESCTL 寄存器 .....	161
16.5.8	DMA_AESKEYx 寄存器 .....	162
第 17 章	UART 控制器 .....	163
17.1	简介 .....	163
17.2	特点 .....	163
17.3	功能框图.....	163
17.4	寄存器地址 .....	163
17.5	寄存器定义 .....	165
17.5.1	UARTx_DATA 寄存器 .....	165
17.5.2	UARTx_STATE 寄存器.....	165
17.5.3	UARTx_CTRL 寄存器 .....	165
17.5.4	UARTx_INTSTS 寄存器.....	166
17.5.5	UARTx_BAUDDIV 寄存器 .....	166
17.5.6	UARTx_CTRL2 寄存器 .....	166
17.5.7	MISC_IREN 寄存器 .....	167
17.5.8	MISC_DUTYL 寄存器.....	167
17.5.9	MISC_DUTYH 寄存器 .....	167
第 18 章	UART32K 控制器.....	168
18.1	简介 .....	168
18.2	特点 .....	168
18.3	功能框图.....	168
18.4	寄存器地址 .....	168
18.5	寄存器定义 .....	169
18.5.1	U32Kx_CTRL0 寄存器 .....	169
18.5.2	U32Kx_CTRL1 寄存器 .....	170
18.5.3	U32Kx_PHASE 寄存器.....	170
18.5.4	U32Kx_DATA 寄存器 .....	170
18.5.5	U32Kx_STS 寄存器.....	171
第 19 章	ISO7816 控制器.....	172
19.1	简介 .....	172
19.2	特点 .....	172
19.3	功能框图.....	172
19.4	寄存器地址 .....	172
19.5	寄存器定义 .....	173
19.5.1	ISO7816_BAUDDIVL 寄存器 .....	173
19.5.2	ISO7816_BAUDDIVH 寄存器.....	173
19.5.3	ISO7816_DATA 寄存器.....	173
19.5.4	ISO7816_INFO 寄存器 .....	173
19.5.5	ISO7816_CFG 寄存器 .....	174
19.5.6	ISO7816_CLK 寄存器.....	174
第 20 章	Timer/PWM 控制器.....	176
20.1	简介 .....	176
20.2	特点 .....	176

20.3	功能框图 .....	176
20.4	寄存器地址 .....	177
20.5	寄存器定义 .....	179
20.5.1	TMRx_CTRL 寄存器 .....	179
20.5.2	TMRx_VALUE 寄存器 .....	179
20.5.3	TMRx_RELOAD 寄存器 .....	179
20.5.4	TMRx_INT 寄存器 .....	179
20.5.5	PWMx_CTL 寄存器 .....	180
20.5.6	PWMx_TAR 寄存器 .....	182
20.5.7	PWMx_CCTLx 寄存器 .....	182
20.5.8	PWMx_CCRx 寄存器 .....	184
20.5.9	PWM_O_SEL 寄存器 .....	184
第 21 章	LCD 控制器 .....	186
21.1	简介 .....	186
21.2	特点 .....	186
21.3	功能框图 .....	186
21.4	LCD 时序 .....	187
21.5	LCD 波形电压 .....	187
21.6	LCD 工作电流 .....	187
21.7	LCD 驱动波形 .....	187
21.8	寄存器地址 .....	192
21.9	寄存器定义 .....	193
21.9.1	LCD_FBx 寄存器 .....	193
21.9.2	LCD_CTRL 寄存器 .....	193
21.9.3	LCD_CTRL2 寄存器 .....	194
21.9.4	LCD_SEGCTRL0 寄存器 .....	195
21.9.5	LCD_SEGCTRL1 寄存器 .....	195
21.9.6	LCD_SEGCTRL2 寄存器 .....	196
21.9.7	ANA_REG6 寄存器 .....	196
第 22 章	SPI 控制器 .....	197
22.1	简介 .....	197
22.2	特点 .....	197
22.3	功能框图 .....	197
22.4	寄存器地址 .....	198
22.5	寄存器定义 .....	198
22.5.1	SPI_CTRL 寄存器 .....	198
22.5.2	SPI_TXSTS 寄存器 .....	199
22.5.3	SPI_TXDAT 寄存器 .....	200
22.5.4	SPI_RXSTS 寄存器 .....	200
22.5.5	SPI_RXDAT 寄存器 .....	202
22.5.6	SPI_MISC 寄存器 .....	202
22.6	应用笔记 .....	203
22.6.1	主机模式 .....	203
22.6.2	从机模式 .....	205
22.6.3	连续字节传输 .....	206
第 23 章	I <sup>2</sup> C 控制器 .....	207
23.1	简介 .....	207
23.2	特点 .....	207
23.3	功能框图 .....	207
23.4	寄存器地址 .....	207
23.5	寄存器定义 .....	208
23.5.1	I2C_DATA 寄存器 .....	208
23.5.2	I2C_ADDR 寄存器 .....	208
23.5.3	I2C_CTRL 寄存器 .....	208
23.5.4	I2C_STS 寄存器 .....	209
23.5.5	I2C_CTRL2 寄存器 .....	213
23.6	应用笔记 .....	214

23.6.1	I <sup>2</sup> C 总线协议：启动/停止的发生	214
23.6.2	数据传输格式	214
23.6.3	ACK 信号传输	215
23.6.4	读写操作	215
23.6.5	接口时序	216
23.6.6	I <sup>2</sup> C SDA 管脚死锁的解决方法	217
23.7	工作流程	218
23.7.1	主机发送模式	218
23.7.2	主机接收模式	219
23.7.3	从机发送模式	220
23.7.4	从机接收模式	221
第 24 章	中断控制器	222
24.1	简介	222
24.2	特点	222
24.3	中断源	222
第 25 章	MISC 控制器	225
25.1	简介	225
25.2	特点	225
25.3	功能框图	225
25.4	寄存器地址	225
25.5	寄存器定义	226
25.5.1	MISC_SRAMINT 寄存器	226
25.5.2	MISC_SRAMINIT 寄存器	226
25.5.3	MISC_PARERR 寄存器	227
25.5.4	MISC_IREN 寄存器	227
25.5.5	MISC_DUTYL 寄存器	227
25.5.6	MISC_DUTYH 寄存器	227
25.5.7	MISC_IRQLAT 寄存器	228
25.5.8	MISC_HIADDR 寄存器	228
25.5.9	MISC_PIADDR 寄存器	228
25.5.10	MISC2_FLASHWC 寄存器	228
25.5.11	MISC2_CLKSEL 寄存器	229
25.5.12	MISC2_CLKDIVH 寄存器	229
25.5.13	MISC2_CLKDIVP 寄存器	229
25.5.14	MISC2_HCLKEN 寄存器	230
25.5.15	MISC2_PCLKEN 寄存器	230
第 26 章	CRYPT 控制器	232
26.1	简介	232
26.2	特点	232
26.3	功能框图	232
26.4	寄存器地址	232
26.5	寄存器定义	233
26.5.1	CRYPT_CTRL 寄存器	233
26.5.2	CRYPT_PTRA 寄存器	233
26.5.3	CRYPT_PTRB 寄存器	234
26.5.4	CRYPT_PTRO 寄存器	234
26.5.5	CRYPT_CARRY 寄存器	234
26.6	使用时的注意事项	234
26.6.1	数据格式	234
26.6.2	运算细节	235
第 27 章	调试特点	236
27.1	特点	236
27.2	SWD 引脚	236
第 28 章	Cortex-M0 Core 简要描述	237
28.1	CMSIS 函数说明	237
第 29 章	电能计量 EM	238



29.1	电能计量功能框图 .....	239
29.2	寄存器列表 .....	239
29.2.1	寄存器总列表 .....	239
29.2.2	模拟控制寄存器 .....	243
29.2.3	系统配置和状态寄存器 .....	245
29.2.4	计量控制寄存器 .....	255
29.2.5	计量数据寄存器 .....	265
29.2.6	能量桶寄存器 .....	266
29.2.7	相位测量寄存器 .....	268
29.2.8	功率潜动阈值寄存器 .....	268
29.2.9	电压骤升骤降寄存器 .....	268
29.2.10	快速检测寄存器 .....	269
29.2.11	波形数据寄存器 .....	269
29.2.12	校表参数寄存器 .....	269
29.2.13	校验和寄存器 .....	271
29.3	电能计量模块电压基准电路（简称 EMBGP） .....	273
29.4	电能计量时钟 .....	274
29.4.1	EM_PCLKDIV 时钟 .....	275
29.4.2	EM_RCH 时钟 .....	275
29.4.3	外部输入低频时钟 .....	275
29.5	模拟信号输入 .....	275
29.6	模拟数字转换 .....	277
29.7	角差校正 .....	278
29.8	原始波形数据产生 .....	279
29.9	有效值计算和校正 .....	280
29.10	有功功率计算和校正 .....	280
29.11	无功功率计算和校正 .....	281
29.12	视在功率计算 .....	281
29.13	功率启动潜动判断 .....	282
29.14	线电压频率测量 .....	283
29.15	相位测量 .....	284
29.16	波形输出和缓存 .....	285
29.16.1	波形输出 .....	287
29.16.2	波形缓存 .....	288
29.17	电信号监测 .....	288
29.17.1	过零点检测 .....	288
29.17.2	电压骤升骤降 .....	289
29.17.3	过压欠压和过流欠流 .....	290
29.18	能量桶 .....	293
29.18.1	高速能量桶 .....	296
29.18.2	低速能量桶 .....	296
29.18.3	能量桶启动潜动判断 .....	296
29.19	CF 输出 .....	297
29.20	信号输出口 .....	298
第 30 章	封装尺寸图 .....	301
30.1	V9400(A)封装尺寸图 .....	301
30.2	V9430 封装尺寸图 .....	302
30.3	V9431(A)封装尺寸图 .....	303
30.4	V9410(A)封装尺寸图 .....	304
30.5	V9420A 封装尺寸图 .....	305
图索引	.....	306
表索引	.....	309
版本更新说明	.....	319





# 第1章 电气特性

## 1.1 绝对最大额定值

除非特别说明，数据都是基于  $T_A=25^{\circ}\text{C}$ ， $V_{DD}=3.3\text{V}$  的测试结果。

超出下列最大值或低于下列最小值的工作条件可能会造成芯片的永久性损伤。长时间工作在最大值条件下可能会影响芯片的可靠性。

表 1-1 绝对最大额定值

符号	参数	最小	最大	单位
$V_{VDD}-V_{SS}$	外部电源电压	-0.3	+3.63	V
$V_{IN}-V_{SS}$	输入到 GPIO 引脚的外部信号	-0.3	+3.63	V
电流采样通道输入电压	IAP/IAN/IBP/IBN	-0.3	+3.3	V
电压采样通道输入电压	UP/UN	-0.3	3.3	V
$S_{VDD}$	IO 上电斜率	3.3V/s	1V/ $\mu\text{s}$	--
$I_{INJ\_PAD}$	单引脚输入电流	-10	+10	mA
$I_{INJ\_SUM}$	所有引脚输入电流的总和	-50	+40	mA
$T_W$	工作温度范围	-40	+85	$^{\circ}\text{C}$
$T_S$	储存温度范围	-55	+150	$^{\circ}\text{C}$
$T_J$	PN 结结温	-40	+125	$^{\circ}\text{C}$

## 1.2 正常工作电压

除非特别说明，数据都是基于  $T_A=25^{\circ}\text{C}$ ， $V_{DD}=3.3\text{V}$  的测试结果。

表 1-2 供电电压

符号	参数	条件（全温度范围）	最小	典型	最大	单位
$V_{IN, VDD}$	$V_{DD}$ 的输入电压*[1]		2.2	3.3	3.6	V

\*[1]: ADC 部分和计量部分支持的工作电压范围: 2.6V~3.6V。

## 1.3 驱动特性

除非特别说明，数据都是基于  $T_A=25^{\circ}\text{C}$ ， $V_{DD}=3.3\text{V}$  的测试结果。

表 1-3 电源管脚的驱动条件

符号	参数	条件（全温度范围）	最小	典型	最大	单位
$I_{DRV, DVCC}$	DVCC 的驱动能力				35	mA
$I_{DRV, VDDOUT}$	VDD_OUT 的驱动能力				20	mA

## 1.4 功耗

除非特别说明，数据都是基于  $T_A=25^{\circ}\text{C}$ ， $V_{DD}=3.3\text{V}$  的测试结果。

表 1-4 常用工作条件



符号	参数	条件（常温下测量，VDD=3.3V（特殊说明除外））	最小	典型	最大	单位
I <sub>ACTIVE</sub>	工作电流	系统时钟：26.2144MHz		5082		μA
		系统时钟：13.1072MHz		2792		
		系统时钟：6.5536MHz		1633		
		系统时钟：3.2768MHz		1016		
		系统时钟：1.6384MHz		713		
		系统时钟：819.2kHz		548		
		系统时钟：409.6kHz		461		
		系统时钟：204.8kHz		418		
		系统时钟：32K RC (FLASH deep-standby)		10.0		
		系统时钟：32K RC (FLASH 正常)		75.0		
I <sub>IDLE</sub>	IDLE 电流	系统时钟：26.2144MHz		1114		μA
		系统时钟：13.1072MHz		625		
		系统时钟：6.5536MHz		379		
		系统时钟：3.2768MHz		256		
		系统时钟：1.6384MHz		195		
		系统时钟：819.2kHz		164		
		系统时钟：409.6kHz		149		
		系统时钟：204.8kHz		141		
		系统时钟：32K RC		5.1		
I <sub>SLP1</sub>	LCD 关闭时的浅睡眠模式电流	RTCCLK 不分频		3.1 6.3@80 °C		μA
		RTCCLK 4 分频		2.8		
I <sub>SLP2</sub>	LCD 开启时的浅睡眠模式电流	RTCCLK 不分频		9.3		μA
		RTCCLK 4 分频		9.0		
I <sub>DSLP</sub>	深睡眠模式电流，VDD=3.3V	RTCCLK 不分频		2.9		μA
		RTCCLK 4 分频		2.6		

表 1-5 各个模块的功耗

模块	电流 (μA) (常温下测量)	状态
CMP A/B	0.08	IT_CMP=00 (偏置电流=20nA)
	0.4	IT_CMP=01 (偏置电流=100nA)
	2	IT_CMP=1* (偏置电流=500nA)
ADCBGP	108	
ADC	230	
TinyADC	0.7	
Tempsensor	370	
LCD	5	LCD 驱动电阻=600k
	9.9	LCD 驱动电阻=300k

	14.5	LCD 驱动电阻=200k
	19	LCD 驱动电阻=150k
32768 XTAL	0.6	
32K RC	0.2	
6.5M RC	80	
6.5M XTAL	120	
PLL_L	20	
PLL_H	40	
EM	2600	IA、IB 和 U 通道同时工作。

## 1.5 复位和电源控制模块特性

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-6 电源和复位控制模块特性

符号	参数	条件（常温下测量）	最小	典型	最大	单位
t <sub>RST</sub>	复位消抖时间			20		μs
V <sub>LPREF</sub>	参考电压 V <sub>LPREF</sub>		1.222	1.3	1.352	V
V <sub>PORH</sub>	PORH 监测电压（VDD）		1.955	2.08	2.163	V
V <sub>PORL</sub>	PORL 监测电压（DVCCLD01）		1.222	1.3	1.352	V
V <sub>VDCIN</sub>	V <sub>VDCIN</sub> 监测电平		1.222	1.3	1.352	V
V <sub>VDDALRAM</sub>	V <sub>DD</sub> 电源监测电压	V <sub>TH</sub> 可配置，V <sub>TH</sub> =2.9V； 参考电压为 V <sub>LPREF</sub>	2.726	2.9	3.016	V
LCDLDO	LCDLDO 电压		3.2	3.3	3.4	V
V <sub>LCDLDO_Drop</sub>	LCDLDO 压降				20	mV

表 1-7 电压回差特性

符号	参数	条件（常温下测量）	最小	典型	最大	单位
V <sub>PORH_HTRIS</sub>	PORH 监测电压回差		53.2	66.5	79.8	mV
V <sub>PORL_HTRIS</sub>	PORL 监测电压回差		33.6	42.0	50.4	mV
V <sub>VDCIN_HTRIS</sub>	V <sub>VDCIN</sub> 监测电压回差		33.6	42.0	50.4	mV
V <sub>VDDALRAM_HTRIS</sub>	V <sub>DD</sub> 电源监测电压回差	V <sub>TH_VDDALARM</sub> = 3.6 V	91.2	114.0	136.8	mV
		V <sub>TH_VDDALARM</sub> = 3.2 V	81.1	101.4	121.6	mV
		V <sub>TH_VDDALARM</sub> = 2.9 V	73.5	91.9	110.2	mV
		V <sub>TH_VDDALARM</sub> = 2.6 V	65.9	82.4	98.8	mV
		V <sub>TH_VDDALARM</sub> = 2.3 V	58.3	72.9	87.4	mV

## 1.6 GPIO 特性

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-8 GPIO 特性



符号	参数	V <sub>VDD</sub>	条件（全温度范围）	最小	最大	单位
V <sub>IH</sub>	输入高电平	3.3V		2		V
V <sub>IL</sub>	输入低电平	3.3V			0.3*V <sub>VDD</sub>	V
V <sub>HSYS</sub>	施密特触发迟滞	3.3V		0.1*V <sub>VDD</sub>		V
I <sub>IH</sub>	输入高电平时电流	3.3V			+1	μA
I <sub>IL</sub>	输入低电平时电流	3.3V		-1		μA
V <sub>OH</sub>	输出高电平	3.3V	驱动电流 5.6mA	2.4	V <sub>VDD</sub>	V
V <sub>OL</sub>	输出低电平	3.3V	驱动电流 5.6mA		0.4	V
C <sub>IN</sub>	输入电容	3.3V			10	pF

## 1.7 通用 ADC 特性

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-9 通用 ADC 特性

符号	参数	条件（全温度范围）	最小	典型	最大	单位
ADCREF	ADCBGP 电压		1.207	1.225	1.243	V
PSRR	ADCBGP 的电源抑制比			-92		dB
V <sub>ADC</sub>	通用 ADC 工作电压		2.7	3.3	3.6	V
I <sub>ADC</sub>	通用 ADC 工作电流		170	230	350	μA
f <sub>ADCLK</sub>	通用 ADC 采样时钟			1.6384		MHz
C <sub>ADC</sub>	内部采样和保持电容			1		pF
INL	非线性集成			2		LSB
DNL	非线性差分			1		LSB
Offset	偏移误差			5		mV
V <sub>WITHSTAND</sub>	管脚耐压值（通用 ADC 输入电压范围）		-0.7		V <sub>DD</sub>	V

## 1.8 模拟比较器特性

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-10 比较器特性

符号	参数	条件（常温）	最小	典型	最大	单位
V <sub>CMP</sub>	比较器工作电压 (VDD)		2.2	3.3	3.6	V
I <sub>CMP</sub>	比较器工作电流	输入偏置电流 20nA，输入 50kHz 方波		0.08		μA
		输入偏置电流 100nA，输入 50kHz 方波		0.4		μA
		输入偏置电流 500nA，输入 50kHz 方波		2		μA
t <sub>d</sub>	传播迟滞	输入偏置电流 20nA，输入 50kHz 方波		1.6		μs

		输入偏置电流 100nA 输入 50kHz 方波		0.63		$\mu\text{s}$
		输入偏置电流 500nA 输入 50kHz 方波		0.27		$\mu\text{s}$
V <sub>CMPIN</sub>	比较器输入电压范围		0.8		V <sub>DD</sub> -0.3	V
V <sub>CMPREF</sub>	比较器参考电压	参考电压为 LPREF	1.222	1.3	1.352	V
		参考电压为 ADCREF	1.18	1.2	1.22	V
V <sub>HTRES</sub>	比较器回差		20.0	25.0	30.0	mV

## 1.9 时钟和锁相环特性

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-11 时钟和锁相环特性

符号	参数	条件（全温度范围）	最小	典型	最大	单位
V <sub>DDPLL</sub>	PLL 工作电压（DVCC）		1.35	1.5	1.65	V
I <sub>VDDPLL</sub>	PLL 工作电流			30		$\mu\text{A}$
V <sub>DDPLLH</sub>	PLLH 工作电压（DVCC）		1.35	1.5	1.65	V
I <sub>VDDPLLH</sub>	PLLH 工作电流			40		$\mu\text{A}$
V <sub>DDRCL</sub>	RCL 工作电压（VDD）		2.2	3.3	3.6	V
I <sub>VDDRCL</sub>	RCL 工作电流			0.2		$\mu\text{A}$
f <sub>RCL</sub>	RCL 频率		29.7	32	35.5	kHz
V <sub>DDRCH</sub>	RCH 工作电压（VDD）		2.2	3.3	3.6	V
I <sub>VDDRCH</sub>	RCH 工作电流			45		$\mu\text{A}$
f <sub>RCH</sub>	RCH 频率		6.357	6.5	6.75	MHz
V <sub>DDXOH</sub>	XOH 工作电压（VDD）		2.2	3.3	3.6	V
I <sub>VDDXOH</sub>	XOH 工作电流			150		$\mu\text{A}$
f <sub>XOH</sub>	XOH 频率			6.5536		MHz

## 1.10 FLASH 和 SRAM 特性

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-12 FLASH 和 SRAM 特性

参数	条件	最小	典型	最大	单位
FLASH 读一个字的时间		38			ns
FLASH 擦写次数	-40~85℃	20000			time
FLASH 数据保留时间	-40~85℃	20			year
FLASH 字节编程时间		6		7.5	$\mu\text{s}$
FLASH 扇区擦除时间（一个扇区有 512 字节）		4		5	ms

FLASH 芯片全擦除时间		30		40	ms
FLASH 读操作电流	26 MHz 访问.		2.5	3.5	mA
FLASH 编程操作电流				3.5	mA
FLASH 擦除操作电流				2	mA
FLASH standby 电流			80	150	$\mu$ A
FLASH deep standby 电流			0.1	6	$\mu$ A
FLASH 工作电压 (DVCC)		1.35	1.5	1.65	V
SRAM 数据保持电压 (DVCC)		1.35	1.5	1.65	V
SRAM 工作温度 (DVCC)		-40	+25	+85	$^{\circ}$ C

## 1.11 晶振的 ESR 特性

除非特别说明，数据都是基于  $T_A=25^{\circ}\text{C}$ ， $V_{DD}=3.3\text{V}$  的测试结果。

表 1-13 晶振的 ESR 特性

参数	条件 (全温度范围)	最小	典型	最大	单位
6.5536M 晶振的 ESR*				100	$\Omega$
32768K 晶振的 ESR				50	K $\Omega$

\*: ESR(Equivalent series resistance): 等效串联电阻

## 1.12 电能计量相关特性

除非特别说明，数据都是基于  $T_A=25^{\circ}\text{C}$ ， $V_{DD}=3.3\text{V}$  的测试结果。

表 1-14 电能计量相关特性

参数	最小	典型	最大	单位	说明
通道间相差					
PF=0.8 容性		$\pm 0.05$		度	
PF=0.5 感性		$\pm 0.05$		度	
有功功率计量误差		0.1		%	动态范围 5000:1 @ $25^{\circ}\text{C}$ 基波频偏 $\pm 25\%$ 以内
有功功率计量带宽	0.4	3.2	6.4	kHz	
无功功率计量误差		0.2		%	动态范围 5000:1 @ $25^{\circ}\text{C}$ 基波频偏 $\pm 25\%$ 以内
无功功率计量带宽	0.4	3.2	6.4	kHz	
秒平均电压有效值误差		0.5		%	动态范围 5000:1 @ $25^{\circ}\text{C}$ 基波频偏 $\pm 25\%$ 以内
电压有效值带宽	0.4	3.2	6.4	kHz	
秒平均电流有效值误差		0.5		%	动态范围 5000:1 @ $25^{\circ}\text{C}$ 基波频偏 $\pm 25\%$ 以内
电流有效值带宽	0.4	3.2	6.4	kHz	

参数	最小	典型	最大	单位	说明
频率测量					
范围	40		70	Hz	
误差		0.01		Hz	
模拟输入信号					
单端输入信号			±200	mV	峰值
ADC 性能					
直流失调			10	mV	
解析度		23		Bit	包括符号位
信号带宽（-3dB）	0.4	3.2	6.4	KHz	可编程
片上参考电压 REF					
偏差	-18		18	mV	@25°C
输出电压		1.21		V	
电源抑制比		92		dB	
温度系数		10	30	ppm/°C	
计量电源输入（VDD2）	2.6	3.3	3.6	V	
数字电源输出（DVCC2）					
电压		1.5		V	
电流			35	mA	
时钟					
计量模块高频时钟 （EM_PCLKDIV）		6.5536		MHz	
计量模块内部高频时钟 （EM_RCH）	-20%	6.5536	+20%	MHz	
计量模块能量桶低频时钟 （EM_X32KIN）		32.768		KHz	

## 1.13 时钟输出稳定时间及唤醒时间

除非特别说明，数据都是基于 TA=25°C，VDD=3.3V 的测试结果。

表 1-15 时钟输出稳定时间及唤醒时间

参数	条件（全温度范围）	最小	典型	最大	单位
----	-----------	----	----	----	----

PLLL 锁定时间				1	ms
PLLH 锁定时间				15	μs
RCL 输出稳定时间				200	μs
RCH 输出稳定时间				5	μs
ADCBGP 输出稳定时间				10	μs
RCH 为系统时钟时的浅睡眠唤醒时间			18.4		μs
PLLL 为系统时钟时的浅睡眠唤醒时间			1.03		ms
PLLH 为系统时钟时的浅睡眠唤醒时间			22.8		μs
RCH 为系统时钟时的 IDLE 中断响应时间			6		μs
PLLL 为系统时钟时的 IDLE 中断响应时间			1.6		μs
PLLH 为系统时钟时的 IDLE 中断响应时间			1.6		μs

## 1.14 通用 ADC 转换时间

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

ADC 时钟频率、CIC 滤波器降采样率、CIC 滤波器忽略采样点数的配置不同，MADC 转换时间不同。详情请参考下表。

表 1-16 ADC 转换时间（CIC 滤波器降采样率与 ADC 时钟频率）

ADC 时钟频率/MHz CIC 滤波器降采样率 ADC 典型转换时间/ms	6.5536	3.2768	1.6384	0.8192	0.4096
1/512	0.937	0.937	1.875	3.750	7.500
1/256	0.468	0.468	0.937	1.875	3.750
1/128	0.234	0.234	0.468	0.937	1.875
1/64	0.117	0.117	0.234	0.468	0.937

其他参数配置：CICSKIP=6，CIC 滤波器忽略前 2 个采样点。

注 1：ADC 为 3 阶 CIC，忽略前两个采样点，数据稳定。

注 2：ADC 转换时间 =  $\frac{1}{\left(\text{CIC降采样率} * \frac{\text{ADC时钟频率}}{2}\right)} * \text{采样点}$ （ADC 时钟频率为 6.5536MHz 时除外），如：CIC

降采样率为 1/512，ADC 时钟频率为 3.2768MHz，忽略前 2 个采样点，

ADC 转换时间 =  $\frac{1}{\left(\frac{1}{512} * \frac{3.2768\text{MHz}}{2}\right)} * 3 = 0.937\text{ms}$

表 1-17 ADC 转换时间（CIC 滤波器忽略采样点数）

参数	条件(忽略采样点数)	典型	单位
ADC 转换时间	2	1.875	ms
	3	2.520	ms
	4	3.150	ms
	5	3.780	ms
	6	4.410	ms



	7	5.040	ms
其他参数配置：降采样率为 1/512，ADC 时钟频率为 1.6384MHz。			

## 1.15 TinyADC 转换时间

除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-18 TinyADC 转换时间

参数	条件	最小	典型	最大	单位
TinyADC 转换时间			40		μs

## 1.16 芯片可靠性分析结果

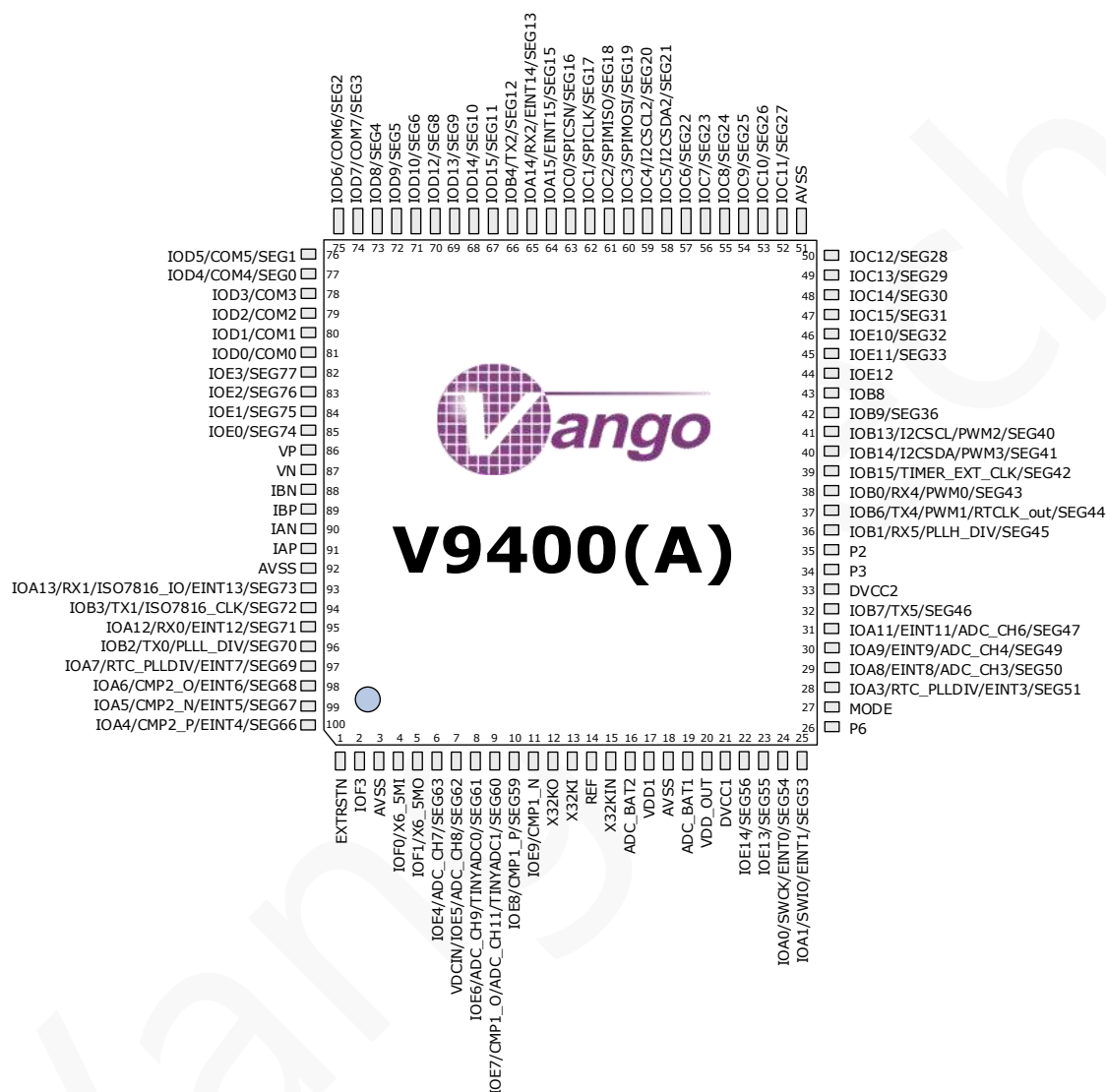
除非特别说明，数据都是基于 TA=25℃，VDD=3.3V 的测试结果。

表 1-19 芯片可靠性分析结果

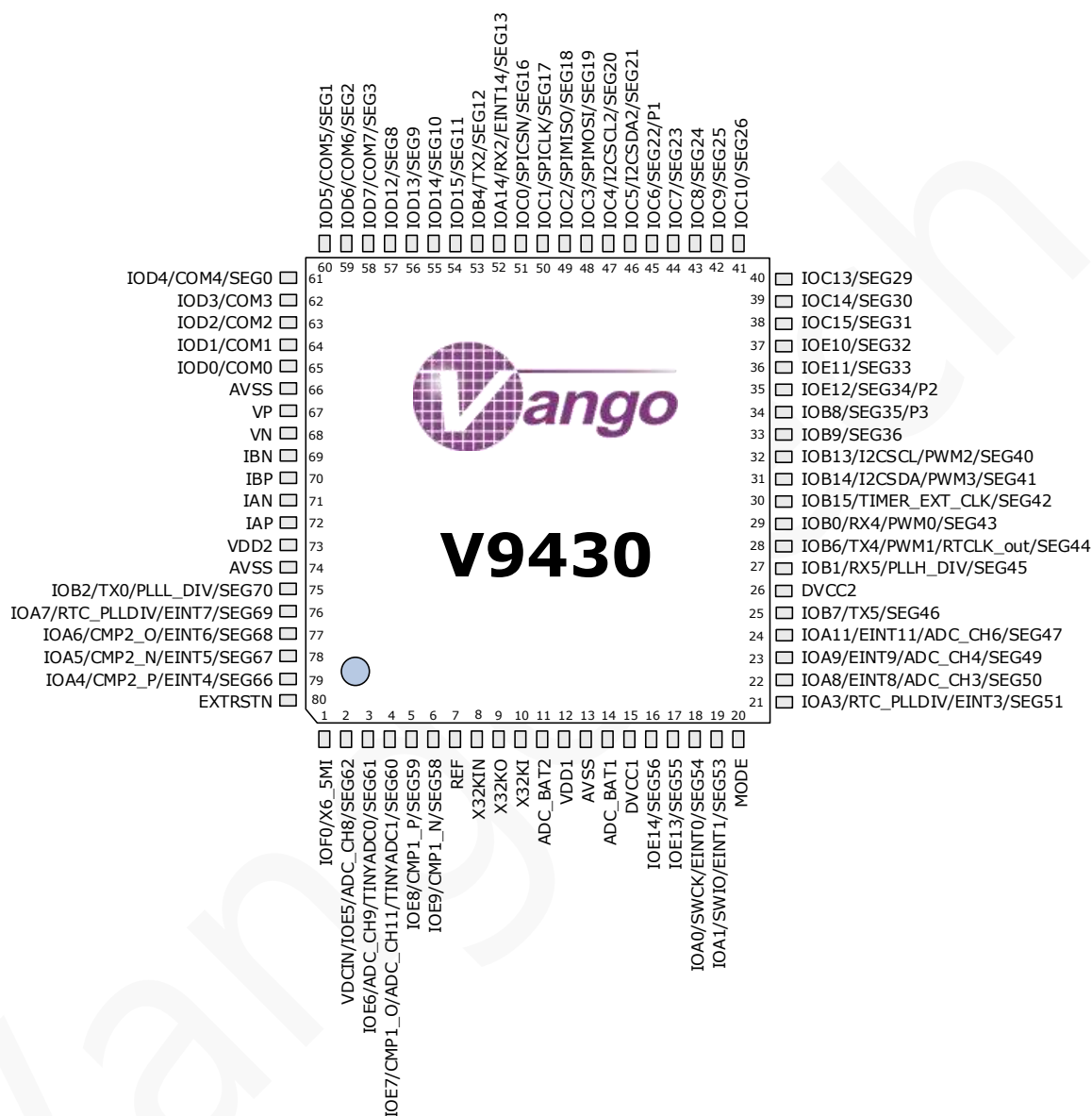
项目	规范	最小值	最大值
HBM	Mil-Std-883J Method 3015.9	-4KV	+4KV
MM	EDEC EIA/JESD22-A115	-300V	+300V
LATCH-UP	JEDEC EIA/JESD78E	-200mA	+200mA

## 第2章 引脚分布图

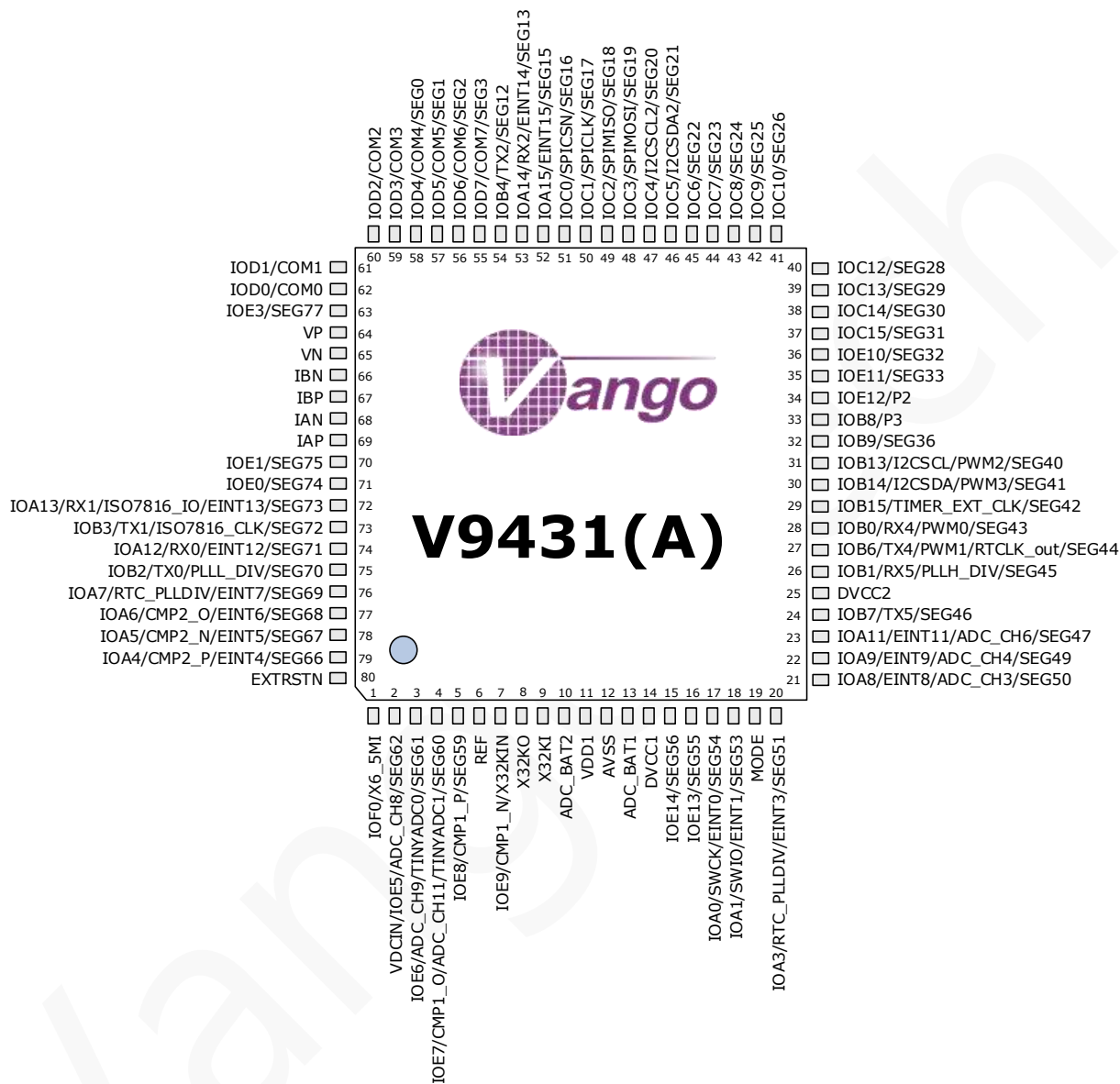
### 2.1 V9400(A)引脚分布图



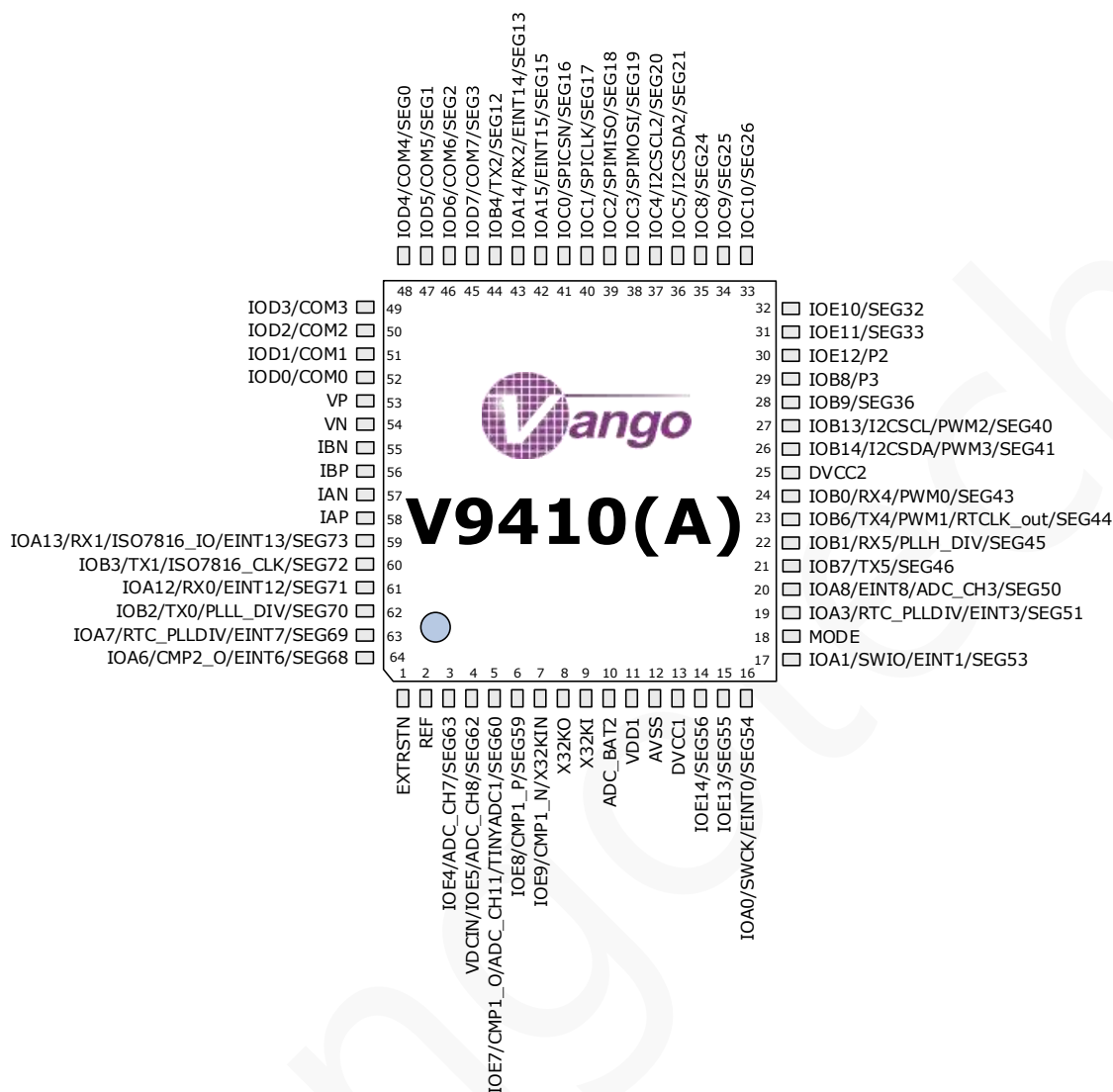
## 2.2 V9430 引脚分布图



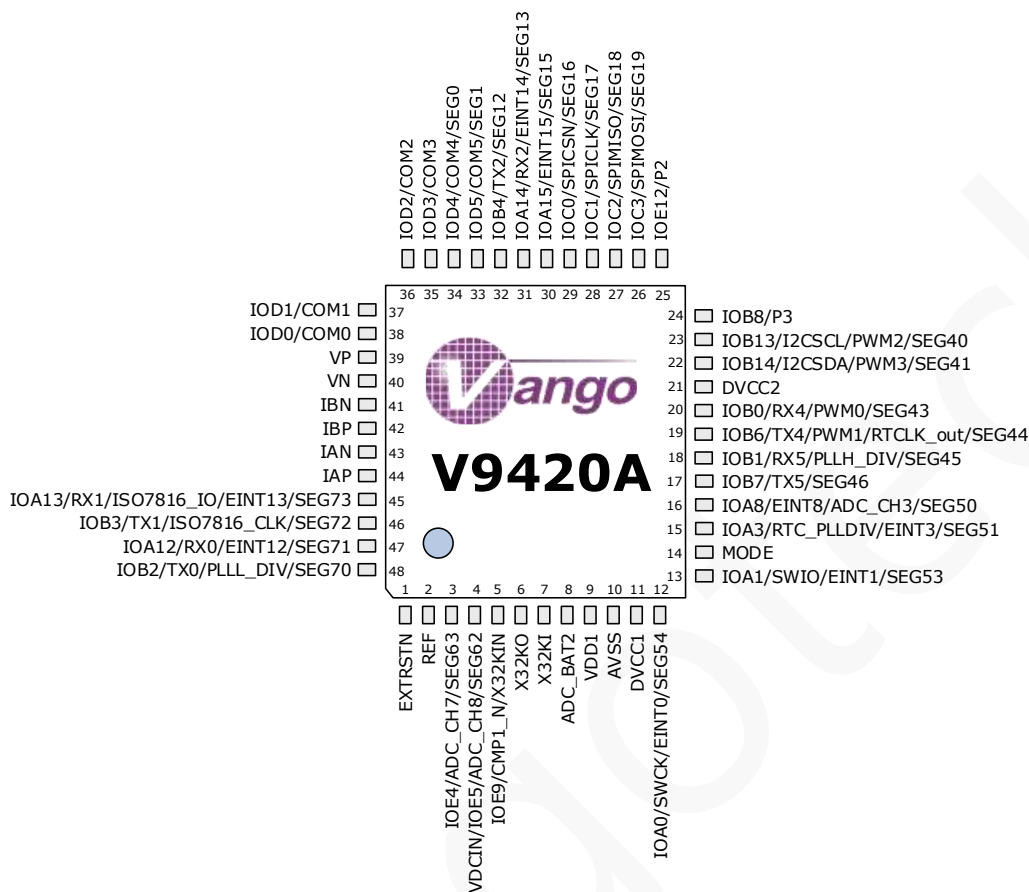
## 2.3 V9431(A)引脚分布图



## 2.4 V9410(A)引脚分布图



## 2.5 V9420A 引脚分布图



2.6 V94XX(A)管脚描述

表 2-1 V94XX(A)管脚描述

芯片型号						引脚名称	类型	描述
V94XX(A)	V9400(A)	V9430	V9431(A)	V9410(A)	V9420A			
1	4	1	1			IOF0	I/O	默认: IOF0 功能 1: 6.5536M 晶体输入
2	5					IOF1	I/O	默认: IOF1 功能 1: 6.5536M 晶体输出
3	6			3	3	IOE4	I/O	默认: IOE4 功能 1: ADC_CH7 输入 功能 2: SEG63
4	7	2	2	4	4	IOE5	I/O	默认: VDCIN 输入 功能 1: ADC_CH8 输入 功能 2: IOE5 功能 3: SEG62
5	8	3	3			IOE6	I/O	默认: IOE6 功能 1: ADC_CH9 输入和 tinyADC 通道 0 输入 功能 2: SEG61
6	9	4	4	5		IOE7	I/O	默认: IOE7 功能 1: 比较器 1 输出 功能 2: ADC_CH11 输入和 tinyADC 通道 1 输入 功能 3: SEG60
7	10	5	5	6		IOE8	I/O	默认: IOE8 功能 1: 比较器 1 P 输入 功能 2: SEG59



8	11	6	7	7	5	IOE9	I/O	默认: IOE9 功能 1: 比较器 1 N 输入 V9431(A)\ V9410(A)\ V9420A 的 IOE9 和 X32KIN 共用一个管脚。
9	14	7	6	2	2	REF	I/O	REF, 计量模块片上基准电压输出, 用户应该在此引脚连接 1uF 电容。
10	15	8	7	7	5	X32KIN	I	X32KIN, 计量模块能量桶的低速时钟输入。 V9431(A)\ V9410(A)\ V9420A 的 IOE9 和 X32KIN 共用一个管脚。
11	12	9	8	8	6	X32KO	O	32768 Hz 晶体输出引脚, 内部匹配电容为 12 pF, 无需外接电容。
12	13	10	9	9	7	X32KI	I	32768 Hz 晶体输入引脚, 内部匹配电容为 12 pF, 无需外接电容。
13	16	11	10	10	8	ADC_BAT2	I	默认: ADC 通道 2 的输入, 用于电池电压测量。
14	17	12	11	11	9	VDD1	P	主电源输入引脚。用户应该在此引脚连接一个 0.1uF 的去耦电容。
15	18	13	12	12	10	AVSS	G	模拟地
16	19	14	13			ADC_BAT1	I	默认: ADC 通道 1 的输入, 用于电池电压测量。
17	20					VDD_OUT	O	默认: 高阻 功能: VDD 电压输出引脚, 该电平与 VDD 管腿保持一致。用户可以用来驱动小功率模块。
18	21	15	14	13	11	DVCC1	O/P	内部数字电源输出引脚, 用户应该在此引脚连接一个 0.1uF 和一个 10uF 的去耦电容。
19	22	16	15	14		IOE14	I/O	默认: IOE14 功能 2: SEG56
20	23	17	16	15		IOE13	I/O	默认: IOE13 功能 2: SEG55
21	24	18	17	16	12	IOA0	I/O	默认: IOA0(MODE=1), SWCK(MODE=0) 功能 1: EINT0 功能 2: SEG54



22	25	19	18	17	13	IOA1	I/O	默认: IOA1(MODE=1), SWIO(MODE=0) 功能 1: EINT1 功能 2: SEG53
23	27	20	19	18	14	MODE	I	调试模式或正常模式选择。 0: 调试模式。 1: 正常模式。 这个引脚应该使用 VDD 上拉, 正常或调试操作过程中该 IO 的状态不应该改变。
24	28	21	20	19	15	IOA3	I/O	默认: IOA3 功能 1: 秒脉冲输出(RTC_PLLDIV 输出) 功能 2: EINT3 功能 3: SEG51
25	26	21				P6	I/O	默认: P6, CF1/CF2/单独中断/全部中断/波形输出
26	29	22	21	20	16	IOA8	I/O	默认: IOA8 功能 1: EINT8 功能 2: ADC_CH3 输入 功能 3: SEG50
27	30	23	22			IOA9	I/O	默认: IOA9 功能 1: EINT9 功能 2: ADC_CH4 输入 功能 3: SEG49
28	31	24	23			IOA11	I/O	默认: IOA11 功能 1: EINT11 功能 2: ADC_CH6 输入 功能 3: SEG47
29	32	25	24	21	17	IOB7	I/O	默认: IOB7

								功能 1: UARTTXD5 功能 2: SEG46
30	33	26	25	25	21	DVCC2	O/P	内部数字电源输出引脚，用户应该在此引脚连接一个 0.1uF 和一个 10uF 的去耦电容。
31	36	27	26	22	18	IOB1	I/O	默认: IOB1 功能 1: UARTRXD5 功能 3: PLLH 经过分频器输出 功能 4: SEG45
32	37	28	27	23	19	IOB6	I/O	默认: IOB6 功能 1: UARTTXD4 功能 2: PWM1 输出 功能 3: RTCCLK 输出 功能 4: SEG44
33	38	29	28	24	20	IOB0	I/O	默认: IOB0 功能 1: UARTRXD4 功能 2: PWM0 输出 功能 3: SEG43
34	39	30	29			IOB15	I/O	默认: IOB15 功能 1: 定时器外部时钟输入。 功能 2: SEG42
35	40	31	30	26	22	IOB14	I/O	默认: IOB14 功能 1: I2CSDA 功能 2: PWM3 输出 功能 3: SEG41
36	41	32	31	27	23	IOB13	I/O	默认: IOB13 功能 1: I2CSCL

								功能 2: PWM2 输出 功能 3: SEG40
37	42	33	32	28		IOB9	I/O	默认: IOB9 功能 1: SEG36
38	43	34	33	29	24	IOB8	I/O	默认: IOB8 V9430/V9431(A)/V9410(A)/V9420A 的 IOB8 和 P3 共用一个管脚
39	34	34	33	29	24	P3	I/O	默认: P3, CF1/CF2/单独中断/全部中断/波形输出 V9430/V9431(A)/V9410(A)/V9420A 的 IOB8 和 P3 共用一个管脚
40	44	35	34	30	25	IOE12	I/O	默认: IOE12 V9430/V9431(A)/V9410(A)/V9420A 的 IOE12 和 P2 共用一个管脚
41	35	35	34	30	25	P2	I/O	默认: P2, CF1/CF2/单独中断/全部中断/波形输出 V9430/V9431(A)/V9410(A)/V9420A 的 IOE12 和 P2 共用一个管脚
42	45	36	35	31		IOE11	I/O	默认: IOE11 功能 1: SEG33
43	46	37	36	32		IOE10	I/O	默认: IOE10 功能 1: SEG32
44	47	38	37			IOC15	I/O	默认: IOC15 功能 1: SEG31
45	48	39	38			IOC14	I/O	默认: IOC14 功能 1: SEG30
46	49	40	39			IOC13	I/O	默认: IOC13 功能 1: SEG29
47	50		40			IOC12	I/O	默认: IOC12 功能 1: SEG28

48	51					AVSS	G	模拟地
49	52					IOC11	I/O	默认: IOC11 功能 1: SEG27
50	53	41	41	33		IOC10	I/O	默认: IOC10 功能 1: SEG26
51	54	42	42	34		IOC9	I/O	默认: IOC9 功能 1: SEG25
52	55	43	43	35		IOC8	I/O	默认: IOC8 功能 1: SEG24
53	56	44	44			IOC7	I/O	默认: IOC7 功能 1: SEG23
54	57	45	45			IOC6	I/O	默认: IOC6 功能 1: SEG22
55	58	46	46	36		IOC5	I/O	默认: IOC5 功能 1: I2CSDA(2) 功能 2: SEG21
56	59	47	47	37		IOC4	I/O	默认: IOC4 功能 1: I2CSCL(2) 功能 2: SEG20
57	60	48	48	38	26	IOC3	I/O	默认: IOC3 功能 1: SPIMOSI 功能 2: SEG19
58	61	49	49	39	27	IOC2	I/O	默认: IOC2 功能 1: SPIMISO 功能 2: SEG18
59	62	50	50	40	28	IOC1	I/O	默认: IOC1

								功能 1: SPICLK 功能 2: SEG17
60	63	51	51	41	29	IOC0	I/O	默认: IOC0 功能 1: SPICSN 功能 2: SEG16
61	64		52	42	30	IOA15	I/O	默认: IOA15 功能 1: 保留 功能 2: 保留 功能 3: EINT15 功能 4: SEG15
62	65	52	53	43	31	IOA14	I/O	默认: IOA14 功能 1: UARTRXD2 功能 2: EINT14 功能 3: SEG13
63	66	53	54	44	32	IOB4	I/O	默认: IOB4 功能 1: UARTTXD2 功能 2: SEG12
64	67	54				IOD15	I/O	默认: IOD15 功能 1: SEG11
65	68	55				IOD14	I/O	默认: IOD14 功能 1: SEG10
66	69	56				IOD13	I/O	默认: IOD13 功能 1: SEG9
67	70	57				IOD12	I/O	默认: IOD12 功能 1: SEG8
68	71					IOD10	I/O	默认: IOD10

								功能 1: SEG6
69	72					IOD9	I/O	默认: IOD9 功能 1: SEG5
70	73					IOD8	I/O	默认: IOD8 功能 1: SEG4
71	74	58	55	45		IOD7	I/O	默认: IOD7 功能 1: COM7/SEG3
72	75	59	56	46		IOD6	I/O	默认: IOD6 功能 1: COM6/SEG2
73	76	60	57	47	33	IOD5	I/O	默认: IOD5 功能 1: COM5/SEG1
74	77	61	58	48	34	IOD4	I/O	默认: IOD4 功能 1: COM4/SEG0
75	78	62	59	49	35	IOD3	I/O	默认: IOD3 功能 1: COM3
76	79	63	60	50	36	IOD2	I/O	默认: IOD2 功能 1: COM2
77	80	64	61	51	37	IOD1	I/O	默认: IOD1 功能 1: COM1
78	81	65	62	52	38	IOD0	I/O	默认: IOD0 功能 1: COM0
79	82		63			IOE3	I/O	默认: IOE3 功能 1: SEG77
80	83					IOE2	I/O	默认: IOE2 功能 1: SEG76
81	84		70			IOE1	I/O	默认: IOE1

								功能 1: SEG75
82	85		71			IOE0	I/O	默认: IOE0 功能 1: SEG74
83		66				AVSS	G	模拟地
84	86	67	64	53	39	UP	I	电压采样信号 P 端输入
85	87	68	65	54	40	UN	I	电压采样信号 N 端输入
86	88	69	66	55	41	IBN	I	B 路电流采样信号 N 端输入
87	89	70	67	56	42	IBP	I	B 路电流采样信号 P 端输入
88	90	71	68	57	43	IAN	I	A 路电流采样信号 N 端输入
89	91	72	69	58	44	IAP	I	A 路电流采样信号 P 端输入
90		73				VDD2	P	计量主电源输入引脚。用户应该在此引脚连接一个 0.1uF 的去耦电容。当该引脚没有引出的时候, 用户可以通过软件控制电源输入。
91	92	74				AVSS	G	模拟地
92	93		72	59	45	IOA13	I/O	默认: IOA13 功能 1: UARTRXD1 功能 2: ISO7816_IO 功能 3: EINT13 功能 4: SEG73
93	94		73	60	46	IOB3	I/O	默认: IOB3 功能 1: UARTTXD1 功能 2: ISO7816_CLK 功能 3: SEG72
94	95		74	61	47	IOA12	I/O	默认: IOA12 功能 1: UARTRXD0 功能 2: EINT12 功能 3: SEG71

95	96	75	75	62	48	IOB2	I/O	默认: IOB2 功能 1: UARTTXD0 功能 2: PLLL_DIV 功能 3: SEG70
96	97	76	76	63		IOA7	I/O	默认: IOA7 功能 1: 秒脉冲输出(RTC_PLLDIV 输出) 功能 2: EINT7 功能 3: SEG69
97	98	77	77	64		IOA6	I/O	默认: IOA6 功能 1: 比较器 2 输出 功能 2: EINT6 功能 3: SEG68
98	99	78	78			IOA5	I/O	默认: IOA5 功能 1: 比较器 2 N 输入 功能 2: EINT5 功能 3: SEG67
99	100	79	79			IOA4	I/O	默认: IOA4 功能 1: 比较器 2 P 输入 功能 2: EINT4 功能 3: SEG66
100	1	80	80	1	1	EXTRSTN	I	外部复位引脚, 低电平有效, RC 滤波电路推荐使用 510K 欧电阻和 0.1uF 电容。
101	2					IOF3	I/O	默认: IOF3
102						IOF2	I/O	默认: IOF2, 芯片内部接地。
103	3					AVSS	G	模拟地

注意: 为保证系统工作正常, **IOF2** 管脚不可操作。



## 第3章 功能框图

### 3.1 系统功能框图

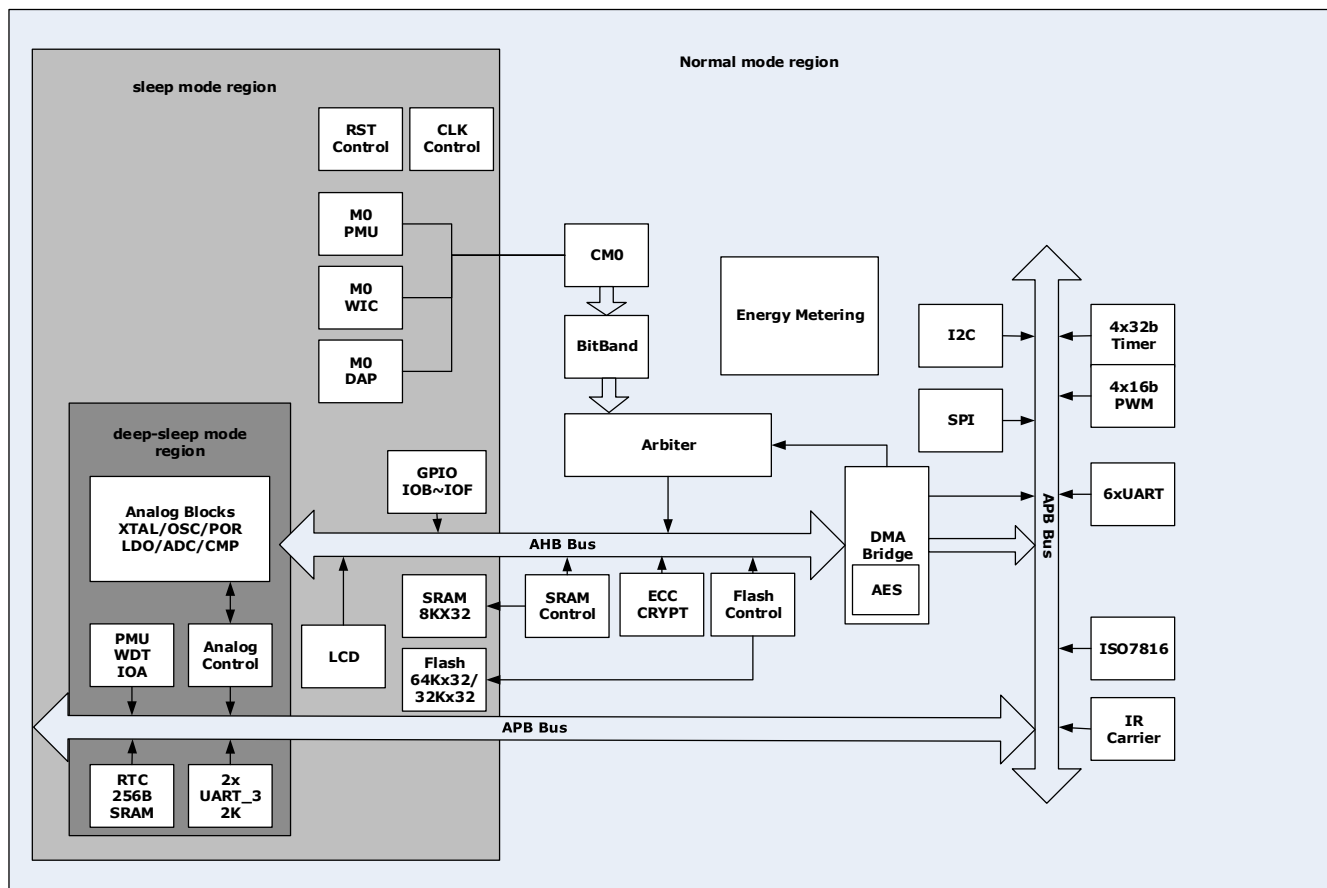


图 3-1 V94XX(A)系统功能框图

## 3.2 电源功能框图

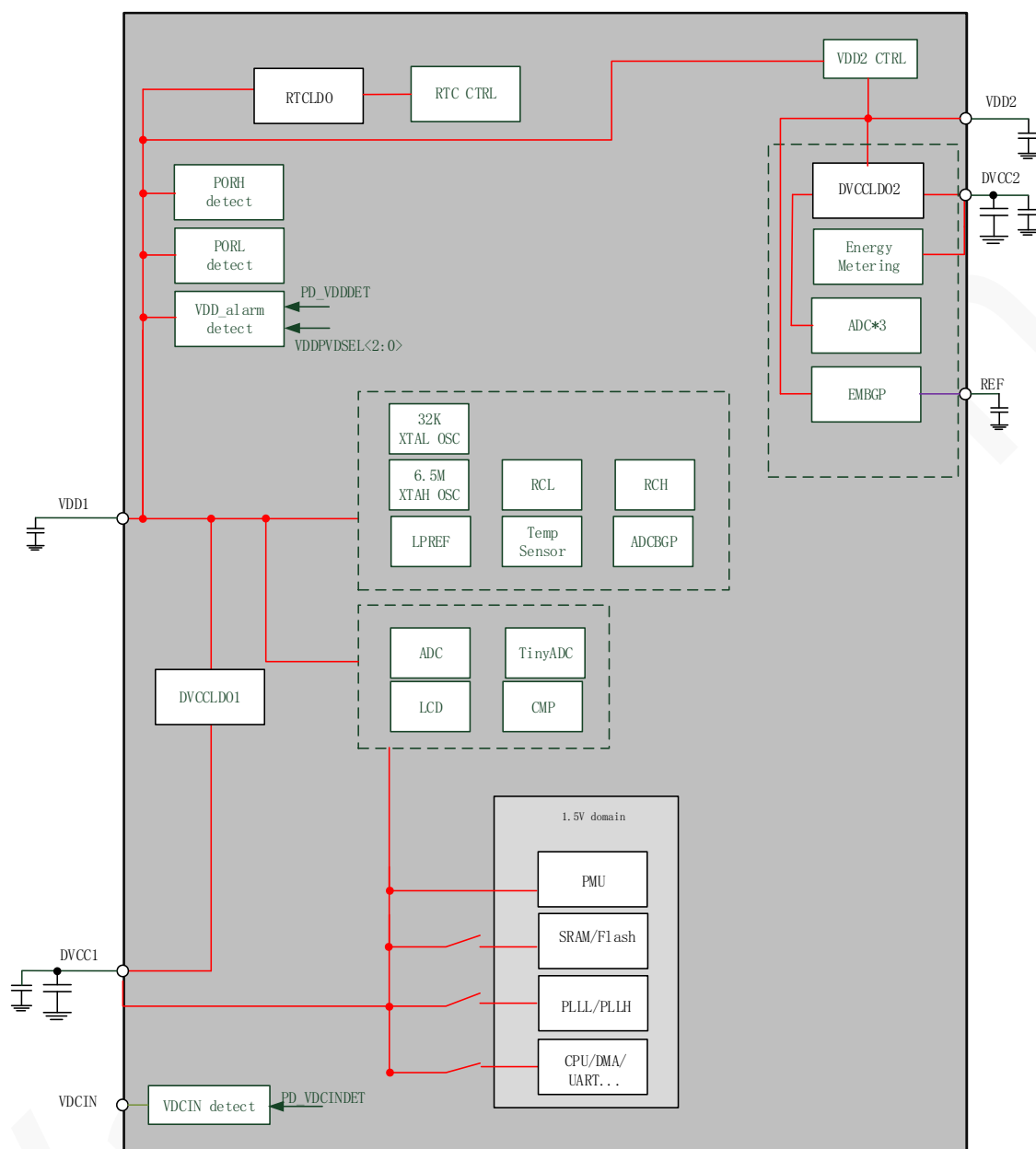


图 3-2 V94XX(A)电源系统功能框图

### 3.3 时钟功能框图

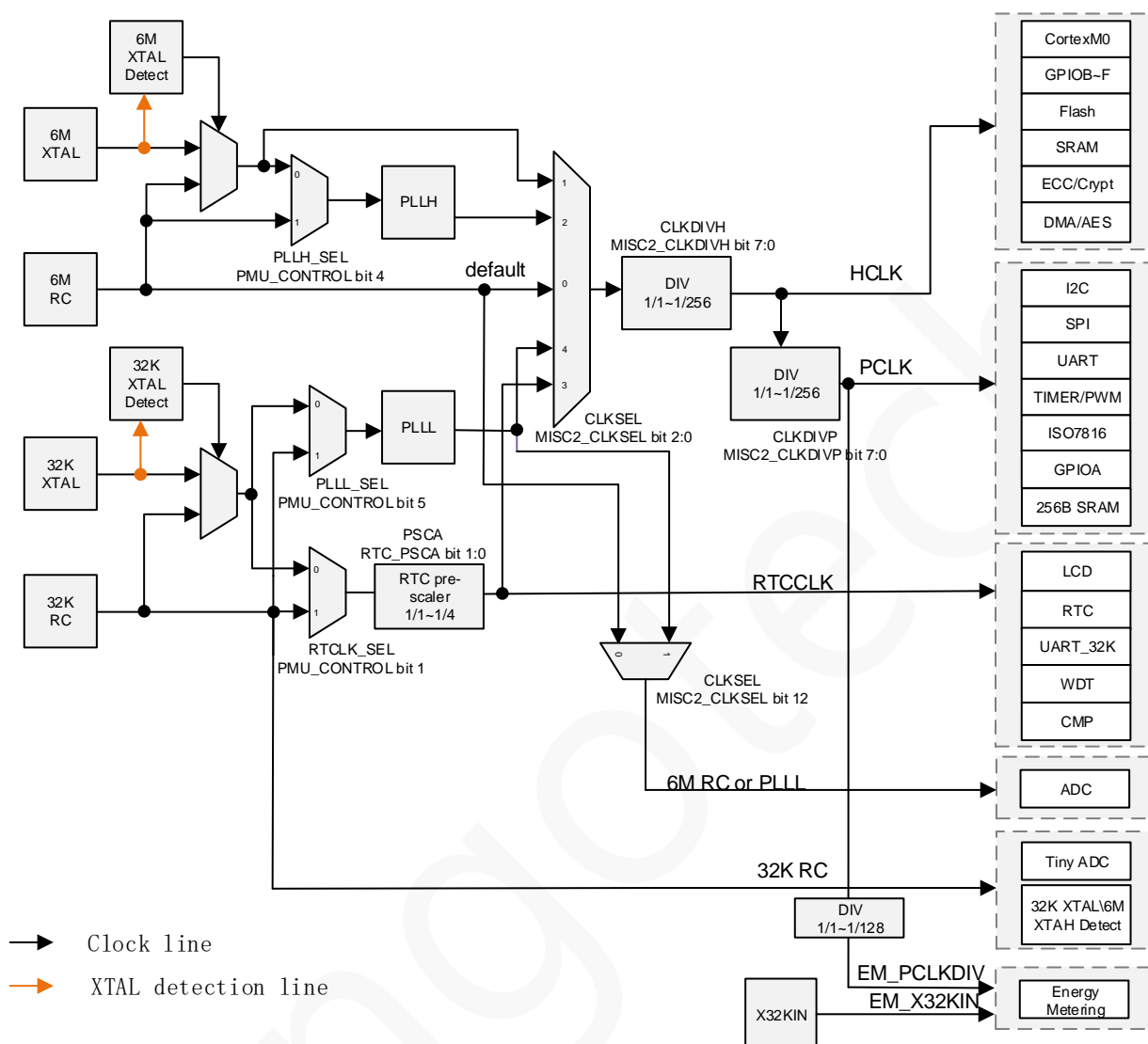


图 3-3 V94XX(A)时钟功能框图

## 第4章 内存映射

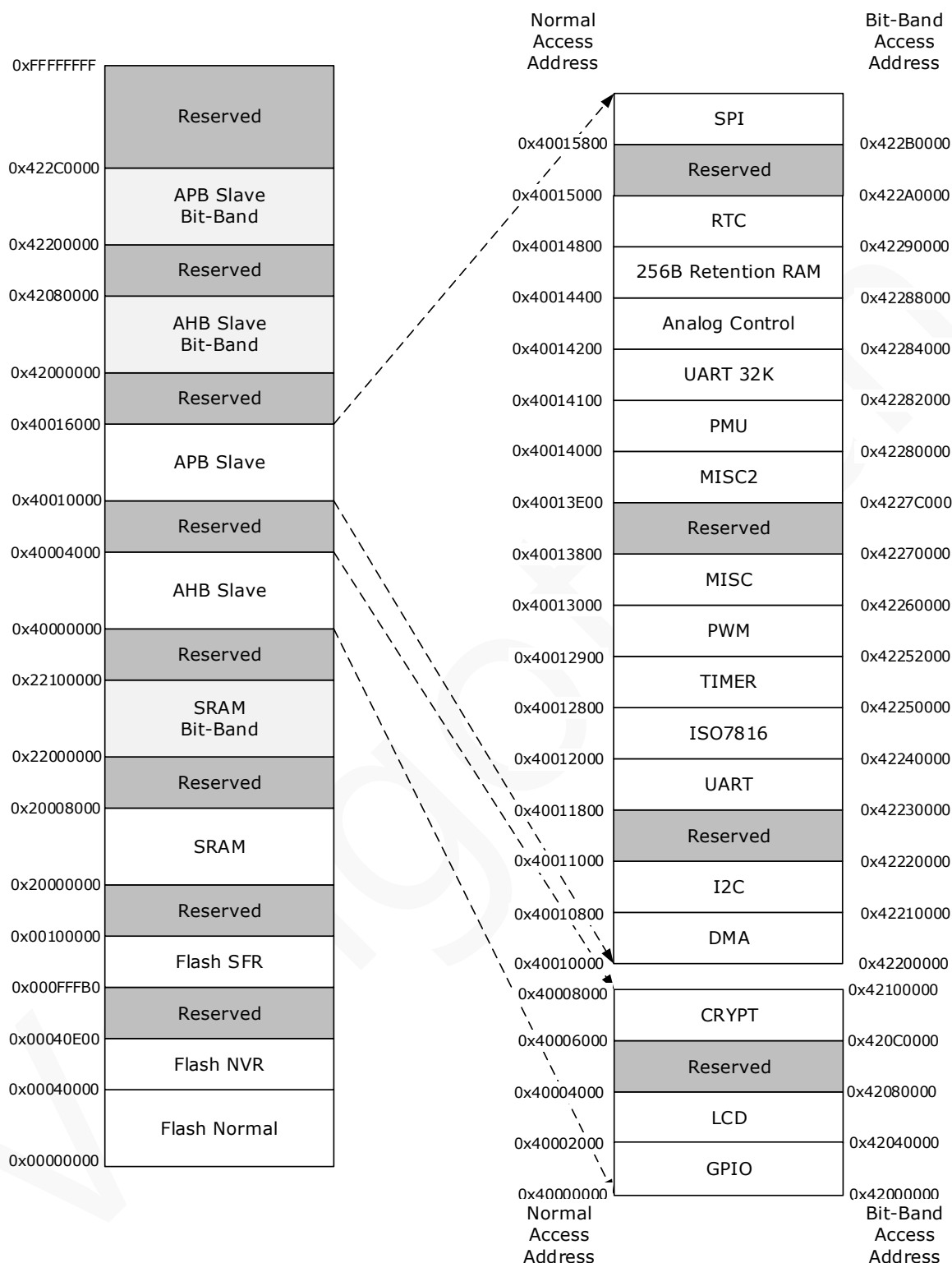


图 4-1 V94XX(A)内存映射

Bit\_band 区域仅能由 Cortex-M0 访问，而 DMA 控制器不能。除此以外的其他区域，Cortex-M0 和 DMA 控制器都可以访问。

## 4.1 寄存器地址

### 4.1.1 PMU 寄存器地址

表 4-1 PMU 寄存器地址(PMU 基地址: 0x40014000)

名称	类型	地址	描述	默认值
PMU_DSLEEPEN	R/W	0x0000	PMU 深睡眠使能寄存器	0x00000000
PMU_DSLEEPPASS	R/W	0x0004	PMU 深睡眠密码寄存器	0x00000000
PMU_CONTROL	R/W	0x0008	PMU 控制寄存器	0x0000
PMU_STS	R/C	0x000C	PMU 状态寄存器	0x0000074
PMU_IOAOEN	R/W	0x0010	IOA 输出使能寄存器	0xFFFF
PMU_IOAIE	R/W	0x0014	IOA 输入使能寄存器	0xFFFF
PMU_IOADAT	R/W	0x0018	IOA 输出数据寄存器	0x0000
PMU_IOAATT	R/W	0x001C	IOA 属性寄存器	0x0000
PMU_IOAWKUEN	R/W	0x0020	IOA 唤醒使能寄存器	0x00000000
PMU_IOASTS	R	0x0024	IOA 输入状态寄存器	--
PMU_IOAINT	R/C	0x0028	IOA 中断状态寄存器	0x0000
PMU_IOASEL	R/W	0x0038	IOA 特殊功能选择寄存器	0x0000
VERSIONID	R	0x003C	V94XX(A)版本号	--
PMU_WDTPASS	R/W	0x0040	看门狗解锁寄存器	0x00000000
PMU_WDTEN	R/W	0x0044	看门狗定时器使能寄存器	0x1
PMU_WDTCLR	W	0x0048	看门狗定时器清零寄存器	0x0000
PMU_IOANODEG	R/W	0x0050	IOA 去抖动电路控制	0x0000

表 4-2 PMU 寄存器地址(PMU 掉电保持 RAM 基地址: 0x40014400)

名称	类型	地址	描述	默认值
PMU_RAM0	R/W	0x0000	PMU 32 位掉电保持 RAM0	--
PMU_RAM1	R/W	0x0004	PMU 32 位掉电保持 RAM1	--
PMU_RAM2	R/W	0x0008	PMU 32 位掉电保持 RAM2	--
PMU_RAM63	R/W	0x00FC	PMU 32 位掉电保持 RAM63	--

### 4.1.2 ANA 寄存器地址

表 4-3 模拟控制寄存器地址(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_REG0	R/W	0x0000	模拟寄存器 0	0x00
ANA_REG1	R/W	0x0004	模拟寄存器 1	0x00
ANA_REG2	R/W	0x0008	模拟寄存器 2	0x00
ANA_REG3	R/W	0x000C	模拟寄存器 3	0x00
ANA_REG4	R/W	0x0010	模拟寄存器 4	0x00
ANA_REG5	R/W	0x0014	模拟寄存器 5	0x00

ANA_REG6	R/W	0x0018	模拟寄存器 6	0x00
ANA_REG7	R/W	0x001C	模拟寄存器 7	0x00
ANA_REG8	R/W	0x0020	模拟寄存器 8	0x00
ANA_REG9	R/W	0x0024	模拟寄存器 9	0x00
ANA_REGA	R/W	0x0028	模拟寄存器 10	0x00
ANA_REGB	R/W	0x002C	模拟寄存器 11	来自 FLASH, 用户不得修改
ANA_REGC	R/W	0x0030	模拟寄存器 12	来自 FLASH, 用户不得修改
ANA_REGD	R/W	0x0034	模拟寄存器 13	来自 FLASH, 用户不得修改
ANA_REGE	R/W	0x0038	模拟寄存器 14	来自 FLASH, 用户不得修改
ANA_REGF	R/W	0x003C	模拟寄存器 15	0x00
ANA_CTRL	R/W	0x0050	模拟控制寄存器	0x00000000
ANA_CMPOUT	R	0x0054	比较器结果寄存器	0x0030
ANA_INTSTS	R/C	0x0060	模拟中断状态寄存器	0x0000
ANA_INTEN	R/W	0x0064	模拟中断使能寄存器	0x0000
ANA_ADCCTRL	R/W	0x0068	ADC 控制寄存器	0x00000000
ANA_ADCDATA0	R	0x0070	ADC 通道 0 数据寄存器	--
ANA_ADCDATA1	R	0x0074	ADC 通道 1 数据寄存器	--
ANA_ADCDATA2	R	0x0078	ADC 通道 2 数据寄存器	--
ANA_ADCDATA3	R	0x007C	ADC 通道 3 数据寄存器	--
ANA_ADCDATA4	R	0x0080	ADC 通道 4 数据寄存器	--
ANA_ADCDATA5	R	0x0084	ADC 通道 5 数据寄存器	--
ANA_ADCDATA6	R	0x0088	ADC 通道 6 数据寄存器	--
ANA_ADCDATA7	R	0x008C	ADC 通道 7 数据寄存器	--
ANA_ADCDATA8	R	0x0090	ADC 通道 8 数据寄存器	--
ANA_ADCDATA9	R	0x0094	ADC 通道 9 数据寄存器	--
ANA_ADCDATAA	R	0x0098	ADC 通道 10 数据寄存器	--
ANA_ADCDATAB	R	0x009C	ADC 通道 11 数据寄存器	--
ANA_CMPCNT1	R/C	0x00B0	比较器 1 计数器	0x00000000
ANA_CMPCNT2	R/C	0x00B4	比较器 2 计数器	0x00000000
ANA_MISC	R/W	0x00B8	模拟 misc. 控制寄存器	0x00

### 4.1.3 RTC 寄存器地址

表 4-4 RTC 控制寄存器地址(RTC 基地址: 0x40014800)

名称	类型	地址	描述	默认值	写保护	读保护
RTC_SEC	R/W	0x0000	RTC 秒钟寄存器	--	V	V
RTC_MIN	R/W	0x0004	RTC 分钟寄存器	--	V	V

RTC_HOUR	R/W	0x0008	RTC 时寄存器	--	V	V
RTC_DAY	R/W	0x000C	RTC 天寄存器	--	V	V
RTC_WEEK	R/W	0x0010	RTC 周寄存器	--	V	V
RTC_MON	R/W	0x0014	RTC 月寄存器	--	V	V
RTC_YEAR	R/W	0x0018	RTC 年寄存器	--	V	V
RTC_WKUSEC	R/W	0x0020	RTC 唤醒秒钟寄存器	0x00	V	
RTC_WKUMIN	R/W	0x0024	RTC 唤醒分钟寄存器	0x00	V	
RTC_WKUHOURL	R/W	0x0028	RTC 唤醒时寄存器	0x00	V	
RTC_WKUCNT	R/W	0x002C	RTC 唤醒计数器寄存器	0x00000000	V	
RTC_CAL	R/W	0x0030	RTC 校正寄存器	--	V	
RTC_DIV	R/W	0x0034	RTC_PLLDIV 分频寄存器	0x00000000		
RTC_CTL	R/W	0x0038	RTC_PLLDIV 分频控制寄存器	0x0		
RTC_PWD	R/W	0x0044	RTC 密码控制寄存器	0x00000000		
RTC_CE	R/W	0x0048	RTC 写使能控制寄存器	0x0		
RTC_LOAD	R/W	0x004C	RTC 读使能控制寄存器	--		
RTC_INTSTS	R/W	0x0050	RTC 中断状态控制寄存器	0x000		
RTC_INTEN	R/W	0x0054	RTC 中断使能控制寄存器	0x000		
RTC_PSCA	R/W	0x0058	RTC 时钟预分频控制寄存器	0x0	V	
RTC_ACTI	R/W	0x0084	RTC 中心温度寄存器	0x1800	V	
RTC_ACF200	R/W	0x0088	RTC 200 倍频率寄存器	0x640000	V	
RTC_ACP0	R/W	0x0090	RTC 校正参数 0 寄存器	0x0000	V	
RTC_ACP1	R/W	0x0094	RTC 校正参数 1 寄存器	0x0000	V	
RTC_ACP2	R/W	0x0098	RTC 校正参数 2 寄存器	0x0000	V	
RTC_ACP3	R	0x009C	RTC 校正参数 3 寄存器	0x0000		
RTC_ACP4	R/W	0x00A0	RTC 校正参数 4 寄存器	0x0000	V	
RTC_ACP5	R/W	0x00A4	RTC 校正参数 5 寄存器	0x0000	V	
RTC_ACP6	R/W	0x00A8	RTC 校正参数 6 寄存器	0x0000	V	
RTC_ACP7	R/W	0x00AC	RTC 校正参数 7 寄存器	0x0000	V	
RTC_ACK1	R/W	0x00B0	RTC 校正参数 k1 寄存器	0x0000	V	
RTC_ACK2	R/W	0x00B4	RTC 校正参数 k2 寄存器	0x0000	V	
RTC_ACK3	R/W	0x00B8	RTC 校正参数 k3 寄存器	0x0000	V	
RTC_ACK4	R/W	0x00BC	RTC 校正参数 k4 寄存器	0x0000	V	
RTC_ACK5	R/W	0x00C0	RTC 校正参数 k5 寄存器	0x0000	V	
RTC_WKUCNTR	R	0x00CC	记录当前 WKUCNT 值	0x000000		
RTC_ACKTEMP	R/W	0x00D0	RTC 校正温度分段区间寄存器	0x3C2800EC	V	

#### 4.1.4 FLASH 寄存器地址

表 4-5 FLASH 控制寄存器地址(FLASH 控制寄存器基地址: 0x00000000)

名称	类型	地址	描述	默认值
FLASH_STS	R	0xFFFFBC	FLASH 编程状态寄存器	0x00
FLASH_INT	R/C	0xFFFFCC	FLASH 校验和中断状态	0x0
FLASH_CSSADDR	R/W	0xFFFFD0	FLASH 校验和开始地址	0x00000
FLASH_CSEADDR	R/W	0xFFFFD4	FLASH 校验和结束地址	0xFFFFF
FLASH_CSVALUE	R	0xFFFFD8	FLASH 校验和值读取寄存器	--
FLASH_CSCVALUE	R/W	0xFFFFDC	FLASH 校验和比较值设置寄存器	0x00000000
FLASH_PASS	R/W	0xFFFFE0	FLASH 密码寄存器	0x00000000
FLASH_CTRL	R/W	0xFFFFE4	FLASH 控制寄存器	0x0
FLASH_PGADDR	R/W	0xFFFFE8	FLASH 程序地址寄存器	0x00000
FLASH_PGDATA	R/W	0xFFFFEC	FLASH 程序字数据寄存器	--
FLASH_PGB0	R/W	0xFFFFEC	FLASH 程序字节数据寄存器 0	--
FLASH_PGB1	R/W	0xFFFFED	FLASH 程序字节数据寄存器 1	--
FLASH_PGB2	R/W	0xFFFFEE	FLASH 程序字节数据寄存器 2.	--
FLASH_PGB3	R/W	0xFFFFEF	FLASH 程序字节数据寄存器 3	--
FLASH_PGHW0	R/W	0xFFFFEC	FLASH 程序半字数据寄存器 0	--
FLASH_PGHW1	R/W	0xFFFFEE	FLASH 程序半字数据寄存器 1	--
FLASH_SERASE	R/W	0xFFFFF4	FLASH 扇区擦除控制寄存器	0x00000000
FLASH_CERASE	R/W	0xFFFFF8	FLASH 芯片擦除控制寄存器	0x00000000
FLASH_DSTB	R/W	0xFFFFFC	FLASH deep standby 控制寄存器	0x00000000

## 4.1.5 GPIO 寄存器地址

表 4-6 GPIO 控制寄存器地址(GPIO 基地址: 0x40000000)

名称	类型	地址	描述	默认值
IOB_OEN	R/W	0x0020	IOB 输出使能寄存器	0xFFFF
IOB_IE	R/W	0x0024	IOB 输入使能寄存器	0xFFFF
IOB_DAT	R/W	0x0028	IOB 输出数据寄存器	0x0000
IOB_ATT	R/W	0x002C	IOB 属性寄存器	0x0000
IOB_STS	R	0x0030	IOB 输入状态寄存器	--
IOC_OEN	R/W	0x0040	IOC 输出使能寄存器	0xFFFF
IOC_IE	R/W	0x0044	IOC 输入使能寄存器	0xFFFF
IOC_DAT	R/W	0x0048	IOC 输出数据寄存器	0x0000
IOC_ATT	R/W	0x004C	IOC 属性寄存器	0x0000
IOC_STS	R	0x0050	IOC 输入状态寄存器	--
IOD_OEN	R/W	0x0060	IOD 输出使能寄存器	0xFFFF
IOD_IE	R/W	0x0064	IOD 输入使能寄存器	0xFFFF
IOD_DAT	R/W	0x0068	IOD 输出数据寄存器	0x0000
IOD_ATT	R/W	0x006C	IOD 属性寄存器	0x0000



IOD_STS	R	0x0070	IOD 输入状态寄存器	--
IOE_OEN	R/W	0x0080	IOE 输出使能寄存器	0xFFFF
IOE_IE	R/W	0x0084	IOE 输入使能寄存器	0xFFFF
IOE_DAT	R/W	0x0088	IOE 输出数据寄存器	0x0000
IOE_ATT	R/W	0x008C	IOE 属性寄存器	0x0000
IOE_STS	R	0x0090	IOE 输入状态寄存器	--
IOF_OEN	R/W	0x00A0	IOF 输出使能寄存器	0x7
IOF_IE	R/W	0x00A4	IOF 输入使能寄存器	0x7
IOF_DAT	R/W	0x00A8	IOF 输出数据寄存器	0x0
IOF_ATT	R/W	0x00AC	IOF 属性寄存器	0x0
IOF_STS	R	0x00B0	IOF 输入状态寄存器	--
IOB_SEL	R/W	0x00C0	IOB 特殊功能选择寄存器	0x00
IOE_SEL	R/W	0x00CC	IOE 特殊功能选择寄存器	0x00
IO_MISC	R/W	0x00E0	IOMISC 控制寄存器	0x00

## 4.1.6 DMA 寄存器地址

表 4-7 DMA 控制寄存器地址(DMA 基地址: 0x40010000)

名称	类型	地址	描述	默认值
DMA_IE	R/W	0x0000	DMA 中断使能寄存器	0x000
DMA_STS	R/W	0x0004	DMA 状态寄存器	0x0000
DMA_C0CTL	R/W	0x0010	DMA 通道 0 控制寄存器	0x00000000
DMA_C0SRC	R/W	0x0014	DMA 通道 0 源寄存器	0x00000000
DMA_C0DST	R/W	0x0018	DMA 通道 0 目的寄存器	0x00000000
DMA_C0LEN	R	0x001C	DMA 通道 0 传输长度寄存器	0x0000
DMA_C1CTL	R/W	0x0020	DMA 通道 1 控制寄存器	0x00000000
DMA_C1SRC	R/W	0x0024	DMA 通道 1 源寄存器	0x00000000
DMA_C1DST	R/W	0x0028	DMA 通道 1 目的寄存器	0x00000000
DMA_C1LEN	R	0x002C	DMA 通道 1 传输长度寄存器	0x0000
DMA_C2CTL	R/W	0x0030	DMA 通道 2 控制寄存器	0x00000000
DMA_C2SRC	R/W	0x0034	DMA 通道 2 源寄存器	0x00000000
DMA_C2DST	R/W	0x0038	DMA 通道 2 目的寄存器	0x00000000
DMA_C2LEN	R	0x003C	DMA 通道 2 传输长度寄存器	0x0000
DMA_C3CTL	R/W	0x0040	DMA 通道 3 控制寄存器	0x00000000
DMA_C3SRC	R/W	0x0044	DMA 通道 3 源寄存器	0x00000000
DMA_C3DST	R/W	0x0048	DMA 通道 3 目的寄存器	0x00000000
DMA_C3LEN	R	0x004C	DMA 通道 3 传输长度寄存器	0x0000
DMA_AESCTL	R/W	0x0050	DMAAES 控制寄存器	0x00000000
DMA_AESKEY0	R/W	0x0060	DMAAES 密钥 0 寄存器	0x00000000

DMA_AESKEY1	R/W	0x0064	DMAAES 密钥 1 寄存器	0x00000000
DMA_AESKEY2	R/W	0x0068	DMAAES 密钥 2 寄存器	0x00000000
DMA_AESKEY3	R/W	0x006C	DMAAES 密钥 3 寄存器	0x00000000
DMA_AESKEY4	R/W	0x0070	DMAAES 密钥 4 寄存器	0x00000000
DMA_AESKEY5	R/W	0x0074	DMAAES 密钥 5 寄存器	0x00000000
DMA_AESKEY6	R/W	0x0078	DMAAES 密钥 6 寄存器	0x00000000
DMA_AESKEY7	R/W	0x007C	DMAAES 密钥 7 寄存器	0x00000000

## 4.1.7 UART 寄存器地址

表 4-8 UART 寄存器地址(UART 基地址: 0x40011800)

名称	类型	地址	描述	默认值
UART0_DATA	R/W	0x0000	UART0 数据寄存器	0x00
UART0_STATE	R/C	0x0004	UART0 状态寄存器	0x00
UART0_CTRL	R/W	0x0008	UART0 控制寄存器	0x000
UART0_INTSTS	R/C	0x000C	UART0 中断状态寄存器	0x00
UART0_BAUDDIV	R/W	0x0010	UART0 波特率分频寄存器	0x00000
UART0_CTRL2	R/W	0x0014	UART0 控制寄存器 2	0x0
UART1_DATA	R/W	0x0020	UART1 数据寄存器	0x00
UART1_STATE	R/C	0x0024	UART1 状态寄存器	0x00
UART1_CTRL	R/W	0x0028	UART1 控制寄存器	0x000
UART1_INTSTS	R/C	0x002C	UART1 中断状态寄存器	0x00
UART1_BAUDDIV	R/W	0x0030	UART1 波特率分频寄存器	0x00000
UART1_CTRL2	R/W	0x0034	UART1 控制寄存器 2	0x0
UART2_DATA	R/W	0x0040	UART2 数据寄存器	0x00
UART2_STATE	R/C	0x0044	UART2 状态寄存器	0x00
UART2_CTRL	R/W	0x0048	UART2 控制寄存器	0x000
UART2_INTSTS	R/C	0x004C	UART2 中断状态寄存器	0x00
UART2_BAUDDIV	R/W	0x0050	UART2 波特率分频寄存器	0x00000
UART2_CTRL2	R/W	0x0054	UART2 控制寄存器 2	0x0
UART4_DATA	R/W	0x0080	UART4 数据寄存器	0x00
UART4_STATE	R/C	0x0084	UART4 状态寄存器	0x00
UART4_CTRL	R/W	0x0088	UART4 控制寄存器	0x000
UART4_INTSTS	R/C	0x008C	UART4 中断状态寄存器	0x00
UART4_BAUDDIV	R/W	0x0090	UART4 波特率分频寄存器	0x00000
UART4_CTRL2	R/W	0x0094	UART4 控制寄存器 2	0x0
UART5_DATA	R/W	0x00A0	UART5 数据寄存器	0x00
UART5_STATE	R/C	0x00A4	UART5 状态寄存器	0x00
UART5_CTRL	R/W	0x00A8	UART5 控制寄存器	0x000

UART5_INTSTS	R/C	0x00AC	UART5 中断状态寄存器	0x00
UART5_BAUDDIV	R/W	0x00B0	UART5 波特率分频寄存器	0x00000
UART5_CTRL2	R/W	0x00B4	UART5 控制寄存器 2	0x0

## 4.1.8 UART32K 寄存器地址

表 4-9 UART32K 寄存器地址(UART32K 基地址: 0x40014100)

名称	类型	地址	描述	默认值
U32K0_CTRL0	R/W	0x0000	UART32K0 控制寄存器 0	0x000
U32K0_CTRL1	R/W	0x0004	UART32K0 控制寄存器 1	0x00
U32K0_PHASE	R/W	0x0008	UART32K0 波特率控制寄存器	0x4B00
U32K0_DATA	R	0x000C	UART32K0 接收数据缓存区	--
U32K0_STS	R/C	0x0010	UART32K0 中断状态寄存器	0x0
U32K1_CTRL0	R/W	0x0080	UART32K1 控制寄存器 0	0x000
U32K1_CTRL1	R/W	0x0084	UART32K1 控制寄存器 1	0x00
U32K1_PHASE	R/W	0x0088	UART32K1 波特率控制寄存器	0x4B00
U32K1_DATA	R	0x008C	UART32K1 接收数据缓存区	--
U32K1_STS	R/C	0x0090	UART32K1 中断状态寄存器	0x0

## 4.1.9 ISO7816 寄存器地址

表 4-10 ISO7816 寄存器地址(ISO7816 基地址: 0x40012000)

名称	类型	地址	描述	默认值
ISO7816_BAUDDIVL	R/W	0x0004	ISO7816 波特率低字节寄存器	0x00
ISO7816_BAUDDIVH	R/W	0x0008	ISO7816 波特率高字节寄存器	0x00
ISO7816_DATA	R/W	0x000C	ISO7816 数据寄存器	0x00
ISO7816_INFO	R/C	0x0010	ISO7816 信息寄存器	0x00
ISO7816_CFG	R/W	0x0014	ISO7816 控制寄存器	0x00
ISO7816_CLK	R/W	0x0018	ISO7816 时钟分频器控制寄存器	0x00

## 4.1.10 TIMER/PWM 寄存器地址

表 4-11 TIMER 寄存器地址(32 位 TIMER 基地址: 0x40012800)

名称	类型	地址	描述	默认值
TMR0_CTRL	R/W	0x0000	Timer0 控制寄存器	0x0
TMR0_VALUE	R/W	0x0004	Timer0 实时计数寄存器	0x00000000
TMR0_RELOAD	R/W	0x0008	Timer0 重载寄存器	0x00000000
TMR0_INT	R/C	0x000C	Timer0 中断状态寄存器	0x0
TMR1_CTRL	R/W	0x0020	Timer1 控制寄存器	0x0
TMR1_VALUE	R/W	0x0024	Timer1 实时计数寄存器	0x00000000

TMR1_RELOAD	R/W	0x0028	Timer1 重载寄存器	0x00000000
TMR1_INT	R/C	0x002C	Timer1 中断状态寄存器	0x0
TMR2_CTRL	R/W	0x0040	Timer2 控制寄存器	0x0
TMR2_VALUE	R/W	0x0044	Timer2 实时计数寄存器	0x00000000
TMR2_RELOAD	R/W	0x0048	Timer2 重载寄存器	0x00000000
TMR2_INT	R/C	0x004C	Timer2 中断状态寄存器	0x0
TMR3_CTRL	R/W	0x0060	Timer3 控制寄存器	0x0
TMR3_VALUE	R/W	0x0064	Timer3 实时计数寄存器	0x00000000
TMR3_RELOAD	R/W	0x0068	Timer3 重载寄存器	0x00000000
TMR3_INT	R/C	0x006C	Timer3 中断状态寄存器	0x0

表 4-12 PWM TIMER 寄存器地址(16 位 PWM TIMER 基地址: 0x40012900)

名称	类型	地址	描述	默认值
PWM0_CTL	R/W	0x0000	PWM Timer0 控制寄存器	0x00
PWM0_TAR	R	0x0004	PWM Timer0 实时计数寄存器	0x0000
PWM0_CCTL0	R/W	0x0008	PWM Timer0 比较寄存器 0	0x000
PWM0_CCTL1	R/W	0x000C	PWM Timer0 比较寄存器 1	0x000
PWM0_CCTL2	R/W	0x0010	PWM Timer0 比较寄存器 2	0x000
PWM0_CCR0	R/W	0x0014	PWM Timer0 比较数据寄存器 0	0x0000
PWM0_CCR1	R/W	0x0018	PWM Timer0 比较数据寄存器 1	0x0000
PWM0_CCR2	R/W	0x001C	PWM Timer0 比较数据寄存器 2	0x0000
PWM1_CTL	R/W	0x0020	PWM Timer1 控制寄存器	0x00
PWM1_TAR	R	0x0024	PWM Timer1 实时计数寄存器	0x0000
PWM1_CCTL0	R/W	0x0028	PWM Timer1 比较数据寄存器 0	0x000
PWM1_CCTL1	R/W	0x002C	PWM Timer1 比较数据寄存器 1	0x000
PWM1_CCTL2	R/W	0x0030	PWM Timer1 比较数据寄存器 2	0x000
PWM1_CCR0	R/W	0x0034	PWM Timer1 比较数据寄存器 0	0x0000
PWM1_CCR1	R/W	0x0038	PWM Timer1 比较数据寄存器 1	0x0000
PWM1_CCR2	R/W	0x003C	PWM Timer1 比较数据寄存器 2	0x0000
PWM2_CTL	R/W	0x0040	PWM Timer2 控制寄存器	0x00
PWM2_TAR	R	0x0044	PWMT Timer2 实时计数寄存器	0x0000
PWM2_CCTL0	R/W	0x0048	PWM Timer2 比较控制寄存器 0	0x000
PWM2_CCTL1	R/W	0x004C	PWM Timer2 比较控制寄存器 1	0x000
PWM2_CCTL2	R/W	0x0050	PWM Timer2 比较控制寄存器 2	0x000
PWM2_CCR0	R/W	0x0054	PWM Timer2 比较数据寄存器 0	0x0000
PWM2_CCR1	R/W	0x0058	PWM Timer2 比较数据寄存器 1	0x0000
PWM2_CCR2	R/W	0x005C	PWM Timer2 比较数据寄存器 2	0x0000

PWM3_CTL	R/W	0x0060	PWM Timer3 控制寄存器	0x00
PWM3_TAR	R	0x0064	PWM Timer3 实时计数寄存器	0x0000
PWM3_CCTL0	R/W	0x0068	PWM Timer3 比较控制寄存器 0	0x0000
PWM3_CCTL1	R/W	0x006C	PWM Timer3 比较控制寄存器 1	0x0000
PWM3_CCTL2	R/W	0x0070	PWM Timer3 比较控制寄存器 2	0x0000
PWM3_CCR0	R/W	0x0074	PWM Timer3 比较数据寄存器 0	0x0000
PWM3_CCR1	R/W	0x0078	PWM Timer3 比较数据寄存器 1	0x0000
PWM3_CCR2	R/W	0x007C	PWM Timer3 比较数据寄存器 2	0x0000
PWM_O_SEL	R/W	0x00F0	PWM 输出选择寄存器	0xDB51

## 4.1.11 LCD 寄存器

表 4-13 LCD 寄存器地址(LCD 基地址: 0x40002000)

名称	类型	地址	描述	默认值
LCD_FB00	R/W	0x0000	LCD 帧缓存 0 寄存器	--
LCD_FB01	R/W	0x0004	LCD 帧缓存 1 寄存器	--
LCD_FB02	R/W	0x0008	LCD 帧缓存 2 寄存器	--
LCD_FB03	R/W	0x000C	LCD 帧缓存 3 寄存器	--
LCD_FB04	R/W	0x0010	LCD 帧缓存 4 寄存器	--
LCD_FB05	R/W	0x0014	LCD 帧缓存 5 寄存器	--
LCD_FB06	R/W	0x0018	LCD 帧缓存 6 寄存器	--
LCD_FB07	R/W	0x001C	LCD 帧缓存 7 寄存器	--
LCD_FB08	R/W	0x0020	LCD 帧缓存 8 寄存器	--
LCD_FB09	R/W	0x0024	LCD 帧缓存 9 寄存器	--
LCD_FB0A	R/W	0x0028	LCD 帧缓存 10 寄存器	--
LCD_FB0B	R/W	0x002C	LCD 帧缓存 11 寄存器	--
LCD_FB0C	R/W	0x0030	LCD 帧缓存 12 寄存器	--
LCD_FB0D	R/W	0x0034	LCD 帧缓存 13 寄存器	--
LCD_FB0E	R/W	0x0038	LCD 帧缓存 14 寄存器	--
LCD_FB0F	R/W	0x003C	LCD 帧缓存 15 寄存器	--
LCD_FB10	R/W	0x0040	LCD 帧缓存 16 寄存器	--
LCD_FB11	R/W	0x0044	LCD 帧缓存 17 寄存器	--
LCD_FB12	R/W	0x0048	LCD 帧缓存 18 寄存器	--
LCD_FB13	R/W	0x004C	LCD 帧缓存 19 寄存器	--
LCD_FB14	R/W	0x0050	LCD 帧缓存 20 寄存器	--
LCD_FB15	R/W	0x0054	LCD 帧缓存 21 寄存器	--
LCD_FB16	R/W	0x0058	LCD 帧缓存 22 寄存器	--

LCD_FB17	R/W	0x005C	LCD 帧缓存 23 寄存器	--
LCD_FB18	R/W	0x0060	LCD 帧缓存 24 寄存器	--
LCD_FB19	R/W	0x0064	LCD 帧缓存 25 寄存器	--
LCD_FB1A	R/W	0x0068	LCD 帧缓存 26 寄存器	--
LCD_FB1B	R/W	0x006C	LCD 帧缓存 27 寄存器	--
LCD_FB1C	R/W	0x0070	LCD 帧缓存 28 寄存器	--
LCD_FB1D	R/W	0x0074	LCD 帧缓存 29 寄存器	--
LCD_FB1E	R/W	0x0078	LCD 帧缓存 30 寄存器	--
LCD_FB1F	R/W	0x007C	LCD 帧缓存 31 寄存器	--
LCD_FB20	R/W	0x0080	LCD 帧缓存 32 寄存器	--
LCD_FB21	R/W	0x0084	LCD 帧缓存 33 寄存器	--
LCD_FB22	R/W	0x0088	LCD 帧缓存 34 寄存器	--
LCD_FB23	R/W	0x008C	LCD 帧缓存 35 寄存器	--
LCD_FB24	R/W	0x0090	LCD 帧缓存 36 寄存器	--
LCD_FB25	R/W	0x0094	LCD 帧缓存 37 寄存器	--
LCD_FB26	R/W	0x0098	LCD 帧缓存 38 寄存器	--
LCD_FB27	R/W	0x009C	LCD 帧缓存 39 寄存器	--
LCD_CTRL	R/W	0x0100	LCD 控制寄存器	0x00
LCD_CTRL2	R/W	0x0104	LCD 控制寄存器 2	0x0000
LCD_SEGCTRL0	R/W	0x0108	LCDSEG 使能控制寄存器 0	0x00000000
LCD_SEGCTRL1	R/W	0x010C	LCDSEG 使能控制寄存器 1	0x00000000
LCD_SEGCTRL2	R/W	0x0110	LCDSEG 使能控制寄存器 2	0x00000000

### 4.1.12 SPI 寄存器地址

表 4-14 SPI 寄存器地址(SPI 基地址: 0x40015800)

名称	类型	地址	描述	默认值
SPI_CTRL	R/W	0x0000	SPI 控制寄存器	0x0000
SPI_TXSTS	R/W	0x0004	SPI 发送状态寄存器	0x8200
SPI_TXDAT	R/W	0x0008	SPI 发送 FIFO 寄存器	--
SPI_RXSTS	R/W	0x000C	SPI 接收状态寄存器	0x0000
SPI_RXDAT	R/W	0x0010	SPI 接收 FIFO 寄存器	--
SPI_MISC	R/W	0x0014	SPI Misc 控制寄存器	0x0003

### 4.1.13 I<sup>2</sup>C 寄存器地址

表 4-15 I<sup>2</sup>C 寄存器地址(I<sup>2</sup>C 基地址: 0x40010800)

名称	类型	地址	描述	默认值
I2C_DATA	R/W	0x0000	I <sup>2</sup> C 数据寄存器	0x00



I2C_ADDR	R/W	0x0004	I <sup>2</sup> C 地址寄存器	0x00
I2C_CTRL	R/W	0x0008	I <sup>2</sup> C 控制/状态寄存器	0x00
I2C_STS	R/W	0x000c	I <sup>2</sup> C 状态寄存器	0xF8
I2C_CTRL2	R/W	0x0018	I <sup>2</sup> C 中断使能寄存器	0x0

### 4.1.14 MISC 寄存器地址

表 4-16 MISC 寄存器地址(MISC 基地址: 0x40013000)

名称	类型	地址	描述	默认值
MISC_SRAMINT	R/C	0x0000	SRAM 奇偶校验错误中断状态寄存器	0x00
MISC_SRAMINIT	R/W	0x0004	SRAM 初始化寄存器	0x01
MISC_PARERR	R	0x0008	SRAM 奇偶校验错误地址寄存器	0x000
MISC_IREN	R/W	0x000C	IR 使能控制寄存器	0x00
MISC_DUTYL	R/W	0x0010	IR 调制波低电平宽度控制寄存器	0x0000
MISC_DUTYH	R/W	0x0014	IR 调制波高电平宽度控制寄存器	0x0000
MISC_IRQLAT	R/W	0x0018	Cortex-M0 IRQ 延迟控制寄存器	0x000
MISC_HIADDR	R	0x0020	AHB 无效访问地址	--
MISC_PIADDR	R	0x0024	APB 无效访问地址	--

表 4-17 MISC2 寄存器地址(MISC2 基地址: 0x40013E00)

名称	类型	地址	描述	默认值
MISC2_FLASHWC	R/W	0x0000	FLASH 等待周期寄存器	0x2100
MISC2_CLKSEL	R/W	0x0004	时钟选择寄存器	0x0
MISC2_CLKDIVH	R/W	0x0008	AHB 时钟分频控制寄存器	0x00
MISC2_CLKDIVP	R/W	0x000C	APB 时钟分频控制寄存器	0x01
MISC2_HCLKEN	R/W	0x0010	AHB 时钟使能控制寄存器	0x1FF
MISC2_PCLKEN	R/W	0x0014	APB 时钟使能控制寄存器	0xFFFFFFFF

### 4.1.15 CRYPT 寄存器地址

表 4-18 CRYPT 寄存器地址(CRYPT 基地址: 0x40006000)

名称	类型	地址	描述	默认值
CRYPT_CTRL	R/W	0x0000	CRYPT 控制寄存器	0x0000
CRYPT_PTRA	R/W	0x0004	CRYPT 指针 A	0x0000
CRYPT_PTRB	R/W	0x0008	CRYPT 指针 B	0x0000
CRYPT_PTRO	R/W	0x000C	CRYPT 指针 O	0x0000
CRYPT_CARRY	R	0x0010	CRYPT 借位/进位寄存器	0x0

### 4.1.16 Info 参数寄存器

Info 信息存储在以 0x00040800 为起始地址的区域, 只能读不能写。所有信息都有双备份, 表格中第一份数据用 1 表示, 第二份用 2 表示。每份数据都带校验 (Checksum), 校验算法: 各份中的数据项的数据相加, 对该相加后的结果取反。

其中地址 0x00040800~0x0004085C 数据，通过专用烧写工具（脱机烧写并开启 RTC 校正功能）写入。其它数据出厂前已写入。

表 4-19 Info 信息寄存器

地址	符号	信息	描述
0x00040000		Count	6
0x00040004		Check_sum1	
0x00040008		address0	GPIOA_IE
0x0004000C		value0	未封装出来的管脚相应的 bit 位写 0, 其他封装出来的相应 bit 位写 1
0x00040010		Check_sum1	
0x00040014		address1	GPIOB_IE
0x00040018		value1	未封装出来的管脚相应的 bit 位写 0, 其他封装出来的相应 bit 位写 1
0x0004001C		Check_sum1	
0x00040020		address2	GPIOC_IE
0x00040024		value2	未封装出来的管脚相应的 bit 位写 0, 其他封装出来的相应 bit 位写 1
0x00040028		Check_sum1	
0x0004002C		address3	GPIOD_IE
0x00040030		value3	未封装出来的管脚相应的 bit 位写 0, 其他封装出来的相应 bit 位写 1
0x00040034		Check_sum1	
0x00040038		address4	GPIOE_IE
0x0004003C		value4	未封装出来的管脚相应的 bit 位写 0, 其他封装出来的相应 bit 位写 1
0x00040040		Check_sum1	
0x00040044		address5	GPIOF_IE
0x00040048		value5	未封装出来的管脚相应的 bit 位写 0, 其他封装出来的相应 bit 位写 1
0x0004004C		Check_sum1	
0x00040800	P4	晶体常温偏移 1	unit(0.1ppm), 低 16 位加载到 RTC_ACP4 寄存器, 比如 0。
0x00040804		Checksum1	INV(SUM(0x40800, 0x40800))
0x00040808	P4	晶体常温偏移 2	unit(0.1ppm), 低 16 位加载到 RTC_ACP4 寄存器, 比如 0。
0x0004080C		Checksum2	INV(SUM(0x40808, 0x40808))
0x00040810	K1	晶体 K1 系数 1	加载到 RTC_ACK1 寄存器, K1 计算公式如下: $K1 = B1 / 1000000 * 65536$ , B1 为晶振曲线二次函数的分段系数。K1 值比如 20827。
0x00040814	K2	晶体 K2 系数 1	加载到 RTC_ACK2 寄存器, K2 计算公式如下: $K2 = B2 / 1000000 * 65536$ , B2 为晶振曲线二次函数的分段系数。K2 值比如 21496。
0x00040818	K3	晶体 K3 系数 1	加载到 RTC_ACK3 寄存器, K3 计算公式如下: $K3 = B3 / 1000000 * 65536$ , B3 为晶振曲线二次函数的分段系数。K3 值比如 22020。
0x0004081C	K4	晶体 K4 系数 1	加载到 RTC_ACK4 寄存器, K4 计算公式如下:



			K4=B4/1000000*65536, B4 为晶振曲线二次函数的分段系数。K4 值比如 24517。
0x00040820	K5	晶体 K5 系数 1	加载到 RTC_ACK5 寄存器, K5 计算公式如下: K5=B5/1000000*65536, B5 为晶振曲线二次函数的分段系数。K5 值比如 25257。
0x00040824		Checksum1	INV(SUM(0x40810, 0x40820))
0x00040828	K1	晶体 K1 系数 2	加载到 RTC_ACK1 寄存器, K1 计算公式如下: K1=B1/1000000*65536, B1 为晶振曲线二次函数的分段系数。K1 值比如 20827。
0x0004082C	K2	晶体 K2 系数 2	加载到 RTC_ACK2 寄存器, K2 计算公式如下: K2=B2/1000000*65536, B2 为晶振曲线二次函数的分段系数。K2 值比如 21496。
0x00040830	K3	晶体 K3 系数 2	加载到 RTC_ACK3 寄存器, K3 计算公式如下: K3=B3/1000000*65536, B3 为晶振曲线二次函数的分段系数。K3 值比如 22020。
0x00040834	K4	晶体 K4 系数 2	加载到 RTC_ACK4 寄存器, K4 计算公式如下: K4=B4/1000000*65536, B4 为晶振曲线二次函数的分段系数。K4 值比如 24517。
0x00040838	K5	晶体 K5 系数 2	加载到 RTC_ACK5 寄存器, K5 计算公式如下: K5=B5/1000000*65536, B5 为晶振曲线二次函数的分段系数。K5 值比如 25257。
0x0004083C		Checksum2	INV(SUM(0x40828, 0x40838))
0x00040840	ACTI	晶体顶点温度 1	加载到 RTC_ACTI 寄存器, 比如: 0x1800。
0x00040844		Checksum1	INV(SUM(0x40840, 0x40840))
0x00040848	ACTI	晶体顶点温度 2	加载到 RTC_ACTI 寄存器, 比如: 0x1800。
0x0004084C		Checksum2	INV(SUM(0x40848, 0x40848))
0x00040850	KTEMP x(x=4 ~1)	晶体 K 系数温度分段区间设置 1	加载到 RTC_ACKTEMP 寄存器, 比如 0x3C2800EC。
0x00040854	—	Checksum1	INV(SUM(0x40850, 0x40850))
0x00040858	KTEMP x(x=4 ~1)	晶体 K 系数温度分段区间设置 2	加载到 RTC_ACKTEMP 寄存器, 比如 0x3C2800EC。
0x0004085C		Checksum2	INV(SUM(0x40858, 0x40858))
...			保留
0x00040CE0		BAT_R offset 1	(3.6-BAT 测量值)*1000, 电阻分压, BAT 包括 BAT1 通道和 BAT2 通道。
0x00040CE4		BAT_C offset 1	(3.6-BAT 测量值)*1000, 电容分压
0x00040CE8		Checksum1	INV(SUM(0x40CE0, 0x40CE4))
0x00040CEC			保留
0x00040CF0		BAT_R offset 2	(3.6-BAT 测量值)*1000, 电阻分压
0x00040CF4		BAT_C offset 2	(3.6-BAT 测量值)*1000, 电容分压
0x00040CF8		Checksum2	INV(SUM(0x40CF0, 0x40CF4))
0x00040CFC			保留
0x00040D00	P1/P0	RTC_ACP1/0 设置 1	高 16 位加载到 RTC_ACP1 寄存器, 比如: 1060; 低 16 位加载到 RTC_ACP0 寄存器, 比如: -214。

0x00040D04	P2'	RTC_ACP2 设置 1	该地址存储值记为 P2'，比如：-19746971。根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2 寄存器。
0x00040D08	P5	RTC_ACP5 设置 1	高 16 位加载到 RTC_ACP5 寄存器，比如：6444；低 16 位舍弃。
0x00040D0C	P7/P6'	RTC_ACP7/6 设置 1	高 16 位加载到 RTC_ACP7 寄存器，比如：0；低 16 位加载到 P6'。比如：1342。 根据公式 $P6=a*P6'$ ，将 P6 加载至 RTC_ACP6 寄存器，其中 $a=PCLK/6553600$ 。
0x00040D10		Checksum1	$INV(SUM(0x40D00, 0x40D0C))$
0x00040D14	P1/P0	RTC_ACP1/0 设置 2	高 16 位加载到 RTC_ACP1 寄存器，比如：1060；低 16 位加载到 RTC_ACP0 寄存器，比如：-214。
0x00040D18	P2'	RTC_ACP2 设置 2	该地址存储值记为 P2'，比如：-19746971。根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2 寄存器。
0x00040D1C	P5	RTC_ACP5 设置 2	高 16 位加载到 RTC_ACP5 寄存器，比如：6444；低 16 位舍弃。
0x00040D20	P7/P6'	RTC_ACP7/6 设置 2	高 16 位加载到 RTC_ACP7 寄存器，比如：0；低 16 位加载到 P6'。比如：1342。 根据公式 $P6=a*P6'$ ，将 P6 加载至 RTC_ACP6 寄存器，其中 $a=PCLK/6553600$ 。
0x00040D24		Checksum2	$INV(SUM(0x40D14, 0x40D20))$
0x00040D28		VDD_OUT gain 1	$Pre-trimresult/3.3*10000$
0x00040D2C		DVCC1 gain 1	$Pre-trimresult/1.5*10000$
0x00040D30		ADCBGP gain 1	$Pre-trimresult/1.2*10000$
0x00040D34		RCL gain 1	$Pre-trimresult/32768*10000$
0x00040D38		RCH gain 1	$Pre-trimresult/6553600*10000$
0x00040D3C		Check sum 1	$INV(SUM(0x40D28, 0x40D38))$
0x00040D40		VDD_OUT gain 2	$Pre-trimresult/3.3*10000$
0x00040D44		DVCC1 gain 2	$Pre-trimresult/1.5*10000$
0x00040D48		ADCBGP gain 2	$Pre-trimresult/1.2*10000$
0x00040D4C		RCL gain 2	$Pre-trimresult/32768*10000$
0x00040D50		RCH gain 2	$Pre-trimresult/6553600*10000$
0x00040D54		Check sum 2	$INV(SUM(0x40D40, 0x40D50))$
0x00040D58		ID word 0 , Backup 1	
0x00040D5C		ID word 1 , Backup 1	
0x00040D60		ID Check sum 1	$INV(SUM(0x40D58, 0x40D5C))$
0x00040D64		ID word 0 , Backup 2	
0x00040D68		ID word 1 , Backup 2	
0x00040D6C		ID Check sum 2	$INV(SUM(0x40D64, 0x40D68))$
0x00040D70	Tr	实际温度 1 (来自 tmp275)	根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2

0x00040D74	Tm	测量温度 1 (来自 ADC)	寄存器。
0x00040D78		Temp Check sum 1	INV(SUM(0x40D70, 0x40D74))
0x00040D7C	Tr	实际温度 2 (来自 tmp275)	根据公式: $P2=P2'+(Tr-Tm)*256$ , 将 P2 加载到 RTC_ACP2 寄存器。
0x00040D80	Tm	测量温度 2 (来自 ADC)	
0x00040D84		Temp Check sum 2	INV(SUM(0x40D7C, 0x40D80))
0x00040D88			保留
0x00040D8C			保留
0x00040D90		Measure LCD LDO gain 1	Pre-trimresult/3.3*10000
0x00040D94		VLCD setting 1	
0x00040D98		LCD LDO Check sum 1	INV(SUM(0x40D90, 0x40D94))
0x00040D9C		Measure LCD LDO gain 2	Pre-trimresult/3.3*10000
0x00040DA0		VLCD setting 2	
0x00040DA4		LCD LDO Check sum 2	INV(SUM(0x40D9C, 0x40DA0))
0x00040DC0		Analog trim data1	CP 测试时自动加载到 ANA_REGx (x=B~E) 的值
0x00040DC4		0xFFFFFFFF	
0x00040DC8		0xFFFFFFFF	
0x00040DCC		Checksum1	INV(SUM(0x40DC0, 0x40DC8)), 其中 0x00040DC8 地址的数据不可读, 使用 0xFFFFFFFF 代替。
0x00040DD0		Analog trim data2	CP 测试时自动加载到 ANA_REGx (x=B~E) 的值
0x00040DD4		0xFFFFFFFF	
0x00040DD8		0xFFFFFFFF	
0x00040DDC		Checksum2	INV(SUM(0x40DD0, 0x40DD8)), 其中 0x00040DD8 地址的数据不可读, 使用 0xFFFFFFFF 代替。
...			
0x00040400	a1	ADC_CHx 通道在 3.3V 系统下不分压标定 1	Vdc=a1/100000000*X+b1/100000000 (用补码方式存放负值)
0x00040404	b1	ADC_CHx 通道在 3.3V 系统下不分压标定 1	
0x00040408	a2	ADC_CHx 通道在 3.3V 系统下电阻分压标定 1	Vdc=a2/100000000*X+b2/100000000 (用补码方式存放负值)
0x0004040C	b2	ADC_CHx 通道在 3.3V 系统下电阻分压标定 1	

0x00040410	a3	ADC_CHx 通道在 3.3V 系统下电容分压标定 1	Vdc=a3/100000000*X+b3/100000000 (用补码方式存放负值)
0x00040414	b3	ADC_CHx 通道在 3.3V 系统下电容分压标定 1	
0x00040418	a4	BAT1 通道在 3.3V 系统下电阻分压标定 1	Vdc=a4/100000000*X+b4/100000000 (用补码方式存放负值)
0x0004041C	b4	BAT1 通道在 3.3V 系统下电阻分压标定 1	
0x00040420	a5	BAT1 通道在 3.3V 系统下电容分压标定 1	Vdc=a5/100000000*X+b5/100000000 (用补码方式存放负值)
0x00040424	b5	BAT1 通道在 3.3V 系统下电容分压标定 1	
0x00040428	a6	BAT2 通道在 3.3V 系统下电阻分压标定 1	Vdc=a6/100000000*X+b6/100000000 (用补码方式存放负值)
0x0004042C	b6	BAT2 通道在 3.3V 系统下电阻分压标定 1	
0x00040430	a7	BAT2 通道在 3.3V 系统下电容分压标定 1	Vdc=a7/100000000*X+b7/100000000 (用补码方式存放负值)
0x00040434	b7	BAT2 通道在 3.3V 系统下电容分压标定 1	
0x00040438		Checksum1	INV(SUM(0x00040400, 0x00040434))
0x0004043C			保留
0x00040440	a1	ADC_CHx 通道在 3.3V 系统下不分压标定 2	Vdc=a1/100000000*X+b1/100000000 (用补码方式存放负值)
0x00040444	b1	ADC_CHx 通道在 3.3V 系统下不分压标定 2	
0x00040448	a2	ADC_CHx 通道在 3.3V 系统下电阻分压标定 2	Vdc=a2/100000000*X+b2/100000000 (用补码方式存放负值)
0x0004044C	b2	ADC_CHx 通道在 3.3V 系统下电阻分压标定 2	
0x00040450	a3	ADC_CHx 通道在 3.3V 系统下电容分压标定 2	Vdc=a3/100000000*X+b3/100000000 (用补码方式存放负值)
0x00040454	b3	ADC_CHx 通道在 3.3V 系统下电容分	

		压标定 2	
0x00040458	a4	BAT1 通道在 3.3V 系统下电阻分压标定 2	Vdc=a4/100000000*X+b4/100000000 (用补码方式存放负值)
0x0004045C	b4	BAT1 通道在 3.3V 系统下电阻分压标定 2	
0x00040460	a5	BAT1 通道在 3.3V 系统下电容分压标定 2	Vdc=a5/100000000*X+b5/100000000 (用补码方式存放负值)
0x00040464	b5	BAT1 通道在 3.3V 系统下电容分压标定 2	
0x00040468	a6	BAT2 通道在 3.3V 系统下电阻分压标定 2	Vdc=a6/100000000*X+b6/100000000 (用补码方式存放负值)
0x0004046C	b6	BAT2 通道在 3.3V 系统下电阻分压标定 2	
0x00040470	a7	BAT2 通道在 3.3V 系统下电容分压标定 2	Vdc=a7/100000000*X+b7/100000000 (用补码方式存放负值)
0x00040474	b7	BAT2 通道在 3.3V 系统下电容分压标定 2	
0x00040478		Checksum2	INV(SUM(0x00040440, 0x00040474))

## 第5章 电源系统

V94XX(A)电源控制器有以下特点:

- VDD 输入电压范围: 2.2V~3.6V
- 其中 ADC 部分和计量部分支持的工作电压范围: 2.6V~3.6V
- GPIO 电源为 VDD
- 模拟电路由 VDD 供电
- 数字电路和 PLL 电路由 DVCC 供电
- 支持低电压监测和实时电池监测

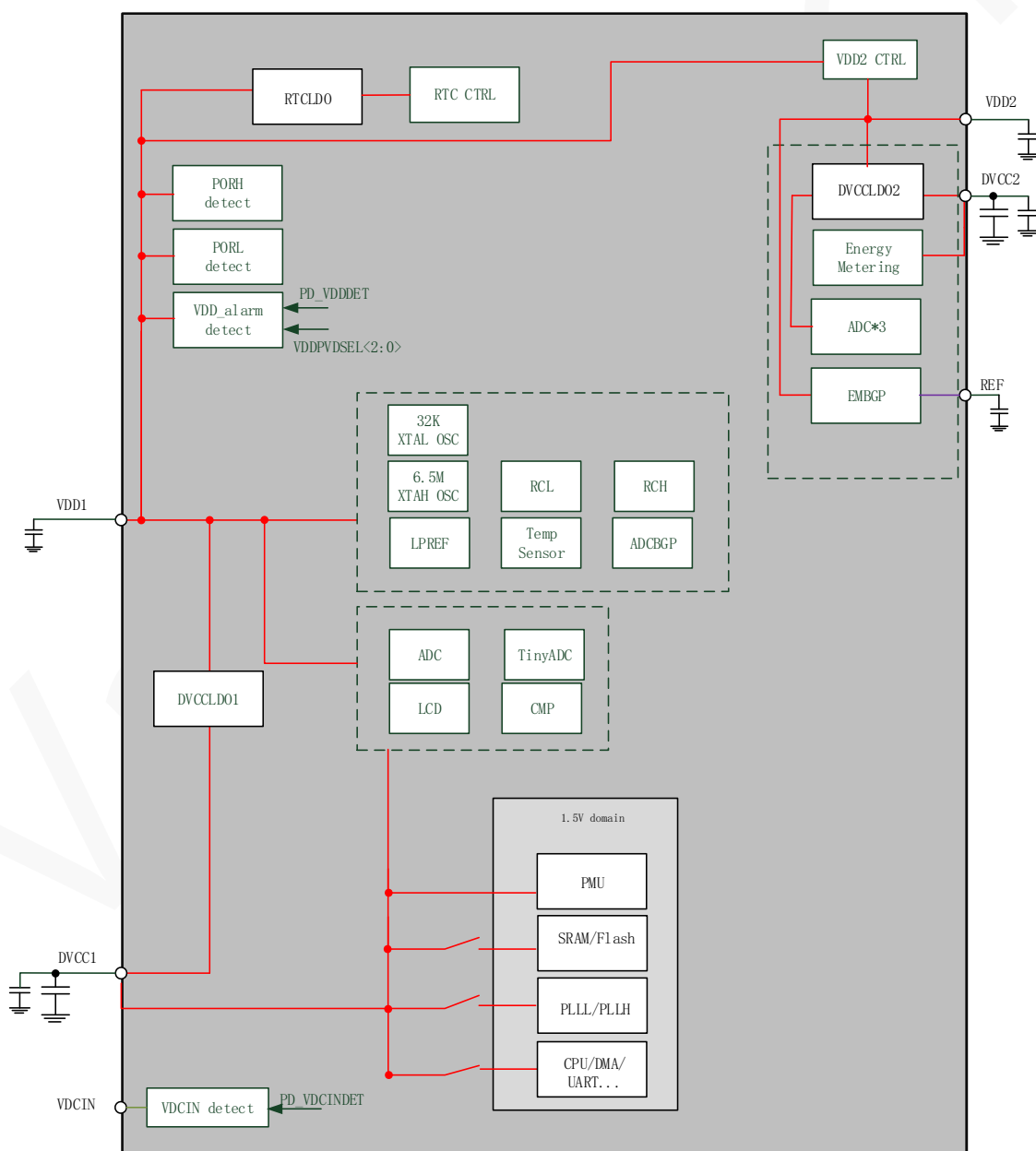


图 5-1 V94XX(A)电源系统功能框图

## 5.1 寄存器地址

表 5-1 ANA 寄存器地址(ANA 基地址: 0x40014200)

寄存器名	类型	地址	描述	默认值
ANA_REG5	R/W	0x0014	模拟寄存器 5	0x00
ANA_REG6	R/W	0x0018	模拟寄存器 6	0x00
ANA_REG7	R/W	0x001C	模拟寄存器 7	0x00
ANA_REG8	R/W	0x0020	模拟寄存器 8	0x00
ANA_REG9	R/W	0x0024	模拟寄存器 9	0x00
ANA_REGA	R/W	0x0028	模拟寄存器 10	0x00
ANA_REGF	R/W	0x003C	模拟寄存器 15	0x00
ANA_CTRL	R/W	0x0050	模拟控制寄存器	0x00000000
ANA_CMPOUT	R	0x0054	比较器结果寄存器	0x0030
ANA_INTSTS	R/C	0x0060	模拟中断状态寄存器	0x0000
ANA_INTEN	R/W	0x0064	模拟中断使能寄存器	0x0000

## 5.2 寄存器定义

### 5.2.1 ANA\_REG5 寄存器

表 5-2 ANA\_REG5 寄存器

Bit	名称	功能	描述
6	VDDLVDETPD	VDD 低电压检测模块电源开关	0: 使能 1: 禁止

### 5.2.2 ANA\_REG6 寄存器

表 5-3 ANA\_REG6 寄存器

Bit	名称	功能	描述
6	BAT1DISC	给 BAT1 电池放电。 放电电阻是 1.7k，放电电流是 Vbat1/1.7k。	0: 禁用从 BAT1 到 GND 的 1.7k 欧姆电阻。 1: 使能从 BAT1 到 GND 的 1.7k 欧姆电阻。
7	BAT2DISC	给 BAT2 电池放电。 放电电阻是 1.7k，放电电流是 Vbat2/1.7k。	0: 禁用从 BAT2 到 GND 的 1.7k 欧姆电阻。 1: 使能从 BAT2 到 GND 的 1.7k 欧姆电阻。

### 5.2.3 ANA\_REG7 寄存器

表 5-4 ANA\_REG7 寄存器

Bit	名称	功能	描述
7: 0	-	保留	默认值为 0，必须配置为 0。

## 5.2.4 ANA\_REG8 寄存器

表 5-5 ANA\_REG8 寄存器

Bit	名称	功能	描述
3: 0	-	保留	0
6: 4	VDDPVDSEL[2: 0]	VDD 电压报警阈值设置寄存器。当 VDD 电压低于该设置值时，将产生 VDDALARM 中断，并置相应中断标志位为 1。	000: 保留 001: 保留 010: 保留 011: 3.6V 100: 3.2V 101: 2.9V 110: 2.6V 111: 2.3V
7	-	保留	默认值是 0，必须配置为 1

## 5.2.5 ANA\_REG9 寄存器

表 5-6 ANA\_REG9 寄存器

Bit	名称	功能	描述
7	PD_VDDDET	关闭 VDD 电压监测模块，该模块由 VDD 供电。	0: 开启。 1: 关闭。

## 5.2.6 ANA\_REGA 寄存器

表 5-7 ANA\_REGA 寄存器

Bit	名称	功能	描述
6:0	-	保留	默认值是 0x0，必须配置为 0x0A。
7	PD_VDCINDET	关闭 VDCIN 监测模块。	0: 开启 1: 关闭

## 5.2.7 ANA\_REGF 寄存器

表 5-8 ANA\_REGF 寄存器

Bit	名称	功能	描述
2	VDDO_EN	VDD_OUT 引脚输出使能控制。	0: 高阻。 1: VDD_OUT 输出 VDD 电平，可以用来驱动小功率模块。

## 5.2.8 ANA\_CTRL 寄存器

表 5-9 ANA\_CTRL 寄存器

Bit	名称	类型	描述	默认值
-----	----	----	----	-----



26	PDNS2	R/W	该位控制 VDDALARM 为 0（同时考虑 PDNS 的设置）时进入深睡眠的行为。  0: 当 VDDALARM 为 0 时不会进入深睡眠。当 VDDALARM 为 1 时，系统可以进入深睡眠模式，并且当 VDDALARM 变为 0 时，系统自动从深睡眠唤醒。  1: VDDALARM 不作为深睡眠条件。	0x0
25: 24	VDCINDEB	R/W	VDCIN 去抖控制寄存器。  0: 无去抖 1: 2 个 RTCCLK 时钟周期消抖。 2: 3 个 RTCCLK 时钟周期消抖。 3: 4 个 RTCCLK 时钟周期消抖。  当使能消抖时，只有当信号在 RTCCLK 时钟的多周期内没有变化时，输入信号才有效。此外，唤醒和中断的响应时间将被延迟，直到信号有效。该电路可以在所有模式下工作，包括浅睡眠和深睡眠模式。	0x0
6	PDNS	R/W	该位控制 VDCINDROP 为 0（同时考虑 PDNS2 的设置）时进入深睡眠的行为。  0: VDCINDROP 为 0 时不会进入深睡眠。当 VDCINDROP 为 1 时，系统可以进入深睡眠，并且当 VDCINDROP 变为 0 时，系统自动从深睡眠唤醒。  1: VDCINDROP 不作为深睡眠条件。	0x0
5: 4	-	-	保留	0

## 5.2.9 ANA\_CMPOUT 寄存器

表 5-10 ANA\_CMPOUT 寄存器

Bit	名称	类型	描述	默认值
10	-	R	保留	0
8	VDCINDROP	R	VDCIN 掉电状态位  0: VDCIN 未掉电（VDCIN 大于阈值） 1: VDCIN 掉电（VDCIN 小于阈值）	0x0
7	VDDALARM	R	VDD 电压报警状态位  0: VDD 电压高于 VDDPVDSEL 设置的电压 1: VDD 电压低于 VDDPVDSEL 设置的电压	0x0
6: 0	-	R	保留	0

## 5.2.10 ANA\_INTSTS 寄存器

表 5-11 ANA\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
11	INTSTS11	R/C	当 VDCINDROP 为 0，进入睡眠模式中断标志位（VDCIN 大于阈值）。当 VDCINDROP 为 0，并且检测到系统进入睡眠的动作，将置位中断标志位。用户可以使能中断使能位（INTEN11），使得系统可以从浅睡眠和深睡眠模式唤醒（VDCINDROP 为 0）。  读 0: 无进入睡眠事件发生（VDCINDROP=0）	0x0

			读 1: 发生进入睡眠事件 (VDCINDROP=0) 写 0: 无影响 写 1: 清除该位	
10	INTSTS10	R/C	VDDLVL 的中断标志, 该中断会在 VDDLVL 上升和下降时产生。 读 0: 无 VDDLVL 中断 读 1: 发生 VDDLVL 中断 写 0: 无效 写 1: 清除该位	0x0
8	INTSTS8	R/C	VDCINDROP 的中断标志, 该中断会在 VDCINDROP 上升和下降时产生。 读 0: 无 VDCIN 中断 读 1: 发生 VDCIN 中断 写 0: 无效 写 1: 清零	0x0
7	INTSTS7	R/C	VDDALARM 的中断标志, 该中断会在 VDDALARM 上升和下降时产生。 读 0: 无 VDDALARM 中断 读 1: 发生 VDDALARM 中断 写 0: 无效 写 1: 清零	0x0
6: 0	-	-	保留	0

## 5.2.11 ANA\_INTEN 寄存器

表 5-12 ANA\_INTEN 寄存器

Bit	名称	类型	描述	默认值
11	INTEN11	R/W	当 VDCINDROP 为 0, 检测到进入睡眠模式的动作, 该事件的中断和唤醒使能控制。 0: 当 VDCINDROP 为 0 时, 检测到进入睡眠, 禁止该中断或唤醒 1: 当 VDCINDROP 为 0 时, 检测到进入睡眠, 使能该中断或唤醒	0x0
10	INTEN10	R/W	VDDLVL 中断或唤醒使能控制。 0: 禁止 VDDLVL 中断或唤醒 1: 使能 VDDLVL 中断或唤醒	0x0
9	-	-	保留	0
8	INTEN8	R/W	上升和下降时, VDCINDROP 中断或唤醒使能控制。 0: 禁止 VDCIN 中断或唤醒 1: 使能 VDCIN 中断或唤醒	0x0
7	INTEN7	R/W	上升和下降时, VDDALARM 中断或唤醒使能控制。 0: 禁止 VDDALARM 中断或唤醒	0x0

			1: 使能 VDDALARM 中断或唤醒	
6: 0	-	-	保留	0

## 5.3 DVCC

V94XX(A)的 PLL 电路和数字电路都由数字电路电源 DVCC。当 DVCC 的输入电压大于 1.7V 时，DVCC 会输出稳定的 1.5V 电压。

数字电路电源的驱动能力为 35mA，当负载电流小于等于 35mA 时，电源输出电压稳定；负载电流大于 35mA 时，负载电流越高，则数字供电越低。

该电源电路始终工作，除非整个芯片系统掉电。

DVCC 管脚外部需要连接去耦电容，建议采用一个 10F 电容与一个 0.1uF 电容并联。

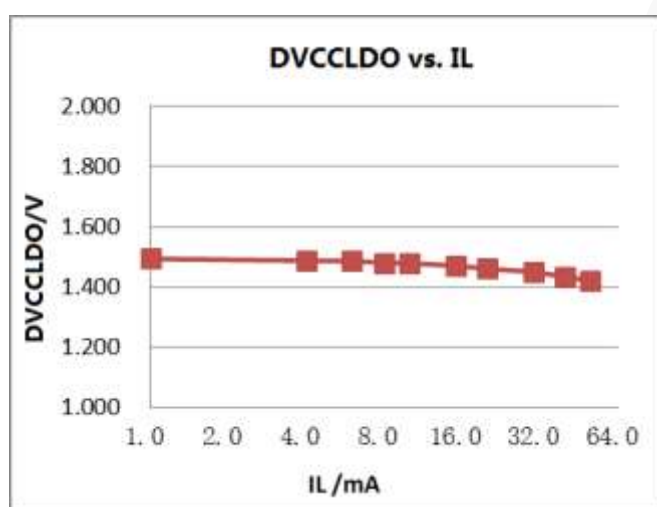


图 5-2 DVCCCLDO 输出电压与负载电流之间的关系

## 5.4 电源监测

### 5.4.1 VDCIN 电源检测

在 V94XX(A)中，主电源电阻分压后送入“VDCIN”管脚。电源检测系统会持续监视 VDCIN 管脚的电压。当 VDCIN 电压低于 1.3V 时，标志位 VDCINDROP (ANACMP\_OUT bit8) 将会被置 1 并产生掉电中断。

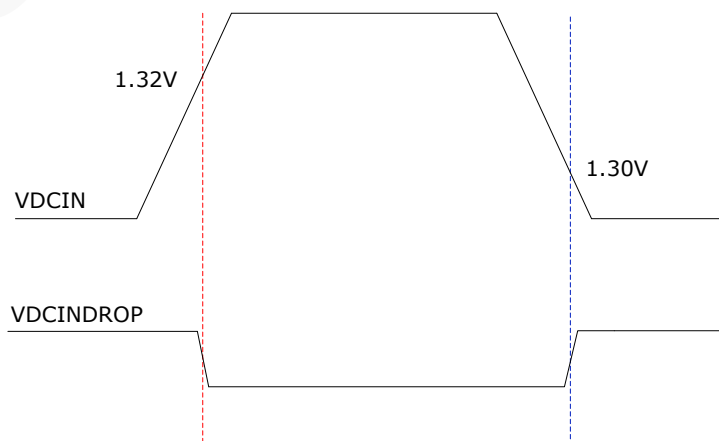


图 5-3 VDCIN 管脚输入信号与位 VDCINDROP 的关系

## 5.4.2 VDD 电源监测

该电源监测模块监测 VDD 的输入电压，当 VDD 的电压低于阈值（寄存器 ANA\_REG8 的位 VDDPVDSEL<2: 0> 设置的电压）时，检测模块将向 MCU 发出一个警告中断。

VDDPVDSEL<2:0>	电压报警阈值设置	000: 保留 001: 保留 010: 保留 011: 3.6V 100: 3.2V 101: 2.9V 110: 2.6V 111: 2.3V
----------------	----------	--

## 第6章 工作模式

### 6.1 简介

PMU 控制器（电源控制器 Power Management Unit，简称 PMU）用于控制 V94XX(A) 的浅睡眠和深睡眠模式。PMU 控制器内支持最多 16 个 IO 唤醒源，可以将芯片从浅睡眠或深睡眠模式唤醒。

浅睡眠模式将掉电部分外设，只有 CPU、SRAM、GPIO、RTC、模拟控制器、PMU 和 UART32K 等模块在此模式下会工作。

深睡眠模式将掉电 CPU 和所有外设（包括 FLASH、SRAM、LCD 控制器和 GPIOB~GPIOF）。只有 GPIOA、RTC、模拟控制器、PMU 和 UART32K 模块在此模式下会工作。

### 6.2 特点

- 深睡眠模式进出可控，并且深睡眠唤醒后复位相应模块，程序从 0 地址开始运行。
- 浅睡眠模式进出可控，浅睡眠唤醒后，MCU 进相应唤醒源中断服务函数，中断事件响应完成后，回到主程序继续往下运行。
- 密码保护，以避免在意外事件中进入深睡眠模式。
- 16 个具有唤醒和中断功能的 GPIO。
- 进入浅睡眠和深度睡眠时会自动关闭时钟，浅睡眠唤醒后会自动打开睡眠前的系统时钟；深睡眠唤醒后会自动打开 RCH 时钟。
- 晶体缺失监测和中断。
- 内置的 256 字节 SRAM 在深睡眠模式下可保存必要的数。
- 深睡眠和浅睡眠唤醒后自动打开看门狗。

## 6.3 功能框图

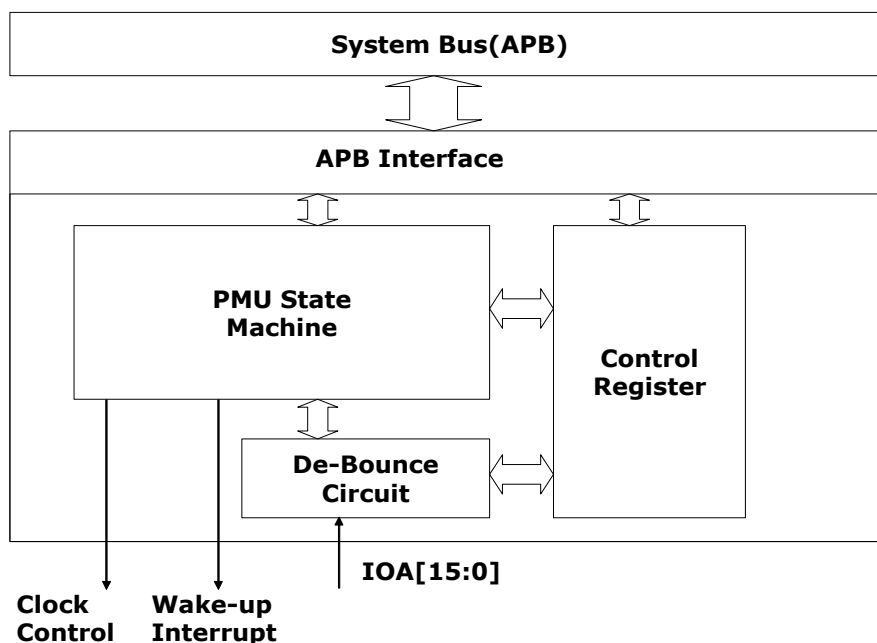


图 6-1 PMU 功能框图

## 6.4 寄存器地址

表 6-1 PMU 工作模式寄存器地址(PMU 基地址: 0x40014000)

名称	类型	地址	描述	默认值
PMU_DSLEEPEN	R/W	0x0000	PMU 深睡眠使能寄存器	0x00000000
PMU_DSLEEPPASS	R/W	0x0004	PMU 深睡眠密码寄存器	0x00000000
PMU_CONTROL	R/W	0x0008	PMU 控制寄存器	0x0000
PMU_STS	R/C	0x000C	PMU 状态寄存器	0x0000074

表 6-2 PMU RAM Retention 寄存器地址(PMU 掉电保持 RAM 基地址: 0x40014400)

名称	类型	地址	描述	默认值
PMU_RAM0	R/W	0x0000	PMU32 掉电保持 RAM0	--
PMU_RAM1	R/W	0x0004	PMU32 掉电保持 RAM1	--
PMU_RAM2	R/W	0x0008	PMU32 掉电保持 RAM2	--
....	....	....	....	--
PMU_RAM63	R/W	0x00FC	PMU32 掉电保持 RAM63	--

表 6-3 ANA 控制器寄存器地址(基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_CTRL	R/W	0x0050	模拟控制寄存器	0x00000000

## 6.5 寄存器定义

### 6.5.1 PMU\_DSLEEPEN 寄存器

为使能深睡眠模式，用户应该先将密码 0xAA5555AA 写入 PMU\_DSLEEPPASS 寄存器，然后写 0x55AAAA55 到 PMU\_DSLEEPEN 寄存器。在调试模式（模式为 0）时，不允许进入深睡眠模式，所有写入到 PMU\_DSLEEPEN 寄存器的值将被丢弃。如果芯片在深睡眠模式且模式从正常模式转换到调试模式，它会自动唤醒，使能 ICE 接口的访问。

表 6-4 PMU\_DSLEEPEN 寄存器

Bit	名称	类型	描述	默认值
31	WKU	R	当前唤醒信号状态。如果是 1，表明有唤醒信号没有清除，用户必须清除相应中断标志位，确保进入深睡眠模式之前该位为 0，否则系统将因未清除唤醒事件而立即唤醒。	0x0
30:0			保留	0

### 6.5.2 PMU\_DSLEEPPASS 寄存器

表 6-5 PMU\_DSLEEPPASS 寄存器

Bit	名称	类型	描述	默认值
31:1			保留	0
0	UNLOCK	R	该位表明深睡眠模式入口已解锁，并准备进入深睡眠模式。为解锁深睡眠模式，用户应该写 0xAA5555AA 到该寄存器。该位会在任意寄存器读或 PMU 寄存器写后立即清除，包括 ICE 的读/写。因而用户应立即在 PMU_DSLEEPEN 中设置正确的密码。	0x0

### 6.5.3 PMU\_CONTROL 寄存器

表 6-6 PMU\_CONTROL 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:8	PWUPCYC	R/W	上电周期计数，该寄存器控制收到唤醒时的上电等待时间。单位为 32K 时钟周期。 建议配置为 0。	0x0
7:6	-	-	保留	0
5	PLLL_SEL	R/W	低速 PLL 输入时钟选择。 0: 32K XTAL。 1: 32K RC。	0x0
4	PL LH_SEL	R/W	高速 PLL 输入时钟选择。 0: 6.5MHz RC。 1: 6.5536MHz XTAL。	0x0
3	INT_6M_EN	R/W	6.5536M XTAL 失效中断使能控制位。该位用于控制输出到 CPU 的中断信号。该位设置为 1 后，如果 6.5536M 晶振被移除或损坏，则会向 CPU 发出中断。	0x0
2	INT_32K_EN	R/W	32K XTAL 失效中断使能控制位。该位用于控制输出到 CPU 的中断信号。该位设置为 1 后，如果 32K 晶振被移除或损坏，则会向 CPU 发出中断。如果这个事件发生在睡眠期间，CPU 将被唤醒。	0x0

1	RTCCLK_SEL	R/W	RTC 时钟选择。 0: 32K XTAL。 1: 32K RC。	0x0
0	INT_IOA_EN	R/W	IOA0~15 中断使能控制位, 用来设置外部中断唤醒 IO 的使能控制。	0x0

## 6.5.4 PMU\_STS 寄存器

表 6-7 PMU\_STS 寄存器

Bit	名称	类型	描述	默认值
31:25	-	-	保留	0
24	MODE	R	该位 MODE 引脚当前状态。 0: Debug 模式。 1: Normal 模式。	--
23:7	-	-	保留	0
6	DPORST	R/C	该位表示上一次复位是否由内部数字上电复位信号引起, 只有第一次上电才会置 1。写 1 清零该位。	0x1
5	PORST	R/C	该位指示上一次复位是 PORH 复位或 PORL 复位, PORH 是 VDD 电压低于 2.08V 时发生的复位; PORL 是 DVCCLDO1 电压低于 1.3V 时发生的复位。写 1 清零该位。	0x1
4	EXTRST	R/C	该位指示上一次复位是否由外部复位信号引起。写 1 清零该位。	0x1
3	EXIST_6M	R	6.5536M XTAL 存在状态寄存器。该位表示 6.5536M XTAL 存在或不存在。开启 6.5536M XTAL (ANA_REG3 的 BIT7 配置为 1) 后, 该状态位才会刷新, 否则保持之前状态。 0: 6.5536M 晶体不存在。 1: 6.5536M 晶体存在。	0x0
2	EXIST_32K	R	32K XTAL 存在状态寄存器。该位表示 32K XTAL 存在或不存在。 0: 32K 晶体不存在。 1: 32K 晶体存在。	0x1
1	INT_6M	R/C	该位表示 6.5536M 晶振失效中断状态。当 EXIST_6M 状态位从 1 变成 0 的时候, 该状态位置 1。 当该位为 1 时, 表示 6.5536M 晶振被去除或损坏。 向此位写入 1 可将该标志清零。	0x0
0	INT_32K	R/C	该位表示 32K 晶振失效中断状态。当 EXIST_32K 状态位从 1 变成 0 的时候, 该状态位置 1。 当该位为 1 时, 表示 32K 晶体被去除或损坏。 向此位写入 1 可将该标志清零。	0x0

## 6.5.5 ANA\_CTRL 寄存器

表 6-8 ANA\_CTRL 寄存器

Bit	名称	类型	描述	默认值
-----	----	----	----	-----



26	PDNS2	R/W	<p>该位控制 VDDALARM 为 0（同时考虑 PDNS 的设置）时进入深睡眠的行为。</p> <p>0：当 VDDALARM 为 0 时不会进入深睡眠。当 VDDALARM 为 1 时，系统可以进入深睡眠模式，并且当 VDDALARM 变为 0 时，系统自动从深睡眠唤醒。</p> <p>1：VDDALARM 不作为深睡眠条件。</p>	0x0
6	PDNS	R/W	<p>该位控制 VDCINDROP 为 0（同时考虑 PDNS2 的设置）时进入深睡眠的行为。</p> <p>0：VDCINDROP 为 0 时不会进入深睡眠。当 VDCINDROP 为 1 时，系统可以进入深睡眠，并且当 VDCINDROP 变为 0 时，系统自动从深睡眠唤醒。</p> <p>1：VDCINDROP 的值，不作为进入深睡眠条件。</p>	0x0

## 6.6 复位

芯片由一种工作模式转为另一种工作模式时，某些硬件模块将自动复位，下表显示每个模块的复位详情。

表 6-9 各个模块在不同复位及睡眠唤醒后的复位情况

模块	外部复位	上电复位	看门狗复位	M0 软复位	唤醒前模式		
					深睡眠	浅睡眠	IDLE
Cortex-M0	√	√	√	√	√	-	-
系统 SRAM	-	-	-	-	数据丢失	-	-
掉电保持 SRAM	-	-	-	-	-	-	-
PMU(IOA)	√	√	√	-	-	-	-
WDT	√	√	√	-	-	-	-
RTC	√*	√*	√*	-	-	-	-
UART32K0	√	√	√	-	-	-	-
UART32K1	√	√	√	-	-	-	-
模拟控制器	√	√	√	-	-	-	-
LCD	√	√	√	√	√	-	-
GPIO(IOB~IOF)	√	√	√	√	√	-	-
MISC2	√	√	√	√	√	-	-
MISC	√	√	√	√	√	√	-
I <sup>2</sup> C	√	√	√	√	√	√	-
SPI	√	√	√	√	√	√	-
UART	√	√	√	√	√	√	-
ISO7816	√	√	√	√	√	√	-
TIMER	√	√	√	√	√	√	-
PWM	√	√	√	√	√	√	-
DMA	√	√	√	√	√	√	-
SRAM 控制器	√	√	√	√	√	√	-

FLASH 控制器	√	√	√	√	√	√	-
电能计量 EM	√	√	√	√	√	√	-

**注意 (√\*)**: RTC 带写保护寄存器不会被上电复位、外部复位、看门狗复位这三种复位源复位, 其他的 RTC 寄存器均可以被这三种唤醒源复位。

不同模式下, 时钟产生模块的使能或禁止由硬件或软件控制, 下表显示不同模式下的时钟状态。

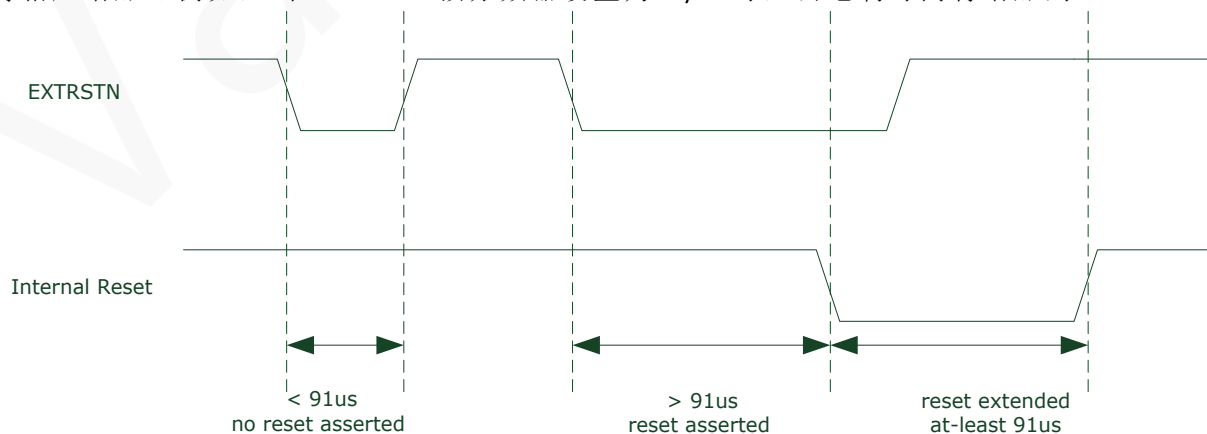
**表 6-10** 在不同模式下的各个时钟的工作状态

时钟源	电源模式			
	深睡眠	浅睡眠	IDLE	工作
6.5536M RC	OFF	OFF	由 RCHPD 控制	
6.5536M XTAL	OFF	OFF	由 XOHPDN 控制	
PLLH	OFF	OFF	由 PLLHPDN 控制	
PLLL	OFF	OFF	由 PLLLPDN 控制	
32K RC	ON	ON	ON	
32K XTAL	ON	ON	ON	

时钟源	不同模式唤醒		
	深睡眠	浅睡眠	IDLE
6.5536M RC	ON	由 RCHPD 控制	
6.5536M XTAL	OFF	由 XOHPDN 控制	
PLLH	OFF	由 PLLHPDN 控制	
PLLL	OFF	由 PLLLPDN 控制	
32K RC	ON	ON	
32K XTAL	ON	ON	

## 6.6.1 片外复位

片外复位引脚(EXTRSTN)能复位 V94XX(A)的大部分模块, 具体可参考表 6-9。为避免外部噪声耦合进此脚, V94XX(A)集成了一个去毛刺电路。下图显示去毛刺这个电路的工作原理。当 RTCCLK 预分频器设置为非零值时, 抗尖峰时间将相应增加。例如, 当 RTCCLK 预分频器设置为 1/4 时, 去毛刺时间将增加到  $91 \times 4 = 364\mu\text{s}$ 。



**图 6-2** 片外复位的去毛刺时序

## 6.6.2 看门狗复位

看门狗复位与上电复位等级相同，提供了芯片的内部复位机制。

## 6.6.3 上电复位

上电复位（POR）模块提供了芯片的内部复位机制，它与片外复位配合工作。

## 6.6.4 M0 软复位

M0 软复位指令可以复位 V94XX(A) 芯片的大多数模块，详情请参考表 6-9。

## 6.6.5 从浅睡眠/深睡眠/IDLE 下唤醒

任何的中断都能将芯片从 IDLE 状态唤醒，其中一些中断能从浅睡眠状态唤醒芯片，一些中断能从深睡眠状态唤醒芯片。

表 6-11 中断源

序列	向量地址	中断序号	描述	外设事件使能位	外设事件标记位	唤醒源	
						深睡眠	浅睡眠
NMI	00000008h	-14	NMI				
HardFault	0000000Ch	-13	HardFault				
SVCall	0000002Ch	-5	SVCall				
PendSV	00000038h	-2	PendSV				
SysTick	0000003Ch	-1	SysTick				
PMU	00000040h	0	IOA0~1	PMU_CONTROL.0 And PMU_IOAWKUEN. 0~1	PMU_IOAINTST S.0~1	V	V
			计量中断 0	PMU_CONTROL.0 And PMU_IOAWKUEN. 2	PMU_IOAINTST S.2	V	V
			IOA3~9	PMU_CONTROL.0 And PMU_IOAWKUEN. 3~9	PMU_IOAINTST S.3~9	V	V
			计量中断 1	PMU_CONTROL.0 And PMU_IOAWKUEN. 10	PMU_IOAINTST S.10	V	V
			IOA11~15	PMU_CONTROL.0	PMU_IOAINTST	V	V

				And PMU_IOAWKUEN. 11~15	S.11~15		
			32K 晶振失效	PMU_CONTROL.2	PMU_STS.0	V	V
			6M 晶振失效	PMU_CONTROL.3	PMU_STS.1		
RTC	00000044h	1	非法时间格式	RTC_INTEN.1	RTC_INTSTS.1	V	V
			多秒中断	RTC_INTEN.2	RTC_INTSTS.2	V	V
			多分钟中断	RTC_INTEN.3	RTC_INTSTS.3	V	V
			多小时中断	RTC_INTEN.4	RTC_INTSTS.4	V	V
			午夜（00：00）中断	RTC_INTEN.5	RTC_INTSTS.5	V	V
			32K 计数器中断	RTC_INTEN.6	RTC_INTSTS.6	V	V
			对 CE 寄存器的非法写入	RTC_INTEN.8	RTC_INTSTS.8		
U32K0 ~1	00000048h 0000004Ch	2\3	接收到数据	U32Kx_CTRL1.0	U32Kx_STS.0	V	V
			接收奇偶校验错误	U32Kx_CTRL1.1	U32Kx_STS.1	V	V
			接收缓存溢出	U32Kx_CTRL1.2	U32Kx_STS.2	V	V
I <sup>2</sup> C	00000050h	4	I <sup>2</sup> C 串行中断	I2C_CTRL2.0	I2C_CTRL.3		
SPI	000000ACh	27	SPI 发送	SPIx_TXSTS.14	SPIx_TXSTS.15		
			SPI 接收	SPIx_RXSTS.14	SPIx_RXSTS.15		
UART0/ 1/2/4/ 5	00000058h/ 0000005Ch/ 00000060h/ 00000068h/ 0000006Ch	6\7\8\ 10\11	接收中断	UARTx_CTRL.3	UARTx_INTSTS. 1		
			发送溢出	UARTx_CTRL.4	UARTx_INTSTS. 2		
			接收溢出	UARTx_CTRL.5	UARTx_INTSTS. 3		
			接收奇偶校验错误	UARTx_CTRL.7	UARTx_INTSTS. 4		
			发送完成	UARTx_CTRL.8	UARTx_INTSTS. 5		
ISO781 6	00000070h	12	接收中断	ISO7816_CFG.5	ISO7816_INFO. 5		
			发送中断	ISO7816_CFG.6	ISO7816_INFO. 6		
			接收溢出	ISO7816_CFG.7	ISO7816_INFO. 7		
Timer0 ~3	00000078h ~ 00000084h	14\15\ 16\17	定时器溢出	TMRx_CTRL.3	TMRx_INT.0		
PWM0 ~3	00000088h ~ 00000094h	18\19\ 20\21	PWM 定时器溢出	PWMx_CTL.1	PWMx_CTL.0		
			比较 0	PWMx_CCTL0.4	PWMx_CCTL0.0		

			比较 1	PWMx_CCTL1.4	PWMx_CCTL1.0		
			比较 2	PWMx_CCTL2.4	PWMx_CCTL2.0		
DMA	00000098h	22	通道 0 包结束	DMA_IE.0	DMA_STS.4		
			通道 1 包结束	DMA_IE.1	DMA_STS.5		
			通道 2 包结束	DMA_IE.2	DMA_STS.6		
			通道 3 包结束	DMA_IE.3	DMA_STS.7		
			通道 0 帧结束	DMA_IE.4	DMA_STS.8		
			通道 1 帧结束	DMA_IE.5	DMA_STS.9		
			通道 2 帧结束	DMA_IE.6	DMA_STS.10		
			通道 3 帧结束	DMA_IE.7	DMA_STS.11		
			通道 0 数据丢弃	DMA_IE.8	DMA_STS.12		
			通道 1 数据丢弃	DMA_IE.9	DMA_STS.13		
			通道 2 数据丢弃	DMA_IE.10	DMA_STS.14		
			通道 3 数据丢弃	DMA_IE.11	DMA_STS.15		
FLASH	0000009Ch	23	校验和错误	FLASH_CTRL.2	FLASH_INT.0		
ANA	000000A0h	24	手动 ADC 转换完成	ANA_INTEN.0	ANA_INTSTS.0		
			自动 ADC 转换完成	ANA_INTEN.1	ANA_INTSTS.1		
			COMP1 的上升和下降沿	ANA_INTEN.2	ANA_INTSTS.2	V	V
			COMP2 的上升和下降沿	ANA_INTEN.3	ANA_INTSTS.3	V	V
			VDDALARM 的上升和下降沿	ANA_INTEN.7	ANA_INTSTS.7	V	V
			VDCIN 的上升和下降沿	ANA_INTEN.8	ANA_INTSTS.8	V	V
			VDDL V 的上升和下降沿	ANA_INTEN.10	ANA_INTSTS.10	√	√
			VDCINDROP=0 同时进入了浅睡眠/深睡眠状态	ANA_INTEN.11	ANA_INTSTS.11	V	V
			ANA_REGx 错误	ANA_INTEN.12	ANA_INTSTS.12	V	V
			TADC 变化超过阈值	ANA_INTEN.13	ANA_INTSTS.13	V	V

## 6.7 工作模式的转换

下图描述了 V94XX(A) 各种工作模式的转换条件和状态。

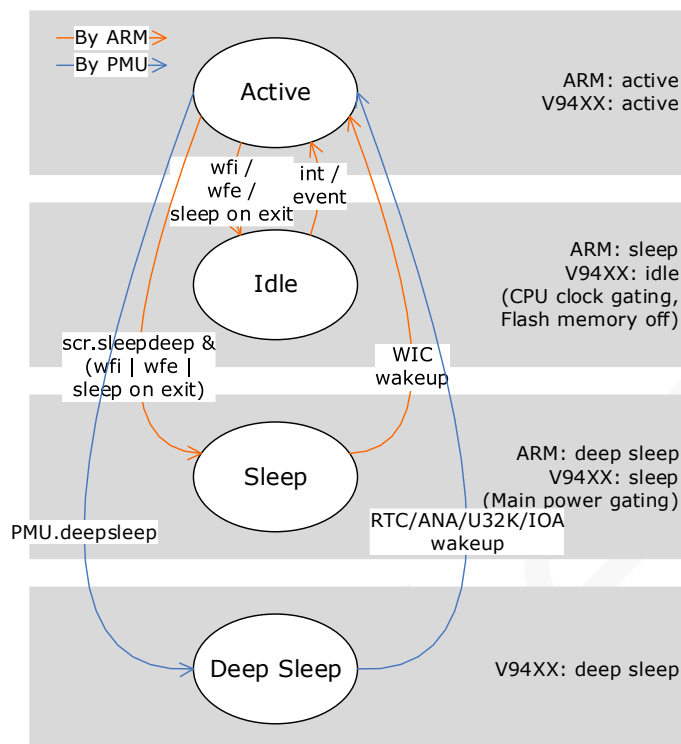


图 6-3 V94XX(A) 工作模式转换

根据上图，IDLE 模式或浅睡眠模式由 CPU 直接控制，用户只需要执行相应的指令（WFI 或 WFE）即可进入这两种模式。PMU 控制器控制深睡眠模式的进出，并使该模式在唤醒后总能返回到工作模式。进入深睡眠模式之前必须设置相应的唤醒源，这部分须由软件用户细致处理。以下显示了每种模式下的唤醒源。

注意：

（1）只有在 MODE=1 时，MCU 才可以进入深睡眠或浅睡眠模式；当 MODE=0 时，MCU 执行进入浅睡眠的指令后，会进入 IDLE 模式；进入深睡眠之前，应该首先配置 PDNS（ANA\_CTRL bit6）、PDNS2(ANA\_CTRL bit24)，或者检查 VDCINDROP 和 VDDALARM 状态，详情请参考 ANA\_CTRL 寄存器。

（2）当 MCU 从浅睡眠唤醒后，MCU 会自动切换系统时钟到进入浅睡眠之前的设置。例如，浅睡眠之前系统时钟为 PLLL，浅睡眠唤醒后，硬件会自动切换系统时钟到 PLLL。当 MCU 从深睡眠唤醒后，硬件会自动开启 RCH 作为系统时钟。IDLE 模式前后，硬件不会自动改变时钟状态。

（3）在进入浅睡眠模式时不能选用 RTCCLK 为系统时钟。系统时钟源由 CLKSEL(MISC2\_CTRL bit2:0)控制。

（4）在浅睡眠唤醒时，PLLL 锁定时间约为 1ms，PLLH 锁定时间为 15μs。

表 6-12 每一种工作模式下的唤醒源

序号	唤醒源	唤醒前模式		
		深睡眠	浅睡眠	IDLE
Event	-	-	-	-
NMI	SRAM 奇偶校验错误	-	-	V
0	PMU(ExtInt)	V	V	V
1	RTC	V	V	V
2	UART32K0	V	V	V
3	UART32K1	V	V	V

4	I <sup>2</sup> C	-	-	V
5	-	-	-	V
6	UART0	-	-	V
7	UART1	-	-	V
8	UART2	-	-	V
9	-	-	-	V
10	UART4	-	-	V
11	UART5	-	-	V
12	ISO7816	-	-	V
13	-	-	-	V
14	TIMER0	-	-	V
15	TIMER1	-	-	V
16	TIMER2	-	-	V
17	TIMER3	-	-	V
18	PWM0	-	-	V
19	PWM1	-	-	V
20	PWM2	-	-	V
21	PWM3	-	-	V
22	DMA	-	-	V
23	FLASH	-	-	V
24	ANA	V	V	V
27	SPI	-	-	V

## 6.8 应用笔记

### 6.8.1 外部 IO 唤醒

所有 16 个 IOA 都可以将系统从浅睡眠或深睡眠模式唤醒。V94XX(A)中有一个去毛刺电路，防止信号上的毛刺意外唤醒系统。

下图是去毛刺电路示意图。去毛刺时间为 4 个 RTCCLK 时钟周期，RTCCLK 不分频时 ( $RTC\_PSCA = 0$ )，抗干扰时间约为  $122\mu s$ 。当 RTCCLK 4 分频时 ( $RTC\_PSCA = 1$ )，去毛刺时间会增加到  $122*4=488\mu s$ 。去毛刺电路仅对浅睡眠、深睡眠模式的 IOA 边沿唤醒信号有效；对正常中断和浅睡眠、深睡眠的 IOA 电平唤醒信号无效。

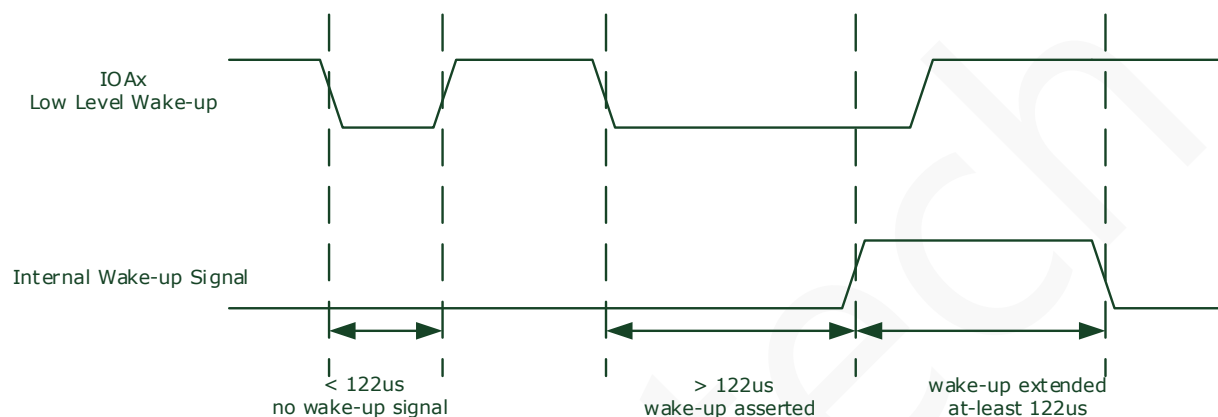


图 6-4 外部唤醒示意图



6.8.2 进入深睡眠模式的流程

下图显示了进入深睡眠模式的步骤。

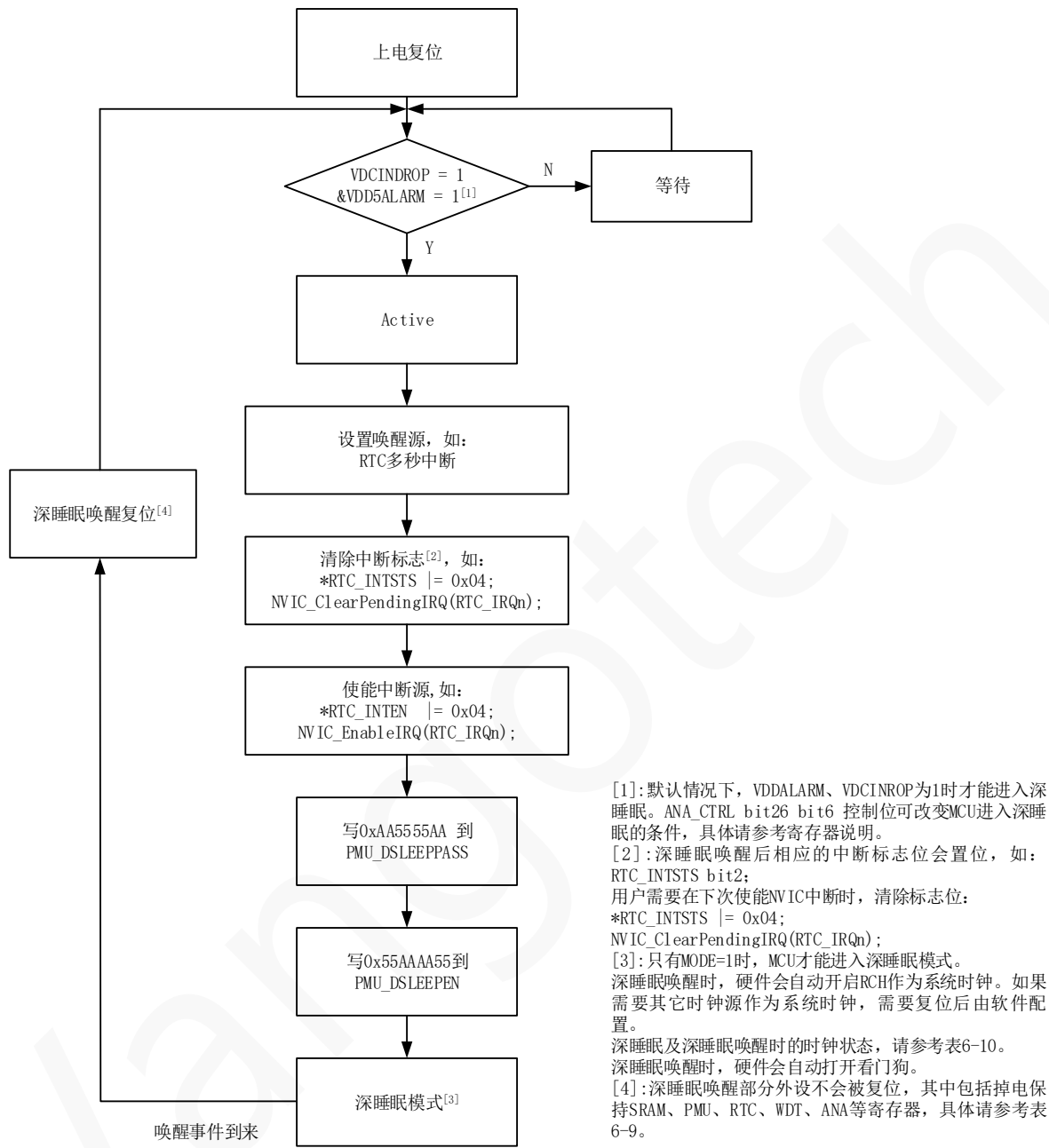
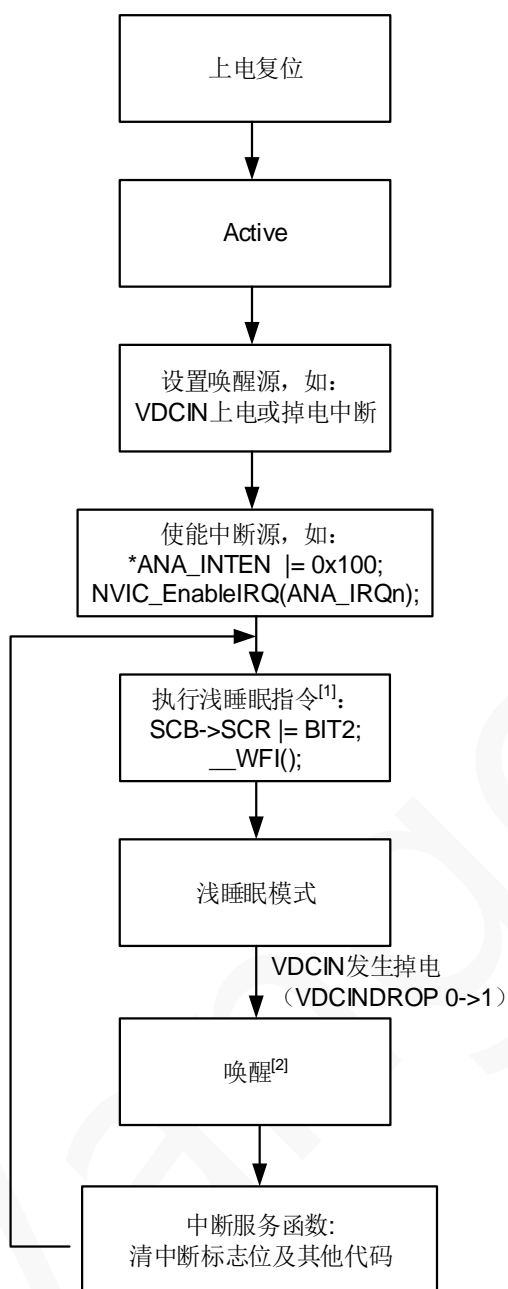


图 6-5 进入深睡眠的流程

### 6.8.3 进入浅睡眠的流程

下图描述了一个进入浅睡眠的步骤示例。



[1]: 只有在MODE=1时, MCU才可以浅睡眠模式; 当MODE=0时, MCU执行浅睡眠的指令后, 会进入Idle模式。

[2]: 浅睡眠唤醒后会复位部分外设, 包括UART、SPI、I2C等; LCD、GPIO、ANA等寄存器不会被复位, 具体请参考表6-9。

浅睡眠唤醒后, MCU会自动切换系统时钟到进入浅睡眠之前的设置。例如, 浅睡眠之前系统时钟为PLLL, 浅睡眠唤醒后, 硬件会自动切换系统时钟到PLLL。

浅睡眠及浅睡眠唤醒后的时钟状态, 请参考表6-10。

浅睡眠唤醒会自动打开看门狗。

注意: (1) 进入浅睡眠时不能选用RTCCLK为系统时钟。系统时钟源由CLKSEL(MISC2\_CTRL bit2:0)控制。

图 6-6 进入浅睡眠的流程

## 6.8.4 进入 IDLE 的流程

下图描述了进入 IDLE 模式的流程。

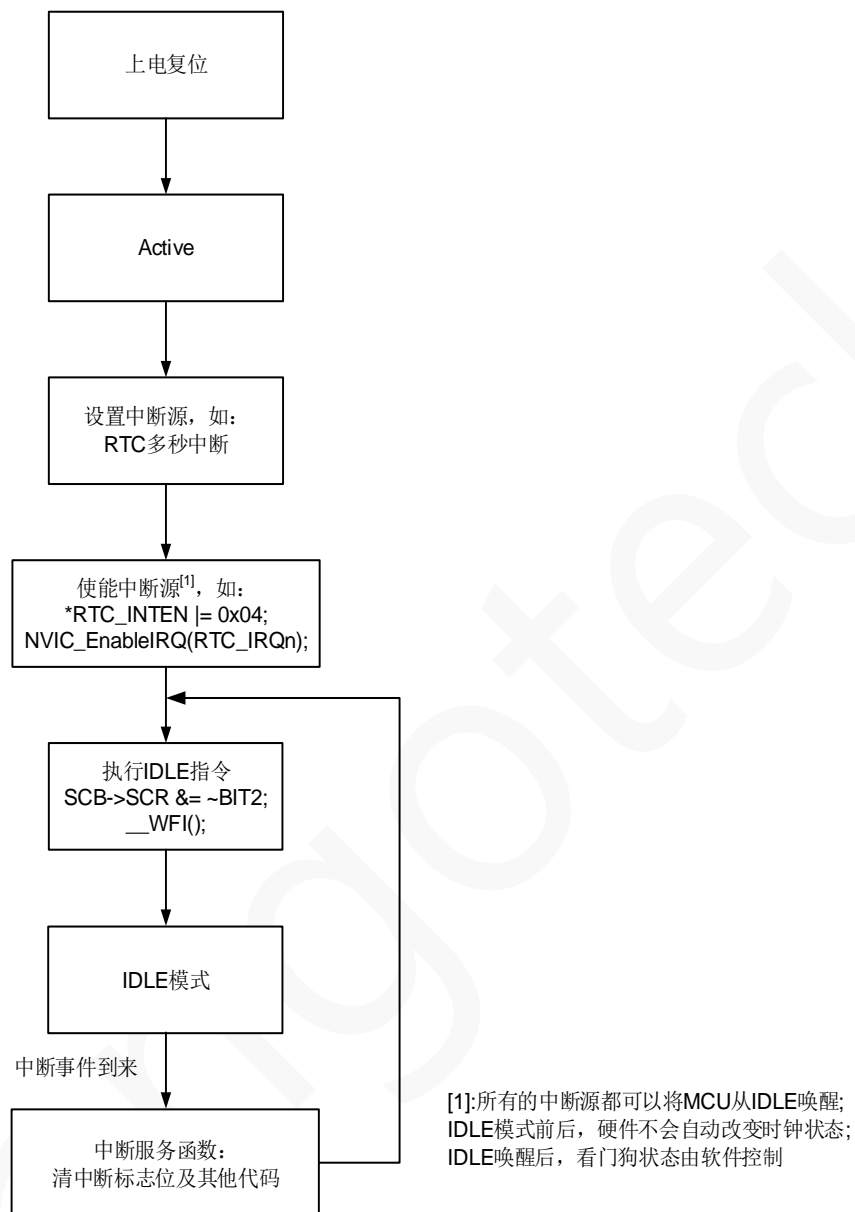


图 6-7 进入 IDLE 的流程

## 第7章 时钟系统

### 7.1 简介

时钟系统的相关设置在系统复位或深睡眠唤醒后会被复位，从深睡眠唤醒后用户需要手动恢复与时钟相关的寄存器设置。

在 V94XX(A)中，有四个时钟源：

--32K RC 振荡电路，用于产生 32.768K RC 时钟。该电路始终运行，且频率可以细调。

--32K 晶体振荡电路，用于产生 32.768K 的晶振时钟。32.768K 晶振时钟始终开启。32.768K 晶体振荡时钟由 32K RC 时钟监测，当其停止振荡时，32K RC 时钟将自动替换它，并且在晶振恢复工作之前，有检测电路始终监测晶振电路。

--6M RC 时钟振荡电路，用于产生 6.5536M RC 时钟。用户可以通过寄存器设置停止该电路的运行，该电路在芯片复位后自动开始运行。

--6M 晶体振荡电路，用于产生 6.5536M 晶振时钟。上电时默认关闭 6M 晶振时钟，用户需手动开启 6M 晶振时钟输出。

在 V94XX(A)中，有三个时钟域：

--HCLK 由 6.5536M 晶振时钟、6.5536M RC 时钟、32.768K 晶振时钟或 32.768K RC 时钟产生或它们的倍频产生，HCLK 为 CPU\FLASH\SRAM\DMA\GPIO\LCD\CRYPT 提供时钟。

--PCLK 由 HCLK 分频产生的，PCLK 为低速外设提供时钟。

--RTCCLK 由 32.768K 晶振时钟或 32.768K RC 时钟产生，为 RTC\WDT\UART32K\LCD 提供时钟。

表 7-1 不同工作模式下的时钟模块开关

时钟源	电源模式			
	深睡眠	浅睡眠	IDLE	工作
6.5536MRC	OFF	OFF	由 RCHPD 控制	
6.5536MXTAL	OFF	OFF	由 XOHPDN 控制	
PLLH	OFF	OFF	由 PLLHPDN 控制	
PLLL	OFF	OFF	由 PLLLPDN 控制	
32KRC	ON	ON	ON	
32KXTAL	ON	ON	ON	
时钟源	不同模式唤醒			
	深睡眠	浅睡眠	IDLE	
6.5536MRC	ON	由 RCHPD 控制		
6.5536MXTAL	OFF	由 XOHPDN 控制		
PLLH	OFF	由 PLLHPDN 控制		
PLLL	OFF	由 PLLLPDN 控制		
32K RC	ON	ON		
32K XTAL	ON	ON		

### 7.2 特点

--子模块的时钟控制

--AHBCLK 和 APBCLK 的时钟分频

--实时晶体检测。晶振检测电路工作时钟为 32K RC。实时检测高频晶体（默认关闭，需要手动开启）与低频晶体状态。当 6.5536M XTAL 停止振荡，6M RC 将自动替换 6.5536M XTAL，并且在 6.5536M XTAL 恢复之后，自动使用 6.5536M XTAL；当 32.768K XTAL 停止振荡，32K RC 将自动替换 32K XTAL，并且在 32.768K XTAL 恢复之后，自动使用 32.768K XTAL。

注意：HCLK 指的是 AHB 时钟，也叫做 AHBCLK；同样，PCLK 指的是 APB 时钟，也叫做 APBCLK。

## 7.3 功能框图

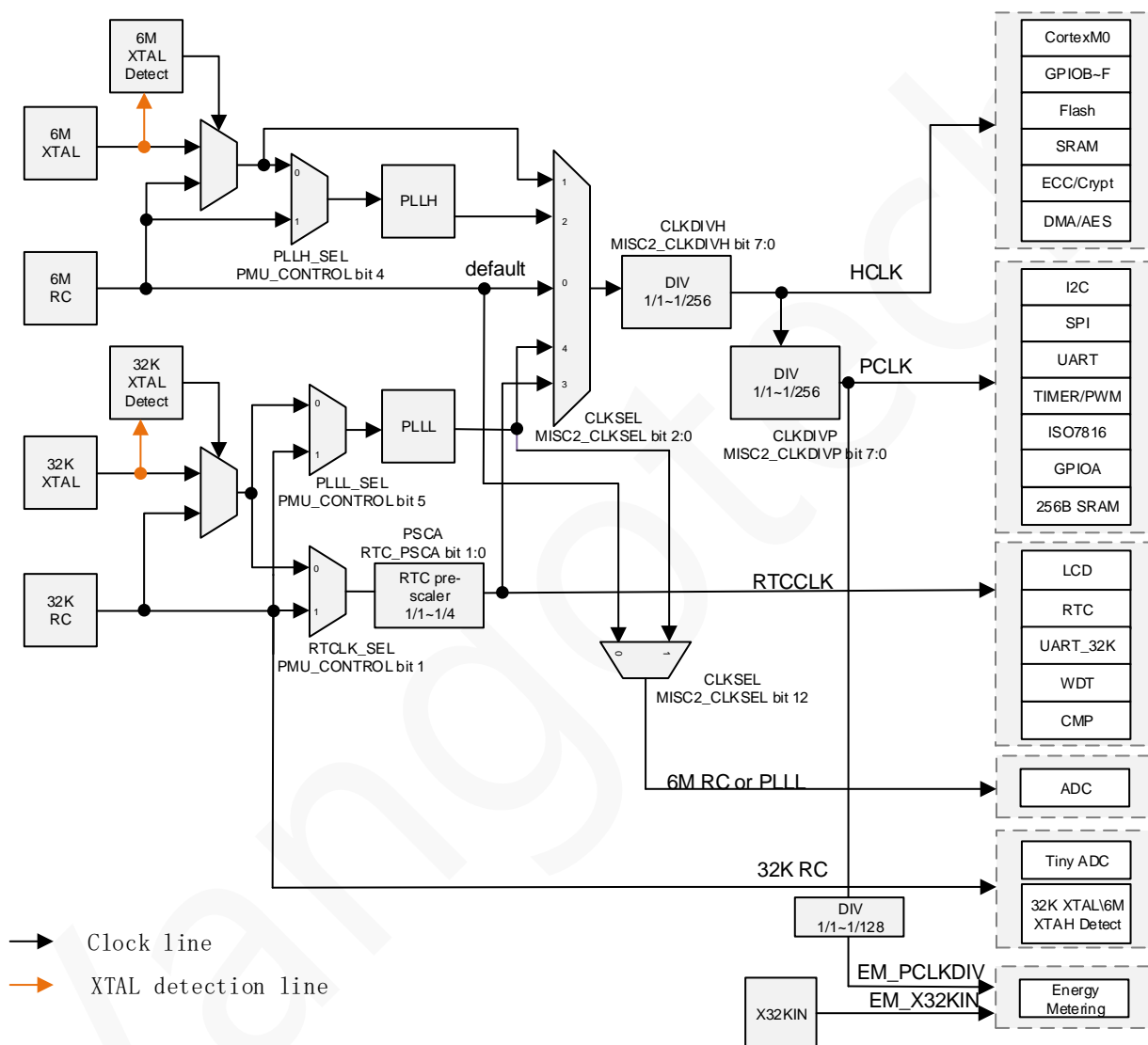


图 7-1 V94XX(A)时钟系统功能框图

## 7.4 寄存器地址

表 7-2 ANA 控制寄存器(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_REG2	R/W	0x0008	模拟寄存器 2	0x00
ANA_REG3	R/W	0x000C	模拟寄存器 3	0x00
ANA_REG9	R/W	0x0024	模拟寄存器 9	0x00
ANA_REGB	R/W	0x002C	模拟寄存器 11	来自 FLASH
ANA_REGC	R/W	0x0030	模拟寄存器 12	来自 FLASH

ANA_CMPOUT	R	0x0054	比较器结果寄存器	0x0030
------------	---	--------	----------	--------

表 7-3 PMU 寄存器地址(PMU 基地址: 0x40014000)

名称	类型	地址	描述	默认值
PMU_CONTROL	R/W	0x0008	PMU 控制寄存器	0x0000
PMU_STS	R/C	0x000C	PMU 状态寄存器	0x0000074

表 7-4 MISC2 控制寄存器(MISC2 基地址: 0x40013E00)

名称	类型	地址	描述	默认值
MISC2_CLKSEL	R/W	0x0004	时钟选择寄存器	0x0
MISC2_CLKDIVH	R/W	0x0008	AHB 时钟分频控制寄存器	0x00
MISC2_CLKDIVP	R/W	0x000C	APB 时钟分频控制寄存器	0x01
MISC2_HCLKEN	R/W	0x0010	AHB 时钟使能控制寄存器	0x1FF
MISC2_PCLKEN	R/W	0x0014	APB 时钟使能控制寄存器	0xFFFFFFFF

## 7.5 寄存器定义

### 7.5.1 ANA\_REG3 寄存器

表 7-5 ANA\_REG3 控制寄存器

位	名称	功能	注意事项
3	ADCBGPPD*	ADCBGP 掉电控制。	0: 开启 ADCBGP 1: 关闭 ADCBGP
4	RCHPD	高频 RC 时钟 RCH(6.5536MRC)开关控制信号。	0: 开启 RCH 1: 关闭 RCH
5	PLLLPDN	PLLL(32768Hz 输入 PLL)开关控制信号。	0: 关闭 PLLL 1: 开启 PLLL
6	PLLHPDN	PLLH(6.5536MHz 输入 PLL)开关控制信号。	0: 关闭 PLLH 1: 开启 PLLH
7	XOHPDN	6.5536M 晶振时钟开关控制信号。	0: 关闭 XOH 1: 开启 XOH

\*: RCH、PLL、ADC 与 ADCBGP 有关系, 在开 RCH、PLL、ADC 之前, 必须开启 ADCBGP。如果系统中没有使用 ADCBGP, 可通过 ADCBGPPD 位关闭 ADCBGP; 如果 ADCBGP 有被用到, 此时, 通过 ADCBGPPD 不能关闭 ADCBGP, 因为系统具有保护功能, 会重新开启 ADCBGP。浅睡眠或深睡眠模式下, 不管 ADCBGP 是否有被用到, 都可以关闭 ADCBGP。

### 7.5.2 ANA\_REG4 寄存器

表 7-6 ANA\_REG4 控制寄存器

位	名称	功能	注意事项
7: 0	-	保留	默认值是 0x00, 必须配置为 0x01。

### 7.5.3 ANA\_REG9 寄存器

表 7-7 ANA\_REG9 控制寄存器

位	名称	功能	注意事项
2: 0	PLLLSEL[2: 0]	选择 PLLL 的时钟频率。	000: 26.2144MHz 001: 13.1072MHz 010: 6.5536MHz 011: 3.2768MHz 100: 1.6384MHz 101: 0.8192MHz 110: 0.4096MHz 111: 0.2048MHz
6: 3	PLLHSEL[3: 0]	PLLH 的时钟频率控制位。PLLH 时钟频率是外部 XOH 或内部 RCH 的倍数。	1000~1011: 保留 1100: 2x 输入时钟频率 1101: 2.5x 输入时钟频率 1110: 3x 输入时钟频率 1111: 3.5x 输入时钟频率 0000: 4x 输入时钟频率 0001: 4.5x 输入时钟频率 0010: 5x 输入时钟频率 0011: 5.5x 输入时钟频率 0100: 6x 输入时钟频率 0101: 6.5x 输入时钟频率 0110: 7x 输入时钟频率 0111: 7.5x 输入时钟频率

### 7.5.4 ANA\_REGB 寄存器

表 7-8 ANA\_REGB 控制寄存器

位	名称	功能	注意事项
4: 0	RCLTRIM[4: 0]	32kHz RC 时钟频率微调。	00000~01111: 增加 4%/步长; 10000~11111: 减少 4%/步长
注意: 用户不要对寄存器 ANA_REGx (x=B~E) 进行配置, 这几个寄存器的值会从 FLASH 自动加载。			

### 7.5.5 ANA\_REGC 寄存器

表 7-9 ANA\_REGC 控制寄存器

位	名称	功能	注意事项
5: 0	RCHTRIM[5: 0]	6.5536MHzRC 时钟频率微调。	000000~011111: 增加 1.25%/步长; 100000~111111: 减少 1.25%/步长。
注意: 用户不要对寄存器 ANA_REGx (x=B~E) 进行配置, 这几个寄存器的值会从 FLASH 自动加载。			

## 7.5.6 ANA\_CMPOUT 寄存器

表 7-10 ANA\_CMPOUT 寄存器

Bit	名称	类型	描述	默认值
1	LOCKL	R	PLLL 锁定状态，锁定时间大约是 1ms。 0: PLLL 未锁定。 1: PLLL 已锁定。	0x0
0	LOCKH	R	PLLH 锁定状态，锁定时间大约是 15μs。 0: PLLH 未锁定。 1: PLLH 已锁定。	0x0

## 7.5.7 PMU\_CONTROL 寄存器

表 7-11 PMU\_CONTROL 寄存器

Bit	名称	类型	描述	默认值
5	PLLL_SEL	R/W	低速 PLL 输入时钟源选择。 0: 32K XTAL 1: 32K RC	0x0
4	PLLH_SEL	R/W	高速 PLL 输入时钟源选择。 0: 6.5MHz RC 1: 6.5536MHz XTAL	0x0
3	INT_6M_EN	R/W	6.5536M XTAL 失效中断使能寄存器。该位用于控制输出到 CPU 的中断信号。该位设置为 1 后，如果 6.5536M 晶振被移除或损坏，则会向 CPU 发出中断。	0x0
2	INT_32K_EN	R/W	32K XTAL 失效中断使能寄存器。该位用于控制输出到 CPU 的中断信号。该位设置为 1 后，如果 32K 晶振被移除或损坏，则会向 CPU 发出中断。如果这个事件发生在睡眠期间，CPU 将被唤醒。	0x0
1	RTCCLK_SEL	R/W	RTC 时钟选择。 0: 32K XTAL 1: 32K RC	0x0

## 7.5.8 PMU\_STS 寄存器

表 7-12 PMU\_STS 寄存器

Bit	名称	类型	描述	默认值
3	EXIST_6M	R	6.5536M XTAL 存在状态寄存器。该位表示 6.5536M XTAL 存在或不存在。开启 6.5536M XTAL (ANA_REG3 的 BIT7 配置为 1) 后，该状态位才会刷新，否则保持之前状态。 0: 6.5536M 晶体不存在。 1: 6.5536M 晶体存在。	0x0
2	EXIST_32K	R	32K XTAL 存在状态寄存器。该位表示 32K XTAL 存在或不存在。 0: 32K 晶体不存在。	0x1



			1: 32K 晶体存在。	
1	INT_6M	R/C	该位表示 6.5536M 晶振失效中断状态。当 EXIST_6M 状态位从 1 变成 0 的时候, 该状态位置 1。 当该位为 1 时, 表示 6.5536M 晶振被去除或损坏。 向此位写入 1 可将该标志清零。	0x0
0	INT_32K	R/C	该位表示 32K 晶振失效中断状态。当 EXIST_32K 状态位从 1 变成 0 的时候, 该状态位置 1。 当该位为 1 时, 表示 32K 晶体被去除或损坏。 向此位写入 1 可将该标志清零。	0x0

## 7.5.9 MISC2\_CLKSEL 寄存器

表 7-13 MISC2\_CLKSEL 寄存器

Bit	名称	类型	描述	默认值
31:3	-	-	保留	0
2:0	CLKSEL	R/W	该寄存器用于控制 AHB 时钟源。 0: RCH (6.5M RC)。 1: XOH (6.5M XTAL)。 2: PLLH。 3: RTCCLK (由 PMU_CONTROL 寄存器中的 RTCCLK_SEL 控制)。 4: PLLL。 在时钟选择到一个时钟源之前, 用户应首先通过设置 PMU_CONTROL 寄存器来使能相应的模块。	0x0

## 7.5.10 MISC2\_CLKDIVH 寄存器

表 7-14 MISC2\_CLKDIVH 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	CLKDIVH	R/W	该寄存器用于控制 AHB 时钟分频。 0: 时钟源除以 1 1: 时钟源除以 2 2: 时钟源除以 3 ... 255: 时钟源除以 256	0x00

## 7.5.11 MISC2\_CLKDIVP 寄存器

表 7-15 MISC2\_CLKDIVP 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	CLKDIVP	R/W	该寄存器用于控制 APB 时钟分频。	0x01

			0: AHB 时钟除以 1 1: AHB 时钟除以 2 2: AHB 时钟除以 3 ... 255: AHB 时钟除以 256	
--	--	--	---	--

## 7.5.12 MISC2\_HCLKEN 寄存器

表 7-16 MISC2\_HCLKEN 寄存器

Bit	名称	类型	描述	默认值
31:9	-	-	保留	0
8:0	HCLKEN	R/W	该寄存器用于控制每个 AHB 模块的时钟使能。对应的模块只有在其功能未被使用时才能被关闭。有关每个模块的详细信息，请参见下表。  0: 关闭; 1: 开启。	0x1FF

表 7-17 HCLK 时钟使能

位	模块	注意事项
0	--	禁止写 0
1	Arbiter&BusMatrix	禁止写 0
2	FLASH 控制器	禁止写 0
3	SRAM 控制器	禁止写 0
4	DMA 控制器	
5	GPIO 控制器	
6	LCD 控制器	
7	--	
8	CRYPT 控制器	

## 7.5.13 MISC2\_PCLKEN 寄存器

表 7-18 MISC2\_PCLKEN 寄存器

Bit	名称	类型	描述	默认值
31:0	PCLKEN	R/W	该寄存器用于控制每个 APB 模块的时钟使能。对应的模块只有在其功能未被使用时才能被关闭。有关每个模块的详细信息，请参见下表。  0: 关闭; 1: 开启。	0xFFFFFFFF

表 7-19 各个模块的 PCLK 时钟使能

Bit	Module	Note
0	AHB2APB Bridge	禁止写 0
1	DMA 控制器	

2	I <sup>2</sup> C	
3	保留	
4	UART0	
5	UART1	
6	UART2	
7	保留	
8	UART4	
9	UART5	
10	ISO7816	
11	保留	
12	Timer	
13	MISC	
14	MISC2	
15	PMU	
16	RTC	
17	ANA	
18	U32K0	
19	U32K1	
20	保留	
21	SPI	
31:22	保留	

## 第8章 模拟控制器

### 8.1 简介

模拟控制器用于控制 V94XX(A)的模拟功能。所有模拟相关控制，如 ADC、比较器和其他模拟监测都由本模块控制。模拟控制器总是处于上电状态，因此所有模拟设置在深睡眠模式下仍会保留，不会被复位。

### 8.2 特点

- 产生模拟监测中断/唤醒信号。
- 产生比较器中断。
- 比较器计数器。
- ADC 手动和自动采样模式。
- 产生 ADC 中断。

### 8.3 功能框图

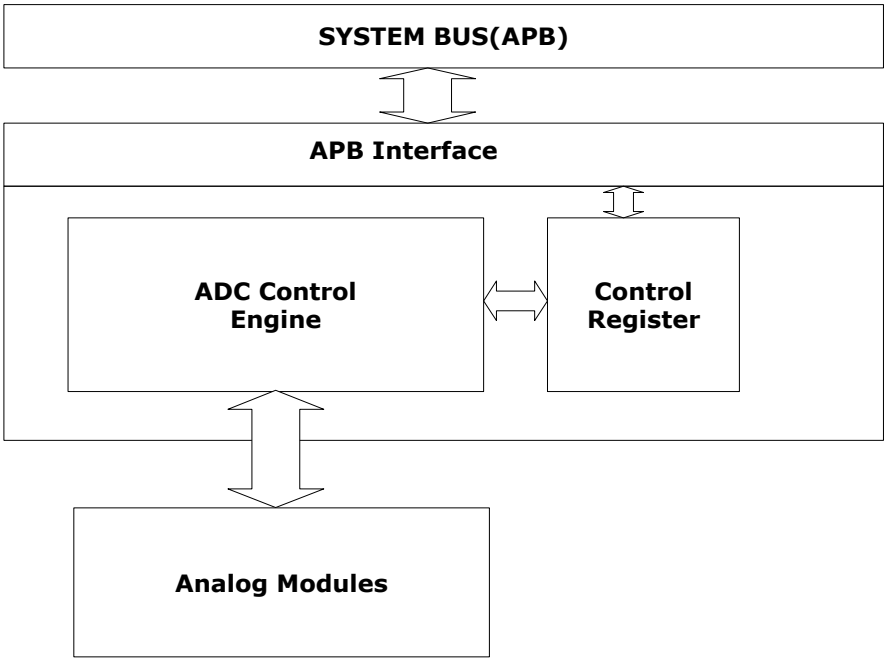


图 8-1 模拟控制器的功能框图

### 8.4 寄存器地址

表 8-1 ANA 控制寄存器(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_REG0	R/W	0x0000	模拟寄存器 0	0x00
ANA_REG1	R/W	0x0004	模拟寄存器 1	0x00
ANA_REG2	R/W	0x0008	模拟寄存器 2	0x00
ANA_REG3	R/W	0x000C	模拟寄存器 3	0x00

ANA_REG4	R/W	0x0010	模拟寄存器 4	0x00
ANA_REG5	R/W	0x0014	模拟寄存器 5	0x00
ANA_REG6	R/W	0x0018	模拟寄存器 6	0x00
ANA_REG7	R/W	0x001C	模拟寄存器 7	0x00
ANA_REG8	R/W	0x0020	模拟寄存器 8	0x00
ANA_REG9	R/W	0x0024	模拟寄存器 9	0x00
ANA_REGA	R/W	0x0028	模拟寄存器 10	0x00
ANA_REGB	R/W	0x002C	模拟寄存器 11	FromFLASH
ANA_REGC	R/W	0x0030	模拟寄存器 12	FromFLASH
ANA_REGD	R/W	0x0034	模拟寄存器 13	FromFLASH
ANA_REGE	R/W	0x0038	模拟寄存器 14	FromFLASH
ANA_REGF	R/W	0x003C	模拟寄存器 15	0x00
ANA_CTRL	R/W	0x0050	模拟控制寄存器	0x00000000
ANA_CMPOUT	R	0x0054	比较器结果寄存器	0x0030
ANA_INTSTS	R/C	0x0060	模拟中断状态寄存器	0x0000
ANA_INTEN	R/W	0x0064	模拟中断使能寄存器	0x0000
ANA_ADCCTRL	R/W	0x0068	ADC 控制寄存器	0x00000000
ANA_ADCDATA0	R	0x0070	ADC 通道 0 数据寄存器	--
ANA_ADCDATA1	R	0x0074	ADC 通道 1 数据寄存器	--
ANA_ADCDATA2	R	0x0078	ADC 通道 2 数据寄存器	--
ANA_ADCDATA3	R	0x007C	ADC 通道 3 数据寄存器	--
ANA_ADCDATA4	R	0x0080	ADC 通道 4 数据寄存器	--
ANA_ADCDATA5	R	0x0084	ADC 通道 5 数据寄存器	--
ANA_ADCDATA6	R	0x0088	ADC 通道 6 数据寄存器	--
ANA_ADCDATA7	R	0x008C	ADC 通道 7 数据寄存器	--
ANA_ADCDATA8	R	0x0090	ADC 通道 8 数据寄存器	--
ANA_ADCDATA9	R	0x0094	ADC 通道 9 数据寄存器	--
ANA_ADCDATAA	R	0x0098	ADC 通道 10 数据寄存器	--
ANA_ADCDATAB	R	0x009C	ADC 通道 11 数据寄存器	--
ANA_CMPCNT1	R/C	0x00B0	比较器 1 计数器	0x00000000
ANA_CMPCNT2	R/C	0x00B4	比较器 2 计数器	0x00000000
ANA_MISC	R/W	0x00B8	模拟 misc.控制寄存器	0x00

## 8.5 寄存器定义

### 8.5.1 ANA\_REGx 寄存器

表 8-2 ANA\_REGx

	Bit								
	7	6	5	4	3	2	1	0	地址

ANA_REG0	-	-	-	-	-	-	0x0000
ANA_REG1	-	-	GDE4	RESDIV	-	-	0x0004
ANA_REG2	-	-	REFSEL_CMP2	REFSEL_CMP1	CMP2_SEL[1:0]	CMP1_SEL[1:0]	0x0008
ANA_REG3	XOHPDN	PLLHPDN	PLLLPDN	RCHPD	ADCBGPP D	CMP2PDN CMP1PDN	0x000C
ANA_REG4	-	-	-	-	-	-	0x0010
ANA_REG5	-	-	-	-	IT_CMP2[1:0]	IT_CMP1[1:0]	0x0014
ANA_REG6	BAT2DISC	BAT1DISC	-	-	VLCD[3:0]	LCD_BMOD	0x0018
ANA_REG7	-	-	-	-	-	-	0x001C
ANA_REG8	-	-	VDDPVDSEL[2:0]	-	-	-	0x0020
ANA_REG9	PD_VDDDET	-	PLLSSEL[3:0]	-	-	PLLSSEL[2:0]	0x0024

ANA_REGA	PD_VDCINDET	-	-	-	-	-	-	-	0x0028
ANA_REGB		-	-	RCLTRIM[4:0]					0x002C
ANA_REGC	-	-	RCHTRIM[5:0]						0x0030
ANA_REGD	-								0x0034
ANA_REGE	-								0x0038
ANA_REGF	ADTREF3_SEL	ADTREF2_SEL	ADTREF1_SEL	SEL_ADT	PDN_ADT	VDDO_EN	-		0x003C

## 8.5.2 ANA\_REG0 寄存器

表 8-3 ANA\_REG0 各个 bit 的功能描述

Bit	名称	功能	描述
7:0	REG0_RSVD	保留	0

## 8.5.3 ANA\_REG1 寄存器

表 8-4 ANA\_REG1 各个 bit 的功能描述

Bit	名称	功能	描述
3:0	-	保留	0
4	RESDIV	使能 ADC 输入信号的电阻分压	0: 不使能 1: 使能 1/4 电阻分压
5	GDE4	使能 ADC 输入信号的电容分压	0: 不使能 1: 使能 1/4 电容分压

7:6	-	保留	0
-----	---	----	---

## 8.5.4 ANA\_REG2 寄存器

表 8-5 ANA\_REG2 各个 bit 的功能描述

Bit	名称	功能	描述
1:0	CMP1_SEL[1: 0]	CMP1 信号输入源选择	00: CMP1_P 与 REF 01: CMP1_N 与 REF 1*: CMP1_P 与 CMP1_N
3:2	CMP2_SEL[1: 0]	CMP2 信号输入源选择	00: CMP2_P 与 REF 01: CMP2_N 与 REF 1*: CMP2_P 与 CMP2_N
4	REFSEL_CMP1	CMP1 的 REF 选择	0: LPREF 1: ADCREF
5	REFSEL_CMP2	CMP2 的 REF 选择	0: LPREF 1: ADCREF
7:6	-	保留	0

## 8.5.5 ANA\_REG3 寄存器

表 8-6 ANA\_REG3 各个 bit 的功能描述

Bit	名称	功能	描述
0	ADCPDN	ADC 开关控制信号。ADC 关闭之前需要先禁止 ADC 自动转换。	0: 关闭 ADC 1: 开启 ADC
1	CMP1PDN	CMP1 开关控制信号	0: 关闭 CMP1 1: 开启 CMP1
2	CMP2PDN	CMP2 开关控制信号	0: 关闭 CMP2 1: 开启 CMP2
3	ADCBGPPD*	ADCBGP 开关控制信号	0: 开启 ADCBGP 1: 关闭 ADCBGP
4	RCHPD	高频 RC 时钟电路 RCH (6.5536MRC) 开关控制信号	0: 开启 RCH 1: 关闭 RCH
5	PLLLPDN	PLLL (32768Hz 输入 PLL) 开关控制信号	0: 关闭 PLLL 1: 开启 PLLL
6	PLLHPDN	PLLH (6.5536MHz 输入 PLL) 开关控制信号	0: 关闭 PLLH 1: 开启 PLLH
7	XOHPDN	6.5536M 晶体振荡电路开关控制信号	0: 关闭 XOH 1: 开启 XOH

\*: RCH、PLL、ADC 与 ADCBGP 有关系, 在开 RCH、PLL、ADC 之前, 必须开启 ADCBGP。如果系统中没有使用 ADCBGP, 可通过 ADCBGPPD 位关闭 ADCBGP; 如果 ADCBGP 有被用到, 此时, 通过 ADCBGPPD 不能关闭 ADCBGP, 因为系统具有保护功能, 会重新开启 ADCBGP。浅睡眠或深睡眠模式下, 不管 ADCBGP 是否有被用到, 都可以关闭 ADCBGP。



## 8.5.6 ANA\_REG4 寄存器

表 8-7 ANA\_REG4 各个 bit 的功能描述

Bit	名称	功能	描述
7:0	REG4_RSVD	保留	默认值是 0x00，必须配置为 0x01。

## 8.5.7 ANA\_REG5 寄存器

表 8-8 ANA\_REG5 各个 bit 的功能描述

Bit	名称	功能	描述
1:0	IT_CMP1[1: 0]	CMP1 偏置电流选择	00: 20nA; 01: 100nA; 1*: 500nA;
3:2	IT_CMP2[1: 0]	CMP2 偏置电流选择	00: 20nA; 01: 100nA; 1*: 500nA;
5:4	-	保留	0
6	VDDLVDETPD	VDD 低电压检测模块电源开关	0: 使能 1: 禁止
7	-	保留	0

## 8.5.8 ANA\_REG6 寄存器

表 8-9 ANA\_REG6 各个 bit 的功能描述

Bit	名称	功能	描述
0	LCD_BMODE	LCD 偏置模式选择	0: 1/3 偏置 1: 1/4 偏置
4:1	VLCD[3:0]	LCD 驱动电压设置	当 VLCD=0: Default 当 VLCD=0~5: 调整幅度=+60mV*VLCD 当 VLCD=6~15 : 调整幅度=-60mV*(VLCD-5)
5	-	保留	0
6	BAT1DISC	给 BAT1 电池放电 放电电阻是 1.7k，放电电流是 Vbat1/1.7k。	1: 使能从 BAT1 到 GND 的 1.7k 电阻
7	BAT2DISC	给 BAT2 电池放电 放电电阻是 1.7k，放电电流是 Vbat2/1.7k。	1: 使能从 BAT2 到 GND 的 1.7k 电阻

## 8.5.9 ANA\_REG7 寄存器

表 8-10 ANA\_REG7 各个 bit 的功能描述

Bit	名称	功能	描述
7:0	REG7_RSVD	保留	默认值为 0，必须配置为 0。

## 8.5.10 ANA\_REG8 寄存器

表 8-11 ANA\_REG8 各个 bit 的功能描述

Bit	名称	功能	描述
3:0	-	保留	0
6:4	VDDPVDSEL[2:0]	VDD 电源的低电压监测阈值设置寄存器，设置将影响 VDDALARM 的状态。	000: 保留 001: 保留 010: 保留 011: 3.6V 100: 3.2V 101: 2.9V 110: 2.6V 111: 2.3V
7	REGB_RSVD	保留	默认值为 0，必须配置为 1

## 8.5.11 ANA\_REG9 寄存器

表 8-12 ANA\_REG9 各个 bit 的功能描述

Bit	名称	功能	描述
2:0	PLLLSEL[2:0]	PLLL 频率选择。	000: 26.2144MHz 001: 13.1072MHz 010: 6.5536MHz 011: 3.2768MHz 100: 1.6384MHz 101: 0.8192MHz 110: 0.4096MHz 111: 0.2048MHz
6:3	PLLHSEL[3:0]	PLLH 频率选择。PLLH 的频率是外部 XOH 或内部 RCH 的倍数。	1000~1011: 保留 1100: 2x 输入时钟频率 1101: 2.5x 输入时钟频率 1110: 3x 输入时钟频率 1111: 3.5x 输入时钟频率 0000: 4x 输入时钟频率 0001: 4.5x 输入时钟频率 0010: 5x 输入时钟频率 0011: 5.5x 输入时钟频率 0100: 6x 输入时钟频率 0101: 6.5x 输入时钟频率

			0110: 7x 输入时钟频率 0111: 7.5x 输入时钟频率
7	PD_VDDDET	关闭 VDD 电压监测模块, 该模块由 VDD 供电。	0: 开启。 1: 关闭。

## 8.5.12 ANA\_REGA 寄存器

表 8-13 ANA\_REGA 各个 bit 的功能描述

Bit	名称	功能	描述
6:0	REGA_RSVD	保留	默认值是 0x0, 必须配置为 0x0A。
7	PD_VDCINDET	关闭 VDCIN 监测模块	0: 开启 1: 关闭

## 8.5.13 ANA\_REGB 寄存器

表 8-14 ANA\_REGB 各个 bit 的功能描述

Bit	名称	功能	描述
4:0	RCLTRIM[4:0]	32kHz RC 频率调节	00000~01111: 频率以 4% 的步长增加; 10000~11111: 频率以 4% 的步长减小
7:5	LPREFTRIM[2:0]	LPREF 电压调节, 它将同步影响 DVCC1 的输出电压值	011: +9%; 010: +6%; 001: +3%; 000: 0%; 100: -3%; 101: -6%; 110: -9%; 111: -12%;

注意: 用户不要对寄存器 ANA\_REGx (x=B~E) 进行配置, 这几个寄存器的值会从 FLASH 自动加载。

## 8.5.14 ANA\_REGC 寄存器

表 8-15 ANA\_REGC 各个 bit 的功能描述

Bit	名称	功能	描述
5:0	RCHTRIM[5:0]	6.5536MHz RC 频率调节	000000~011111: 频率以 1.25% 的步长增加; 100000~111111: 频率以 1.25% 的步长减小
7:6	-	保留	0

注意: 用户不要对寄存器 ANA\_REGx (x=B~E) 进行配置, 这几个寄存器的值会从 FLASH 自动加载。

## 8.5.15 ANA\_REGD 寄存器

表 8-16 ANA\_REGD 各个 bit 的功能描述

Bit	名称	功能	描述
7:0	REGD_RSVD	保留	0

注意：用户不要对寄存器 ANA\_REGx (x=B~E) 进行配置，这几个寄存器的值会从 FLASH 自动加载。

## 8.5.16 ANA\_REGE 寄存器

表 8-17 ANA\_REGE 各个 bit 的功能描述

Bit	名称	功能	描述
7:0	REGE_RSVD	保留	0

注意：用户不要对寄存器 ANA\_REGx (x=B~E) 进行配置，这几个寄存器的值会从 FLASH 自动加载。

## 8.5.17 ANA\_REGF 寄存器

表 8-18 ANA\_REGF 各个 bit 的功能描述

Bit	名称	功能	描述
1:0	-	保留	0
2	VDDO_EN	VDD_OUT 引脚输出使能控制	0: 输出高阻 1: 允许 VDD_OUT 管脚输出内部的 VDD 电源电压
3	PDN_ADT	Tiny ADC 开关控制	0: 关闭 1: 开启
4	SEL_ADT	Tiny ADC 信号输入源选择	0: 连接到 IOE6 1: 连接到 IOE7
5	ADTREF1_SEL	ADTREF1 电压选择	0: 0.9V 1: 0.7V
6	ADTREF2_SEL	ADTREF2 电压选择	0: 1.8V 1: 1.6V
7	ADTREF3_SEL	ADTREF3 电压选择	0: 2.7V 1: 2.5V

## 8.5.18 ANA\_CTRL 寄存器

表 8-19 ANA\_CTRL 寄存器

Bit	名称	类型	描述	默认值
31:27	-	-	保留	0
26	PDNS2	R/W	该位控制 VDDALARM 为 0（同时考虑 PDNS 的设置）时进入深睡眠的行为。 0: 当 VDDALARM 为 0 时不会进入深睡眠。当 VDDALARM 为 1 时，系统可以进入深睡眠模式，并且当 VDDALARM 变为 0 时，系统自动从深睡眠唤醒。	0x0

			1: VDDALARM 不作为深睡眠条件。	
25:24	VDCINDEB	R/W	<p>VDCIN 去抖控制寄存器。</p> <p>0: 无去抖</p> <p>1: 2 个 RTCCLK 时钟周期消抖。</p> <p>2: 3 个 RTCCLK 时钟周期消抖。</p> <p>3: 4 个 RTCCLK 时钟周期消抖。</p> <p>当使能消抖时, 只有当信号在 RTCCLK 时钟的多周期内没有变化时, 输入信号才有效。此外, 唤醒和中断的响应时间将被延迟, 直到信号有效。该电路可以在所有模式下工作, 包括浅睡眠和深睡眠模式。</p>	0x0
23:22	CMP2DEB	R/W	<p>比较器 2 消抖控制寄存器。</p> <p>0: 不消抖。</p> <p>1: 2 个 RTCCLK 时钟周期消抖。</p> <p>2: 3 个 RTCCLK 时钟周期消抖。</p> <p>3: 4 个 RTCCLK 时钟周期消抖。</p> <p>当使能消抖时, 只有当信号在 RTCCLK 时钟的多周期内没有变化时, 输入信号才有效。此外, 唤醒和中断的响应时间将被延迟, 直到信号有效。该电路可以在所有模式下工作, 包括浅睡眠和深睡眠模式。</p>	0x0
21:20	CMP1DEB	R/W	<p>比较器 1 消抖控制寄存器。</p> <p>0: 不消抖。</p> <p>1: 2 个 RTCCLK 时钟周期消抖。</p> <p>2: 3 个 RTCCLK 时钟周期消抖。</p> <p>3: 4 个 RTCCLK 时钟周期消抖。</p> <p>当使能消抖时, 只有当信号在 RTCCLK 时钟的多周期内没有变化时, 输入信号才有效。此外, 唤醒和中断的响应时间将被延迟, 直到信号有效。该电路可以在所有模式下工作, 包括浅睡眠和深睡眠模式。</p>	0x0
19:16	-	-	保留	0
15:8	RCHTGT	R/W	高频时钟 RCH 自动校正频率设置。这个寄存器用于存储用户设置的高频时钟目标频率值, 如果用户要设置的目标频率值是 6.5536MHz 那么用户需要 在该寄存器中填入 $6553600/32768=200$ 。	0x0
7	-	-	保留	0
6	PDNS	R/W	<p>该位控制 VDCINDROP 为 0 (同时考虑 PDNS2 的设置) 时进入深睡眠的行为。</p> <p>0: VDCINDROP 为 0 时不会进入深睡眠。当 VDCINDROP 为 1 时, 系统可以进入深睡眠, 并且当 VDCINDROP 变为 0 时, 系统自动从深睡眠唤醒。</p> <p>1: 不管 VDCINDROP 的值, 都能进入深睡眠</p>	0x0
5:4	-	-	保留	0
3:2	COMP2_SEL	R/W	<p>该寄存器用于控制比较器 COMP2 的中断和唤醒。</p> <p>0: 关闭</p> <p>1: 上升沿</p>	0x0

			2: 下降沿 3: 上下沿	
1:0	COMP1_SEL	R/W	该寄存器用于控制比较器 COMP1 的中断和唤醒。 0: 关闭 1: 上升沿 2: 下降沿 3: 上下沿	0x0

## 8.5.19 ANA\_CMPOUT 寄存器

表 8-20 ANA\_CMPOUT 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:14	TADCO	R	TinyADC 输出。	-
13:9	-	-	保留	0
8	VDCINDROP	R	VDCIN 掉电状态。 0: VDCIN 高于阈值电压，没有掉电。 1: VDCIN 低于阈值电压，发生掉电。	0x0
7	VDDALARM	R	电源 VDD 电压预警信号 VDDALARM 输出。 0: 电源 VDD 电压高于 VDDPVDSEL 设置的值。 1: 电源 VDD 电压低于 VDDPVDSEL 设置的值。	0x0
6:4	-	-	保留	0
3	COMP2	R	比较器 2 输出。	0x0
2	COMP1	R	比较器 1 输出。	0x0
1	LOCKL	R	PLLL 锁定状态，锁定时间大约是 1ms。 0: PLLL 未锁定。 1: PLLL 已锁定。	0x0
0	LOCKH	R	PLLH 锁定状态，锁定时间大约是 15μs。 0: PLLH 未锁定。 1: PLLH 已锁定。	0x0

## 8.5.20 ANA\_INSTS 寄存器

表 8-21 ANA\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
31:14	-	-	保留	0
13	INTSTS13	R/C	TADC 过阈值中断。TADC 在上升或下降时，并且相比上一周期的变化值大于等于阈值（TADCTH），中断标志位将会置位。 读 0: 无过阈值中断 读 1: 发生过阈值中断 写 0: 无效	0x0

			写 1: 清零	
12	INTSTS12	R/C	模拟控制寄存器 ANA_REGx 错误标记位。芯片内部自动对所有模拟寄存器 ANA_REGx 进行校验和校验和奇偶校验, 当外部干扰引起模拟寄存器 ANA_REGx 被错误改写时, 该标记位将置 1。 读 0: 未发生中断 读 1: 发生中断 写 0: 无影响 写 1: 清除该位	0x0
11	INTSTS11	R/C	当 VDCINDROP 为 0, 进入睡眠模式中断标志位 (VDCIN 大于阈值)。当 VDCINDROP 为 0, 并且检测到系统进入睡眠的动作, 将置位中断标志位。用户可以使能中断使能位 (INTEN11), 使得系统可以从浅睡眠或深睡眠模式唤醒 (VDCINDROP 为 0)。 读 0: 无进入睡眠事件发生 (VDCINDROP=0) 读 1: 发生进入睡眠事件 (VDCINDROP=0) 写 0: 无影响 写 1: 清除该位	0x0
10	INTSTS10	R/C	VDDLVL 的中断标志, 该中断会在 VDDLVL 上升和下降时产生。 读 0: 无 VDDLVL 中断 读 1: 发生 VDDLVL 中断 写 0: 无效 写 1: 清除该位	0x0
9	-	-	保留	0
8	INTSTS8	R/C	VDCINDROP 的中断标志, 该中断会在 VDCINDROP 上升和下降时产生。 读 0: 无 VDCIN 中断 读 1: 发生 VDCIN 中断 写 0: 无效 写 1: 清零	0x0
7	INTSTS7	R/C	VDDALARM 的中断标志, 该中断会在 VDDALARM 上升和下降时产生。 读 0: 无 VDDALARM 中断 读 1: 发生 VDDALARM 中断 写 0: 无效 写 1: 清零	0x0
6:4	-	-	保留	0
3	INTSTS3	R/C	COMP2 的中断标志, 中断产生条件受 COMP2_SEL 控制。读 读 0: 无 COMP2 中断 读 1: 发生 COMP2 中断 写 0: 无效 写 1: 清零	0x0
2	INTSTS2	R/C	COMP1 的中断标志, 中断产生条件受 COMP1_SEL 控制。读 读 0: 无 COMP1 中断	0x0

			读 1: 发生 COMP1 中断 写 0: 无效 写 1: 清零	
1	INTSTS1	R/C	ADC 自动转换完成或 ADC 转换发生硬件超时的中断标记位。 读 0: 未发生中断 读 1: 发生中断 写 0: 无影响 写 1: 清除该位	0x0
0	INTSTS0	R/C	手动设置 ADC 转换完成或 ADC 转换发生硬件超时的中断标记位。 读 0: 未发生中断 读 1: 发生中断 写 0: 无影响 写 1: 清除该位	0x0

## 8.5.21 ANA\_INTEN 寄存器

表 8-22 ANA\_INTEN 寄存器

Bit	名称	类型	描述	默认值
31:14	-	-	保留	0
13	INTEN13	R/W	TADC 的中断和唤醒使能 (TADC 输出值的变化大于等于阈值)。 0: 禁止 TADC 中断和唤醒。 1: 使能 TADC 中断和唤醒。	0x0
12	INTEN12	R/W	模拟寄存器 ANA_REGx 状态错误中断唤醒使能控制位。 0: 禁止 1: 使能	0x0
11	INTEN11	R/W	当 VDCINDROP 为 0, 检测到进入睡眠模式的动作, 该事件的中断和唤醒使能控制。 0: 当 VDCINDROP 为 0 时, 检测到进入睡眠, 禁止该中断或唤醒 1: 当 VDCINDROP 为 0 时, 检测到进入睡眠, 使能该中断或唤醒	0x0
10	INTEN10	R/W	VDDLVL 中断或唤醒使能控制。 0: 禁止 VDDLVL 中断或唤醒 1: 使能 VDDLVL 中断或唤醒	0x0
9	-	-	保留	0
8	INTEN8	R/W	上升和下降时, VDCINDROP 中断或唤醒使能控制。 0: 禁止 VDCIN 中断或唤醒 1: 使能 VDCIN 中断或唤醒	0x0
7	INTEN7	R/W	上升和下降时, VDDALARM 中断或唤醒使能控制。 0: 禁止 VDDALARM 中断或唤醒	0x0



			1: 使能 VDDALARM 中断或唤醒	
6:4	-	-	保留	0
3	INTEN3	R/W	COMP2 的中断和唤醒使能控制。 0: 禁止 COMP2 中断和唤醒。 1: 使能 COMP2 中断和唤醒。	0x0
2	INTEN2	R/W	COMP1 的中断和唤醒使能控制。 0: 禁止 COMP1 中断和唤醒。 1: 使能 COMP1 中断和唤醒。	0x0
1	INTEN1	R/W	ADC 自动转换完成和 ADC 硬件超时的中断使能控制。 0: 禁止 ADC 自动转换和 ADC 硬件超时中断。 1: 使能 ADC 自动转换和 ADC 硬件超时中断。	0x0
0	INTEN0	R/W	ADC 手动转换完成和 ADC 硬件超时的中断使能控制。 0: 禁止 ADC 手动转换和 ADC 硬件超时中断。 1: 使能 ADC 手动转换和 ADC 硬件超时中断。	0x0

## 8.5.22 ANA\_ADCCTRL 寄存器

表 8-23 ANA\_ADCCTRL 寄存器

Bit	名称	类型	描述	默认值
31	MTRIG	R/W	手动控制 ADC 转换。 写 0: 无影响 写 1: 使能一次手动 ADC 转换 读 0: 当前手动设置的 ADC 转换已完成 读 1: 当前手动设置的 ADC 转换未完成	0x0
30	-	-	保留	0
29	CICAON	R/W	CIC 滤波器开启控制寄存器 0: 只有在 ADC 转换进行时开启 CICI 滤波器 1: CIC 滤波器始终开启	0x0
28	CICINV	R/W	CIC 滤波器输入反向控制 0: 无需反向 1: 将 CIC 输入信号反向	0x0
27	CICSCA	R/W	CIC 输出值缩小 2 倍选择 0: 输出值不变 1: 输出值缩小 2 倍	0x0
26:24	CICSKIP	R/W	鉴于 ADC 和 CIC 滤波器需要一定的稳定时间, 该寄存器用来选择忽略最开始的 CIC 输出采样点的数量。如果 CICAON=1 同时 ADC 采样通道没有变过, 那么该功能将失效。 0: 忽略最开始的 4 个采样点 1: 忽略最开始的 5 个采样点 2: 忽略最开始的 6 个采样点 3: 忽略最开始的 7 个采样点	0x0

			4: 忽略最开始的 0 个采样点 5: 忽略最开始的 1 个采样点 6: 忽略最开始的 2 个采样点 7: 忽略最开始的 3 个采样点	
23:22	DSRSEL	R/W	CIC 降采样率控制寄存器。降采样率越高，数据输出结果越稳定，收敛时间越长。 0: 1/512 1: 1/256 2: 1/128 3: 1/64	0x0
21	AMODE	R/W	ADC 自动转换模式控制。 0: 根据 ACH 的设置做单通道转换 1: 12 通道转换	0x0
20	MMODE	R/W	手动设置 ADC 转换模式控制。 0: 根据 MCH 的设置做单通道转换 1: 12 通道转换	0x0
19	-	-	保留。该寄存器位必须保持为 0。	0x0
18:16	AEN	R/W	ADC 自动转换使能控制寄存器。 0: 禁止 ADC 自动转换 4: ADC 自动转换由定时器 0 的溢出触发 5: ADC 自动转换由定时器 1 的溢出触发 6: ADC 自动转换由定时器 2 的溢出触发 7: ADC 自动转换由定时器 3 的溢出触发	0x0
15:13	-	-	保留	0
12	CLKSEL	R/W	ADC 时钟源选择 0: 6.5M RCH 1: PLLL	0x0
11:8	CLKDIV	R/W	ADC 时钟分频设置。ADC 时钟来自芯片内部高频 RC 时钟的分频，典型工作条件下，内部 RC 频率为 6.5536MHz，用户需设置寄存器 CLKDIV=3 来获得 ADC 的时钟频率 1.6384MHz。	0x0
7:4	ACH	R/W	ADC 自动转换通道控制 0: 通道 0 1: 通道 1 ... 11: 通道 11 12~15: 保留 只有当 AMODE=0 时该寄存器有效	0x0
3:0	MCH	R/W	ADC 手动转换通道控制 0: 通道 0 1: 通道 1 ...	0x0

			11: 通道 11 12~15: 保留 只有当 MMODE=0 时该寄存器有效	
--	--	--	---	--

## 8.5.23 ANA\_ADCDATAx 寄存器

表 8-24 ANA\_ADCDATAx 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	ADCDATAx	R	ADC 的转换结果数据寄存器。 DATA0: 存储 ADC 通道 0 转换结果。 DATA1: 存储 ADC 通道 1 转换结果。 ... DATA11: 存储 ADC 通道 11 转换结果。 单通道转换 (MMODE=0 或 AMODE=0) 时, 只有被 MCH 或 ACH 选择通道对应的 DATA 值更新。多通道转换 (MMODE=1 或 AMODE=1), 更新 12 个采样通道的转换值。	0x--

## 8.5.24 ANA\_CMPCNTx 寄存器

表 8-25 ANA\_CMPCNTx 寄存器

Bit	名称	类型	描述	默认值
31:0	CNTx	R/C	该寄存器用于存储 COMPx_SEL 设置的比较器 x 的事件次数的计数器。比如 COMPx_SEL 被设置为 1, 当 COMPx 发生上升沿时, 该计数器加一。 通过写 0 可以将该寄存器清零, 但这个操作只有在 CMPPDNx=1 时有效。	0x0000000

## 8.5.25 ANA\_MISC 寄存器

表 8-26 ANA\_MISC 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5:4	TADCTH	R/W	TADC 阈值设置。该寄存器控制 TADC 产生中断的阈值。TADC 时钟选用内部 32KRC 作为时钟源采样外部信号。当 TADC 的中断使能, 并且连续两个周期采样值的差值大于阈值, INTSTS13 就会置 1 并产生中断。	0x0
3:0	-	-	保留	0

# 第9章 通用 ADC

## 9.1 简介

芯片中集成了一个二阶  $\Sigma$ - $\Delta$ ADC，最多支持 9 通道的外部信号输入。所有通道的采样量程在不分压情况下均为 0~1.2V。分压情况下，3.3V 供电时，各通道量程为 0~3.6V。ADC 采样通道包含 GND、温度、BAT1、BAT2 电压以及外部输入信号。ADC 的时钟来自 6.5536M RC 或 PLLL。ADC 控制器工作在主域，因此与 ADC 相关的寄存器设置在深睡眠下可保持，不会被复位。

## 9.2 特点

- ADC 支持手动转换和自动转换模式。
- 支持 ADC 转换完成中断。
- 支持 12 通道全扫描功能。

## 9.3 ADC 功能框图

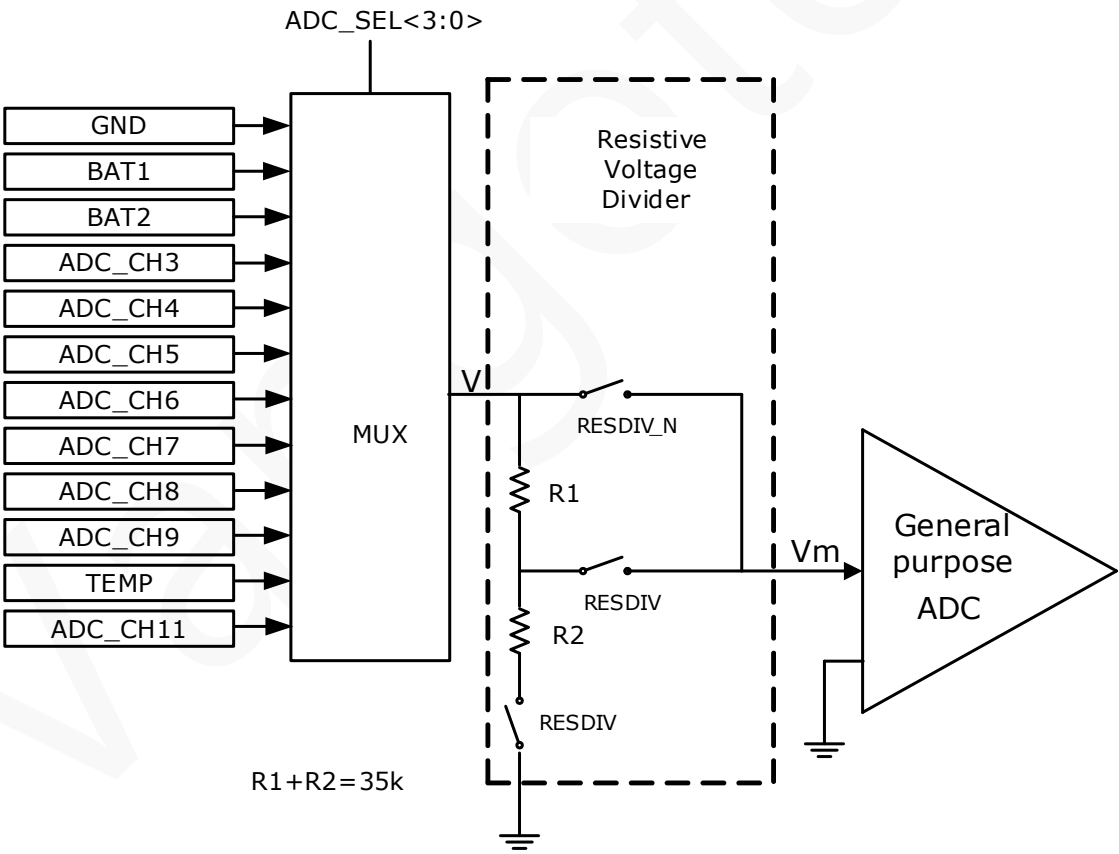


图 9-1 ADC 功能框图

## 9.4 寄存器地址

表 9-1 ANA 控制寄存器(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
----	----	----	----	-----

ANA_REG0	R/W	0x0000	模拟寄存器 0	0x00
ANA_REG1	R/W	0x0004	模拟寄存器 1	0x00
ANA_REG3	R/W	0x000C	模拟寄存器 3	0x00
ANA_INTSTS	R/C	0x0060	模拟中断状态寄存器	0x0000
ANA_INTEN	R/W	0x0064	模拟中断使能寄存器	0x0000
ANA_ADCCTRL	R/W	0x0068	ADC 控制寄存器	0x00000000
ANA_ADCDATA0	R	0x0070	ADC 通道 0 数据寄存器	--
ANA_ADCDATA1	R	0x0074	ADC 通道 1 数据寄存器	--
ANA_ADCDATA2	R	0x0078	ADC 通道 2 数据寄存器	--
ANA_ADCDATA3	R	0x007C	ADC 通道 3 数据寄存器	--
ANA_ADCDATA4	R	0x0080	ADC 通道 4 数据寄存器	--
ANA_ADCDATA5	R	0x0084	ADC 通道 5 数据寄存器	--
ANA_ADCDATA6	R	0x0088	ADC 通道 6 数据寄存器	--
ANA_ADCDATA7	R	0x008C	ADC 通道 7 数据寄存器	--
ANA_ADCDATA8	R	0x0090	ADC 通道 8 数据寄存器	--
ANA_ADCDATA9	R	0x0094	ADC 通道 9 数据寄存器	--
ANA_ADCDATAA	R	0x0098	ADC 通道 10 数据寄存器	--
ANA_ADCDATAB	R	0x009C	ADC 通道 11 数据寄存器	--

## 9.5 寄存器定义

### 9.5.1 ANA\_REG1 寄存器

表 9-2 ANA\_REG1 寄存器

Bit	名称	功能	描述
3:0	-	保留	0
4	RESDIV	使能 ADC 输入信号的电阻分压	0: 不使能 1: 使能 1/4 电阻分压
5	GDE4	使能 ADC 输入信号的电容分压	0: 不使能 1: 使能 1/4 电容分压
7:6	-	保留	0

### 9.5.2 ANA\_REG3 寄存器

表 9-3 ANA\_REG3 各个 bit 的功能描述

Bit	名称	功能	描述
0	ADCPDN	ADC 开关控制信号。ADC 关闭之前需要先禁止 ADC 自动转换。	0: 关闭 1: 开启
3	ADCBGPPD	ADCBGP 掉电控制	0: 开启 ADCBGP 1: 关闭 ADCBGP

4	RCHPD	高频 RC 时钟 RCH(6.5536MRC)开关控制信号	0: 开启 RCH 1: 关闭 RCH
---	-------	-------------------------------	------------------------

\*: RCH、PLL、ADC 与 ADCBGP 有关系, 在开 RCH、PLL、ADC 之前, 必须开启 ADCBGP。如果系统中没有使用 ADCBGP, 可通过 ADCBGPPD 位关闭 ADCBGP; 如果 ADCBGP 有被用到, 此时, 通过 ADCBGPPD 不能关闭 ADCBGP, 因为系统具有保护功能, 会重新开启 ADCBGP。浅睡眠或深睡眠模式下, 不管 ADCBGP 是否有被用到, 都可以关闭 ADCBGP。

### 9.5.3 ANA\_INSTS 寄存器

表 9-4 ANA\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
1	INTSTS1	R/C	ADC 自动转换完成或 ADC 转换发生硬件超时的中断标记位。 读 0: 未发生中断 读 1: 发生中断 写 0: 无影响 写 1: 清除该位	0x0
0	INTSTS0	R/C	手动设置 ADC 转换完成或 ADC 转换发生硬件超时的中断标记位。 读 0: 未发生中断 读 1: 发生中断 写 0: 无影响 写 1: 清除该位	0x0

### 9.5.4 ANA\_INTEN 寄存器

表 9-5 ANA\_INTEN 寄存器

Bit	名称	类型	描述	默认值
1	INTEN1	R/W	ADC 自动转换完成和 ADC 转换硬件超时的中断使能控制。 0: 禁止 ADC 自动转换和 ADC 转换硬件超时中断。 1: 使能 ADC 自动转换和 ADC 转换硬件超时中断。	0x0
0	INTEN0	R/W	ADC 手动转换完成和 ADC 转换硬件超时的中断使能控制。 0: 禁止 ADC 手动转换和 ADC 转换硬件超时中断。 1: 使能 ADC 手动转换和 ADC 转换硬件超时中断。	0x0

### 9.5.5 ANA\_ADCCTRL 寄存器

表 9-6 ANA\_ADCCTRL 寄存器

Bit	名称	类型	描述	默认值
31	MTRIG	R/W	触发 ADC 手动转换 写 0: 无影响 写 1: 开始 ADC 手动转换 读 0: 当前 ADC 手动转换已完成 读 1: 当前 ADC 手动转换未完成	0x0

30	-	-	保留	0
29	CICAON	R/W	CIC 滤波器始终开启控制寄存器。 0: 只有在 ADC 转换进行时开启 CICI 滤波器 1: CIC 滤波器始终开启	0x0
28	CICINV	R/W	CIC 滤波器反向输入控制 0: 无需反向 1: 将 CIC 输入信号反向	0x0
27	CICSCA	R/W	CIC 输出值缩小 2 倍选择 0: 输出值不变 1: 输出值缩小为 1/2	0x0
26:24	CICSKIP	R/W	鉴于 ADC 和 CIC 滤波器需要一定的稳定时间, 该寄存器用来选择忽略最开始的 CIC 输出采样点的数量。如果 CICAON=1 同时 ADC 采样通道没有变过, 那么该功能将失效。 0: 忽略起始的 4 个采样点 1: 忽略起始的 5 个采样点 2: 忽略起始的 6 个采样点 3: 忽略起始的 7 个采样点 4: 忽略起始的 0 个采样点 5: 忽略起始的 1 个采样点 6: 忽略起始的 2 个采样点 7: 忽略起始的 3 个采样点	0x0
23:22	DSRSEL	R/W	CIC 降采样率控制寄存器。降采样率越高, 数据输出越稳定, 采样率越低。 0: 1/512 1: 1/256 2: 1/128 3: 1/64	0x0
21	AMODE	R/W	ADC 自动模式控制 0: 根据 ACH 的设置做单通道转换 1: 12 通道转换	0x0
20	MMODE	R/W	手动设置 ADC 转换模式控制 0: 根据 MCH 的设置做单通道转换 1: 12 通道转换	0x0
19	-	-	保留。该寄存器位必须保持为 0。	0x0
18:16	AEN	R/W	ADC 自动转换使能控制 0: 禁止 ADC 自动转换 4: 定时器 0 溢出时触发 ADC 自动转换 5: 定时器 1 溢出时触发 ADC 自动转换 6: 定时器 2 溢出时触发 ADC 自动转换 7: 定时器 3 溢出时触发 ADC 自动转换	0x0

15:13	-	-	保留	0
12	CLKSEL	R/W	ADC 时钟源选择 0: 6.5M RCH 1: PLLL	0x0
11:8	CLKDIV	R/W	ADC 时钟分频设置。典型工作时钟为 1.6384MHz。 ADC 时钟频率=ADC 时钟源频率/(CLKDIV+1)。 比如: ADC 时钟源为内部高频 RCH (6.5536MHz), 此时, 应该设置寄存器 CLKDIV=3, 以使得 ADC 的时钟频率 1.6384MHz。	0x0
7:4	ACH	R/W	ADC 自动转换通道控制 0: 通道 0 1: 通道 1 ... 11: 通道 11 12~15: 保留 只有当 AMODE=0 时该寄存器有效 ADC 采样通道参见下面的表格	0x0
3:0	MCH	R/W	ADC 手动转换通道控制 0: 通道 0 1: 通道 1 ... 11: 通道 11 12~15: 保留 只有当 MMODE=0 时该寄存器有效 ADC 采样通道参见下面的表格	0x0

表 9-7 ADC 采样通道

Bit	名称	通道
7:4	ACH	0000: GND; 0001: BAT1 0010: BAT2
3:0	MCH	0011: ADC_CH3 0100: ADC_CH4 0101: ADC_CH5 0110: ADC_CH6; 0111: ADC_CH7 1000: ADC_CH8 1001: ADC_CH9 1010: TEMP; 1011: ADC_CH11 1100~ 1111: 保留



		当选择 TEMP 通道时，温度传感器自动打开；当选择其它通道时，温度传感器自动关闭。
--	--	--

## 9.5.6 ANA\_ADCDATAx 寄存器

表 9-8 ANA\_ADCDATAx 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	-
15:0	ADCDATAx	R	ADC 的转换结果数据寄存器，是 16 位补码。 DATA0：存储 ADC 通道 0 转换结果。 DATA1：存储 ADC 通道 1 转换结果。 ... DATA11：存储 ADC 通道 11 转换结果。 单通道转换（MMODE=0 或 AMODE=0）时，只有被 MCH 或 ACH 选择通道对应的 DATA 值更新。多通道转换（MMODE=1 或 AMODE=1），更新 12 个采样通道的转换值。	0x--

## 9.6 电池电压测量和外部电压测量

在 V94XX(A)中，ADC 能通过外部输入管脚测量外部信号，也能测量 BAT1、BAT2 等信号的电压。

ADC 工作模式：定时器自动触发 ADC 转换、手动触发 ADC 转换。

ADC 采样通道选择：单通道、12 通道。

ADC 分压模式：不分压、电阻分压、电容分压。电阻分压有额外功耗。因为内部分压电路有一个 35kΩ 的电阻，这个电路消耗的功率：

$$P = U_D \times I_D = V \times \frac{V}{R_1 + R_2} = \frac{V^2}{35000} \quad \text{Equation 9-1}$$

测量外部电压（ADC\_CHx），包括通道 3、4、5、6、7、8、9、11。

测量电池电压通道，即 BAT1、BAT2 测量（通道 1、2）。

用户需要检查用于存储 ADC 系数的 a、b（A、B）存储区域参数是否合法，如果合法，采用该存储区域数据做运算，如果非法，ADC 系数采用固定参数做运算，运算公式见表 9-9 电压测量量程与公式。ADC 系数的 a、b（A、B）见表 4-19 Info 信息寄存器。

表 9-9 电压测量量程与公式

Power	Channel	Divider Mode	Signal range (V)	Formula
3.3V	ADC_CHx	No divider	-0.2~1.2	如果 ADC 的 a、b 参数合法， $V_{DC} = a1 \div 100000000 * X + b1 \div 100000000$ 否则 $V_{DC} = 0.00003680 * X + 0.00205011$
		Resistive	-0.2~3.6	如果 ADC 的 a、b 参数合法， $V_{DC} = a2 \div 100000000 * X + b2 \div 100000000$ 否则 $V_{DC} = 0.00016425 * X + 0.03739179$
		Capacitive	-0.2~3.6	如果 ADC 的 a、b 参数合法，

				$V_{DC} = a3 \div 100000000 * X + b3 \div 100000000$ 否则 $V_{DC} = 0.00014051 * X - 0.00023322$
	BAT1 channels	Resistive	2.5~4.8	如果 ADC 的 a、b 参数合法， $V_{DC} = a4 \div 100000000 * X + b4 \div 100000000$ 否则 $V_{DC} = 0.00015392 * X + 0.06667986 + \text{OffsetBatR} \div 1000$
		Capacitive	2.5~4.6	如果 ADC 的 a、b 参数合法， $V_{DC} = a5 \div 100000000 * X + b5 \div 100000000$ 否则 $V_{DC} = 0.00014107 * X - 0.00699515 + \text{OffsetBatC} \div 1000$
	BAT2 channels	Resistive	0.7~4.8	如果 ADC 的 a、b 参数合法， $V_{DC} = a6 \div 100000000 * X + b6 \div 100000000$ 否则 $V_{DC} = 0.00015392 * X + 0.06667986 + \text{OffsetBatR} \div 1000$
		Capacitive	0.7~4.6	如果 ADC 的 a、b 参数合法， $V_{DC} = a7 \div 100000000 * X + b7 \div 100000000$ 否则 $V_{DC} = 0.00014107 * X - 0.00699515 + \text{OffsetBatC} \div 1000$

上述公式中的 X 值计算公式如下：

$$X = \text{ADC\_DATAx}$$

**Equation 9-2**

其中：

ADC\_DATAx 是地址(0x40014270+4x)(16bit)中的值，x 是对应的通道号 0-11。

测量电池电压和外部电压的步骤如下：

#### 1. 设置 ADC

ADC 的时钟频率推荐设置为 1.6384Mhz，同时设置 CIC 的降采样率 DSRSEL 为 1/512，并且根据测量的电压大小范围设置分压或不分压。设置手动或自动触发 ADC。

- 等待 ANA\_INSTS(0x40014260)寄存器的位 ANA\_INTSTS0(手动触发) /ANAINSTS1(自动触发)值为 1 时，查询 ANA\_INSTS(0x40014260)寄存器的位 ANA\_INTSTS14 是否为 0，如果是 0，读寄存器 ADC\_DATAx 的值；如果是 1，说明 ADC 转换发生了硬件超时，重复步骤 1。

- 根据上表计算电池电压或外部电压。

## 9.7 温度测量

CH10 为温度测量通道，温度测量范围是-40~+85°C。

测量温度的步骤如下：

#### 1. 设置 ADC

ADC 的时钟频率推荐设置为 1.6384Mhz，同时设置 CIC 的降采样率 DSRSEL 为 1/512，并且不分压或电阻分压(不支持电容分压模式下测温)。设置手动或自动触发 ADC。

- 等待 ANA\_INSTS(0x40014260)寄存器的位 ANA\_INTSTS0(手动触发) /ANAINSTS1(自动触发)值为 1 时，查询 ANA\_INSTS(0x40014260)寄存器的位 ANA\_INTSTS14 是否为 0，如果是 0，读寄存器 ADC\_DATAA(0x40014298)的值；如果是 1，说明 ADC 转换发生了硬件超时，重复步骤 1。

- 计算温度 T：(单位：1/256°C)，T 是一个 16 位的补码，实际温度值等于 T/256.0。例如 0x1880 代表 24.5 摄氏度。

计算公式:

$$T = ((P0 * ((X * X) >> 16)) + P1 * X + P2) >> 8$$

Equation 9-3

X 是从寄存器 ADC\_DATAA(16bit)读出来的值

P0 是地址 0x40D00(16bit)中的值

P1 是地址 0x40D02(16bit)中的值

P2 是由以下公式计算出的值:

$$P2 = P2' + (Tr - Tm) * 256$$

Equation 9-4

P2'是地址 0x40D04(32bit)中的值

Tr 是地址 0x40D70(32bit)中的值

Tm 是地址 0x40D74(32bit)中的值

注意:

对于通道 10, 硬件内部做了 4 个点求平均, 内部用于求平均的四个缓存是个先入先出队列, 会被 DPOR 复位为 0。用户每做一次 ADC 转换, 硬件就往缓存写入一个数据, 并将内部四个缓存的数据的平均值计算出来, 放在 ADC 通道 10 寄存器。因此, 为了得到某个温度点准确的温度值, 需要用户连续转换 ADC 四次, 取最后一次 ADC 通道 10 寄存器数据作为 ADC 数据。

## 第10章 比较器控制器

### 10.1 简介

V94XX(A)集成了两个比较器，用来比较模拟信号的大小，有以下几种比较模式：

- 比较器正端接管脚 CMP\_P，负端信号接管脚 CMP\_N；
- 比较器正端接管脚 CMP\_P，负端信号接 ADCREF 或 LPREF；
- 比较器正端接管脚 CMP\_N，负端信号接 ADCREF 或 LPREF。

每个比较器有 20mv 的电压回差。

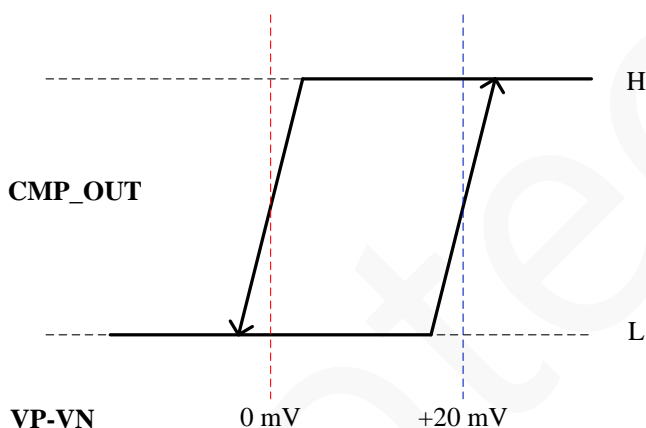


图 10-1 比较器的迟滞窗口

### 10.2 特点

- 比较器有中断/唤醒功能
- 有多种比较模式
- 有比较器计数器
- 在睡眠下也能工作

### 10.3 功能框图

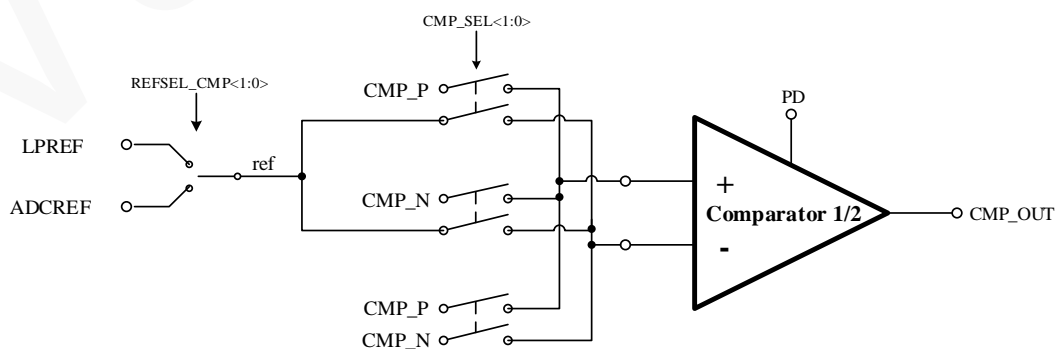


图 10-2 比较器功能框图

## 10.4 寄存器地址

表 10-1 ANA 控制寄存器(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_REG2	R/W	0x0008	模拟寄存器 2	0x00
ANA_REG3	R/W	0x000C	模拟寄存器 3	0x00
ANA_REG5	R/W	0x0014	模拟寄存器 5	0x00
ANA_REG6	R/W	0x0018	模拟寄存器 6	0x00
ANA_CTRL	R/W	0x0050	模拟控制寄存器	0x00000000
ANA_CMPOUT	R	0x0054	比较结果寄存器	0x0030
ANA_INTSTS	R/C	0x0060	模拟中断状态寄存器	0x00
ANA_INTEN	R/W	0x0064	模拟中断使能寄存器	0x00
ANA_CMPCNT1	R/C	0x00B0	比较器 1 计数器	0x00000000
ANA_CMPCNT2	R/C	0x00B4	比较器 2 计数器	0x00000000

## 10.5 寄存器定义

### 10.5.1 ANA\_REG2 寄存器

表 10-2 ANA\_REG2 寄存器

Bit	名称	功能	描述
1:0	CMP1_SEL[1: 0]	CMP1 信号源选择	00: CMP1_P 与 REF 01: CMP1_N 与 REF 1*: CMP1_P 与 CMP1_N
3:2	CMP2_SEL[1: 0]	CMP2 信号源选择	00: CMP2_P 与 REF 01: CMP2_N 与 REF 1*: CMP2_P 与 CMP2_N
4	REFSEL_CMP1	CMP1 的参考电压	0: LPREF 1: ADCREF
5	REFSEL_CMP2	CMP2 的参考电压	0: LPREF 1: ADCREF

### 10.5.2 ANA\_REG3 寄存器

表 10-3 ANA\_REG3 寄存器

Bit	名称	功能	描述
1	CMP1PDN	CMP1 掉电控制信号	0: 关闭 CMP1 1: 开启 CMP1
2	CMP2PDN	CMP2 掉电控制信号	0: 关闭 CMP2

			1: 开启 CMP2
3	ADCBGPPD	ADCBGP 掉电控制信号	0: 上电 ADCBGP 1: 掉电 ADCBGP

\*: RCH、PLL、ADC 与 ADCBGP 有关系, 在开 RCH、PLL、ADC 之前, 必须开启 ADCBGP。如果系统中没有使用 ADCBGP, 可通过 ADCBGPPD 位关闭 ADCBGP; 如果 ADCBGP 有被用到, 此时, 通过 ADCBGPPD 不能关闭 ADCBGP, 因为系统具有保护功能, 会重新开启 ADCBGP。浅睡眠或深睡眠模式下, 不管 ADCBGP 是否有被用到, 都可以关闭 ADCBGP。

### 10.5.3 ANA\_REG5 寄存器

表 10-4 ANA\_REG5 寄存器

Bit	名称	功能	描述
1:0	IT_CMP1[1:0]	CMP1 的偏置电流选择。	00: 20nA; 01: 100nA; 1*: 500nA;
3:2	IT_CMP2[1:0]	CMP2 的偏置电流选择	00: 20nA; 01: 100nA; 1*: 500nA;

注意: 当 CMP 偏置电流越小时, 功耗越低以及传播延迟时间越长; 当 CMP 偏置电流越大时, 功耗越高以及传播延迟时间越短。

### 10.5.4 ANA\_CTRL 寄存器

表 10-5 ANA\_CTRL 寄存器

Bit	名称	类型	描述	默认值
23:22	CMP2DEB	R/W	比较器 2 消抖控制寄存器。 0: 无消抖。 1: 2 个 RTCCLK 时钟周期消抖。 2: 3 个 RTCCLK 时钟周期消抖。 3: 4 个 RTCCLK 时钟周期消抖。 当使能消抖时, 只有当信号在 RTCCLK 时钟的多周期内没有改变时, 输入信号才有效。此外, 唤醒和中断的响应时间将被延迟, 直到信号有效。该电路可以在所有模式下工作, 包括浅睡眠和深睡眠模式。	0x0
21:20	CMP1DEB	R/W	比较器 1 消抖控制寄存器。 0: 不消抖。 1: 2 个 RTCCLK 时钟周期消抖。 2: 3 个 RTCCLK 时钟周期消抖。 3: 4 个 RTCCLK 时钟周期消抖。 当使能消抖时, 只有当信号在 RTCCLK 时钟的多周期内没有变化时, 输入信号才有效。此外, 唤醒和中断的响应时间将被延迟,	0x0

			直到信号有效。该电路可以在所有模式下工作，包括浅睡眠和深睡眠模式。	
19:18	-	-	保留	0
3:2	COMP2_SEL	R/W	该寄存器用来控制比较器 2 中断和唤醒信号。 0: 关闭 1: 比较器 2 的上升沿。 2: 比较器 2 的下降沿。 3: 比较器 2 的双边沿。	0x0
1:0	COMP1_SEL	R/W	该寄存器用来控制比较器 1 的中断和唤醒信号。 0: 关闭 1: 比较器 1 的上升沿。 2: 比较器 1 的下降沿。 3: 比较器 1 的双边沿。	0x0

### 10.5.5 ANA\_CMPOUT 寄存器

表 10-6 ANA\_CMPOUT 寄存器

Bit	名称	类型	描述	默认值
3	COMP2	R	该位显示比较器 2 的结果。	0x0
2	COMP1	R	该位显示比较器 1 的结果。	0x0

### 10.5.6 ANA\_INTSTS 寄存器

表 10-7 ANA\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
3	INTSTS3	R/C	COMP2 的中断标志，中断产生条件受 COMP2_SEL 控制。读 0: 无 COMP2 中断。 读 1: 发生 COMP2 中断。 写 0: 无效。 写 1: 清零。	0x0
2	INTSTS2	R/C	COMP1 的中断标志，中断产生条件受 COMP1_SEL 控制。读 0: 无 COMP1 中断。 读 1: 发生 COMP1 中断。 写 0: 无效。 写 1: 清零。	0x0

### 10.5.7 ANA\_INTEN 寄存器

表 10-8 ANA\_INTEN 寄存器

Bit	名称	类型	描述	默认值
3	INTEN3	R/W	COMP2 的中断和唤醒使能控制。 0: 禁止 COMP2 中断和唤醒。 1: 使能 COMP2 中断和唤醒。	0x0

2	INTEN2	R/W	COMP1 的中断和唤醒使能控制。 0：禁止 COMP1 中断和唤醒。 1：使能 COMP1 中断和唤醒。	0x0
---	--------	-----	---	-----

10.5.8 ANA\_CMPCNTx 寄存器

表 10-9 ANA\_CMPCNTx 寄存器

Bit	名称	类型	描述	默认值
31:0	CNTx	R/C	根据 COMPx_SEL 的设置,该寄存器储存比较器 X 的发生次数。例如,当 COMPx_SEL 设置为 1,当 COMPx 上升,那么这个计数器将增加 1。该寄存器可以写 0 清零,但仅当相应的 CMPPDNx 设置为 1 时有效 (即相应的比较器也被使能时)。	0x0000000



# 第11章 TinyADC 控制器

## 11.1 简介

TinyADC（2位的ADC）控制器用来控制V94XX(A)的TinyADC功能，TinyADC是低精度模数转换器。TinyADC由VDD供电。TinyADC控制器位于电源主域，所以它在深睡眠下也可以正常工作。

## 11.2 特点

--产生 TinyADC 中断。

## 11.3 功能框图

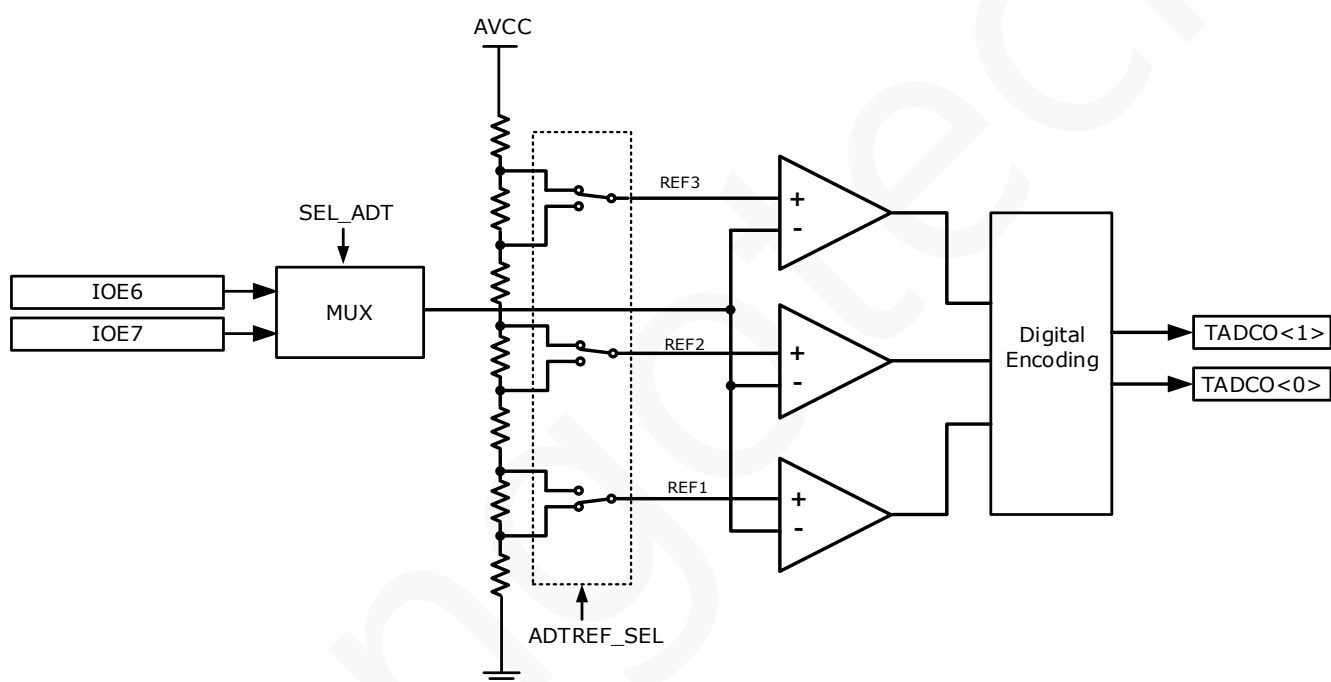


图 11-1 模拟控制器功能框图

## 11.4 寄存器地址

表 11-1 ANA 控制器(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_REGF	R/W	0x003C	模拟控制寄存器 15	0x00
ANA_CMPOUT	R	0x0054	比较器结果寄存器	0x0030
ANA_INTSTS	R/C	0x0060	模拟中断状态寄存器	0x0000
ANA_INTEN	R/W	0x0064	模拟中断使能寄存器	0x0000
ANA_MISC	R/W	0x00B8	模拟 misc.控制寄存器	0x00

## 11.5 寄存器定义

### 11.5.1 ANA\_REGF 寄存器

表 11-2 ANA\_REGF 寄存器

Bit	名称	功能	描述
3	PDN_ADT	Tiny ADC 开关控制	0: 掉电 1: 上电
4	SEL_ADT	Tiny ADC 信号输入源选择	0: 从 IOE6 输入 1: 从 IOE7 输入
5	ADTREF1_SEL	TADC 参考电压 1 的选择	0: 0.9V 1: 0.7V
6	ADTREF2_SEL	TADC 参考电压 2 的选择	0: 1.8V 1: 1.6V
7	ADTREF3_SEL	TADC 参考电压 3 的选择	0: 2.7V 1: 2.5V

### 11.5.2 ANA\_CMPOUT 寄存器

表 11-3 有关 TADC 的 ANA\_CMPOUT 寄存器

Bit	名称	类型	描述	默认值
15:14	TADCO	R	TinyADC 输出。 例如：如果 ADTREFx_SEL 三个位配置为 0，那么： 信号 < 0.9，TADCO 输出 0。 信号 > 0.9 && < 1.8，TADCO 输出 1。 信号 > 1.8 && < 2.7，TADCO 输出 2。 信号 > 2.7，TADCO 输出 3。	-

### 11.5.3 ANA\_INTSTS 寄存器

表 11-4 有关 TADC 的 ANA\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
31:14	-	-	保留	0
13	INTSTS13	R/C	TADC 过阈值中断。TADC 在上升或下降时，并且相比上一周期的变化值大于等于阈值（TADCTH），中断标志位将会置位。 读 0：无过阈值中断。 读 1：发生过阈值中断。 写 0：无效。 写 1：清零。	0x0

### 11.5.4 ANA\_INTEN 寄存器

表 11-5 ANA\_INTEN 寄存器 forTADC

Bit	名称	类型	描述	默认值
31:14	-	-	保留	0
13	INTEN13	R/W	TADC 的中断和唤醒使能（TADC 的变化超过阈值）。 0：禁止 TADC 中断和唤醒。 1：使能 TADC 中断和唤醒。	0x0

### 11.5.5 ANA\_MISC 寄存器

表 11-6 ANA\_MISC 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5:4	TADCTH*	R/W	TADC 阈值设置。该寄存器控制 TADC 产生中断的阈值。TADC 时钟选用内部 32K RC 作为时钟源采样外部信号。当 TADC 的中断使能，并且两个连续周期采样值的差值大于等于阈值，INTSTS13 就会置 1 并产生中断。	0x0
3:0	-	-	保留	0

\*：ANA\_MISC 寄存器的 3：0 位必须写 0。

## 第12章 看门狗

### 12.1 简介

V94XX(A)内部集成16位的看门狗（WDT）电路，WDT时钟源是32kHz RTCCLK。当发生程序跑飞的情况，该定时器将会溢出并产生系统复位。WDT复位等级与POR复位等级相同，可提供芯片内部的复位。

### 12.2 特点

--WDT 的复位周期可选为：2s\1s\0.5s\0.25s。

--WDT 具有密码保护机制，防止对 WDT 的非法访问。

### 12.3 看门狗在不同模式下的状态

看门狗的使能或禁止在不同模式下分别由硬件或是软件控制，下表为 MODE=1（PMU\_STS bit24）时看门狗的状态控制。当 MODE=0 时，看门狗在任何模式下都是无效的。

表 12-1 看门狗在不同模式下状态(MODE=1)

看门狗	电源模式			
	深睡眠	浅睡眠	IDLE	工作
开/关	ON(WDTEN=1) OFF(WDTEN=0)	ON(WDTEN=1) OFF(WDTEN=0)	ON(WDTEN=1) OFF(WDTEN=0)	ON(WDTEN=1) OFF(WDTEN=0)
看门狗	不同模式唤醒			
	深睡眠	浅睡眠	IDLE	
开/关	ON	ON	ON(WDTEN=1) OFF(WDTEN=0)	

### 12.4 寄存器地址

表 12-2 PMU\_WDT 控制器(PMU 基地址: 0x40014000)

名称	类型	地址	描述	默认值
PMU_WDTPASS	R/W	0x0040	看门狗解锁寄存器	0x00000000
PMU_WDTEN	R/W	0x0044	看门狗定时器使能寄存器	0x1
PMU_WDTCLR	W	0x0048	清看门狗寄存器	0x0000

### 12.5 寄存器定义

#### 12.5.1 PMU\_WDTPASS 寄存器

表 12-3 PMU\_WDTPASS 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	UNLOCK	R	该位表示看门狗使能寄存器已被解锁，并准备设置看门狗使能控制寄存器。向该寄存器写入 0xAA5555AA，会将标志位 UNLOCK	0x0

			置为 1。任何对 PMU 控制寄存器（包括 ICE）的写操作，都会清零该位，因此用户应在 UNLOCK 位置 1 后，立即置位 PMU_WDTEN，否则只能重新继续解锁步骤。	
--	--	--	---	--

## 12.5.2 PMU\_WDTEN 寄存器

表 12-4 PMU\_WDTEN 寄存器

Bit	名称	类型	描述	默认值
31:4	-	-	保留	0
3:2	WDTSEL	R/W	设置 WDT 的计数周期。 0: 2 秒 1: 1 秒 2: 0.5 秒 3: 0.25 秒 在设置该寄存器之前，首先应将 PMU_WDTPASS 的 UNLOCK 位置 1。	0x0
1	-	-	保留	0
0	WDTEN	R/W	该位表示看门狗定时器使能。在设置该寄存器之前，首先应将 PMU_WDTPASS 的 UNLOCK 位置 1。	0x1

## 12.5.3 PMU\_WDTCLR 寄存器

表 12-5 PMU\_WDTCLR 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	WDTCNT	R/C	该寄存器表示看门狗定时器的当前计数值。当该定时器计数到达 WDTSEL 设置的极限值时，将会发生看门狗系统复位。因此，用户应定时清空此定时器，防止看门狗复位。向该寄存器写入 0x55AAAA55 即可清空该寄存器。该寄存器在调试模式下会自动清零，即调试模式下不会发生看门狗复位。	0x0

## 第13章 RTC 控制器

### 13.1 简介

RTC 控制器用来控制时间的计算和实时时钟校正。时间计算功能可以实现年、月、星期、日、小时、分、秒自动计算功能和闰年监测。RTC 校正功能，可通过软件校正 RTC，包括监测温度和调整秒时钟周期以补偿由温度变化引起的时钟偏移。RTC 引擎内置多秒、多分、多小时和 RTC 计数器自动唤醒定时器。

### 13.2 特点

- 支持从秒到年的 BCD 格式的时间计算。
- 自动闰年监测。
- 支持晶体时钟频率补偿。
- 可编程的多唤醒源。
- 为厂家提供校正功能。
- RTC 分频在浅睡眠/深睡眠模式下实现低功耗。

### 13.3 功能框图

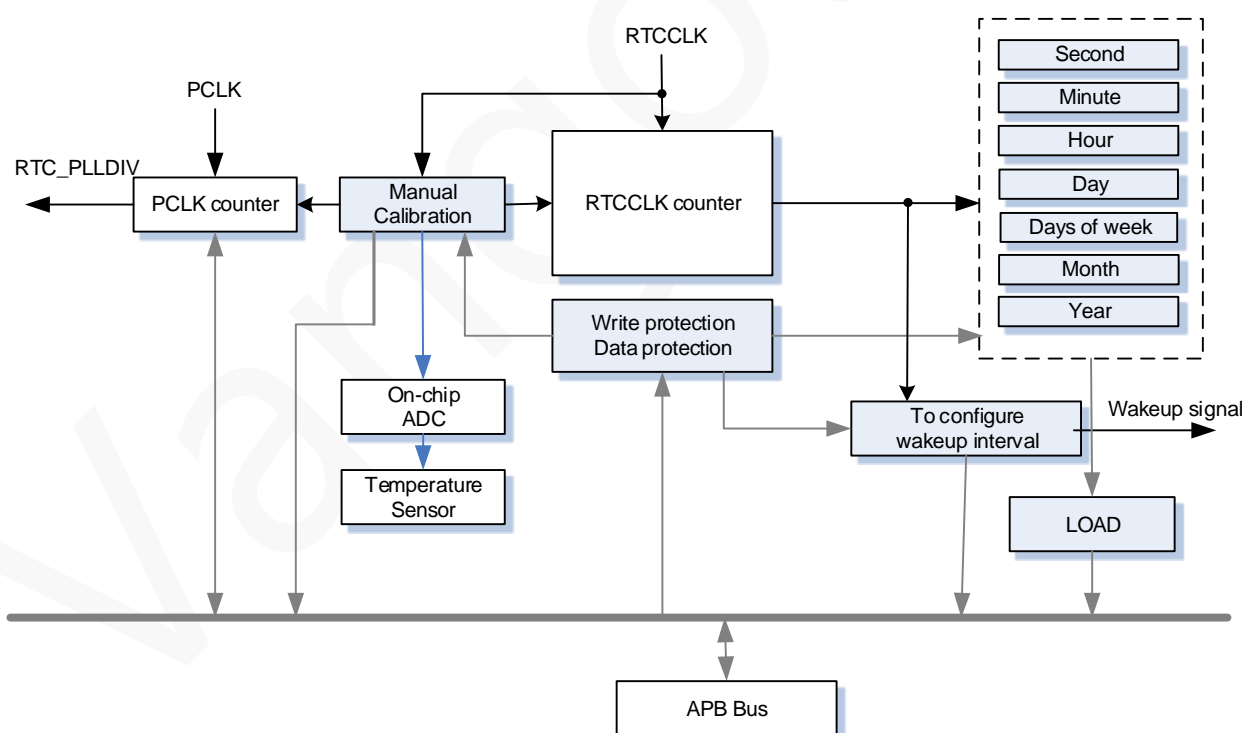


图 13-1 RTC 控制器功能框图

### 13.4 寄存器读写

#### 13.4.1 RTC 寄存器写操作

V94XX(A)芯片中，部分 RTC 寄存器受写保护，包括 RTC 时间寄存器、RTC 校正寄存器和另外几个寄存器，详情

可参考表 13-1。

MCU 应按以下步骤对写保护寄存器进行写操作：

1. 首先用户必须等待 BSY（RTC\_CE 寄存器的 bit1）为 0，以保证 RTC 处于空闲状态；
2. 对 RTC\_PWD 寄存器写 0x5AA55AA5，使能对 RTC\_CE 寄存器的访问；
3. 对 RTC\_CE 寄存器写 0xA55AA55B，使能对带写保护寄存器的写操作；
4. 配置写保护寄存器，其中时间寄存器 RTC\_SEC~RTC\_YEAR 应该同时进行配置；
5. 对 RTC\_PWD 寄存器写 0x5AA55AA5，使能对 RTC\_CE 寄存器的访问；
6. 对 RTC\_CE 寄存器写 0xA55AA55A，清除 CE(RTC\_CE 寄存器的 bit0)标志。当 CE 从 1 清为 0 时，BSY（RTC\_CE 寄存器的 bit1）置 1。当 CE 清为 0 时，写保护寄存器的值将开始更新到 RTC 内核，更新步骤需要 3 个 32K 周期，大约 100μs；
7. 步骤 6 之后，用户需要等待 BSY（RTC\_CE 寄存器的 bit1）自动清 0，以保证 RTC 的更新完成。

不受写保护的 RTC 寄存器，MCU 可以直接进行写操作。

## 13.4.2 RTC 寄存器读操作

RTC 时间寄存器受读保护，详情可参考表 13-1。

MCU 应按以下步骤完成 RTC 时间寄存器的读：

1. 首先用户必须等待 BSY（RTC\_CE 寄存器的 bit1）为 0，以保证 RTC 处于空闲状态；
2. MCU 读 RTC\_LOAD 寄存器，RTC\_LOAD 寄存器的读操作开始，BSY（RTC\_CE 寄存器的 bit1）会立马置 1。BSY 在读操作后会自动清 0，读操作需要 3 个 32K 周期，大约 100μs；
3. RTC\_LOAD 寄存器的读操作结束后，当前时间会被锁定，用户可以读取 RTC\_SEC~RTC\_YEAR 寄存器获得时间值。

不受读保护的 RTC 寄存器，MCU 可以直接进行读操作。

## 13.5 寄存器地址

下表列出了每个寄存器的位置。“写保护”是指该寄存器只当 CE 设置为 1 时可写，当 CE 清零时值才会更新到 RTC 寄存器。“读保护”是指仅当 RTC\_LOAD 读操作发生后且 BSY 位清 0，这些寄存器的值会被更新。

表 13-1 RTC 控制器(RTC 基地址：0x40014800)

名称	类型	地址	描述	默认值	写保护	读保护
RTC_SEC	R/W	0x0000	RTC 秒寄存器	--	V	V
RTC_MIN	R/W	0x0004	RTC 分寄存器	--	V	V
RTC_HOUR	R/W	0x0008	RTC 时寄存器	--	V	V
RTC_DAY	R/W	0x000C	RTC 天寄存器	--	V	V
RTC_WEEK	R/W	0x0010	RTC 周寄存器	--	V	V
RTC_MON	R/W	0x0014	RTC 月寄存器	--	V	V
RTC_YEAR	R/W	0x0018	RTC 年寄存器	--	V	V
RTC_WKUSEC	R/W	0x0020	RTC 秒唤醒寄存器	0x00	V	
RTC_WKUMIN	R/W	0x0024	RTC 分唤醒寄存器	0x00	V	
RTC_WKUHOURL	R/W	0x0028	RTC 时唤醒寄存器	0x00	V	

RTC_WKUCNT	R/W	0x002C	RTC 计数唤醒寄存器	0x0000000	V	
RTC_CAL	R/W	0x0030	RTC 校正寄存器	--	V	
RTC_DIV	R/W	0x0034	RTC_PLLDIV 分频寄存器	0x0000000		
RTC_CTL	R/W	0x0038	RTC_PLLDIV 分频器控制寄存器	0x0		
RTC_PWD	R/W	0x0044	RTC 密码控制寄存器	0x00000000		
RTC_CE	R/W	0x0048	RTC 写使能控制寄存器	0x0		
RTC_LOAD	R/W	0x004C	RTC 读使能控制寄存器	--		
RTC_INTSTS	R/W	0x0050	RTC 中断状态控制寄存器	0x000		
RTC_INTEN	R/W	0x0054	RTC 中断使能控制寄存器	0x000		
RTC_PSCA	R/W	0x0058	RTC 时钟预分频控制寄存器	0x0	V	
RTC_ACTI	R/W	0x0084	晶体顶点温度寄存器	0x1800	V	
RTC_ACF200	R/W	0x0088	RTC200 倍频率寄存器	0x640000	V	
RTC_ACP0	R/W	0x0090	RTC 参数 0 寄存器	0x0000	V	
RTC_ACP1	R/W	0x0094	RTC 参数 1 寄存器	0x0000	V	
RTC_ACP2	R/W	0x0098	RTC 参数 2 寄存器	0x0000	V	
RTC_ACP3	R	0x009C	RTC 参数 3 寄存器	0x0000		
RTC_ACP4	R/W	0x00A0	RTC 参数 4 寄存器	0x0000	V	
RTC_ACP5	R/W	0x00A4	RTC 参数 5 寄存器	0x0000	V	
RTC_ACP6	R/W	0x00A8	RTC 参数 6 寄存器	0x0000	V	
RTC_ACP7	R/W	0x00AC	RTC 参数 7 寄存器	0x0000	V	
RTC_ACK1	R/W	0x00B0	RTC 参数 k1 寄存器	0x0000	V	
RTC_ACK2	R/W	0x00B4	RTC 参数 k2 寄存器	0x0000	V	
RTC_ACK3	R/W	0x00B8	RTC 参数 k3 寄存器	0x0000	V	
RTC_ACK4	R/W	0x00BC	RTC 参数 k4 寄存器	0x0000	V	
RTC_ACK5	R/W	0x00C0	RTC 参数 k5 寄存器	0x0000	V	
RTC_WKUCNTR	R	0x00CC	记录当前 WKUCNT 值	0x000000		
RTC_ACKTEMP	R/W	0x00D0	RTC K 系数温度分段点控制寄存器	0x3C2800EC	V	

## 13.6 寄存器定义

### 13.6.1 RTC\_SEC/MIN/DAY/WEEK/MONTH/YEAR 寄存器

时间寄存器为 BCD 编码，其中 7~4 位代表十位数，3~0 位为个位数。这些寄存器的值只有当 CE 为高时可以设置，仅当 CE 从 1 清为 0 时，才更新到 RTC 核心。为读取当前时间，用户应该读操作 RTC\_LOAD 寄存器并等待 BSY 清零，否则该寄存器值无效。

表 13-2 RTC\_SEC/MIN/DAY/WEEK/MONTH/YEAR 寄存器

寄存器		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x0000	RTC_SEC, 0~59	-	S40	S20	S10	S8	S4	S2	S1
0x0004	RTC_MIN, 0~59	-	M40	M20	M10	M8	M4	M2	M1
0x0008	RTC_HOUR, 0~23	-	-	H20	H10	H8	H4	H2	H1



寄存器		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x000C	RTC_DAY, 1~31	-	-	D20	D10	D8	D4	D2	D1
0x0010	RTC_WEEK, 0~6	-	-	-	-	-	W4	W2	W1
0x0014	RTC_MON, 1~12	-	-	-	Mo10	Mo8	Mo4	Mo2	Mo1
0x0018	RTC_YEAR, 00~99	Y80	Y40	Y20	Y10	Y8	Y4	Y2	Y1
默认值		X	X	X	X	X	X	X	X
<p>“周”只能由用户设置，RTC 不会自动监测。初始化设置后，RTC 将增加周自动计数。例如，当用户设置 2010/1/1 星期五，RTC 将监测到 2010/01/02 为星期六。W4/W2/W1: 000，星期日；001，星期一；010、星期二；011、星期三；100、星期四；101、星期五；110、星期六。</p> <p>系统只能设置年的最后两位数字，例如 2010 年，RTC_YEAR 应设置为 0b00010000。</p>									

## 13.6.2 RTC\_WKUSEC 寄存器

表 13-3 RTC\_WKUSEC 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5:0	WKUSEC	R/W	该寄存器用于控制多秒唤醒功能。唤醒周期为(WKUSEC+1)*1 秒。当 INTEN[2] 为 1 且内部唤醒秒计数达到目标值，INTSTS[2]将被置 1，而唤醒信号也会生效到 PMU 控制器。由 WKUSEC 产生的第一次中断，如果新的 WKUSEC 值不等于当前 WKUSEC 数将会有<1 秒的误差。如果新的 WKUSEC 等于当前 WKUSEC，第一次中断时间可能有 0~(WKUSEC+1)的变化。为了避免这个问题，设置一个替代的值（如 0）到该寄存器和更新它的 RTC_CE 操作，然后设置正确的值。	0x00

## 13.6.3 RTC\_WKUMIN 寄存器

表 13-4 RTC\_WKUMIN 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5:0	WKUMIN	R/W	该寄存器用于控制多分唤醒功能。唤醒周期为(WKUMIN+1)*1 分。当 INTEN[3] 为 1 且内部唤醒分计数达到目标值，INTSTS[3]将被置 1，而唤醒信号也会生效到 PMU 控制器。由 WKUMIN 产生的第一次中断，如果新的 WKUMIN 值不等于当前 WKUMIN 数，将会有<1 分钟的误差。如果新的 WKUMIN 等于当前 WKUMIN，第一次中断时间可能有 0~(WKUMIN+1)的变化。为了避免这个问题，向这个寄存器设置一个替代的值（如 0）并通过 RTC_CE 操作更新它的值，然后设置正确的值。	0x00

## 13.6.4 RTC\_WKUHOURL 寄存器

表 13-5 RTC\_WKUHOURL 寄存器

Bit	名称	类型	描述	默认值
31:5	-	-	保留	0
4:0	WKUHOURL	R/W	该寄存器用于控制多小时唤醒功能。唤醒周期为(WKUHOURL+1)*1 小时。当 INTEN[4]为 1 且内部唤醒小时	0x00

			计数达到目标值，INTSTS[4]将被置 1，而唤醒信号也会生效到 PMU 控制器。由 WKU HOUR 产生的第一次中断，如果新的 WKU HOUR 值不等于当前 WKU HOUR 数将会有 <1 小时的误差。如果新的 WKU HOUR 等于当前 WKU HOUR，第一次中断时间可能有 0~(WKU HOUR +1)的变化。为了避免这个问题，向这个寄存器设置一个替代的值（如 0）并通过 RTC_CE 操作更新它的值，然后设置正确的值。	
--	--	--	--	--

### 13.6.5 RTC\_WKUCNT 寄存器

表 13-6 RTC\_WKUCNT 寄存器

Bit	名称	类型	描述	默认值
31:26	-	-	保留	0
25:24	CNTSEL	R/W	该寄存器是用来设置 WKUCNT 计数器时钟。 当 PSCA 是 0: 0: 计数器时钟为 32768 Hz。 1: 计数器的时钟 2048 Hz。 2: 计数器的时钟 512 Hz。 3: 计数器的时钟 128 Hz。 当 PSCA 是 1: 0: 计数器的时钟 8192 Hz。 1: 计数器的时钟 2048 Hz。 2: 计数器的时钟 512 Hz。 3: 计数器的时钟 128 Hz。	0x0
23:0	WKUCNT	R/W	该寄存器是用来控制 32K 计数器唤醒功能。唤醒时间为： (WKUCNT+1)*计数器时钟周期，计数器时钟由 CNTSEL 控制。当 INTEN[6]为 1 且内部计数器达到 WKUCNT 设定值，将产生 WKUCNT 周期计数中断或唤醒信号。	0x000000

### 13.6.6 RTC\_CAL 寄存器

表 13-7 RTC\_CAL 寄存器

Bit	名称	类型	描述	默认值
31:14	-	-	保留	0
13:0	CAL	R/W	RTC 32768 校正寄存器，该寄存器是一个 14 位的补码。RTC 会每 30 秒进行一次校正，内部计数器会在 1~29 秒计数，在第 30 秒计为[32768-(CAL-1)]，这样方可让 30 秒内的平均 1 秒脉冲成为精确的 1 秒脉冲。CAL 寄存器的 PPM 分辨率为 1.02ppm，可调范围为±8332.3ppm（±12 分钟/天）。RTC 手动校正时，需手动更新该寄存器的值。	--

### 13.6.7 RTC\_DIV 寄存器

表 13-8 RTC\_DIV 寄存器

Bit	名称	类型	描述	默认值
31:26	-	-	保留	0

25:0	RTCDIV	R/W	该寄存器用来配置 RTC_PLLDIV 输出频率。设置 IOA3 (PMU_IOASEL bit3 置 1) 或 IOA7 (PMU_IOASEL bit7 置 1) 为 RTC_PLLDIV 输出功能, 同时设置 RTC_CTL bit2, 则对应 IOA 输出分频脉冲。输出频率为 $PCLK/(2*(RTCDIV+1))$ 。RTC 手动校正时, 该寄存器需手动更新。	0x00000
------	--------	-----	---	---------

### 13.6.8 RTC\_CTL 寄存器

表 13-9 RTC\_CTL 寄存器

Bit	名称	类型	描述	默认值
31:3	-	-	保留	0
2	RTCPLLOE	R/W	RTC_PLLDIV 输出使能, 该寄存器用于控制 RTC_PLLDIV 分频器输出。 0: 禁止 RTC_PLLDIV 输出。 1: 使能 RTC_PLLDIV 输出。 RTC_PLLDIV 输出引脚为 IOA[3]或 IOA[7], 应该设置 IOA3 (PMU_IOASEL bit3 置 1) 或 IOA7 (PMU_IOASEL bit7 置 1) 为 RTC_PLLDIV 输出使能。	0x0
1:0	-	-	保留	0x0

### 13.6.9 RTC\_PWD 寄存器

表 13-10 RTC\_PWD 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	PWDEN	R	该寄存器用于保护 RTC_CE 寄存器的访问。在访问 RTC_CE 之前, 应写密码 0x5AA55AA5 到 RTC_PWD 寄存器, 此时 PWDEN 将置为 1。任何对 RTC_CE 寄存器的写操作, 都会使 PWDEN 自动清零, 这意味着用户在下次访问 RTC_CE 寄存器之前应该再次向该寄存器写 0x5AA55AA5。	0x0

### 13.6.10 RTC\_CE 寄存器

表 13-11 RTC\_CE 寄存器

Bit	名称	类型	描述	默认值
31:2	-	-	保留	0
1	BSY	R	该位在 CE 从 1 清零或当 RTC_LOAD 寄存器被 CPU 读取后立即置 1, 标志着 RTC 更新过程或 RTC 读取过程正在进行, 该位在读或写过程完成后自动清零。用户可以查询该位检查 RTC 更新是否完成。更新或读取过程需要大约 3 个 32K 周期, 大约 100μs。	0x0
0	CE	R	该寄存器用于解锁对 RTC 寄存器的访问。该位只能在 PWDEN 置为 1 且将 0xA55AA55B 写入 RTC_CE 寄存器后置 1。该位置为 1 后, 可以对 RTC 寄存器进行编程, 但当该位清零后寄存器值才会真正进行更新。要清除该位, 应将 PWDEN 设置为 1, 并向 RTC_CE 寄存器写入 0xA55AA55A。	0x0

## 13.6.11 RTC\_LOAD 寄存器

表 13-12 RTC\_LOAD 寄存器

Bit	名称	类型	描述	默认值
31:0	LOAD	R/W	用户读该寄存器后，当前时间将被锁存到时间寄存器，用户可以从 RTC_SEC~RTC_YEAR 寄存器读取数据。读取过程将需要 3 个 32K 周期，用户可以检查 BSY (RTC_CE bit1) 位判断读取过程是否完成。从该寄存器读取的数据无效。	--

## 13.6.12 RTC\_INTSTS 寄存器

表 13-13 RTC\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
31:9	-	-	保留	0
8	INTSTS8	R/C	中断状态 8，发生非法写 RTC_CE 寄存器时，该位置 1。非法写入是指 BSY 位为 1 时，CE (RTC_CE bit0) 被置 1 或读取 RTC_LOAD 寄存器。写 1 清零该位。	0x0
7	-	-	保留	0
6	INTSTS6	R/C	中断状态 6，当 32K 计数器到达由 WKUCNT 设置的周期时，该位置 1。写 1 清零该位。	0x0
5	INTSTS5	R/C	中断状态 5，该时间到达午夜 (00: 00) 时，该位置 1。写 1 清零该位。	0x0
4	INTSTS4	R/C	中断状态 4，当到达由 WKUHOURL 设置的多小时中断周期时，该位置 1。写 1 清零该位。	0x0
3	INTSTS3	R/C	中断状态 3，当到达由 WKUMIN 设置的多分钟中断周期时，该位置 1。写 1 清零该位。	0x0
2	INTSTS2	R/C	中断状态 2，当到达由 WKUSEC 设置的多秒中断周期时，该位置 1。写 1 清零该位。	0x0
1	INTSTS1	R/C	中断状态 1，当非法格式的时间写入 RTC 寄存器时，该位置 1。写 1 清零该位。	0x0
0	-	-	保留	0

## 13.6.13 RTC\_INTEN 寄存器

表 13-14 RTC\_INTEN 寄存器

Bit	名称	类型	描述	默认值
31:9	-	-	保留	0
8	INTEN8	R/W	中断使能 8，当该位为 1 且 INTSTS8 置 1 时，将产生 CE 非法访问中断。	0x0
7	-	-	保留	0
6	INTEN6	R/W	中断使能 6，当该位为 1 且 INTSTS6 置 1 时，将产生 WKUCNT 设置周期计数中断或唤醒信号。	0x0
5	INTEN5	R/W	中断使能 5，当该位为 1 且 INTSTS5 置 1 时，将产生午夜计时中断或唤醒信号。	0x0
4	INTEN4	R/W	中断使能 4，当该位为 1 且 INTSTS4 置 1 时，将产生多小时中断或唤醒信号。	0x0

3	INTEN3	R/W	中断使能 3, 当该位为 1 且 INTSTS3 置 1 时, 将产生多分钟中断或唤醒信号。	0x0
2	INTEN2	R/W	中断使能 2, 当该位为 1 且 INTSTS2 置 1 时, 将产生多秒中断或唤醒信号。	0x0
1	INTEN1	R/W	中断使能 1, 当该位为 1 且 INTSTS1 置 1 时, 将产生非法格式时间写入中断或唤醒信号。	0x0
0			保留	0

### 13.6.14 RTC\_PSCA 寄存器

表 13-15 RTC\_PSCA 寄存器

Bit	名称	类型	描述	默认值
31:2	-	-	保留	0
1:0	PSCA	R/W	该寄存器用于控制 RTCCLK 预分频器。RTCCLK 时钟速率慢时, 可以显著降低浅睡眠或深睡眠模式下的功耗。 0: 无预分频器, RTCCLK 为 32768 Hz。 1: 1/4 预分频器, RTCCLK 为 8192 Hz。 2~3: 保留。 当该寄存器配置为 1 时, 所有使用 RTCCLK 的模块都将受到影响。硬件会自动调整 RTC 计数器时钟和 LCD 帧速率, 因此不需手动修改设置。但是 RTC_WKUCNT 的输入时钟将会改变, 用户需要修改相应的配置; UART32K 波特率寄存器需要根据新的 RTCCLK 重新计算。	0x0

### 13.6.15 RTC\_ACTI 寄存器

表 13-16 RTC\_ACTI 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	ACTI	R/W	ACTI 表示 RTC 校正期间晶体中心温度的 Ti 值, 其值需要从 Info 区加载。该寄存器是 16 位的补码, 其中, 高 8 位表示整数部分, 低 8 位表示小数部分。例如, 0x1880 表示 24.5 度: $24.5 \times 256 = 6272 = 0x1880$ ; 0xE780 表示 -24.5 度: $\sim 0x1880 + 1 = 0xE780$ 。 该寄存器受 RTC 写保护。	0x1800

### 13.6.16 RTC\_ACF200 寄存器

表 13-17 RTC\_ACF200 寄存器

Bit	名称	类型	描述	默认值
31:26	-	-	保留	0
25:0	F200	R/W	该寄存器存储了用于计算 PLLDIV 的值, 即 PCLK/2。例如, 当 PCLK 为 13107200, $PCLK/2 = 6553600 = 0x640000$ 。该寄存器受 RTC 写保护。	0x640000

## 13.6.17 RTC\_ACP0 寄存器

表 13-18 RTC\_ACP0 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	符号扩展位	-
15:0	P0	R/W	P0 是 16 位的补码，用于温度校正。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0x0000

## 13.6.18 RTC\_ACP1 寄存器

表 13-19 RTC\_ACP1 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	符号扩展位	-
15:0	P1	R/W	P1 是 16 位的补码，用于温度校正。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0x0000

## 13.6.19 RTC\_ACP2 寄存器

表 13-20 RTC\_ACP2 寄存器

Bit	名称	类型	描述	默认值
31:0	P2	R/W	P2 是 32 位的补码，用于温度校正。需从 Info 区加载数据，详情参考表 13-33。该寄存器的值计算公式如下： $P2 = P2' + (Tr - Tm) * 256$ P2' 为 Info 区 0x40D04 地址存储值。 Tr 为 Info 区 0x40D70 地址存储值。 Tm 为 Info 区 0x40D74 地址存储值。 该寄存器受 RTC 写保护。	0x00000000

## 13.6.20 RTC\_ACP3 寄存器

表 13-21 RTC\_ACP3 寄存器

Bit	名称	类型	描述	默认值
31:0	P3	R	P3 是 16 位补码，用于频率校正。需从 Info 区加载数据，详情参考表 13-33。RTC_ACKTEMP 寄存器中的 4 个温度值划分温度段为 5 个区间，匹配校正后的温度 T 处于哪个区间，确定温度区间系数 Kx(x=1~5)并将系数存储至 RTC_ACP3 寄存器。该寄存器受 RTC 写保护。	0x00000000

## 13.6.21 RTC\_ACP4 寄存器

表 13-22 RTC\_ACP4 寄存器

Bit	名称	类型	描述	默认值
31:16	-	--	符号扩展位	-
15:0	P4	R/W	P4 是晶体常温偏移量，是 16 位补码。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0x0000



### 13.6.22 RTC\_ACP5 寄存器

表 13-23 RTC\_ACP5 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	符号扩展位	-
15:0	P5	R/W	P5 是 16 位补码, 用于 RTC_CAL 寄存器校正值计算。需从 Info 区加载数据, 详情参考表 13-33。该寄存器受 RTC 写保护。	0x0000

### 13.6.23 RTC\_ACP6 寄存器

表 13-24 RTC\_ACP6 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	符号扩展位	-
15:0	P6	R/W	P6 是 16 位补码, 用于 RTC_DIV 寄存器存储值计算。P6 计算公式如下: $P6 = a * P6'$ $a = PCLK / 6553600$ 。P6' 为 Info 区加载值, 详情参考表 13-33。该寄存器受 RTC 写保护。	0x0000

### 13.6.24 RTC\_ACP7 寄存器

表 13-25 RTC\_ACP7 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	符号扩展位	-
15:0	P7	R/W	P7 是 16 位补码, 用于频率校正。需从 Info 区加载数据, 详情参考表 13-33。该寄存器受 RTC 写保护。	0x0000

### 13.6.25 RTC\_ACKx 寄存器

表 13-26 RTC\_ACKx 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	符号扩展位	-
15:0	K1~K5	R/W	K1~K5 是 16 位补码, 用于频率校正。需从 Info 区加载数据, 详情参考表 13-33。RTC 手动温度补偿时, 用户需根据计算的温度 T 手动匹配 Kx (x=1~5), 并根据 Kx 进行频率校正。不需存储值到 RTC_ACP3 寄存器。该寄存器受 RTC 写保护。  K1: $T < KTEMP1$ 。 K2: $KTEMP1 \leq T < KTEMP2$ 。 K3: $KTEMP2 \leq T < KTEMP3$ 。 K4: $KTEMP3 \leq T < KTEMP4$ 。 K5: $T \geq KTEMP4$ 。	0x0000

### 13.6.26 RTC\_ACKTEMP 寄存器

表 13-27 RTC\_ACKTEMP 寄存器



Bit	名称	类型	描述	默认值
31:24	KTEMP4	R/W	KTEMP4 是一个 8 位的补码，用来控制温度分段节点 4。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0x3C
23:16	KTEMP3	R/W	KTEMP3 是一个 8 位的补码，用来控制温度分段节点 3。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0x28
15:8	KTEMP2	R/W	KTEMP2 是一个 8 位的补码，用来控制温度分段节点 2。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0x00
7:0	KTEMP1	R/W	KTEMP1 是一个 8 位的补码，用来控制温度分段节点 1。需从 Info 区加载数据，详情参考表 13-33。该寄存器受 RTC 写保护。	0xEC

### 13.6.27 RTC\_WKUCNTR 寄存器

表 13-28 RTC\_WKUCNTR 寄存器

Bit	名称	类型	描述	默认值
31:24	-	-	保留。	0
23:0	WKUCNTR	R	记录当前 WKUCNT 值。数值递增，数据范围：0 ~ WKUCNT。从 0 开始累加，到达 WKUCNT 设定值后，从 0 开始继续累加。	--

## 13.7 Info 参数寄存器(与 RTC 温度补偿有关)

Info 信息存储在以 0x00040800 为起始地址的区域，只能读不能写。所有信息都有双备份，表格中第一份数据用 1 表示，第二份用 2 表示。每份数据都带校验（Checksum），校验算法：各份中的数据项的数据相加，对该相加后的结果取反。

其中地址 0x00040800~0x0004085C 数据，通过专用烧写工具（脱机烧写并开启 RTC 校正功能）写入。其它数据出厂前已写入。

表 13-29 Info 信息寄存器(与 RTC 温度补偿有关)

地址	符号	信息	描述
0x00040800	P4	晶体常温偏移 1	unit(0.1ppm)，低 16 位加载到 RTC_ACP4 寄存器，比如 0。
0x00040804		Checksum1	INV(SUM(0x40800, 0x40800))
0x00040808	P4	晶体常温偏移 2	unit(0.1ppm)，低 16 位加载到 RTC_ACP4 寄存器，比如 0。
0x0004080C		Checksum1	INV(SUM(0x40808, 0x40808))
0x00040810	K1	晶体 K1 系数 1	加载到 RTC_ACK1 寄存器，K1 计算公式如下： $K1 = B1 / 1000000 * 65536$ ，B1 为晶振曲线二次函数的分段系数。K1 值比如 20827。
0x00040814	K2	晶体 K2 系数 1	加载到 RTC_ACK2 寄存器，K2 计算公式如下： $K2 = B2 / 1000000 * 65536$ ，B2 为晶振曲线二次函数的分段系数。K2 值比如 21496。
0x00040818	K3	晶体 K3 系数 1	加载到 RTC_ACK3 寄存器，K3 计算公式如下： $K3 = B3 / 1000000 * 65536$ ，B3 为晶振曲线二次函数的分段系数。K3 值比如 22020。
0x0004081C	K4	晶体 K4 系数 1	加载到 RTC_ACK4 寄存器，K4 计算公式如下： $K4 = B4 / 1000000 * 65536$ ，B4 为晶振曲线二次函数的分段系数。K4 值比如 24517。



0x00040820	K5	晶体 K5 系数 1	加载到 RTC_ACK5 寄存器，K5 计算公式如下： $K5=B5/1000000*65536$ ，B5 为晶振曲线二次函数的分段系数。K5 值比如 25257。
0x00040824		Checksum1	INV(SUM(0x40810, 0x40820))
0x00040828	K1	晶体 K1 系数 2	加载到 RTC_ACK1 寄存器，K1 计算公式如下： $K1=B1/1000000*65536$ ，B1 为晶振曲线二次函数的分段系数。K1 值比如 20827。
0x0004082C	K2	晶体 K2 系数 2	加载到 RTC_ACK2 寄存器，K2 计算公式如下： $K2=B2/1000000*65536$ ，B2 为晶振曲线二次函数的分段系数。K2 值比如 21496。
0x00040830	K3	晶体 K3 系数 2	加载到 RTC_ACK3 寄存器，K3 计算公式如下： $K3=B3/1000000*65536$ ，B3 为晶振曲线二次函数的分段系数。K3 值比如 22020。
0x00040834	K4	晶体 K4 系数 2	加载到 RTC_ACK4 寄存器，K4 计算公式如下： $K4=B4/1000000*65536$ ，B4 为晶振曲线二次函数的分段系数。K4 值比如 24517。
0x00040838	K5	晶体 K5 系数 2	加载到 RTC_ACK5 寄存器，K5 计算公式如下： $K5=B5/1000000*65536$ ，B5 为晶振曲线二次函数的分段系数。K5 值比如 25257。
0x0004083C		Checksum2	INV(SUM(0x40828, 0x40838))
0x00040840	ACTI	晶体顶点温度 1	加载到 RTC_ACTI 寄存器，比如：0x1800。
0x00040844		Checksum1	INV(SUM(0x40840, 0x40840))
0x00040848	ACTI	晶体顶点温度 2	加载到 RTC_ACTI 寄存器，比如：0x1800。
0x0004084C		Checksum2	INV(SUM(0x40848, 0x40848))
0x00040850	KTEMPx (x=4~1)	晶体 K 系数温度分段区间设置 1	加载到 RTC_ACKTEMP 寄存器，比如 0x3C2800EC。
0x00040854		Checksum1	INV(SUM(0x40850, 0x40850))
0x00040858	KTEMPx (x=4~1)	晶体 K 系数温度分段区间设置 2	加载到 RTC_ACKTEMP 寄存器，比如 0x3C2800EC。
0x0004085C		Checksum2	INV(SUM(0x40858, 0x40858))
0x00040D00	P1/P0	RTC_ACP1/0 设置 1	高 16 位加载到 RTC_ACP1 寄存器，比如：1060；低 16 位加载到 RTC_ACP0 寄存器，比如：-214。
0x00040D04	P2'	RTC_ACP2 设置 1	该地址存储值记为 P2'，比如：-19746971。根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2 寄存器。
0x00040D08	P5	RTC_ACP5 设置 1	高 16 位加载到 RTC_ACP5 寄存器，比如：6444；低 16 位舍弃。
0x00040D0C	P7/P6'	RTC_ACP7/6 设置 1	高 16 位加载到 RTC_ACP7 寄存器，比如：0；低 16 位加载到 P6'。比如：1342。  根据公式 $P6=a*P6'$ ，将 P6 加载至 RTC_ACP6 寄存器，其中 $a=PCLK/6553600$ 。
0x00040D10		Checksum1	INV(SUM(0x40D00, 0x40D0C))
0x00040D14	P1/P0	RTC_ACP1/0 设置 2	高 16 位加载到 RTC_ACP1 寄存器，比如：1060；低 16 位加载到 RTC_ACP0 寄存器，比如：-214。
0x00040D18	P2'	RTC_ACP2 设置 2	该地址存储值记为 P2'，比如：-19746971。根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2 寄存器。

0x00040D1C	P5	RTC_ACP5 设置 2	高 16 位加载到 RTC_ACP5 寄存器，比如：6444；低 16 位舍弃。
0x00040D20	P7/P6'	RTC_ACP7/6 设置 2	高 16 位加载到 RTC_ACP7 寄存器，比如：0；低 16 位加载到 P6'。比如：1342。 根据公式 $P6=a*P6'$ ，将 P6 加载至 RTC_ACP6 寄存器，其中 $a=PCLK/6553600$ 。
0x00040D24		Checksum2	$INV(SUM(0x40D14, 0x40D20))$
0x00040D70	Tr	实际温度 1 (来自 tmp275)	根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2 寄存器。
0x00040D74	Tm	测量温度 1 (来自 ADC)	
0x00040D78		Temp Check sum 1	$INV(SUM(0x40D70, 0x40D74))$
0x00040D7C	Tr	实际温度 2 (来自 tmp275)	根据公式： $P2=P2'+(Tr-Tm)*256$ ，将 P2 加载到 RTC_ACP2 寄存器。
0x00040D80	Tm	测量温度 2 (来自 ADC)	
0x00040D84		Temp Check sum 2	$INV(SUM(0x40D7C, 0x40D80))$

## 13.8 应用笔记

### 13.8.1 温度校正

跟据下式计算温度  $T$ ， $T$  为实际温度的 256 倍。。

$$T = ((P0 * ((X * X) >> 16)) + P1 * X + P2) >> 8$$

其中  $X$  为 ADC 温度采样值，为 16 位补码。

$P0$ - $P2$  是寄存器 RTC\_ACP0~RTC\_ACP2 的存储值，被用作温度校正。

实际温度计算结果为： $T' = T / 256$

### 13.8.2 频率误差校正

(1) 芯片频率误差计算公式如下：

$$Delta = (P3 * ((T - Ti)^2 >> 16)) >> 16 + P4$$

其中， $Delta$  是 32K 晶体的频率偏差率（单位 0.1ppm）。

$T$ =温度计算值。

$Ti$  为 RTC 校正期间晶体中心温度的  $Ti$  值，即 RTC\_ACTI 寄存器存储值。

$P4$  位晶体常温偏差值。

$P3$  是当前温度  $T$  所在温度区间的  $K$  系数。RTC 手动温度补偿时，用户需根据计算的温度  $T$  手动匹配  $Kx$  ( $x=1\sim5$ )，并根据  $Kx$  进行频率校正。不需存储值到 RTC\_ACP3 寄存器。

(2) 频率校正值计算公式如下：

$$RTC\_CAL = (Delta * P5) >> 16$$

$P5$  是  $RTC\_ACP5$  寄存器的值，或根据公式： $P5 = 65536 / 10 / (1 / 30 / 32768 / RTCCLK\text{分频值})$  计算，即  $RTC\_CAL$  值为 1 时  $P5$  的值，式中  $(1 / 30 / 32768 / RTCCLK\text{分频值})$  为寄存器  $RTC\_CAL$  的 PPM 分辨率。

(3) PLL 分频秒脉冲输出计算公式如下：

$$RTC\_DIV = PCLK / 2 - (\Delta * P6) \gg 12 - 1$$

$P6$  是  $RTC\_ACP6$  寄存器的值，或根据公式： $P6 = PCLK * 2.048 * 10^{-4}$  计算。

### 13.8.3 RTC 手动温度补偿

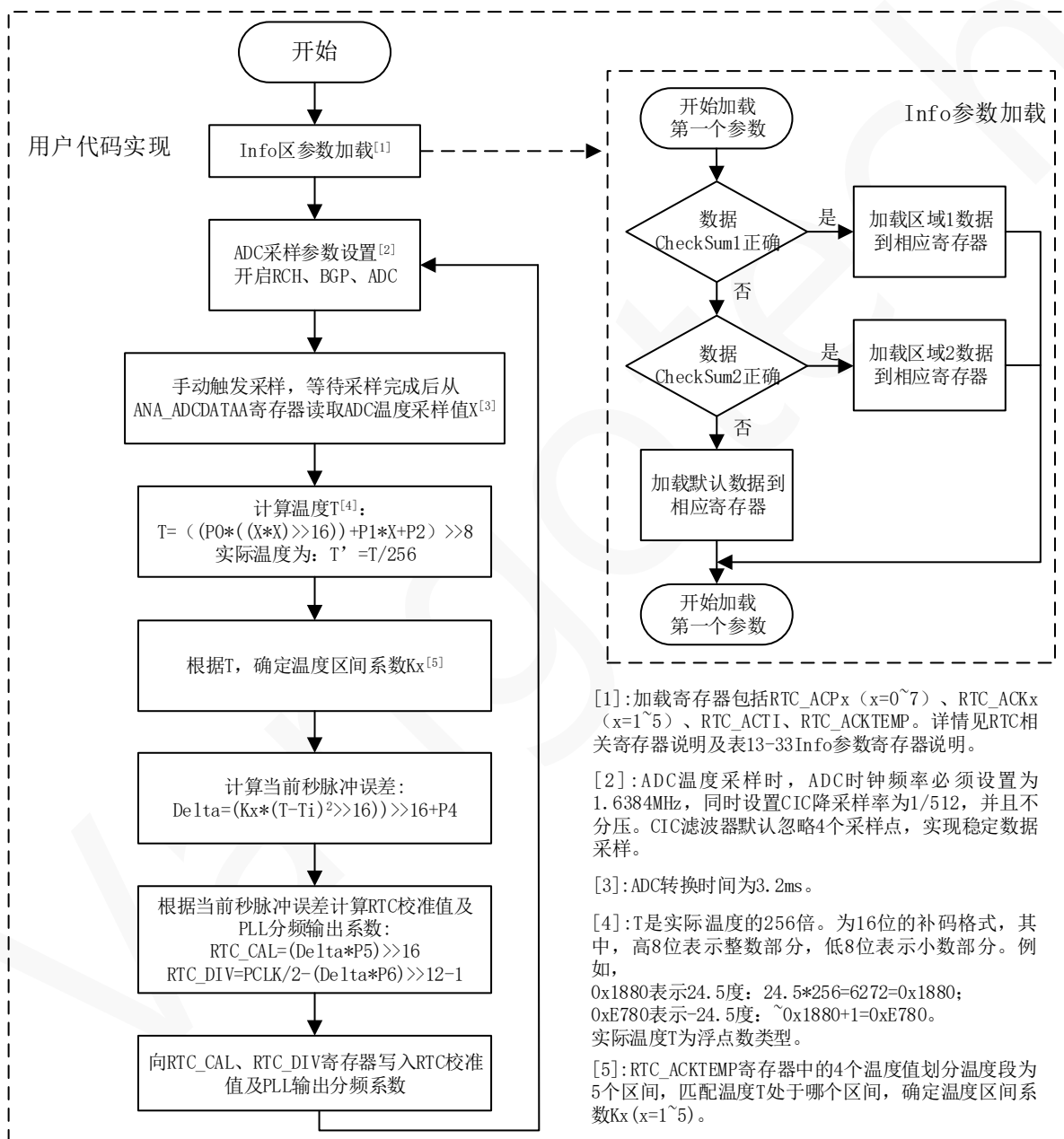


图 13-2 RTC 手动温度补偿流程图

# 第14章 FLASH 控制器

## 14.1 简介

FLASH 控制器用于控制内置 FLASH 的读/写程序。它支持字节/半字/字程序和扇区/芯片擦除。可编程时序可由 MISC 控制器中的 1USCYCLE 控制。FLASH 控制器中的设置将在从浅睡眠或深睡眠模式唤醒后复位，用户应在从这两种状态唤醒后手动恢复设置。

## 14.2 特点

- 支持字节/半字/字编程。
- 支持扇区/芯片擦除。
- 支持 standby（普通待机）模式，当 MCU 没有访问 flash，flash 自动进入 standby 模式。任何对 flash 的访问操作，都会让 flash 从 standby 唤醒，唤醒时间为 0。
- 支持 deep-standby（深度待机）模式，当 MCU 进入 sleep 模式或 deepsleep 模式，flash 自动进入 deep-standby 模式。同时支持手动配置 flash 的 FLASH\_DSTB 寄存器进入 deep-standby 模式。任何对 flash 的访问操作，都会让 flash 从 deep-standby 唤醒，唤醒时间为 10 个 1μstick 时间。
- 当程序在 FLASH 中运行，并且 FLASH 未准备就绪时，MCU 会自动挂起。当 FLASH 准备就绪，MCU 自动从挂起中恢复。
- 支持 Info 扇区读。
- 支持 FLASH 配置寄存器读/写。
- 可编程的编程速度。
- 支持硬件自动校验和比对功能，支持校验和错误中断。

## 14.3 功能框图

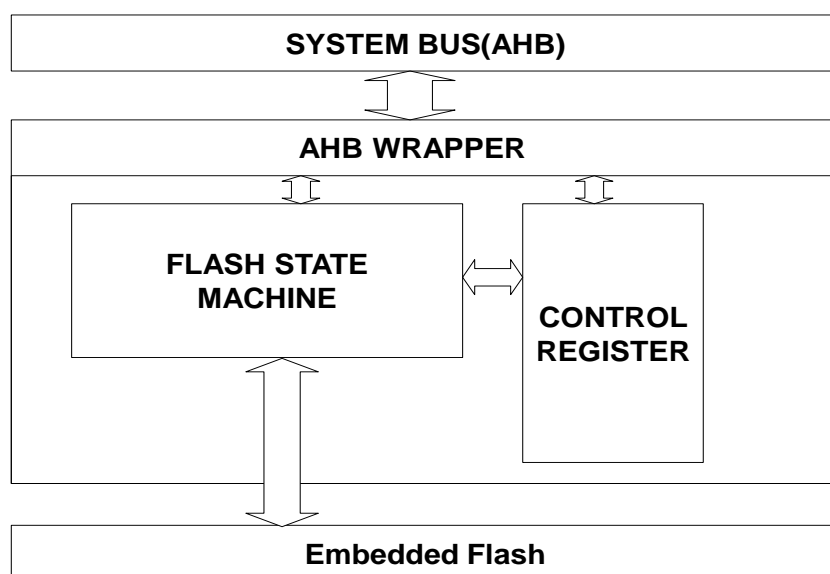


图 14-1 FLASH 控制器的功能框图

## 14.4 寄存器地址

表 14-1 FLASH 控制器地址(FLASH 控制器基地址: 0x00000000)

名称	类型	地址	描述	默认值
FLASH_STS	R	0xFFFFBC	FLASH 编程状态寄存器	0x00
FLASH_INT	R/C	0xFFFFCC	FLASH 校验和中断状态	0x0
FLASH_CSSADDR	R/W	0xFFFFD0	FLASH 校验和开始地址	0x00000
FLASH_CSEADDR	R/W	0xFFFFD4	FLASH 校验和结束地址	0x3FFFC
FLASH_CSVALUE	R	0xFFFFD8	FLASH 校验和计算值寄存器	--
FLASH_CSCVALUE	R/W	0xFFFFDC	FLASH 校验和预期值设置寄存器	0x00000000
FLASH_PASS	R/W	0xFFFFE0	FLASH 密码寄存器	0x00000000
FLASH_CTRL	R/W	0xFFFFE4	FLASH 控制寄存器	0x0
FLASH_PGADDR	R/W	0xFFFFE8	FLASH 程序地址寄存器	0x00000
FLASH_PGDATA	R/W	0xFFFFEC	FLASH 程序字数据寄存器	--
FLASH_PGB0	R/W	0xFFFFEC	FLASH 程序字节数据寄存器 0	--
FLASH_PGB1	R/W	0xFFFFED	FLASH 程序字节数据寄存器 1	--
FLASH_PGB2	R/W	0xFFFFEE	FLASH 程序字节数据寄存器 2	--
FLASH_PGB3	R/W	0xFFFFEF	FLASH 程序字节数据寄存器 3	--
FLASH_PGHW0	R/W	0xFFFFEC	FLASH 程序半字数据寄存器 0	--
FLASH_PGHW1	R/W	0xFFFFEE	FLASH 程序半字数据寄存器 1	--
FLASH_SERASE	R/W	0xFFFFF4	FLASH 扇区擦除控制寄存器	0x00000000
FLASH_CERASE	R/W	0xFFFFF8	FLASH 芯片擦除控制寄存器	0x00000000
FLASH_DSTB	R/W	0xFFFFFC	FLASH deep-standby 控制寄存器	0x00000000

表 14-2 MISC2 控制器 for FLASH(MISC2 基地址: 0x40013E00)

名称	类型	地址	描述	默认值
MISC2_FLASHWC	R/W	0x0000	FLASH 等待周期寄存器	0x2100

## 14.5 寄存器定义

### 14.5.1 FLASH\_PASS 寄存器

表 14-3 FLASH\_PASS 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	UNLOCK	R	UNLOCK 位用于指示 FLASH 是否已解锁。要设置此位, 用户应将 0x55AAAA55 写入此寄存器。该位将保持高电平, 直到用户将任何其他值写入该寄存器。如果 UNLOCK 位为 0, 则将禁止所有 FLASH 编程或擦除或写配置操作。	0x0

## 14.5.2 FLASH\_CTRL 寄存器

表 14-4 FLASH\_CTRL 寄存器

Bit	名称	类型	描述	默认值
31:4	-	-	保留	0
2	CSINTEN	R/W	该寄存器用于控制校验和错误的中断使能。 0: 禁止校验和错误中断。 1: 使能校验和错误中断。	0x0
1:0	CSMODE	R/W	该寄存器用于控制校验和工作模式，用于控制校验和计算以及比对。 0: 关闭校验和模式。 1: 校验和计算和比对工作始终进行。 2: 每一次定时器 2 溢出时做一次校验和计算和比对。 3: 每个 RTC 秒脉冲的上升沿做一次校验和计算和比对。	0x0

## 14.5.3 FLASH\_PGADDR 寄存器

表 14-5 FLASH\_PGADDR 寄存器

Bit	名称	类型	描述	默认值
31:18	-	-	保留	0
17:0	PGADDR	R/W	flash 编程地址设置。这是 FLASH IP 的字节地址，有关每种模式下的地址的详细信息，请参见表 14-6。当编程成功完成后，该地址将自动递增。	0x000000

表 14-6 PGADDR 及写数据寄存器

模式	PGADDR[17:2]	PGADDR[1:0]	写数据寄存器
字编程模式	0x0000~0xFFFF	0	FLASH_PGDATA
字节编程模式	0x0000~0xFFFF	0	FLASH_PGB0
		1	FLASH_PGB1
		2	FLASH_PGB2
		3	FLASH_PGB3
半字编程模式	0x0000~0xFFFF	0	FLASH_PGHWO
		2	FLASH_PGHW1
页擦除	0x0000~0xFFFF	0	FLASH_SERASE

## 14.5.4 FLASH\_PGDATA 寄存器

表 14-7 FLASH\_PGDATA 寄存器

Bit	名称	类型	描述	默认值
31:0	PGDATA	R/W	该寄存器用于控制字编程数据。当 UNLOCK 为 1 时，写入该寄存器将自动触发一个 4 字节的编程。在此期间禁止对 FLASH IP 的任何访问，直到编程完成。FLASH 控制器内部有两个 32 位 FIFO，因此用户可以在没有中断的情况下进行连续编程。但是如果用户需要访问 FLASH IP，比如执行程序，在这段	



			时间内 FLASH 程序的速度会变慢，因为 FLASH IP 需要更多的启动时间。建议将您的编程程序放在 SRAM 中，这样可以实现最大程序速度。	
--	--	--	---	--

### 14.5.5 FLASH\_PGBx 寄存器

表 14-8 FLASH\_PGBx 寄存器

Bit	名称	类型	描述	默认值
7:0	PGBx	R/W	该寄存器用于控制字节编程数据。当 UNLOCK 为 1 时，写入该寄存器将自动触发一个 1 字节的编程。在此期间对 FLASH IP 的任何访问将封闭，直到编程完成。FLASH 控制器内部有一个双 8 位 FIFO，因此用户可以在没有中断的情况下进行连续编程。但是如果用户需要访问 FLASH IP，比如执行程序，在这段时间内 FLASH 程序的速度会变慢，因为 FLASH IP 需要更多的启动时间。建议将您的编程程序放在 SRAM 中，这样可以实现最大程序速度。	

### 14.5.6 FLASH\_PGHWx 寄存器

表 14-9 FLASH\_PGHWx 寄存器

Bit	名称	类型	描述	默认值
15:0	PGHWx	R/W	该寄存器用于控制半字编程数据。当 UNLOCK 为 1 时，写入该寄存器将自动触发一个 2 字节的编程。在此期间对 FLASH IP 的任何访问将封闭，直到程序完成。FLASH 控制器内部有一个双 16 位 FIFO，因此用户可以在没有中断的情况下进行连续编程。但是如果用户需要访问 FLASH IP，比如执行程序，在这段时间内 FLASH 程序的速度会变慢，因为 FLASH IP 需要更多的启动时间。建议将您的编程程序放在 SRAM 中，这样可以实现最大程序速度。	

### 14.5.7 FLASH\_SERASE 寄存器

表 14-10 FLASH\_SERASE 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	SERASE	R	该位用于指示扇区擦除是否正在进行。扇区大小是 512Byte。当 UNLOCK 为 1 且用户将 0xAA5555AA 写入该寄存器时，该位可以置 1，并且在扇区擦除完成后它将自动清零。扇区擦除的地址由 PGADDR[17: 2]控制。扇区擦除时间为 4000*1μs。	0x0

### 14.5.8 FLASH\_CERASE 寄存器

表 14-11 FLASH\_CERASE 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	CERASE	R	该位用来表示芯片全擦除是否正在进行。当 UNLOCK 为 1 并且用户将 0xAA5555AA 写入寄存器，该位可以被置 1，芯片擦除完成后该位将自动清除。芯片擦除时间为 30000*1μs。	0x0

## 14.5.9 FLASH\_DSTB 寄存器

表 14-12 FLASH\_DSTB 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	DSTB	R	<p>该位用于指示 FLASH IP 是否正在进入 deep-standby。当 UNLOCK 为 1 且用户将 0xAA5555AA 写入该寄存器时，该位可以置 1，并且在 FLASH IP 从 deep-standby 模式唤醒后，该位自动清零。Deep-standby 模式可以通过对 FLASH IP 的任何访问退出，但是需要 10 个 1μs 滴答的时间才能使 FLASH IP 恢复正常状态。在此期间，对 FLASH 的任何访问将被过滤，直到 FLASH 准备就绪。进入 deep-standby 后功耗可以降低 65μA。</p> <p>深睡眠和浅睡眠模式下，FLASH 自动进入 deep-standby。</p> <p>IDLE 模式下，用户需要在 SRAM 中运行代码，手动配置该寄存器。</p>	0x0

## 14.5.10 FLASH\_INT 寄存器

表 14-13 FLASH\_INT 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	CSERR	R/C	校验和错误状态位。该标志用于指示以前的校验和错误，也就是说与 CSVALUE/CSCVALUE 不一致。当 CSINTEN 为 1 时，将产生一个中断给 CPU。写 1 可清除该标志。	0x0

## 14.5.11 FLASH\_CSSADDR 寄存器

表 14-14 FLASH\_CSSADDR 寄存器

Bit	名称	类型	描述	默认值
31:18	-	-	保留	0
17:0	CSSADDR	R/W	校验和起始地址寄存器。该寄存器用于控制校验和的起始地址。该值为该寄存器的字节地址，但最低 2 位始终为 0。	0x00000

## 14.5.12 FLASH\_CSEADDR 寄存器

表 14-15 FLASH\_CSEADDR 寄存器

Bit	名称	类型	描述	默认值
31:18	-	-	保留	0
17:0	CSEADDR	R/W	校验和结束地址寄存器。该寄存器用于控制校验和的结束地址。该值是校验和结束地址的字节地址，但最低 2 位始终为 0。校验和范围为 (CSSADDR=<ADDR=<CSEADDR)。	0x3FFFC

## 14.5.13 FLASH\_CSVALUE 寄存器

表 14-16 FLASH\_CSVALUE 寄存器

Bit	名称	类型	描述	默认值
-----	----	----	----	-----



31:0	CSCVALUE	R	校验和计算值寄存器。该寄存器用于表示当前 MCU 自动计算的校验和结果。	0x00000000
------	----------	---	--------------------------------------	------------

## 14.5.14 FLASH\_CSCVALUE 寄存器

表 14-17 FLASH\_CSCVALUE 寄存器

Bit	名称	类型	描述	默认值
31:0	CSCVALUE	R/W	校验和预期值设置寄存器。该寄存器用于设置预期的校验和结果。当校验和完成，MCU 自动计算的校验和值 CSCVALUE 与用户设置的预期校验值 CSCVALUE 不匹配时，CSERR 标志将被置 1。校验和预期值的计算方法：将该范围（CSSADDR=<ADDR=<CSEADDR）内的值（无符号 32 位数据类型）全部累加，累加结果保留低 32 位。	0x00000000

## 14.5.15 FLASH\_STS 寄存器

表 14-18 FLASH\_STS 寄存器

Bit	名称	类型	描述	默认值
31:5	-	-	保留	0
4:0	STS	R	FLASH 控制器状态寄存器。 1: FLASH 操作完成。 其他: FLASH 控制器忙碌状态或处于 IDLE 状态。 用户可以查看这个寄存器，确保在禁用 FLASH_PASS 寄存器之前，所有的 FLASH 操作已完成。	0x0

## 14.5.16 MISC2\_FLASHWC 寄存器

表 14-19 MISC2\_FLASHWC 寄存器

Bit	名称	类型	描述	默认值
31:14	-	-	保留	0
13:8	1USCYCLE	R/W	这个寄存器用于 FLASH 控制器以 AHBClock 计算 1μs 的滴答。该设置与 FLASH 的唤醒时间以及 FLASH 的擦除时间、FLASH 的编程时间有关。FLASH 唤醒时间=1μs 的滴答时间*10。必须满足 1μstick>=1μs。 $1\mu\text{stick} = (\text{AHB 时钟周期}) * (1\text{USCYCLE} + 1)$ 。 比如 AHB 时钟频率为 26.2144M，为了保证 1μs 的滴答时间，以及最小的唤醒时间，1USCYCLE 应该设置为 26。此时 FLASH 的唤醒时间为 $27 / 26214400 * 10$ ，大约为 10μs。 比如 AHB 时钟频率为 32.768k，为了保证 1μs 的滴答时间，以及最小的唤醒时间，1USCYCLE 应该设置为 0。此时 FLASH 的唤醒时间是 $1 / 32768 * 10$ ，大约为 305μs。	0x21
7:0	-	-	保留	0

# 第15章 GPIO 控制器

## 15.1 简介

V94XX(A)中最多有 75 个 IO，其中 14 个外部 IO（GPIOA）由 PMU 控制器控制，可以唤醒浅睡眠与深睡眠。其余 IO（GPIOB~GPIOF）由 GPIO 控制器控制，不可以唤醒浅睡眠与深睡眠；浅睡眠下状态保持，深睡眠下状态丢失，深睡眠模式唤醒后用户应手动恢复设置。

## 15.2 特点

- 每个 IO 都可以为输入或输出模式
- 每个 IO 都可以为开漏模式
- 所有 IO 无上拉、下拉电阻
- GPIOA 可以唤醒浅睡眠与深睡眠

## 15.3 功能框图

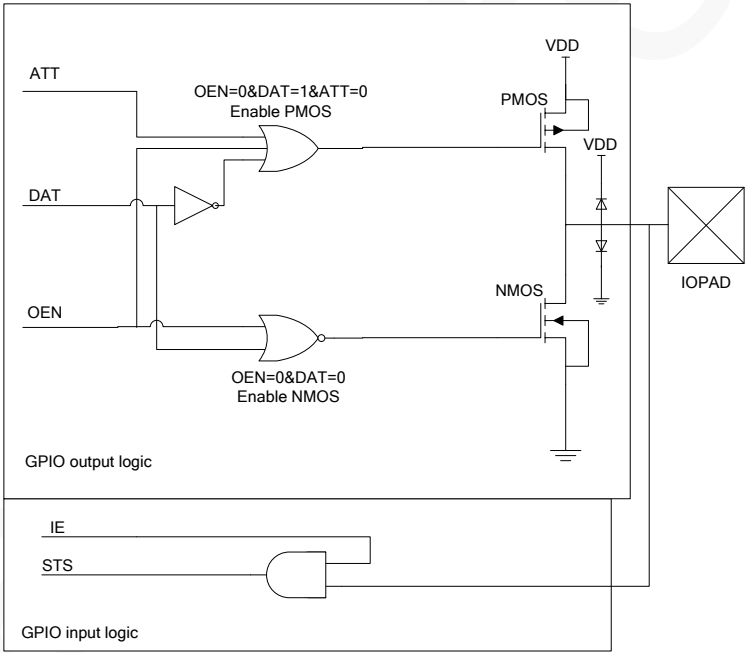


图 15-1 GPIO 控制器的功能框图

## 15.4 寄存器地址

表 15-1 PMU\_IOA 控制器(PMU 基地址: 0x40014000)

名称	类型	地址	描述	默认值
PMU_IOAOEN	R/W	0x0010	IOA 输出使能寄存器	0xFFFF
PMU_IOAIE	R/W	0x0014	IOA 输入使能寄存器	0xFFFF
PMU_IOADAT	R/W	0x0018	IOA 数据寄存器	0x0000
PMU_IOAATT	R/W	0x001C	IOA 属性寄存器	0x0000
PMU_IOAWKUEN	R/W	0x0020	IOA 唤醒使能寄存器	0x00000000

PMU_IOASTS	R	0x0024	IOA 输入状态寄存器	--
PMU_IOAINTSTS	R/C	0x0028	IOA 中断状态寄存器	0x0000
PMU_IOASEL	R/W	0x0038	IOA 特殊功能选择寄存器	0x0000
PMU_IOANODEG	R/W	0x0050	IOA 去抖动电路控制	0x0000

表 15-2 GPIO 控制器(GPIO 基地址: 0x40000000)

名称	类型	地址	描述	默认值
IOB_OEN	R/W	0x0020	IOB 输出使能寄存器	0xFFFF
IOB_IE	R/W	0x0024	IOB 输入使能寄存器	0xFFFF
IOB_DAT	R/W	0x0028	IOB 输出数据寄存器	0x0000
IOB_ATT	R/W	0x002C	IOB 属性寄存器	0x0000
IOB_STS	R	0x0030	IOB 输入状态寄存器	--
IOC_OEN	R/W	0x0040	IOC 输出使能寄存器	0xFFFF
IOC_IE	R/W	0x0044	IOC 输入使能寄存器	0xFFFF
IOC_DAT	R/W	0x0048	IOC 输出数据寄存器	0x0000
IOC_ATT	R/W	0x004C	IOC 属性寄存器	0x0000
IOC_STS	R	0x0050	IOC 输入状态寄存器	--
IOD_OEN	R/W	0x0060	IOD 输出使能寄存器	0xFFFF
IOD_IE	R/W	0x0064	IOD 输入使能寄存器	0xFFFF
IOD_DAT	R/W	0x0068	IOD 输出数据寄存器	0x0000
IOD_ATT	R/W	0x006C	IOD 属性寄存器	0x0000
IOD_STS	R	0x0070	IOD 输入使能寄存器	--
IOE_OEN	R/W	0x0080	IOE 输出使能寄存器	0xFFFF
IOE_IE	R/W	0x0084	IOE 输入使能寄存器	0xFFFF
IOE_DAT	R/W	0x0088	IOE 输出数据寄存器	0x0000
IOE_ATT	R/W	0x008C	IOE 属性寄存器	0x0000
IOE_STS	R	0x0090	IOE 输入使能寄存器	--
IOF_OEN	R/W	0x00A0	IOF 输出使能寄存器	0x7
IOF_IE	R/W	0x00A4	IOF 输入使能寄存器	0x7
IOF_DAT	R/W	0x00A8	IOF 输出数据寄存器	0x0
IOF_ATT	R/W	0x00AC	IOF 属性寄存器	0x0
IOF_STS	R	0x00B0	IOF 输入使能寄存器	--
IOB_SEL	R/W	0x00C0	IOB 特殊功能选择寄存器	0x00
IOE_SEL	R/W	0x00CC	IOE 特殊功能选择寄存器	0x00
IO_MISC	R/W	0x00E0	Iomisc.控制寄存器	0x00

## 15.5 寄存器定义

### 15.5.1 PMU\_IOAOEN 寄存器

表 15-3 PMU\_IOAOEN 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOAOEN	R/W	每个位控制 IOA 的输出使能信号，有关 IO 状态的详细信息，请参见表 15-7。 0：使能 IO 输出模式。 1：禁止 IO 输出模式。	0xFFFF

### 15.5.2 PMU\_IOAIE 寄存器

表 15-4 PMU\_IOAIE 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOAIE	R/W	每个位控制 IOA 的输入使能信号，有关 IO 状态的详细信息，请参见表 15-7。 0：禁止 IO 输入功能。 1：使能 IO 输入功能。	0xFFFF

### 15.5.3 PMU\_IOADAT 寄存器

表 15-5 PMU\_IOADAT 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOADAT	R/W	每个位控制 IOA 的输出数据和拉低/高功能，有关 IO 状态的详细信息，请参见表 15-7。	0x0000

### 15.5.4 PMU\_IOAATT 寄存器

表 15-6 PMU\_IOAATT 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOAATT	R/W	每一位控制 IOA 的属性，下表是不同设置下的 IO 状态的详细信息。当一个 IO 口设置为输出模式（GPIO 或特殊功能），该位用来控制 IO 为 CMOS 或开漏模式。 0：CMOS 模式。 1：开漏模式(禁止 PMOS 输出)。	0x0000

表 15-7 不同设置下的 IO 状态

IOx 设置			IO 状态
IOxOEN	IOxDAT	IOxATT	

1	0	0	禁止输出
1	1	0	禁止输出
1	0	1	禁止输出
1	1	1	禁止输出
0	0	0	输出低电平
0	1	0	输出高电平
0	0	1	开漏输出低电平
0	1	1	开漏拉高

### 15.5.5 PMU\_IOAWKUEN 寄存器

表 15-8 PMU\_IOAWKUEN 寄存器

Bit	名称	类型	描述	默认值
31:0	IOAWKUEN	R/W	每 2 位控制 IOA 的中断和唤醒功能。 位[1:0]: IOA0WKUEN[1:0] 位[3:2]: IOA1WKUEN[1:0] ..... 位[31:30]: IOA15WKUEN[1:0] 有关每种唤醒模式的详细信息, 请参见表 15-9。	0x00000000

表 15-9 IO 唤醒模式

IOAyWKUEN[1: 0]	IOAyDAT	唤醒事件
0	X	禁用唤醒功能
1	0	上升沿唤醒
	1	下降沿唤醒
2	0	高电平唤醒
	1	低电平唤醒
3	X	双边沿唤醒

其中  $y=0\ldots 15$ , 表示某个 IO 口。

硬件在进入浅睡眠或深睡眠模式前锁存 IOA 最后时刻的状态, 当选择上升沿唤醒时, 用户一定要保证在进入睡眠前该 IOA 为低电平, 确保正确检测到 IOA 的上升沿, 否则不能正常唤醒。当选择下降沿唤醒模式时, 用户应保证进入睡眠前该 IOA 状态为高电平, 确保正确检测到 IOA 的下降沿, 否则不能正常唤醒。

电平中断没有去毛刺电路, 包括正常中断和浅睡眠、深睡眠的 IOA 电平唤醒。去毛刺电路仅对浅睡眠、深睡眠模式的 IOA 边沿唤醒信号有效, 去毛刺时间约为 4 个 RTCCLK 时钟周期。

### 15.5.6 PMU\_IOASTS 寄存器

表 15-10 PMU\_IOASTS 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOASTS	R	每个位表示当前 IO 的输入数据值。	--

## 15.5.7 PMU\_IOAINTSTS 寄存器

表 15-11 PMU\_IOAINTSTS 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOAINTSTS	R/C	IO 中断状态寄存器中的每一位代表 IOA 的中断状态。当检测到相应 IOA 唤醒事件时，对应的中断位会置 1。写 1 会清零对应的标志位。	0x0000

## 15.5.8 PMU\_IOASEL 寄存器

表 15-12 PMU\_IOASEL 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	IOA_SEL7	R/W	IOA7 特殊功能选择寄存器。 0: GPIO。 1: 特殊功能 1 (RTC_PLLDIV)。	0x0
6	IOA_SEL6	R/W	IOA6 特殊功能选择寄存器。 0: GPIO。 1: 特殊功能 2 (CMP2_O)。	0x0
5:4	-	-	保留	0
3	IOA_SEL3	R/W	IOA3 特殊功能选择寄存器。 0: GPIO。 1: 特殊功能 1 (RTC_PLLDIV)。	0x0
2:0	-	-	保留	0

## 15.5.9 PMU\_IOANODEG 寄存器

表 15-13 PMU\_IOANODEGL 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	
15:0	IOANODEG	R/W	每一位控制 IOA 边沿唤醒信号是否有去抖动电路。该位设置为 1，相应 IOA 引脚的去抖动电路就失效，唤醒速度变快。去抖动电路影响浅睡眠、深睡眠模式下的边沿唤醒信号。 0: IOA 唤醒信号有去抖动电路。 1: IOA 唤醒信号无去抖动电路。	0x0000

## 15.5.10 IOX\_OEN 寄存器

表 15-14 IOX\_OEN 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0

15:0	IOXOEN*	R/W	每个位控制 IOX 的输出使能信号，有关 IO 状态的详细信息，请参见表 15-7。 0：使能 IO 输出模式。 1：禁止 IO 输出模式。	0xFFFF
------	---------	-----	--	--------

\*：应该保持 IOF\_OEN 寄存器的 bit2 为 1。

## 15.5.11 IOX\_IE 寄存器

表 15-15 IOX\_IE 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOXIE*	R/W	每个位控制 IOX 的输入使能信号，有关 IO 状态的详细信息，请参见表 15-7。 0：禁止 IO 输入功能。 1：使能 IO 输入功能。	0xFFFF

\*：应该保持 IOF\_IE 寄存器的 bit2 为 1。

## 15.5.12 IOX\_DAT 寄存器

表 15-16 IOX\_DAT 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOXDAT	R/W	每个位控制 IOX 的输出数据和拉低/高功能，有关 IO 状态的详细信息，请参见表 15-7。	0x0000

## 15.5.13 IOX\_ATT 寄存器

表 15-17 IOX\_ATT 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOXATT	R/W	该寄存器的每一位用来控制相应 IO 的属性。 0：CMOS 输出 1：开漏输出 当某一个 IO 被配置为一个特殊功能，比如 UART 或 SPI，用户可以设置相应的 IOXATT 为 1，让它处于开漏模式，通过外部上拉输出高。	0x0000

## 15.5.14 IOX\_STS 寄存器

表 15-18 IOX\_STS 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	IOXSTS	R	每个位显示当前 IOX 输入数据值。必须使能输入，才能读到有效的外部电平状态。	--

## 15.5.15 IOB\_SEL 寄存器

表 15-19 IOB\_SEL 寄存器

Bit	名称	类型	描述	默认值
31:7	-	-	保留	0
6	IOB_SEL6	R/W	IOB6 特殊功能选择寄存器。 0: GPIO 或特殊功能 1/2。 1: 特殊功能 1/2 未使能时的特殊功能 3 (RTCCLK 输出)。	0x0
5:3	-	-	保留	0
2	IOB_SEL2	R/W	IOB2 特殊功能选择寄存器。 0: GPIO 或特殊功能 1/2。 1: 特殊功能 1/2 未使能时的特殊功能 3 (PLLL 经过分频器输出)。	0x0
1	IOB_SEL1	R/W	IOB1 特殊功能选择寄存器。 0: GPIO 或特殊功能 1/2。 1: 特殊功能 1/2 未使能时的特殊功能 3 (PLLH 经过分频器输出)。	0x0
0	-	-	保留	0

## 15.5.16 IOE\_SEL 寄存器

表 15-20 IOE\_SEL 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	IOE_SEL7	R/W	IOE7 特殊功能选择寄存器。 0: GPIO 或高阻。 1: 特殊功能 1 (CMP1_O)。	0x0
6:0	-	-	保留	0

## 15.5.17 IO\_MISC 寄存器

表 15-21 IO\_MISC 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5	I2CIOC	R/W	该寄存器用于控制 IOB 或 IOC 的 I <sup>2</sup> C 功能。 0: I <sup>2</sup> C 在 IOB13~IOB14。 1: I <sup>2</sup> C 在 IOC4~IOC5。	0x0
4:3	-	-	保留	0
2:0	PLLHDIV	R/W	当 IOB1 被选择为特殊功能 3 时或 IOB2 被选择为特殊功能 3 时, 该寄存器用于控制 PLLH、PLLL 输出的分频比。 0: /1	0x0



		1: /2 2: /4 3: /8 4: /16 其他: 保留。	
--	--	--	--

## 15.6 特殊功能 IO

### 15.6.1 特殊功能 IOA

下表显示的特殊功能 IOA 的特殊功能将开启时，对应的模块功能。

表 15-22 特殊功能 IOA

IO	特殊功能 1	特殊功能 2	特殊功能 3	特殊功能 4	特殊功能 5
IOA0	SWDCLK (When MODE is 0)		LCDSEG54		
IOA1	SWDIO (When MODE is 0)		LCDSEG53		
IOA3	RTC_PLLDIV		LCDSEG51		
IOA4			LCDSEG61	CMP2_P	
IOA5			LCDSEG62	CMP2_N	
IOA6	CMP2_O		LCDSEG63		
IOA7	RTC_PLLDIV		LCDSEG69		
IOA8			LCDSEG50	ADC_CH3	
IOA9			LCDSEG49	ADC_CH4	
IOA11			LCDSEG47	ADC_CH6	
IOA12	UARTRX0		LCDSEG71		
IOA13	UARTRX1	ISO7816_IO	LCDSEG73		
IOA14	UARTRX2		LCDSEG13		
IOA15			LCDSEG15		

### 15.6.2 特殊功能 IOB

下表显示了 IOB 的特殊功能，特殊功能会在相应模块被使能后打开。但是当特殊功能是 LCD COM/SEG 或 ADC 或 CMP 或 X6\_5M 或 VDCIN 时，用户应该手动禁止相应 GPIO 的输入输出功能，并禁用此 GPIO 上的其他特殊功能，以确保这些特殊功能的正确性。

表 15-23 特殊功能 IOB

IO	特殊功能 1	特殊功能 2	特殊功能 3	特殊功能 4
IOB0	UARTRX4	PWM0		LCDSEG43
IOB1	UARTRX5		PLLH divider	LCDSEG45
IOB2	UARTTX0		PLLL divider	LCDSEG70
IOB3	UARTTX1	ISO7816_CLK		LCDSEG72
IOB4	UARTTX2			LCDSEG12

IOB6	UARTTX4	PWM1	RTCCLK	LCDSEG44
IOB7	UARTTX5			LCDSEG46
IOB9				LCDSEG36
IOB13	I2CSCL (I2CIOCI=0)	PWM2		LCDSEG40
IOB14	I2CSDA (I2CIOCI=0)	PWM3		LCDSEG41
IOB15	TIMER EXT CLK			LCDSEG42

### 15.6.3 特殊功能 IOC

下表显示了 IOC 的特殊功能，特殊功能会在相应模块被使能后打开。但是当特殊功能是 LCD COM/SEG 或 ADC 或 CMP 或 X6\_5M 或 VDCIN 时，用户应该手动禁止相应 GPIO 的输入输出功能，并禁用此 GPIO 上的其他特殊功能，以确保这些特殊功能的正确性。

表 15-24 特殊功能 IOC

IO	特殊功能 1	特殊功能 2	特殊功能 3
IOC0	SPICSN		LCDSEG16
IOC1	SPICLK		LCDSEG17
IOC2	SPIMISO		LCDSEG18
IOC3	SPIMOSI		LCDSEG19
IOC4		I2CSCL(I2CIOCI=1)	LCDSEG20
IOC5		I2CSDA(I2CIOCI=1)	LCDSEG21
IOC6			LCDSEG22
IOC7			LCDSEG23
IOC8			LCDSEG24
IOC9			LCDSEG25
IOC10			LCDSEG26
IOC11			LCDSEG27
IOC12			LCDSEG28
IOC13			LCDSEG29
IOC14			LCDSEG30
IOC15			LCDSEG31

### 15.6.4 特殊功能 IOD

下表显示了 IOD 的特殊功能，特殊功能会在相应模块被使能后打开。但是当特殊功能是 LCD COM/SEG 或 ADC 或 CMP 或 X6\_5M 或 VDCIN 时，用户应该手动禁止相应 GPIO 的输入输出功能，并禁用此 GPIO 上的其他特殊功能，以确保这些特殊功能的正确性。

表 15-25 特殊功能 IOD

IO	特殊功能 1	特殊功能 2
IOD0	LCDCOM0	
IOD1	LCDCOM1	
IOD2	LCDCOM2	
IOD3	LCDCOM3	
IOD4	LCDCOM4	LCDSEG0

IOD5	LCDCOM5	LCDSEG1
IOD6	LCDCOM6	LCDSEG2
IOD7	LCDCOM7	LCDSEG3
IOD8	LCDSEG4	
IOD9	LCDSEG5	
IOD10	LCDSEG6	
IOD12	LCDSEG8	
IOD13	LCDSEG9	
IOD14	LCDSEG10	
IOD15	LCDSEG11	

### 15.6.5 特殊功能 IOE

下表显示了 IOE 的特殊功能，特殊功能会在相应模块被使能后打开。但是当特殊功能是 LCD COM/SEG 或 ADC 或 CMP 或 X6\_5M 或 VDCIN 时，用户应该手动禁止相应 GPIO 的输入输出功能，并禁用此 GPIO 上的其他特殊功能，以确保这些特殊功能的正确性。

表 15-26 特殊功能 IOE

IO	特殊功能 1	特殊功能 2	特殊功能 3
IOE0		LCDSEG74	
IOE1		LCDSEG75	
IOE2		LCDSEG76	
IOE3		LCDSEG77	
IOE4		LCDSEG63	ADC_CH7
IOE5		LCDSEG62	ADC_CH8
IOE6		LCDSEG61	ADC_CH9, Tiny ADC 0
IOE7	CMP1_O	LCDSEG60	ADC_CH11, Tiny ADC 1
IOE8		LCDSEG59	CMP1_P
IOE9			CMP1_N
IOE10		LCDSEG32	
IOE11		LCDSEG33	
IOE13		LCDSEG55	
IOE14		LCDSEG56	

### 15.6.6 特殊功能 IOF

下表显示了 IOF 的特殊功能，特殊功能会在相应模块被使能后打开。但是当特殊功能是 LCD COM/SEG 或 ADC 或 CMP 或 X6\_5M 或 VDCIN 时，用户应该手动禁止相应 GPIO 的输入输出功能，并禁用此 GPIO 上的其他特殊功能，以确保这些特殊功能的正确性。

表 15-27 特殊功能 IOF

IO	特殊功能 1
IOF0	X6_5MI
IOF1	X6_5MO

## 第16章 DMA 控制器

### 16.1 简介

直接存储器存取（DMA）用于提供存储器到存储器或存储器到外设之间的高速数据传输。DMA 控制器集成了 AHB 到 APB 桥接器，因此当 DMA 进行低速外设数据传输时，高速外设总线（AHB）性能不会受到低速外设的影响。在 V94XX(A) 中集成了 4 个 DMA 通道，4 个通道的优先级是循环的。当系统从浅睡眠或深睡眠唤醒时，DMA 控制器的设置会被复位，所以从这两种状态唤醒后应该手动恢复相关设置。

### 16.2 特点

- 支持存储器间的传输；
- 支持存储器到外设传输；
- 支持外设到存储器传输；
- 软件模式或硬件 DMA 请求模式；
- 支持固定地址、单个包传输完成后地址循环或所有包传输完成后地址循环模式；
- 支持单个包传输完成/所有包传输完成/数据传输中止 IRQ 产生；
- 支持总线数据中止检测；
- 集成 AHB 到 APB 桥接；
- 支持 AES128/192/256 编解码。

### 16.3 功能框图

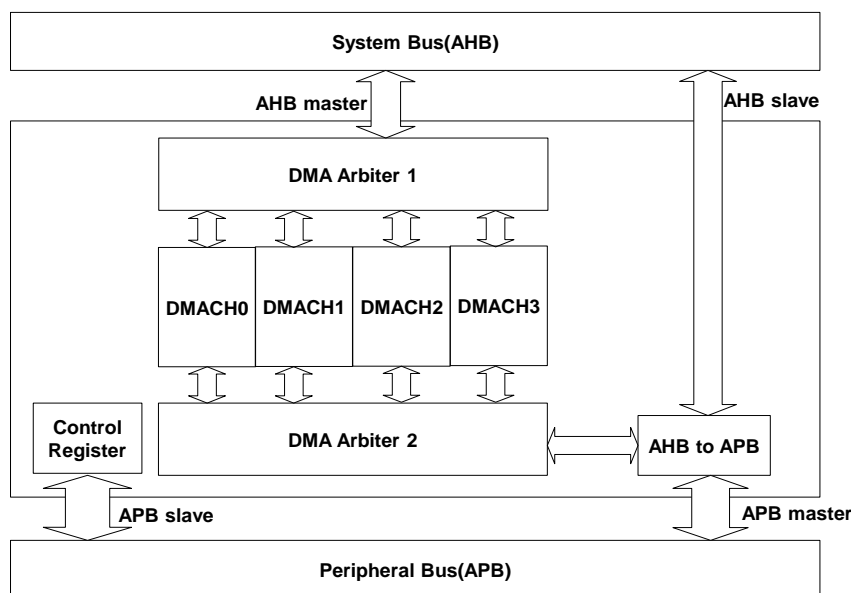


图 16-1 DMA 控制器功能框图

### 16.4 寄存器地址

表 16-1 DMA 控制器(DMA 基地址: 0x40010000)

名称	类型	地址	描述	默认值
----	----	----	----	-----

DMA_IE	R/W	0x0000	DMA 中断使能寄存器	0x000
DMA_STS	R/W	0x0004	DMA 状态寄存器	0x0000
DMA_C0CTL	R/W	0x0010	DMA 通道 0 控制寄存器	0x00000000
DMA_C0SRC	R/W	0x0014	DMA 通道 0 源寄存器	0x00000000
DMA_C0DST	R/W	0x0018	DMA 通道 0 目的寄存器	0x00000000
DMA_C0LEN	R	0x001C	DMA 通道 0 传输长度寄存器	0x0000
DMA_C1CTL	R/W	0x0020	DMA 通道 1 控制寄存器	0x00000000
DMA_C1SRC	R/W	0x0024	DMA 通道 1 源寄存器	0x00000000
DMA_C1DST	R/W	0x0028	DMA 通道 1 目标寄存器	0x00000000
DMA_C1LEN	R	0x002C	DMA 通道 1 传输长度寄存器	0x0000
DMA_C2CTL	R/W	0x0030	DMA 通道 2 控制寄存器	0x00000000
DMA_C2SRC	R/W	0x0034	DMA 通道 2 源寄存器	0x00000000
DMA_C2DST	R/W	0x0038	DMA 通道 2 目标寄存器	0x00000000
DMA_C2LEN	R	0x003C	DMA 通道 2 传输长度寄存器。	0x0000
DMA_C3CTL	R/W	0x0040	DMA 通道 3 控制寄存器	0x00000000
DMA_C3SRC	R/W	0x0044	DMA 通道 3 源寄存器	0x00000000
DMA_C3DST	R/W	0x0048	DMA 通道 3 目的寄存器	0x00000000
DMA_C3LEN	R	0x004C	DMA 通道 3 传输长度寄存器	0x0000
DMA_AESCTL	R/W	0x0050	DMAAES 控制寄存器	0x00000000
DMA_AESKEY0	R/W	0x0060	DMAAES 密钥 0 寄存器	0x00000000
DMA_AESKEY1	R/W	0x0064	DMAAES 密钥 1 寄存器	0x00000000
DMA_AESKEY2	R/W	0x0068	DMAAES 密钥 2 寄存器	0x00000000
DMA_AESKEY3	R/W	0x006C	DMAAES 密钥 3 寄存器	0x00000000
DMA_AESKEY4	R/W	0x0070	DMAAES 密钥 4 寄存器	0x00000000
DMA_AESKEY5	R/W	0x0074	DMAAES 密钥 5 寄存器	0x00000000
DMA_AESKEY6	R/W	0x0078	DMAAES 密钥 6 寄存器	0x00000000
DMA_AESKEY7	R/W	0x007C	DMAAES 密钥 7 寄存器	0x00000000

## 16.5 寄存器定义

### 16.5.1 DMA\_IE 寄存器

表 16-2 DMA\_IE 寄存器

Bit	名称	类型	描述	默认值
31:12	-	-	保留	0
11	C3DAIE	R/W	通道 3 数据中止中断使能。 0: 禁用 1: 启用	0x0
10	C2DAIE	R/W	通道 2 数据中止中断使能。 0: 禁用	0x0

			1: 启用	
9	C1DAIE	R/W	通道 1 数据中止中断使能。 0: 禁用 1: 启用	0x0
8	C0DAIE	R/W	通道 0 数据中止中断使能。 0: 禁用 1: 启用	0x0
7	C3FEIE	R/W	通道 3 所有包传输完成中断使能。 0: 禁用 1: 启用	0x0
6	C2FEIE	R/W	通道 2 所有包传输完成中断使能。 0: 禁用 1: 启用	0x0
5	C1FEIE	R/W	通道 1 所有包传输完成中断使能。 0: 禁用 1: 启用	0x0
4	C0FEIE	R/W	通道 0 所有包传输完成中断使能。 0: 禁用 1: 启用	0x0
3	C3PEIE	R/W	通道 3 单个包传输完成中断使能。 0: 禁用 1: 启用	0x0
2	C2PEIE	R/W	通道 2 单个包传输完成中断使能。 0: 禁用 1: 启用	0x0
1	C1PEIE	R/W	通道 1 单个包传输完成中断使能。 0: 禁用 1: 启用	0x0
0	C0PEIE	R/W	通道 0 单个包传输完成中断使能。 0: 禁用 1: 启用	0x0

## 16.5.2 DMA\_STS 寄存器

表 16-3 DMA\_STS 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15	C3DA	R/C	通道 3 数据中止中断标志, 写 1 清除该标志。	0x0
14	C2DA	R/C	通道 2 数据中止中断标志, 写 1 清除该标志。	0x0
13	C1DA	R/C	通道 1 数据中止中断标志, 写 1 清除该标志。	0x0

12	C0DA	R/C	通道 0 数据中止中断标志，写 1 清除该标志。	0x0
11	C3FE	R/C	通道 3 所有包传输完成中断标志，写 1 清除该标志。	0x0
10	C2FE	R/C	通道 2 所有包传输完成中断标志，写 1 清除该标志。	0x0
9	C1FE	R/C	通道 1 所有包传输完成中断标志，写 1 清除该标志。	0x0
8	C0FE	R/C	通道 0 所有包传输完成中断标志，写 1 清除该标志。	0x0
7	C3PE	R/C	通道 3 单个包传输完成中断标志，写 1 清除该标志。	0x0
6	C2PE	R/C	通道 2 单个包传输完成中断标志，写 1 清除该标志。	0x0
5	C1PE	R/C	通道 1 单个包传输完成中断标志，写 1 清除该标志。	0x0
4	C0PE	R/C	通道 0 单个包传输完成中断标志，写 1 清除该标志。	0x0
3	C3BUSY	R	DMA 通道 3 工作状态寄存器。 0: 空闲 1: 工作	0x0
2	C2BUSY	R	DMA 通道 2 工作状态寄存器。 0: 空闲 1: 工作	0x0
1	C1BUSY	R	DMA 通道 1 工作状态寄存器。 0: 空闲 1: 工作	0x0
0	C0BUSY	R	DMA 通道 0 工作状态寄存器。 0: 空闲 1: 工作	0x0

### 16.5.3 DMA\_CxCTL 寄存器

表 16-4 DMA\_CxCTL 寄存器用于控制 DMA，该寄存器中的数据将在传输开始后锁存到 DMA 通道，因此对该寄存器的任何修改都不会影响正在进行的 DMA 传输。只有 STOP 位可以立即停止 DMA 传输。

表 16-4 DMA\_CxCTL 寄存器

Bit	名称	类型	描述	默认值
31:24	FLEN	R/W	包个数寄存器，DMA 传输数据帧中所有包的包个数为 (FLEN+1)。DMA 传输的总长度为 (FLEN+1) * (PLEN+1)。	0x0
23:16	PLEN	R/W	包长度寄存器，单个包的数据个数为 (PLEN+1)。DMA 传输的总长度为 (FLEN+1) * (PLEN+1)。	0x0
15	STOP	R/W	强制结束 DMA 传输，向此位写入 1 将强制 DMA 通道返回空闲状态，如果另一个 DMA 传输需要分配到同一通道，用户应将此位写零。 0: 默认工作状态 1: 强制结束 DMA 传输	0x0
14	AESEN	R/W	使能 DMA 通道的 AES 加密/解密功能。只有 DMA 通道 3 支持 AES 功能。当启用 AES 功能时，SIZE 应设置为 32 位模式，SMODE 和 DMODE 应设置为 1 或 2。使能 AES 功能时，PLEN 应配置为 4 的整数倍。 0: 禁止 1: 使能	0x0



13	CONT	R/W	连续模式，DMA 传输会持续直到 STOP 位设置为 1 才停止。每次 DMA 完成所有包和帧传输后，它将从头开始执行并连续传输。 0：非连续模式 1：连续模式	0x0
12	TMODE	R/W	传送模式选择寄存器。 0：DMA 请求源每触发一次只传输一个数据。 1：DMA 请求源每触发一次传输一个包数据。	0x0
11:7	DMASEL	R/W	DMA 请求源选择。有关每个 DMA 请求源的详细信息，请参见表 16-5	0x0
6:5	DMODE	R/W	目的地址模式。 0：固定。 1：自动增加但在单个包传输完成后返回起始值。 2：自动增加但在所有包传输完成后返回起始值。 3：保留。	0x0
4:3	SMODE	R/W	源地址模式。 0：固定。 1：自动增加但在单个包传输完成后返回起始值。 2：自动增加但在所有包传输完成后返回起始值。 3：保留。	0x0
2:1	SIZE	R/W	传输长度模式。 0：字节(8 bits) 1：半字(16 bits) 2：全字(32 bits) 3：保留。	0x0
0	EN	R/W	DMA 通道使能寄存器。 0：禁止 DMA 通道(当 CONT 为 1 时无效)。 1：使能 DMA 通道。 当在 CONT 为 0 时，DMA 传输完毕后，该位将由硬件自动清零。	0x0

表 16-5 DMA 请求源选择

DMASEL	请求源	DMASEL	请求源	DMASEL	请求源	DMASEL	Source
0	Software	8	-	16	-	24	UART 32K 0
1	-	9	-	17	-	25	UART 32K 1
2	UART0 TX	10	UART4 TX	18	TIMER 0	26	CMP1
3	UART0 RX	11	UART4 RX	19	TIMER 1	27	CMP2
4	UART1 TX	12	UART5 TX	20	TIMER 2	28	-
5	UART1 RX	13	UART5 RX	21	TIMER 3	29	-
6	UART2 TX	14	ISO7816 TX	22	-	30	SPI TX
7	UART2 RX	15	ISO7816 RX	23	-	31	SPI RX



## 16.5.4 DMA\_CxSRC 寄存器

表 16-6 DMA\_CxSRC 寄存器

Bit	名称	类型	描述	默认值
31:0	SRC	R/W	DMA 源地址寄存器。当 DMA_CxCTL 中的 SIZE 为 1 时，该寄存器必须为半字对齐。当 DMA_CxCTL 中的 SIZE 为 2 时，该寄存器必须全字对齐。当 SRC 设置为 0x4001xxxx 时，它将自动切换到 IO 读取。	0x00000000

## 16.5.5 DMA\_CxDST 寄存器

表 16-7 DMA\_CxDST 寄存器

Bit	名称	类型	描述	默认值
31:0	DST	R/W	DMA 目标地址寄存器。当 DMA_CxCTL 中的 SIZE 为 1 时，该寄存器必须为半字对齐。当 DMA_CxCTL 中的 SIZE 为 2 时，该寄存器必须全字对齐。当 DST 设置为 0x4001xxxx 时，它将自动切换到 IO 写入。	0x00000000

## 16.5.6 DMA\_CxLEN 寄存器

表 16-8 DMA\_CxLEN 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:8	CFLEN	R	当前已传输的包个数。 DMA 已传输的总长度为 $CFLEN * (PLEN + 1) + CPLEN$ 。 在 DMA 完成所有包传输完成后自动清零。	0x00
7:0	CPLEN	R	当前单个传输包中已传输的数据个数。 DMA 已传输的总长度为 $CFLEN * (PLEN + 1) + CPLEN$ 。 在 DMA 完成所有包传输完成后自动清零。	0x00

## 16.5.7 DMA\_AESCTL 寄存器

表 16-9 DMA\_AESCTL 寄存器

Bit	名称	类型	描述	默认值
31:4	-	-	保留	0
3:2	MODE	R/W	AES 模式选择寄存器。 0: AES128 1: AES192 2: AES256 3: 保留。	0x0
1	-	-	保留	0
0	ENC	R/W	AES 编码/解码选择寄存器。 0: 解码 1: 编码	0x0

## 16.5.8 DMA\_AESKEYx 寄存器

表 16-10 DMA\_AESKEYx 寄存器

Bit	名称	类型	描述	默认值
31:0	KEYx	R/W	AES 密钥寄存器： KEY0: 位 31~0 KEY1: 位 63~32 KEY2: 位 95~64 KEY3: 位 127~96 KEY4: 位 159~128 KEY5: 位 191~160 KEY6: 位 223~192 KEY7: 位 255~224  当模式为 AES128 时，仅使用位 127~0。 当模式为 AES192 时，仅使用位 191~0。 当模式为 AES256 时，使用位 255~0。	0x00000000

## 第17章 UART 控制器

### 17.1 简介

UART 控制器用于通过 UART 协议发送/接收数据。V94XX(A)最多有 5 个 UART 控制器，每个 UART 控制器可以单独编程，并有自己的 CPU 中断。UART 控制器可支持无奇偶校验的 8 位数据或有奇偶校验的 9 位数据传输。发送机和接收机都有一个数据缓存器和一个移位寄存器。UART 控制器中的设置将在从浅睡眠或深睡眠模式唤醒后复位，用户应在从这两种状态唤醒后手动恢复设置。UART 的波特率支持范围 300~819200bps。

### 17.2 特点

- 可编程波特率发生器；
- 数据长度为 8 位；
- 带奇偶校验或不带奇偶校验；
- 奇偶校验错误监测；
- 发送/接收中断标志；
- 发送/接收溢出中断标志。

### 17.3 功能框图

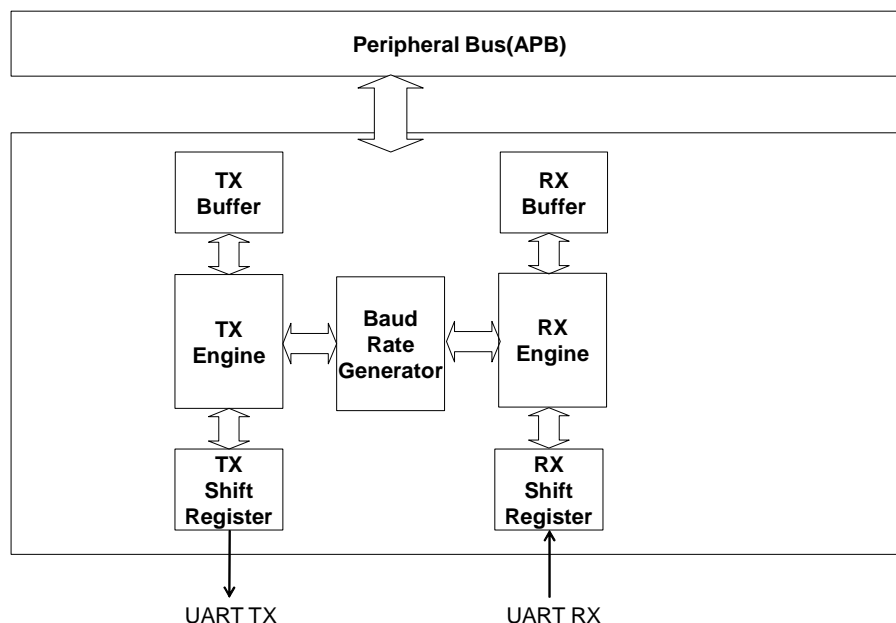


图 17-1 UART 控制器的功能框图

### 17.4 寄存器地址

表 17-1 UART 控制器(UART 基地址: 0x40011800)

名称	类型	地址	描述	默认值
UART0_DATA	R/W	0x0000	UART0 数据寄存器	0x00

UART0_STATE	R/C	0x0004	UART0 状态寄存器	0x00
UART0_CTRL	R/W	0x0008	UART0 控制寄存器	0x000
UART0_INTSTS	R/C	0x000C	UART0 中断状态寄存器	0x00
UART0_BAUDDIV	R/W	0x0010	UART0 波特率分频寄存器	0x00000
UART0_CTRL2	R/W	0x0014	UART0 控制寄存器 2	0x0
UART1_DATA	R/W	0x0020	UART1 数据寄存器	0x00
UART1_STATE	R/C	0x0024	UART1 状态寄存器	0x00
UART1_CTRL	R/W	0x0028	UART1 控制寄存器	0x000
UART1_INTSTS	R/C	0x002C	UART1 中断状态寄存器	0x00
UART1_BAUDDIV	R/W	0x0030	UART1 波特率分频寄存器	0x00000
UART1_CTRL2	R/W	0x0034	UART1 控制寄存器 2	0x0
UART2_DATA	R/W	0x0040	UART2 数据寄存器	0x00
UART2_STATE	R/C	0x0044	UART2 状态寄存器	0x00
UART2_CTRL	R/W	0x0048	UART2 控制寄存器	0x000
UART2_INTSTS	R/C	0x004C	UART2 中断状态寄存器	0x00
UART2_BAUDDIV	R/W	0x0050	UART2 波特率分频寄存器	0x00000
UART2_CTRL2	R/W	0x0054	UART2 控制寄存器 2	0x0
UART4_DATA	R/W	0x0080	UART4 数据寄存器	0x00
UART4_STATE	R/C	0x0084	UART4 状态寄存器	0x00
UART4_CTRL	R/W	0x0088	UART4 控制寄存器	0x000
UART4_INTSTS	R/C	0x008C	UART4 中断状态寄存器	0x00
UART4_BAUDDIV	R/W	0x0090	UART4 波特率分频寄存器	0x00000
UART4_CTRL2	R/W	0x0094	UART4 控制寄存器 2	0x0
UART5_DATA	R/W	0x00A0	UART5 数据寄存器	0x00
UART5_STATE	R/C	0x00A4	UART5 的状态寄存器	0x00
UART5_CTRL	R/W	0x00A8	UART5 控制寄存器	0x000
UART5_INTSTS	R/C	0x00AC	UART5 中断状态寄存器	0x00
UART5_BAUDDIV	R/W	0x00B0	UART5 波特率分频寄存器	0x00000
UART5_CTRL2	R/W	0x00B4	UART5 控制寄存器 2	0x0

表 17-2 MISC 控制器(MISC 基地址: 0x40013000)

名称	类型	地址	描述	默认值
MISC_IREN	R/W	0x000C	IR 使能控制寄存器	0x00
MISC_DUTYL	R/W	0x0010	IR 调制波低电平宽度控制寄存器	0x0000
MISC_DUTYH	R/W	0x0014	IR 调制波高电平宽度控制寄存器	0x0000

## 17.5 寄存器定义

### 17.5.1 UARTx\_DATA 寄存器

表 17-3 UARTx\_数据寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	DATA	R/W	读：接收数据。 写：发送数据。	0x0

### 17.5.2 UARTx\_STATE 寄存器

表 17-4 UARTx\_STATE 寄存器

Bit	名称	类型	描述	默认值
31:7	-	-	保留	0
6	RXPSTS	R	接收奇偶校验标志。该位表示最后接收到的数据的奇偶校验位。	0x0
5	TXDONE	R/C	发送完成标志。当单字节数据发送完成时，该标志将被置 1。向该位写 1 或向 UARTx_INTSTS 寄存器中的 TXDONEIF 位写 1，该位将被清零。	0x0
4	RXPE	R/C	接收数据奇偶校验错误标志位。当接收数据的奇偶校验值与期望不同时，该标志会被置 1。向该位写 1 或向 UARTx_INTSTS 寄存器中的 RXPEIF 位写 1，该位将被清零。	0x0
3	RXOV	R/C	接收缓存区溢出标志。当 RXFULL 为 1 并且又从 RX 接收机接收到一个数据时，该标志将被置 1。向该位写 1 或向 UARTx_INTSTS 寄存器中的 RXOVIF 位写 1，该位将被清零。	0x0
2	TXOV	R/C	发送缓存区溢出标志。当 TXFULL 为 1 并且有一个新数据写入到 UARTx_DATA 寄存器时，该标志将被置 1。向该位写 1 或向 UARTx_INTSTS 寄存器中的 TXOVIF 位写 1，该位将被清零。	0x0
1	RXFULL	R	接收缓存器满标志。 0：接收缓存器为空。 1：接收缓存器已满。 当 UARTRX 接收机接收到数据时，该标志位将被置 1。用户从 UARTx_DATA 读取数据后，该位将被清零。	0x0
0	-	-	保留	0

### 17.5.3 UARTx\_CTRL 寄存器

表 17-5 UARTx\_CTRL 寄存器

Bit	名称	类型	描述	默认值
31:9	-	-	保留	0
8	TXDONEIE	R/W	发送完成中断使能。	0x0
7	RXPEIE	R/W	接收数据奇偶校验错误中断使能。	0x0
6	-	-	保留	0
5	RXOVIE	R/W	接收溢出中断使能。	0x0

4	TXOVIE	R/W	发送溢出中断使能。	0x0
3	RXIE	R/W	接收中断使能。	0
2	-	-	保留。	0x0
1	RXEN	R/W	接收使能。	0x0
0	TXEN	R/W	发送使能。	0x0

## 17.5.4 UARTx\_INTSTS 寄存器

表 17-6 UARTx\_INTSTS 寄存器

Bit	名称	类型	描述	默认值
31:5	-	-	保留	0
5	TXDONEIF	R/C	发送完成标志。当发送单个字节并且 TXDONEIE 为 1 时，该标志位将被置 1。向该位写 1 或向 UARTx_STATE 寄存器中的 TXDONE 位写 1，该位将被清零。	0x0
4	RXPEIF	R/C	接收数据奇偶校验错误标志。当接收数据的奇偶校验值与期望不同时，该标志会被置 1。向该位写 1 或向 UARTx_STATE 寄存器的 RXPE 位写入 1 时，该位将被清零。	0x0
3	RXOVIF	R/C	接收缓存区溢出标志。当 RXOVIE 为 1、RXFULL 为 1 并且从 RX 接收机收到一个新数据时，该标志将被置 1。向该位或向 UARTx_STATE 寄存器中的 RXOV 位写 1，该位将被清零。	0x0
2	TXOVIF	R/C	发送缓存区溢出标志。当 TXOVIE 为 1、TXFULL 为 1 且向 UARTx_DATA 寄存器写入一个新数据，该标志将置 1。向该位写 1 或向 UARTx_STATE 寄存器中的 TXOV 位写 1，该位将被清零。	0x0
1	RXIF	R/C	接收中断标志，当 UARTx_DATA 寄存器接收到数据时，该位置 1。向该位写 1 即可清零。	0x0
0	-	-	保留。	0

## 17.5.5 UARTx\_BAUDDIV 寄存器

表 17-7 UARTx\_BAUDDIV 寄存器

Bit	名称	类型	描述	默认值
31:20	-	-	保留	0
19:0	BAUDDIV	R/W	波特率分频寄存器。在使能 UART 之前必须先设置好该寄存器。 $BAUDDIV = APBCLK / \text{Baud-rate}$ 。 例如当 $APBCLK = 6.5536\text{MHz}$ ，波特率是 9600bps 时，该寄存器值等于 $6553600 / 9600 = 682$ 。	0x0

## 17.5.6 UARTx\_CTRL2 寄存器

表 17-8 UARTx\_CTRL2 寄存器

Bit	名称	类型	描述	默认值
31:4	-	-	保留	0
3:2	PMODE	R/W	奇偶校验模式选择寄存器。该寄存器仅在 MODE 设置为 1 时有效。	0x0

			0: 偶校验。 1: 奇校验。 2: 奇偶校验位始终为 0。 3: 奇偶校验位始终为 1。	
1	MODE	R/W	UART 数据模式控制寄存器。 0: 无校验, 1+8+1 模式。 1: 奇偶校验, 1+8+1+1 模式, 奇偶校验位由 PMODE 寄存器控制。	0x0
0	MSB	R/W	数据位传输顺序控制寄存器。 0: 低位先传 (LSB)。 1: 高位先传 (MSB)。	0x0

## 17.5.7 MISC\_IREN 寄存器

表 17-9 MISC\_IREN 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5:0	IREN	R/W	IR 使能控制寄存器。该寄存器中的每个位对应于 1 个 UARTTX 通道。当 IREN[x]置 1 时, 表示 UARTTX[x]输出为红外载波 (IR) 调制信号。	0x00

## 17.5.8 MISC\_DUTYL 寄存器

表 17-10 MISC\_DUTYL 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	DUTYL	R/W	红外载波 (IR) 调制波低电平宽度控制寄存器。低脉冲宽度将是 (DUTYL + 1) * APBCLK 周期。	0x0000

## 17.5.9 MISC\_DUTYH 寄存器

表 17-11 MISC\_DUTYH 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	DUTYH	R/W	红外载波 (IR) 调制波高电平宽度控制寄存器。高脉冲宽度为 (DUTYH+1) * APBCLK 周期。	0x0000

# 第18章 UART32K 控制器

## 18.1 简介

UART32K 控制器使用 UART 协议接收数据。V94XX(A)中有两个独立的 UART32K 控制器，其输入引脚从普通串口 UART0~UART2 的输入脚（rx0~rx2）中选择。UART32K 控制器的工作频率是 RTCCLK，因此可以在浅睡眠或深睡眠模式下工作。UART32K 支持的典型波特率为 9600，支持数据长度为 8 位，支持无奇偶校验的 8 位数据和有奇偶校验的 9 位数据传输，接收机带一个数据缓冲器和一个移位寄存器。UART32K 的波特率支持范围 300~9600bps。

## 18.2 特点

- 可在浅睡眠或深睡眠模式下工作。
- 具有浅睡眠或深睡眠唤醒功能。
- 可编程波特率发生器。
- 支持 8 位数据传输。
- 支持无奇偶校验的 8 位数据和有奇偶校验的 9 位数据传输。
- 奇偶校验错误检测。
- 接收中断标志。
- 接收奇偶校验错误标志。
- 接收溢出中断标志。

## 18.3 功能框图

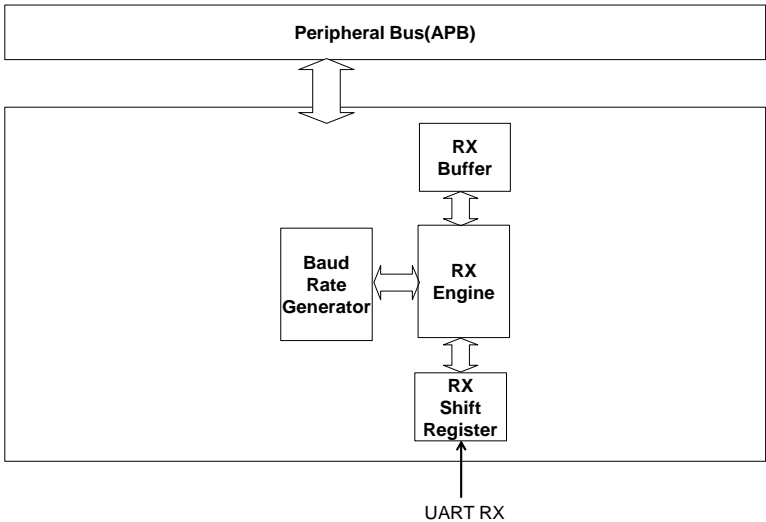


图 18-1 UART32K 控制器的功能框图

## 18.4 寄存器地址

表 18-1 UART32K 控制器(UART32K0 基地址：0x40014100)

名称	类型	地址	描述	默认值
U32K0_CTRL0	R/W	0x0000	UART32K0 控制寄存器 0	0x000



U32K0_CTRL1	R/W	0x0004	UART32K0 控制寄存器 1	0x00
U32K0_PHASE	R/W	0x0008	UART32K0 波特率控制寄存器	0x4B00
U32K0_DATA	R	0x000C	UART32K0 接收数据缓存区	--
U32K0_STS	R/C	0x0010	UART32K0 中断状态寄存器	0x0

**UART32K1 基地址: 0x40014180**

名称	类型	地址	描述	默认值
U32K1_CTRL0	R/W	0x0000	UART32K1 控制寄存器 0	0x00
U32K1_CTRL1	R/W	0x0004	UART32K1 控制寄存器 1	0x00
U32K1_PHASE	R/W	0x0008	UART32K1 波特率控制寄存器	0x4B00
U32K1_DATA	R	0x000C	UART32K1 接收数据缓存区	--
U32K1_STS	R/C	0x0010	UART32K1 中断状态寄存器	0x0

## 18.5 寄存器定义

### 18.5.1 U32Kx\_CTRL0 寄存器

表 18-2 U32Kx\_CTRL0 寄存器

Bit	名称	类型	描述	默认值
31:9	-	-	保留	0
8	WKUMODE	R/W	唤醒模式控制。RXIE=1 时这个寄存器才会生效。 0: 只要接收到数据就会唤醒（不管校验是否正确）。 1: 只有当奇偶校验位和停止位正确才会唤醒。	0x0
7:6	DEBSEL	R/W	消抖控制。 0: 无消抖。 1: 2 个 RTCCLK 时钟周期消抖。 2: 3 个 RTCCLK 时钟周期消抖。 3: 4 个 RTCCLK 时钟周期消抖。 当 UART 总线上有干扰时, 启用消抖功能可在接收数据时降低干扰。但是消抖周期大时可能会使接收数据丢失。	0x0
5:4	PMODE	R/W	奇偶校验模式控制, 仅在 MODE 设置为 1 时有效。 0: 偶校验。 1: 奇校验。 2: 奇偶校验位始终为 0。 3: 奇偶校验位始终为 1。	0x0
3	MODE	R/W	UART 数据模式控制寄存器。 0: 无校验。1+8+1 模式。 1: 有奇偶校验。1+8+1+1 模式, 奇偶校验位由 PMODE 控制。	0x0
2	MSB	R/W	数据位接收顺序控制。 0: 低位先收 (LSB)。 1: 高位先收 (MSB)。	0x0

1	ACOFF	R/W	自动校正控制。为了在 RTCCLK 时钟下正确接收数据，控制器内部有一个硬件自动校正机制。默认情况下，硬件自动校正机制是打开的，向该位写 1 可以关闭硬件自动校正机制。  0：开启自动校正 1：关闭自动校正。	0x0
0	EN	R/W	UART32K 控制器使能。将该位置 1，UART32K 控制器将在两个 RTCCLK 时钟周期后开始接收数据。因此在将该位置 1 之前，必须为各个 U32K 控制器设置正确的参数。  0：禁止 1：使能	0x0

## 18.5.2 U32Kx\_CTRL1 寄存器

表 18-3 U32Kx\_CTRL1 寄存器

Bit	名称	类型	描述	默认值
31:6	-	-	保留	0
5:4	RXSEL	R/W	接收数据源控制。  0：来自 UART RX0(IOA12)。 1：来自 UART RX1(IOA13)。 2：来自 UART RX2(IOA14)。 3：保留。	0x0
3	-	-	保留	0
2	RXOVIE	R/W	接收溢出中断/唤醒使能。	0x0
1	RXPEIE	R/W	接收奇偶校验错误中断/唤醒使能。	0x0
0	RXIE	R/W	接收中断/唤醒使能。使能后，当 WKUMODE 为 0 时，只要接收到串口数据就会唤醒；当 WKUMODE 为 1 时，只有当奇偶校验位和停止位正确才会唤醒。	0x0

## 18.5.3 U32Kx\_PHASE 寄存器

表 18-4 U32Kx\_PHASE 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	PHASE	R/W	波特率分频寄存器，在使能 UART32K 之前必须设置这个寄存器。  $PHASE = 65536 * \text{Baud-rate} / \text{RTCCLK}$ 。  例如，当波特率是 9600bps，RTCCLK 频率为 32768， $65536 * 9600 / 32768 = 19200$ 。当 RTC_PSCA 寄存器的 PSCA=1，PHASE 应该使用 RTCCLK 的分频后的频率进行计算。	0x4B00

## 18.5.4 U32Kx\_DATA 寄存器

表 18-5 U32Kx\_数据寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0

7:0	DATA	R	读：接收数据。	--
-----	------	---	---------	----

18.5.5 U32Kx\_STS 寄存器

表 18-6 U32Kx\_STS 寄存器

Bit	名称	类型	描述	默认值
31:3	-	-	保留	0
2	RXOV	R/C	接收缓存器溢出标志。当接收 FIFO 满并且从 RX 接收机接收到一个新数据时，该标志位置 1。向该位写 1 可将标志位清零。	0x0
1	RXPE	R/C	接收数据奇偶校验错误标志。当接收数据的奇偶校验值错误时，该标志位置 1。向该位写 1 可将标志位清零。	0x0
0	RXIF	R/C	接收中断标志位，当 WKUMODE 为 0 时，只要接收到串口数据就会置 1；当 WKUMODE 为 1 时，只有当奇偶校验位和停止位正确才会置 1。向该位写 1 可将标志位清零。	0x0



ISO7816_CFG	R/W	0x0014	ISO7816 控制寄存器	0x00
ISO7816_CLK	R/W	0x0018	ISO7816 时钟分频器控制寄存器	0x00

## 19.5 寄存器定义

### 19.5.1 ISO7816\_BAUDDIVL 寄存器

表 19-2 ISO7816\_BAUDDIVL 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	BAUDDIVL	R/W	波特率分频器的低字节。	0x0

### 19.5.2 ISO7816\_BAUDDIVH 寄存器

表 19-3 ISO7816\_BAUDDIVH 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	BAUDDIVH	R/W	波特率分频的高字节。波特率计数器是从 BAUDDIV 数到 0xFFFF，所以波特率计算公式为： $BAUDDIV = 0x10000 - (APBCLK / \text{波特率})$ 例如：当 APBCLK=6.5536MHz、波特率=9600bps，寄存器值应该为： $0x10000 - (6553600 / 9600) = 0xFD56$ 。	0x0

### 19.5.3 ISO7816\_DATA 寄存器

表 19-4 ISO7816\_DATA 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	DATA	R/W	读：接收数据，当接收数据没有被取走，又接收到一个新的数据时，OVIF 标志位将会置 1。 写：发送数据，对该端口写将会触发 ISO7816 总线的一次数据发送。	0x0

### 19.5.4 ISO7816\_INFO 寄存器

表 19-5 ISO7816\_INFO 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	OVIF	R/C	接收溢出标志。当已写入数据未被 CPU 读取时，接收到另一个数据，该位将置 1。向该位写 1 可将标志位清零。	0x0
6	SDIF	R/C	发送中断标志。当成功发送一字节数据并收到应答（ACK），该位将置 1。这里，不关心接收应答是 0 还是 1。向该位写 1 可将标志位清零。	0x0

5	RCIF	R/C	接收中断标志。当成功接收一字节数据并发送应答信号 (ACK), 该位将置 1。向该位写 1 可将标志位清零。	0x0
4	LSB	R/W	数据位发送/接收顺序控制寄存器。 0: 高位先传 (MSB)。 1: 低位先传 (LSB)。	0x0
3	SDERR	R/C	发送模式下接收到的应答 (ACK) 为 0 时, 该位将被置 1。向该位写 1 可将标志位清零。	0x0
2	RCERR	R/C	接收数据有校验和错误时, 该位将被置为 1。向该位写 1 可将标志位清零。	0x0
1	CHKSUM	R/W	发送或接收的数据的校验和位。这个位可以被 CPU 读/写, 但原来的信息会丢失。	0x0
0	RCACK	R/W	在发送结束时收到的 ACK。这个位可以被 CPU 读/写, 但原来的信息会丢失。	0x0

### 19.5.5 ISO7816\_CFG 寄存器

表 19-6 ISO7816\_CFG 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	OVIE	R/W	接收溢出中断使能寄存器。	0x0
6	SDIE	R/W	发送中断使能寄存器。	0x0
5	RCIE	R/W	接收中断使能寄存器。	0x0
4	ACKLEN	R/W	接收错误数据时 ACK 低电平长度。 0: 1 位 1: 2 位	0x0
3:2	-	-	保留	0x0
1	CHKP	R/W	奇偶校验模式控制寄存器。 0: 偶校验 1: 奇校验。	0x0
0	EN	R/W	ISO7816 使能寄存器。 0: 禁用 ISO7816 功能。 1: 使能 ISO7816 功能。	0x0

### 19.5.6 ISO7816\_CLK 寄存器

表 19-7 ISO7816\_CLK 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	CLKEN	R/W	ISO7816 时钟输出使能。 0: 禁止时钟输出。 1: 使能时钟输出。	0x0

6:0	CLKDIV	R/W	ISO7816 时钟分频值。 0: APBCLK/1 1: APBCLK/2 2: APBCLK/3 ... 127: APBCLK/128	0x0
-----	--------	-----	---	-----

下表显示了 ISO7816 的数据包格式。

表 19-8 ISO7816 数据包格式

bit	0	1~8	9	10	11	12	0	...
发送方向	发送机→接收机			接收机→发送机		--	发送机→接收机	
定义	Start bit	Data	Parity bit	ACK*		Wait	Start bit	...
值	0	MSB→LSB	P	ACK*		1	0	...
TX	0	MSB→LSB	P	1			0	...
RX	1	1	1	ACK*		1	1	...
描述	10 位数据			3 位保护时间			下一帧	

\*如果 ACK 为 0，则 RX 将在第 10 位中间从 1 复位为 0，在第 11 或第 12 位置为 1，由 ISO7816\_CFG 中的 ACKLEN 控制。

发送包含总共 10 位，包括 1 位起始位、8 位数据和 1 个奇偶校验位。有 3 位的保护时间，接收器检查接收机的正确性并发送 1 位或 2 位的响应 ACK。发送器和接收器在位 12 的位置都应该保持在高电平状态，等待下一个数据包。

## 第20章 Timer/PWM 控制器

### 20.1 简介

V94XX(A)共有 8 个定时器。其中四个是通用 32 位定时器，用于定时功能。另外 4 个定时器是 16 位定时器，具有 PWM 比较功能。每个定时器可以产生 CPU 中断。每个 PWM 可以输出具有不同输出波形的 3 个输出。定时器/PWM 控制器中的设置将在从浅睡眠或深睡眠模式唤醒后复位，用户应在从这两种状态唤醒后手动恢复设置。

### 20.2 特点

- 4 个 32 位通用定时器
- 4 个 16 位 PWM 定时器
- 每个 PWM 定时器最多可以有 3 个输出
- 有 8 个输出模式可选择

### 20.3 功能框图

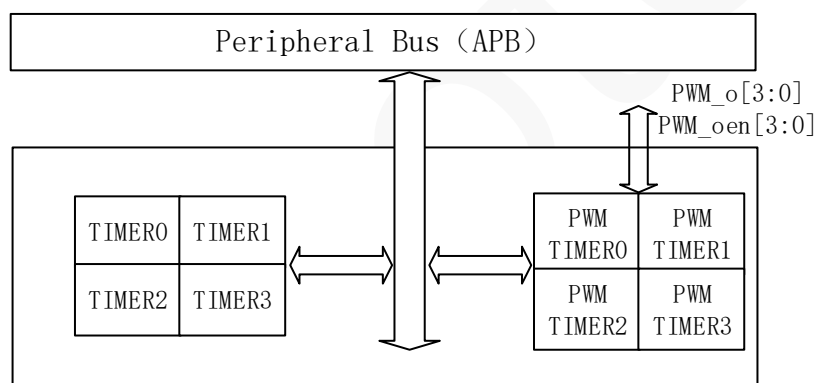


图 20-1 定时器功能框图



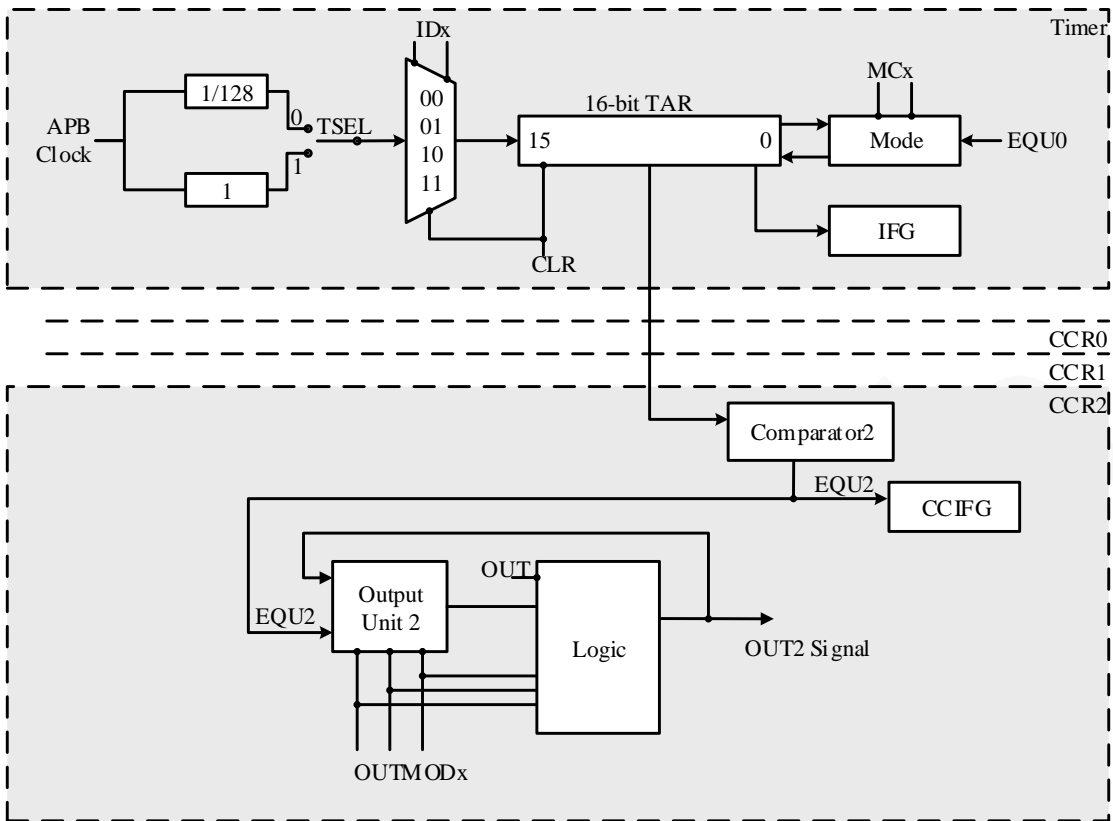


图 20-2 PWM 定时器功能框图

20.4 寄存器地址

表 20-1 32 位 TIMER 控制器(32 位 TIMER 基地址: 0x40012800)

名称	类型	地址	描述	默认值
TMR0_CTRL	R/W	0x0000	Timer0 控制寄存器	0x0
TMR0_VALUE	R/W	0x0004	Timer0 实时计数寄存器	0x00000000
TMR0_RELOAD	R/W	0x0008	Timer0 重载寄存器	0x00000000
TMR0_INT	R/C	0x000C	Timer0 中断状态寄存器	0x0
TMR1_CTRL	R/W	0x0020	Timer1 控制寄存器	0x0
TMR1_VALUE	R/W	0x0024	Timer1 实时计数寄存器	0x00000000
TMR1_RELOAD	R/W	0x0028	Timer1 重载寄存器	0x00000000
TMR1_INT	R/C	0x002C	Timer1 中断状态寄存器	0x0
TMR2_CTRL	R/W	0x0040	Timer2 控制寄存器	0x0
TMR2_VALUE	R/W	0x0044	Timer2 实时计数寄存器	0x00000000
TMR2_RELOAD	R/W	0x0048	Timer2 重载寄存器	0x00000000
TMR2_INT	R/C	0x004C	Timer2 中断状态寄存器	0x0
TMR3_CTRL	R/W	0x0060	Timer3 控制寄存器	0x0
TMR3_VALUE	R/W	0x0064	Timer3 实时计数寄存器	0x00000000

TMR3_RELOAD	R/W	0x0068	Timer3 重载寄存器	0x00000000
TMR3_INT	R/C	0x006C	Timer3 中断状态寄存器	0x0

表 20-2 16 位 PWM TIMER 控制器(16bPWM TIMER 基地址: 0x40012900)

名称	类型	地址	描述	默认值
PWM0_CTL	R/W	0x0000	PWM Timer0 控制寄存器	0x00
PWM0_TAR	R	0x0004	PWM Timer0 实时计数寄存器	0x0000
PWM0_CCTL0	R/W	0x0008	PWM Timer0 比较寄存器 0	0x000
PWM0_CCTL1	R/W	0x000C	PWM Timer0 比较寄存器 1	0x000
PWM0_CCTL2	R/W	0x0010	PWM Timer0 比较寄存器 2	0x000
PWM0_CCR0	R/W	0x0014	PWM Timer0 比较数据寄存器 0	0x0000
PWM0_CCR1	R/W	0x0018	PWM Timer0 比较数据寄存器 1	0x0000
PWM0_CCR2	R/W	0x001C	PWM Timer0 比较数据寄存器 2	0x0000
PWM1_CTL	R/W	0x0020	PWM Timer1 控制寄存器	0x00
PWM1_TAR	R	0x0024	PWM Timer1 实时计数寄存器	0x0000
PWM1_CCTL0	R/W	0x0028	PWM Timer1 比较数据寄存器 0	0x000
PWM1_CCTL1	R/W	0x002C	PWM Timer1 比较数据寄存器 1	0x000
PWM1_CCTL2	R/W	0x0030	PWM Timer1 比较数据寄存器 2	0x000
PWM1_CCR0	R/W	0x0034	PWM Timer1 比较数据寄存器 0	0x0000
PWM1_CCR1	R/W	0x0038	PWM Timer1 比较数据寄存器 1	0x0000
PWM1_CCR2	R/W	0x003C	PWM Timer1 比较数据寄存器 2	0x0000
PWM2_CTL	R/W	0x0040	PWM Timer2 控制寄存器	0x00
PWM2_TAR	R	0x0044	PWM Timer2 实时计数寄存器	0x0000
PWM2_CCTL0	R/W	0x0048	PWM Timer2 比较控制寄存器 0	0x000
PWM2_CCTL1	R/W	0x004C	PWM Timer2 比较控制寄存器 1	0x000
PWM2_CCTL2	R/W	0x0050	PWM Timer2 比较控制寄存器 2	0x000
PWM2_CCR0	R/W	0x0054	PWM Timer2 比较数据寄存器 0	0x0000
PWM2_CCR1	R/W	0x0058	PWM Timer2 比较数据寄存器 1	0x0000
PWM2_CCR2	R/W	0x005C	PWM Timer2 比较数据寄存器 2	0x0000
PWM3_CTL	R/W	0x0060	PWM Timer3 控制寄存器	0x00
PWM3_TAR	R	0x0064	PWM Timer3 实时计数寄存器	0x0000
PWM3_CCTL0	R/W	0x0068	PWM Timer3 比较控制寄存器 0	0x000
PWM3_CCTL1	R/W	0x006C	PWM Timer3 比较控制寄存器 1	0x000
PWM3_CCTL2	R/W	0x0070	PWM Timer3 比较控制寄存器 2	0x000
PWM3_CCR0	R/W	0x0074	PWM Timer3 比较数据寄存器 0	0x0000
PWM3_CCR1	R/W	0x0078	PWM Timer3 比较数据寄存器 1	0x0000
PWM3_CCR2	R/W	0x007C	PWM Timer3 比较数据寄存器 2	0x0000
PWM_O_SEL	R/W	0x00F0	PWM 输出选择寄存器	0xDB51

## 20.5 寄存器定义

### 20.5.1 TMRx\_CTRL 寄存器

表 20-3 TMRx\_CTRL 寄存器

Bit	名称	类型	描述	默认值
31:4	-	-	保留	0
3	INTEN	R/W	Timerx 中断使能寄存器。	0x0
2	EXTCLK	R/W	选择外部时钟信号 ext_clk (IOB15)*作为时钟源。Ext_clk 的上升沿将作为内部计数器向下计数的使能信号。每到来一个外部时钟 ext_clk 上升沿，内部计数器减 1。该模式下，外部时钟信号 ext_clk 的频率应低于 PCLK/6。	0x0
1	EXTEN	R/W	选择 ext_clk (IOB15)管腿作为时钟使能 IO 口。当 EXTEN 为 1，并且 ext_clk 为 1 时，内部计数器将以 PCLK 的时钟速率递减 1。当 EXTEN 为 1，并且 ext_clk 为 0，内部计数器不会变化。Ext_clk 会有两个周期的同步时钟，以避免 ext_clk 上的毛刺。当 EXTCLK 和 EXTEN 同时使能，EXTCLK 优先级更高。	0x0
0	EN	R/W	Timerx 使能控制寄存器。	0x0

注 (\*)：ext\_clk (IOB15)是名为 TIMEREXT\_CLK (IOB15)的引脚

### 20.5.2 TMRx\_VALUE 寄存器

表 20-4 TMRx\_VALUE 寄存器

Bit	名称	类型	描述	默认值
31:0	VALUE	R/W	Timerx 实时值寄存器。	0x00000000

### 20.5.3 TMRx\_RELOAD 寄存器

表 20-5 TMRx\_RELOAD 寄存器

Bit	名称	类型	描述	默认值
31:0	RELOAD	R/W	Timerx 重载值寄存器。对该寄存器写值即设置当前值。每当定时器向下计数到 0，RELOAD 的值将被重载到计数器 TMRx_VALUE 中，并重新开始向下计数。同时，INT 位将被置 1。  定时器 RELOAD 计算公式如下： $RELOAD = Period * APBCLK - 1。$	0x00000000

### 20.5.4 TMRx\_INT 寄存器

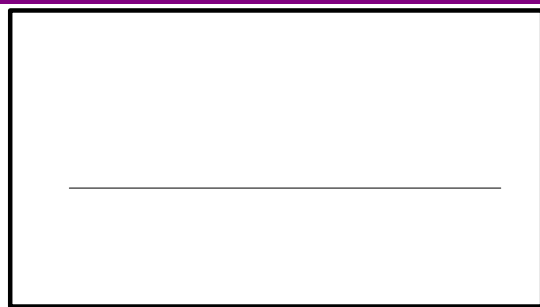
表 20-6 TMRx\_INT 寄存器

Bit	名称	类型	描述	默认值
31:1	-	-	保留	0
0	INT	R/C	Timerx 中断状态寄存器，写 1 清零。当内部向下计数器计数到 0 时，该位被置 1。	0x0

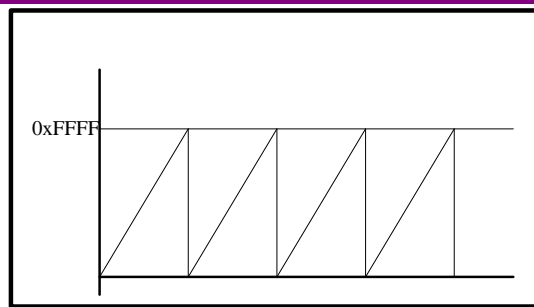
## 20.5.5 PWMx\_CTL 寄存器

表 20-7 PWMx\_CTL 寄存器

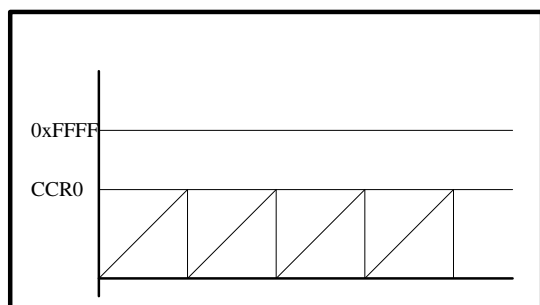
Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:6	ID	R/W	PWM timerx 输入时钟分频控制。 0: 输入时钟 1/2 分频。 1: 输入时钟 1/4 分频。 2: 输入时钟 1/8 分频。 3: 输入时钟 1/16 分频。	0x0
5:4	MC	R/W	PWM Timer 模式控制。 0: 停止计数。 1: 向上计数模式: 计数器向上计数到 CCR0, 并从 0 重新开始。 2: 连续计数模式: 计数器向上计数到 0xFFFF, 并从 0 重新开始。 3: 向上/向下计数模式: 计数器向上计数到 CCR0, 并从 CCR0 向下计数到 0。	0x0
3	TSEL	R/W	时钟源选择。 0: APB 时钟/128。 1: APB 时钟。	0x0
2	CLR	R/W	TAR 清零寄存器, 当该位设置为 1 时, TAR 将清零。在 TAR 清零后, 该位将自动清零。	0x0
1	IE	R/W	PWM Timerx 中断使能寄存器。	0x0
0	IFG	R/C	PWM Timerx 中断状态标志, 写 1 清零该标志。 向上计数模式: 当计数器从 CCR0 变为 0 时标志位置 1 连续计数模式: 当计数器从 0xFFFF 变为 0 时标志位置 1 向上/向下计数模式: 当计数器从 0x0001 到 0 时标志位置 1	0x0



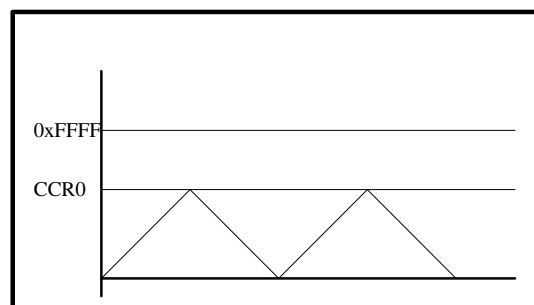
**Stop Mode**  
Timer stop.



**Continuous Mode**  
Timer will count to 0xFFFF, and restart from 0 and count to 0xFFFF.



**Up Mode**  
Timer will count to CCR0, and restart from 0 to CCR0.



**Up/Down Mode**  
Timer will count to CCR0, and then down count to 0x0, and then up\_count to CCR0 again.

图 20-3 PWM 定时器模式

MC 设置为非零值时, 定时器将开始工作。PWM Timer 工作模式为向上计数模式和向上/向下计数模式时, 若设置 PWMx\_CCR0 为 0, 定时器 x 将停止计数。如果重新向 PWMx\_CCR0 写入非零值, 定时器将重新开始计数。

向上计数模式下, CCR0 的值在计数期间被改变并且 CCR0 的新值大于当前计数值, 则定时器将计数到新的 CCR0, 然后返回到 0。如果 CCR0 的新值小于当前计数值, 定时器将计数一个节拍后重置计数器值为 0 并重新开始计数。

向上/向下计数模式下, 向上计数期间, 如果 CCR0 的值在计数期间被改变并且 CCR0 的新值大于当前计数值, 则定时器将计数到新的 CCR0, 然后递减计数到 0。如果新的 CCR0 小于当前计数值, 则定时器将计数一个节拍后开始向下计数到 0。向下计数期间, 如果 CCR0 的值在计数期间被改变, 定时器将连续完成向下计数过程。

连续模式下, 用户可以动态设置 CCR0~CCR2 的值控制输出频率。下图显示了此应用的示例。

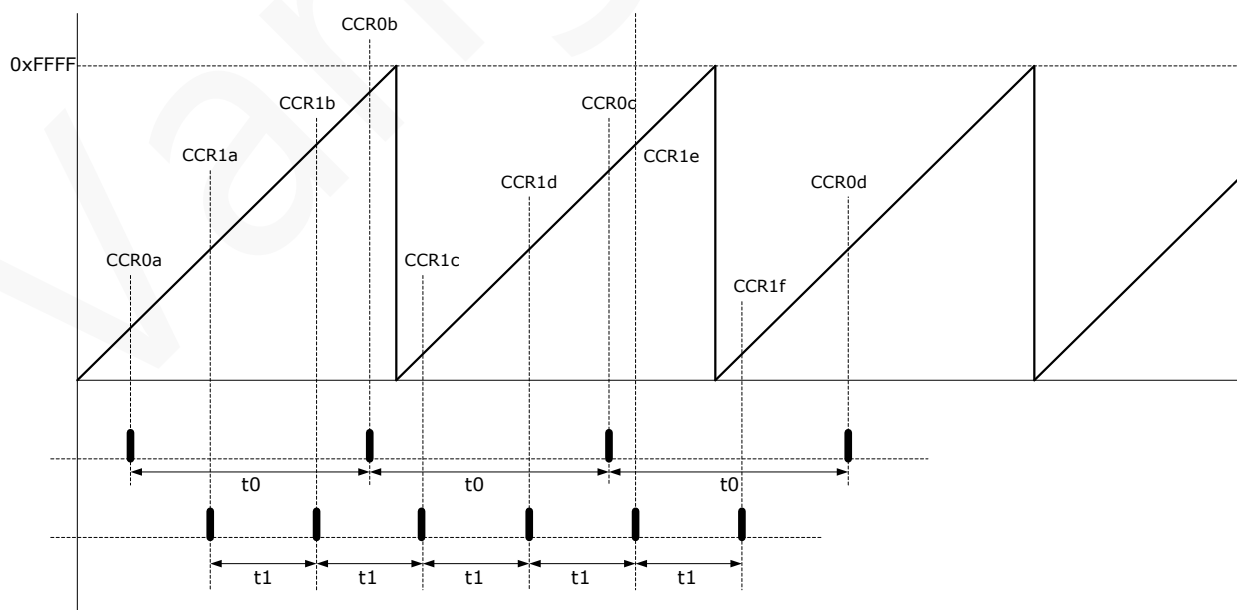


图 20-4 连续模式示例

上图中, CCR0a、CCR1a 分别是 CCR0、CCR1 在时刻 Ta0、Ta1 的值, CCR0b、CCR1b 分别是 CCR0、CCR1 在时刻 Tb0、Tb1 的值。(Tb0=Ta0+t0, Tb1=Ta1+t1)

上图展示在连续模式下，用户可以使用 CCIFG（CCTLx 的位 0）和 CCIE（CCTLx 的位 4），并设置 CCR0a 和 CCR1，则当 TAR=CCR0a 或 TAR=CCR1a 时可产生中断信号。当 TAR=CCR0b 或 TAR=CCR1b 时，可以产生另一个具有不同周期的中断。T0 和 t1 完全独立，所以一个 PWM 定时器最多可设置 3 个完全独立的中断。在此模式下，当定时器计数从 0xFFFF 到 0x0 时，IFG 标志将被置 1

## 20.5.6 PWMx\_TAR 寄存器

表 20-8 PWMx\_TAR 寄存器

位	名称	类型	描述	默认值
31: 0	TAR	R	PWM Timerx 实时计数寄存器	0x00000000

## 20.5.7 PWMx\_CCTLx 寄存器

表 20-9 PWMx\_CCTLx 寄存器

Bit	名称	类型	描述	默认值
31:10	-	-	保留	0
9	OUTEN	R/W	输出 OUTx 使能控制寄存器。 0: OUTx 输出禁止。 1: OUTx 输出使能。	0x0
8	-	-	保留	0
7:5	OUTMOD	R/W	输出模式选择。 0: 常数模式，OUTx 输出 OUT 位的值。 1: 置位模式，当 TAR=CCR <sub>x</sub> (x=0~2)时，OUTx 将置 1。 2: 翻转/复位模式，当 TAR=CCR <sub>x</sub> (x=1~2)时，OUTx 翻转，当 TAR=CCR0 时，OUTx 复位为 0。此模式不能用于 OUT0。 3: 置位/复位模式，当 TAR=CCR <sub>x</sub> (x=1~2)时，OUTx 置 1，当 TAR=CCR0 时，OUTx 复位为 0。此模式不能用于 OUT0。 4: 翻转模式，当 TAR=CCR <sub>x</sub> (x=1~2)时，OUTx 将会被翻转。 5: 复位模式，当 TAR=CCR <sub>x</sub> (x=1~2)时，OUTx 复位为 0。 6: 翻转/置位模式，当 TAR=CCR <sub>x</sub> (x=1~2)时，OUTx 被翻转，当 TAR=CCR0 时，OUTx 置为 1。此模式不能用于 OUT0。 7: 复位/置位模式，当 TAR=CCR <sub>x</sub> (x=1~2)时，OUTx 复位为 0，当 TAR=CCR0 时，OUTx 将置为 1。此模式不能用于 OUT0。	0x0
4	CCIE	R/W	比较中断使能寄存器。	0x0
3	-	-	保留	0
2	OUT	R/W	当 OUTMOD 设置为 0 时，该位用于控制 OUTx 的输出值。	0x0
1	-	-	保留	0
0	CCIFG	R/W	比较模式下，当 TAR=CCR <sub>x</sub> 时，该位将被置 1。 向该位写 1 可将标志位清零。	0

下图是在向上计数模式下的 PWM 输出：

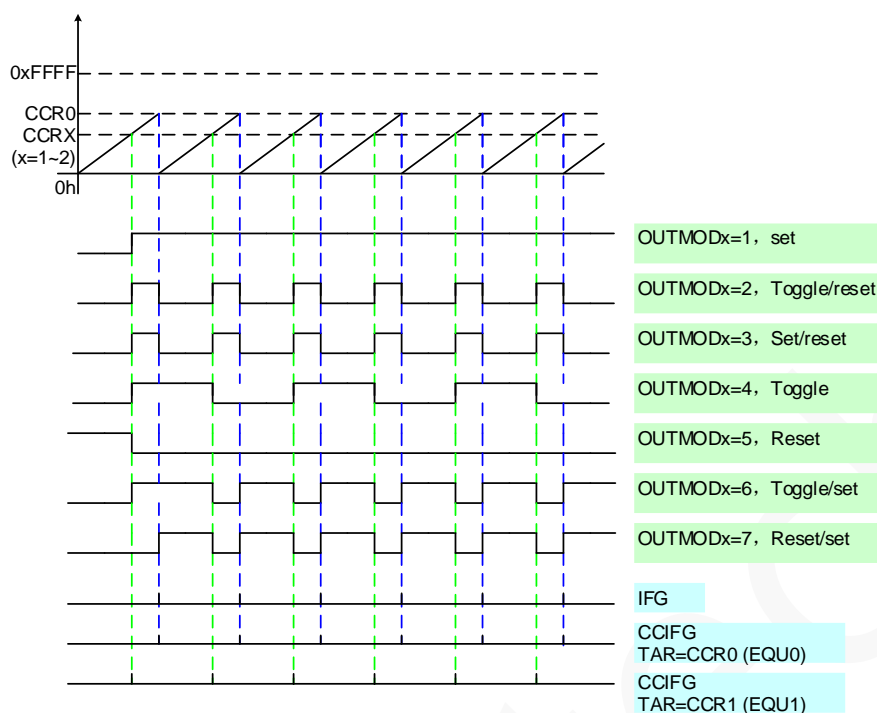


图 20-5 向上计数模式下的 PWM 输出

下图是在连续计数模式下的 PWM 输出：

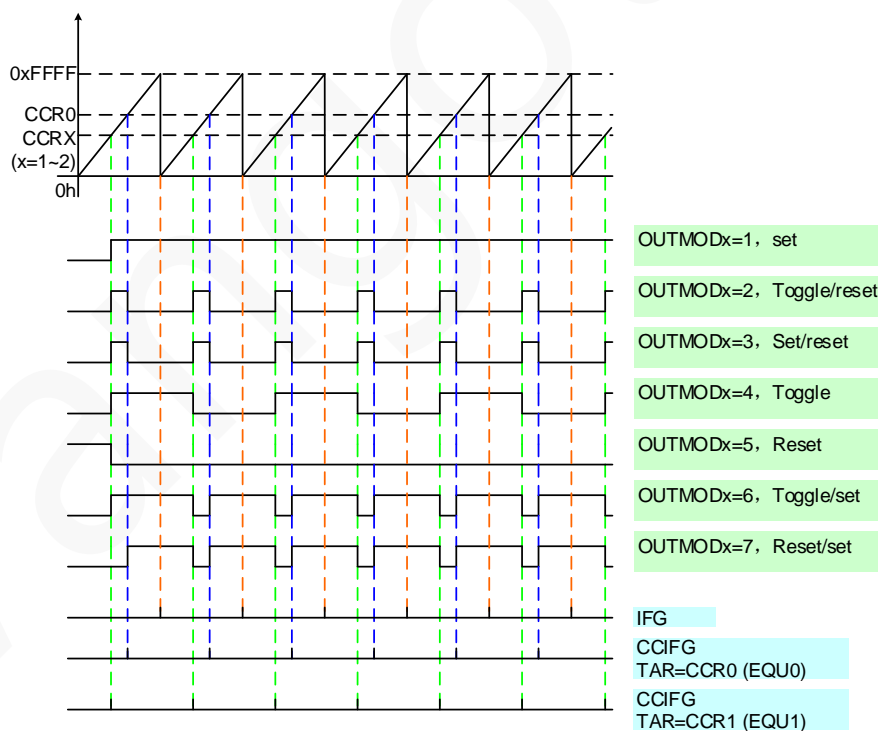


图 20-6 连续计数模式下的 PWM 输出

下图是在向上/向下计数模式下的 PWM 输出：

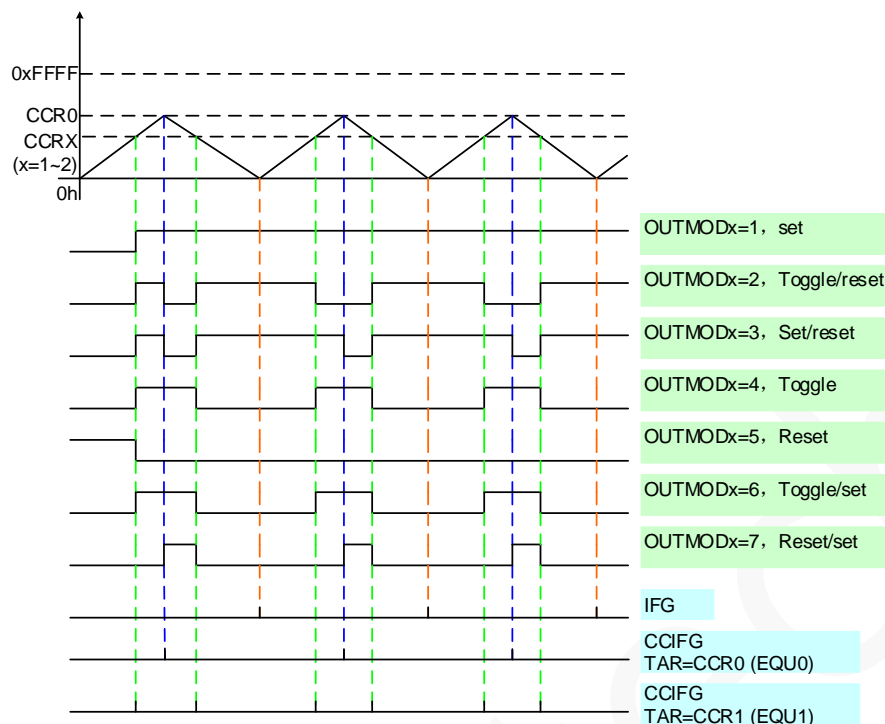


图 20-7 向上/向下计数模式下的 PWM 输出

## 20.5.8 PWMx\_CCRx 寄存器

表 20-10 PWMx\_CCRx 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	CCR <sub>x</sub>	R/W	比较数据寄存器。该寄存器用于与 TAR 比较以生成 PWM 输出。	0x0

## 20.5.9 PWM\_O\_SEL 寄存器

表 20-11 PWM\_O\_SEL 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:12	O_SEL3	R/W	外部引脚输出 PWM3 的输出选择寄存器。与 O_SEL0 的定义相同。	0xD
11:8	O_SEL2	R/W	外部引脚输出 PWM2 的输出选择寄存器。与 O_SEL0 的定义相同。	0xB
7:4	O_SEL1	R/W	外部引脚输出 PWM1 的输出选择寄存器。与 O_SEL0 的定义相同。	0x5
3:0	O_SEL0	R/W	外部输出引脚 PWM0 的输出选择寄存器。 0x0: 来自 PWM0 的 OUT0 0x1: 来自 PWM0 的 OUT1 0x2: 来自 PWM0 的 OUT2 0x4: 来自 PWM1 的 OUT0	0x1



			0x5: 来自 PWM1 的 OUT1 0x6: 来自 PWM1 的 OUT2 0x8: 来自 PWM2 的 OUT0 0x9: 来自 PWM2 的 OUT1 0xA: 来自 PWM2 的 OUT2 0xC: 来自 PWM3 的 OUT0 0xD: 来自 PWM3 的 OUT1 0xE: 来自 PWM3 的 OUT2 其他: 保留。	
--	--	--	---	--

## 第21章 LCD 控制器

### 21.1 简介

LCD控制器用于在LCD面板上显示内容。LCD控制器支持4/6/8 COM模式，最多65段。LCD控制器内有两个具有自动切换功能的帧缓存器。LCD控制器中的设置在从深睡眠模式唤醒后会被复位，用户应在深睡眠模式唤醒后手动恢复设置。

### 21.2 特点

- 4/6/8 COM 模式。
- 支持最多 4COM×65SEG，6COM×63SEG 或 8COM×61SEG
- 支持 1/3 Bias 或 1/4 Bias 模式
- LCD 的偏置电压由一个独立的 3.3 VLDO 产生，在 3.6V-2.7V 间以 60mV/LSB 的步长可调
- 内置电阻分压电路产生 LCD 波形电压
- LCD 扫描频率由 RTCCLK 时钟产生
- 两个帧缓存器，支持自动切换功能。
- 可调帧速率。

### 21.3 功能框图

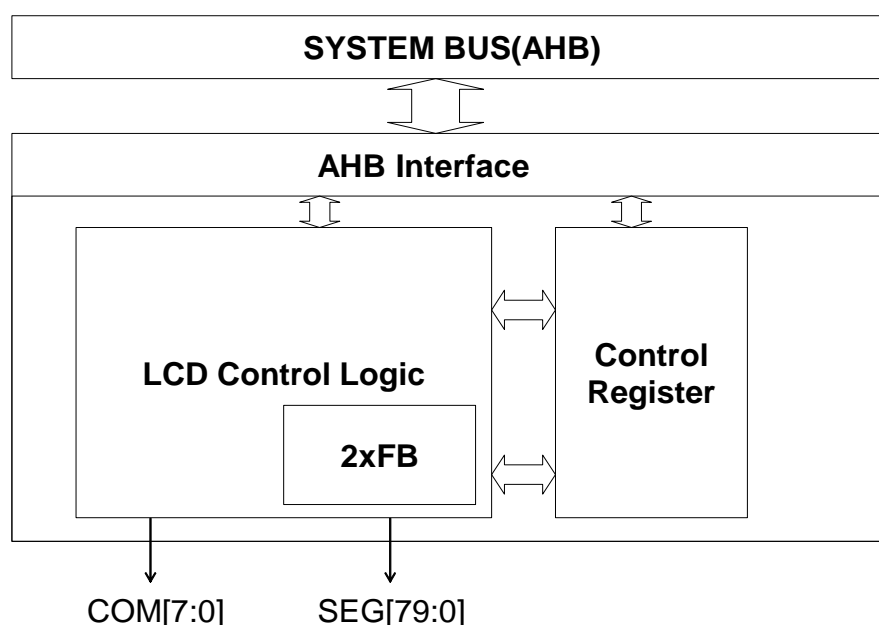


图 21-1 LCD 控制器的功能框图

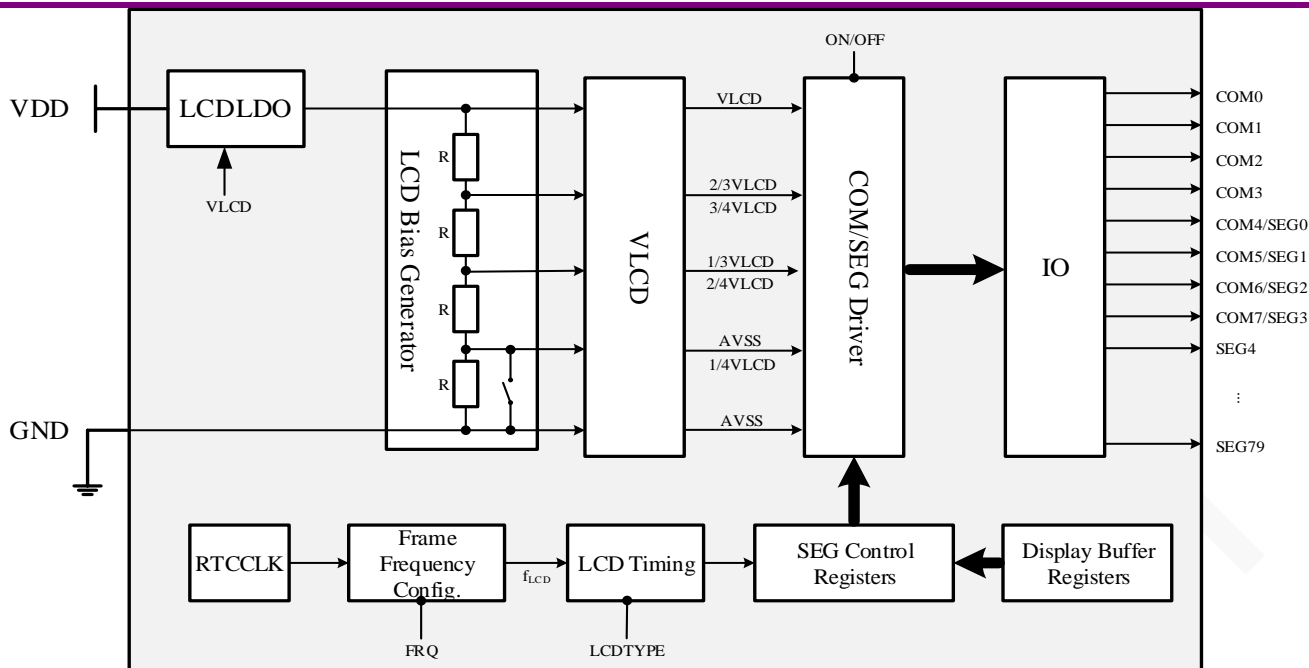


图 21-2 LCD 控制器结构图

## 21.4 LCD 时序

LCD 控制器用于在 LCD 面板上显示内容。LCD 控制器支持 4/6/8COM 模式，最多 65 段。LCD 控制器内有两个具有自动切换功能的帧缓存器。LCD 控制器中的设置在从深睡眠模式唤醒后会被复位，用户应在深睡眠模式唤醒后手动恢复设置。

## 21.5 LCD 波形电压

V94XX(A)中，LCD 驱动电路的偏置电压由 VDD 通过内部分压电阻电路产生 LCD 偏置电压 (VLCD)，用于产生 LCD 驱动波形。用户可通过 ANA\_REG6 寄存器中的控制位 VLCD[4: 1]对波形电压进行调整。

## 21.6 LCD 工作电流

用户可以改变寄存器 LCD\_CTRL 中的控制位 DRV，配置偏置电压产生电路中的各分压电阻的阻值，改变电路电流大小，从而调整液晶屏显示亮度。默认情况下各分压电阻的阻值为 300kΩ。

## 21.7 LCD 驱动波形

LCD 偏置电压产生电路中有 4 个串联电阻，可以配置为 1/3 Bias 模式或 1/4 Bias 模式。用户可通过配置 LCD\_BMOD 位 (ANA\_REG6 的 bit0) 开启或关闭一个电阻，从而配置 LCD 为 1/3 偏置或 1/4 偏置模式。

下图为[1/4Duty, 1/3Bias]的 LCD 驱动波形（示例）。

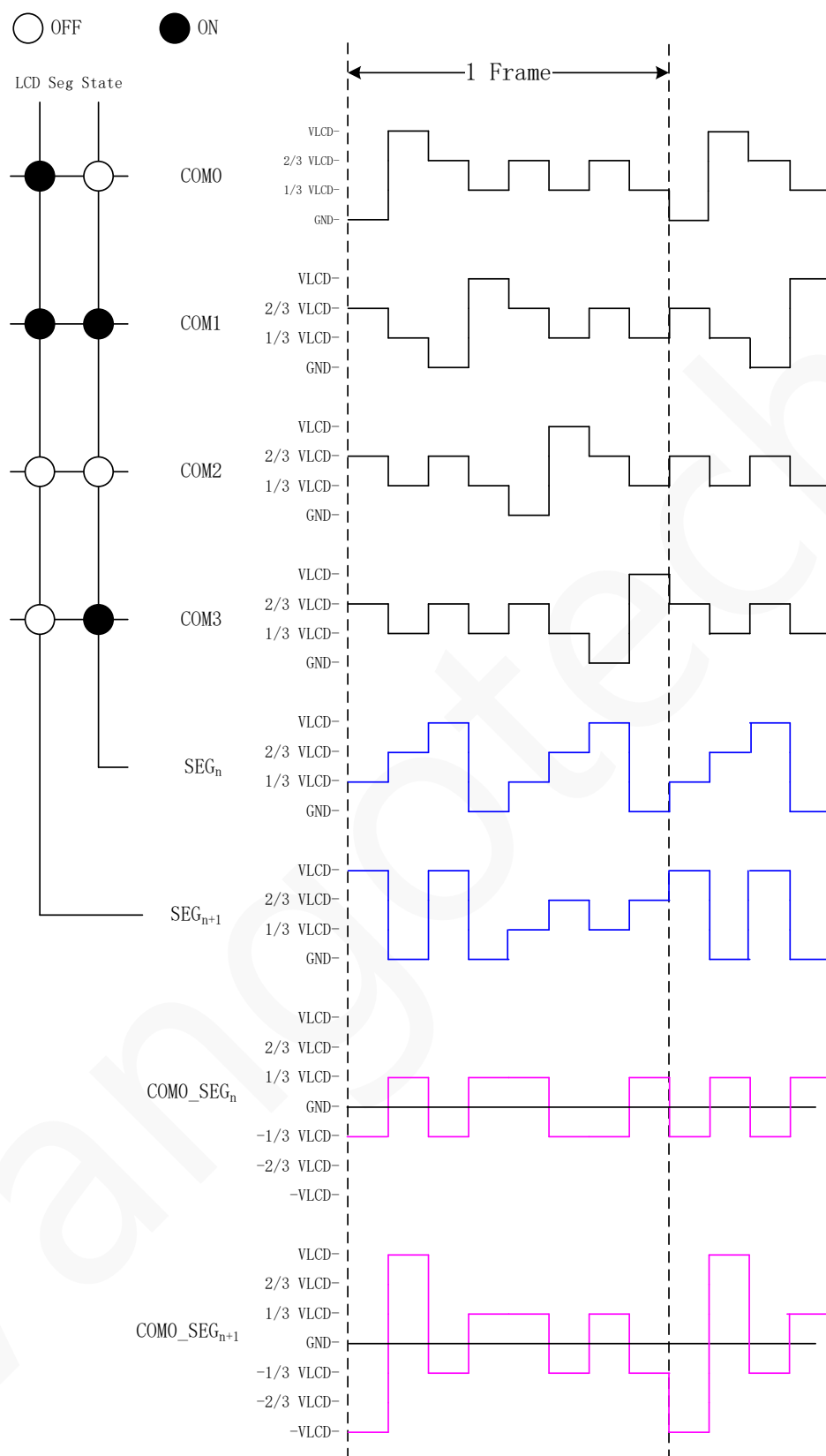


图 21-3 LCD 波形[1/4 Duty, 1/3 Bias]

下图为[1/6Duty, 1/3Bias]的 LCD 驱动波形（示例）。

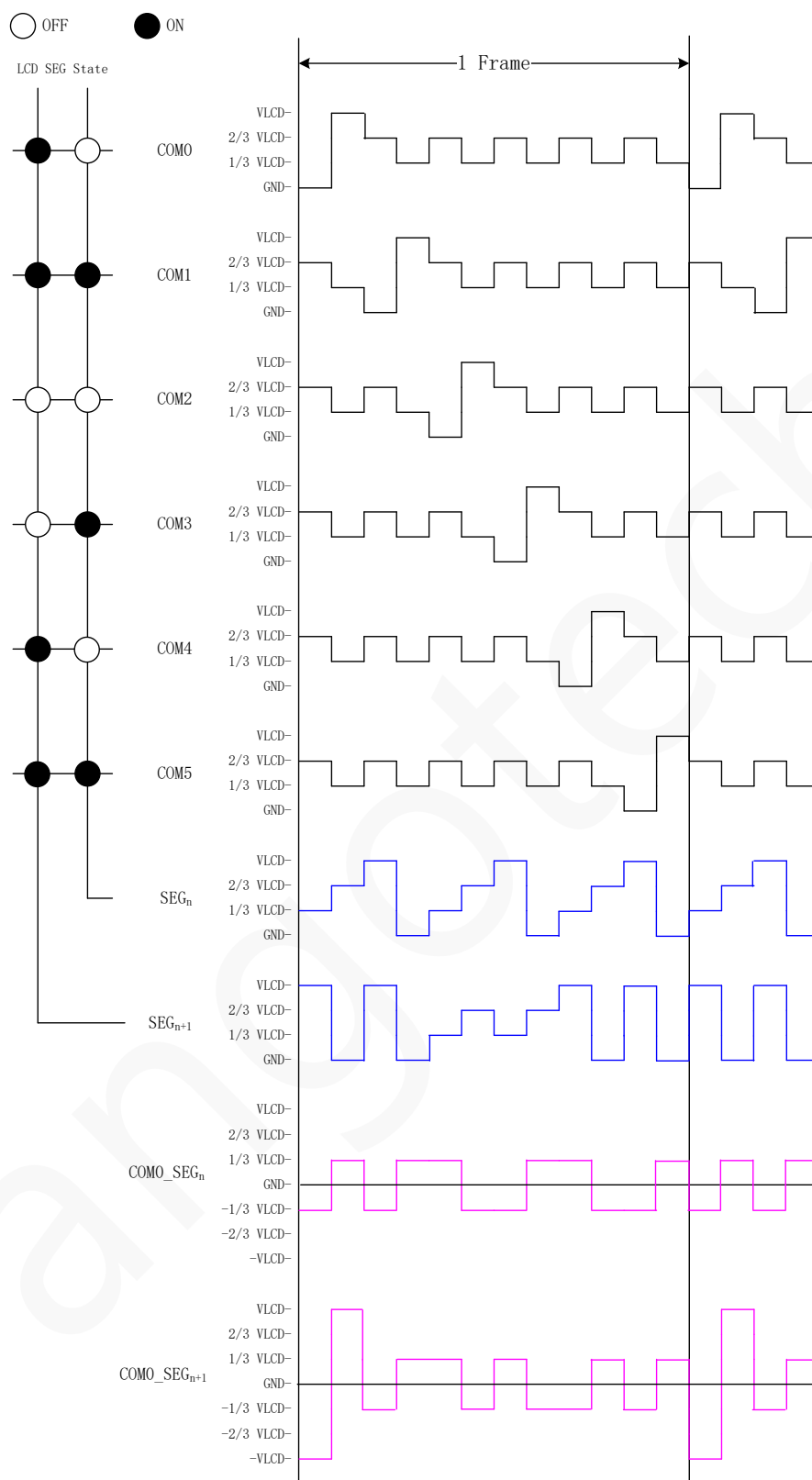


图 21-4 LCD 波形[1/6 Duty, 1/3 Bias]

下图为[1/8Duty, 1/3Bias]的 LCD 驱动波形（示例）

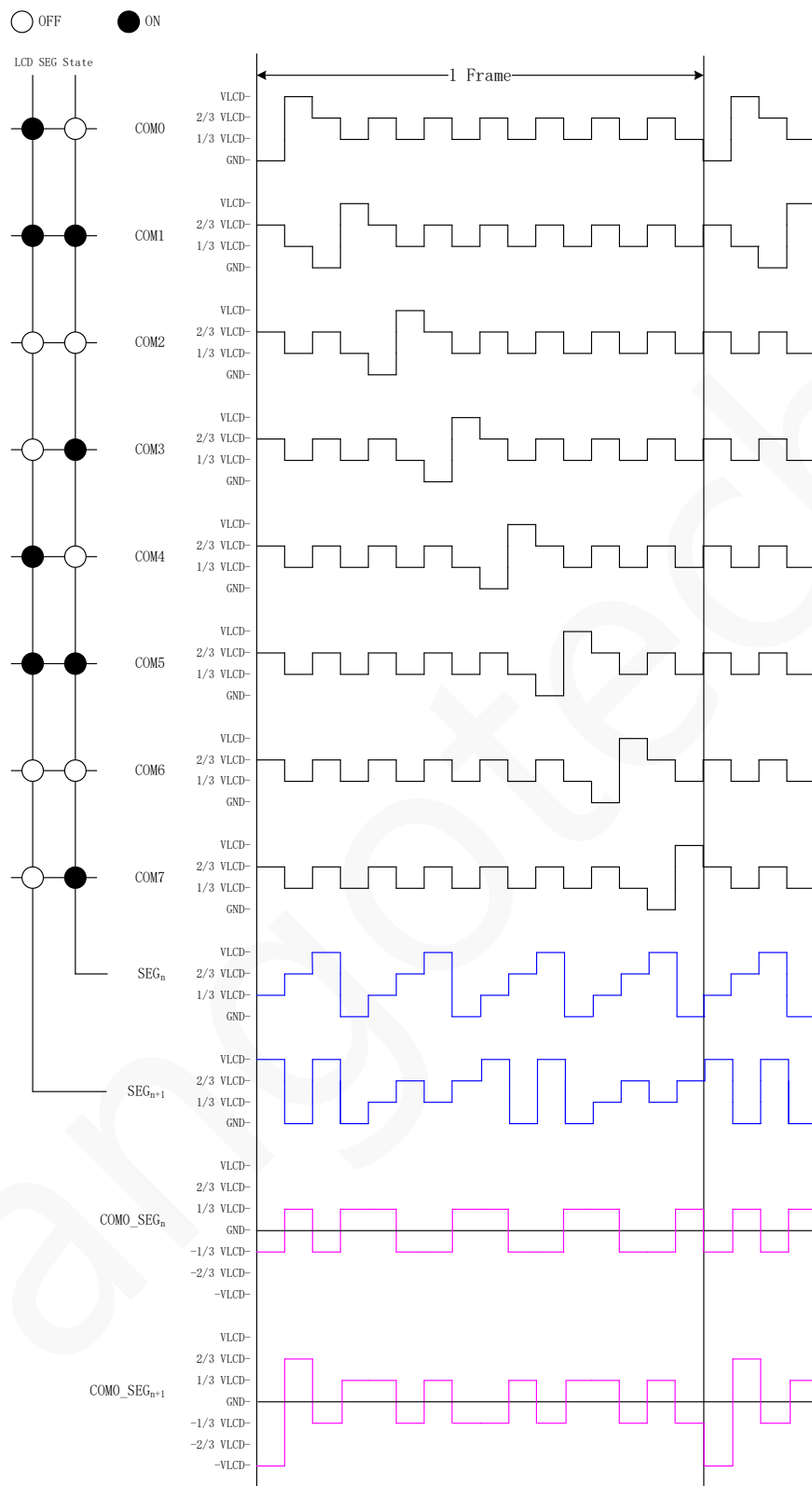


图 21-5 LCD 波形[1/8 Duty, 1/3 Bias]

下图为[1/8Duty, 1/4Bias]的 LCD 驱动波形（示例）

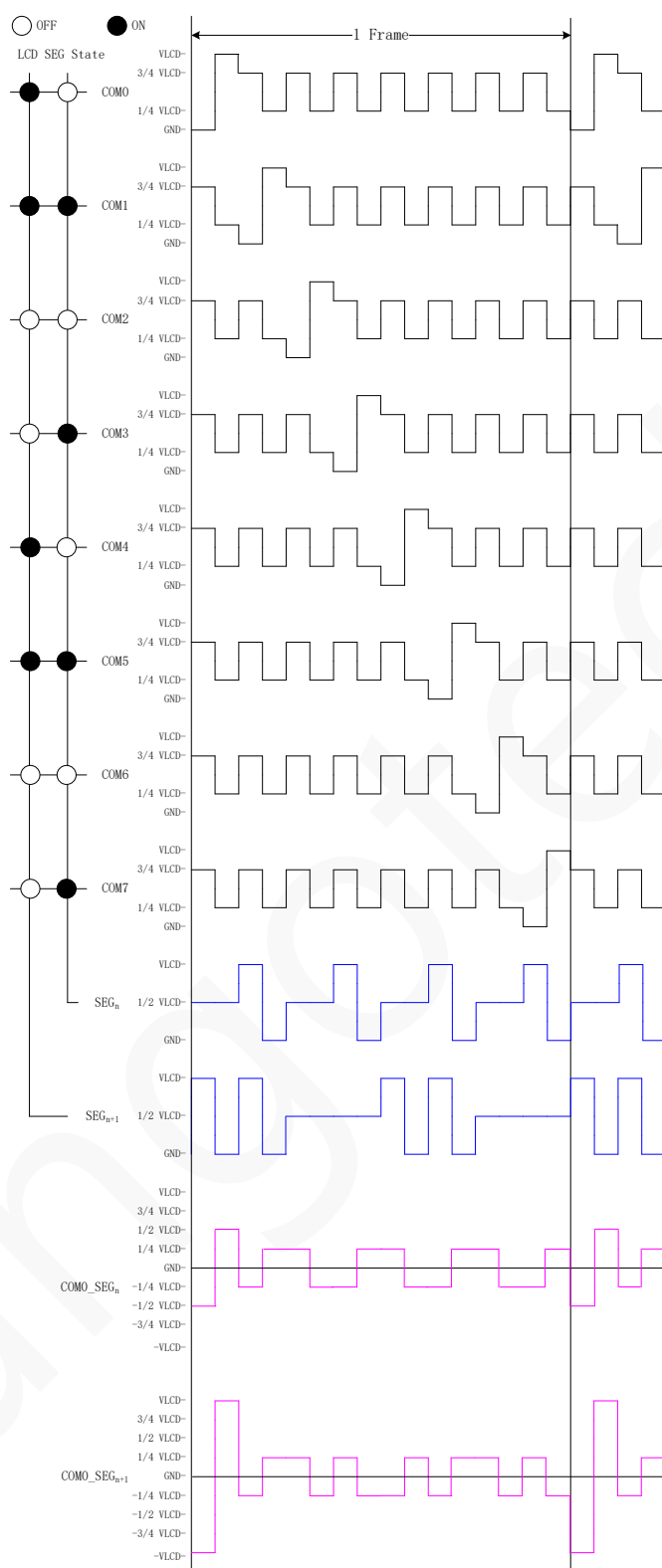


图 21-6 LCD 波形[1/8 Duty , 1/4 Bias]

## 21.8 寄存器地址

表 21-1 与 LCD 相关的 ANA 控制器(ANA 基地址: 0x40014200)

名称	类型	地址	描述	默认值
ANA_REG6	R/W	0x0018	模拟寄存器 6	0x00

表 21-2 LCD 控制器(LCD 基地址: 0x40002000)

名称	类型	地址	描述	默认值
LCD_FB00	R/W	0x0000	LCD 帧缓存 0 寄存器	--
LCD_FB01	R/W	0x0004	LCD 帧缓存 1 寄存器	--
LCD_FB02	R/W	0x0008	LCD 帧缓存 2 寄存器	--
LCD_FB03	R/W	0x000C	LCD 帧缓存 3 寄存器	--
LCD_FB04	R/W	0x0010	LCD 帧缓存 4 寄存器	--
LCD_FB05	R/W	0x0014	LCD 帧缓存 5 寄存器	--
LCD_FB06	R/W	0x0018	LCD 帧缓存 6 寄存器	--
LCD_FB07	R/W	0x001C	LCD 帧缓存 7 寄存器	--
LCD_FB08	R/W	0x0020	LCD 帧缓存 8 寄存器	--
LCD_FB09	R/W	0x0024	LCD 帧缓存 9 寄存器	--
LCD_FB0A	R/W	0x0028	LCD 帧缓存 10 寄存器	--
LCD_FB0B	R/W	0x002C	LCD 帧缓存 11 寄存器	--
LCD_FB0C	R/W	0x0030	LCD 帧缓存 12 寄存器	--
LCD_FB0D	R/W	0x0034	LCD 帧缓存 13 寄存器	--
LCD_FB0E	R/W	0x0038	LCD 帧缓存 14 寄存器	--
LCD_FB0F	R/W	0x003C	LCD 帧缓存 15 寄存器	--
LCD_FB10	R/W	0x0040	LCD 帧缓存 16 寄存器	--
LCD_FB11	R/W	0x0044	LCD 帧缓存 17 寄存器	--
LCD_FB12	R/W	0x0048	LCD 帧缓存 18 寄存器	--
LCD_FB13	R/W	0x004C	LCD 帧缓存 19 寄存器	--
LCD_FB14	R/W	0x0050	LCD 帧缓存 20 寄存器	--
LCD_FB15	R/W	0x0054	LCD 帧缓存 21 寄存器	--
LCD_FB16	R/W	0x0058	LCD 帧缓存 22 寄存器	--
LCD_FB17	R/W	0x005C	LCD 帧缓存 23 寄存器	--
LCD_FB18	R/W	0x0060	LCD 帧缓存 24 寄存器	--
LCD_FB19	R/W	0x0064	LCD 帧缓存 25 寄存器	--
LCD_FB1A	R/W	0x0068	LCD 帧缓存 26 寄存器	--
LCD_FB1B	R/W	0x006C	LCD 帧缓存 27 寄存器	--
LCD_FB1C	R/W	0x0070	LCD 帧缓存 28 寄存器	--
LCD_FB1D	R/W	0x0074	LCD 帧缓存 29 寄存器	--
LCD_FB1E	R/W	0x0078	LCD 帧缓存 30 寄存器	--
LCD_FB1F	R/W	0x007C	LCD 帧缓存 31 寄存器	--



LCD_FB20	R/W	0x0080	LCD 帧缓存 32 寄存器	--
LCD_FB21	R/W	0x0084	LCD 帧缓存 33 寄存器	--
LCD_FB22	R/W	0x0088	LCD 帧缓存 34 寄存器	--
LCD_FB23	R/W	0x008C	LCD 帧缓存 35 寄存器	--
LCD_FB24	R/W	0x0090	LCD 帧缓存 36 寄存器	--
LCD_FB25	R/W	0x0094	LCD 帧缓存 37 寄存器	--
LCD_FB26	R/W	0x0098	LCD 帧缓存 38 寄存器	--
LCD_FB27	R/W	0x009C	LCD 帧缓存 39 寄存器	--
LCD_CTRL	R/W	0x0100	LCD 控制寄存器	0x00
LCD_CTRL2	R/W	0x0104	LCD 控制寄存器 2	0x0000
LCD_SEGCTRL0	R/W	0x0108	LCDSEG 使能控制寄存器 0	0x00000000
LCD_SEGCTRL1	R/W	0x010C	LCDSEG 使能控制寄存器 1	0x00000000
LCD_SEGCTRL2	R/W	0x0110	LCDSEG 使能控制寄存器 2	0x00000000

## 21.9 寄存器定义

### 21.9.1 LCD\_FBx 寄存器

表 21-3 LCD\_FBx 寄存器

Bit	名称	类型	描述	默认值
31:0	DATAx	R/W	每个位表示显示阵列中的数据，不同模式下数据排列的详细信息，请参见表 3~6。这些寄存器可以进行字节读或写，因此用户可以使用 CPU 或 DMA 来读写这些寄存器。	--

LCD\_FB00~LCD\_FB13 是帧缓存器 A。

LCD\_FB14~LCD\_FB27 是帧缓存器 B。

下表仅显示帧缓存器 A，如果 FBMODE 被选择为 1，则帧缓存器 B 呈现与帧缓存器 A 相同的数据。

表 21-4 LCD\_FBx 寄存器的数据排布

寄存器	数据 bit			
	31:24	23:16	15:8	7:0
LCD_FB00	SEG3 对应的 COM[7:0]	SEG2 对应的 COM[7:0]	SEG1 对应的 COM[7:0]	SEG0 对应的 COM[7:0]
LCD_FB01	SEG7 对应的 COM[7:0]	SEG6 对应的 COM[7:0]	SEG5 对应的 COM[7:0]	SEG4 对应的 COM[7:0]
....				
LCD_FB13	SEG79 对应的 COM[7:0]	SEG78 对应的 COM[7:0]	SEG77 对应的 COM[7:0]	SEG76 对应的 COM[7:0]

当选择 4/6 COM 模式时，COM 的高位将被丢弃，因此数据排布格式与 8COM 模式相同。

### 21.9.2 LCD\_CTRL 寄存器

表 21-5 LCD\_CTRL 寄存器



Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	EN	R/W	LCD 使能寄存器。 0: 禁用 1: 启用	0x0
6	-	-	保留	0
5:4	TYPE	R/W	LCD 模式控制寄存器。 0: 4 COM 模式 1: 6 COM 模式 2: 8 COM 模式 3: 保留。	0x0
3:2	DRV	R/W	LCD 驱动电阻控制寄存器。 0: 300k ohm 1: 600k ohm 2: 150k ohm 3: 200 kohm	0x0
1:0	FRQ	R/W	LCD 扫描频率。 0: 64Hz 1: 128Hz 2: 256Hz 3: 512Hz 这里的扫描频率是指每个 COM 的扫描速度，因此如果 COM 总数为 4，则整个帧速率等于该值除以 4。	0x0

**注意：**切换 COM 类型时，首先关闭 LCD 控制器（LCD\_CTRL 寄存器的第 7 位），然后等待一个延时，以保证当前帧传输完成。延时与当前帧速率相关（LCD\_CTRL 寄存器的第 1: 0 位、第 7 位）。

例如：LCD 的当前扫描速率是 64Hz，将 COM 类型从 4COM 切换到 6COM，延时时间计算如下：

$$T_{\text{Delay}} = 1/(64/4) = 62.5\text{ms}$$

## 21.9.3 LCD\_CTRL2 寄存器

表 21-6 LCD\_CTRL2 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:8	SWPR	R/W	帧缓存器转换周期。转换周期计算公式如下： $0.5 \text{ 秒} * (\text{SWPR} + 1)$	0x0
7:6	FBMODE	R/W	LCD 帧缓存转换模式控制寄存器。 0: 始终显示帧缓存器 A 1: 在帧缓存器 A 和帧缓存器 B 之间切换。 2: 在帧缓存器 A 和空白之间切换。	0x0

			3: 保留。	
5	-	-	保留	0
4	BKFILL	R/W	空白周期填充值。该寄存器控制空白周期内的填充值。	0x0
3:0	-	-	保留	0

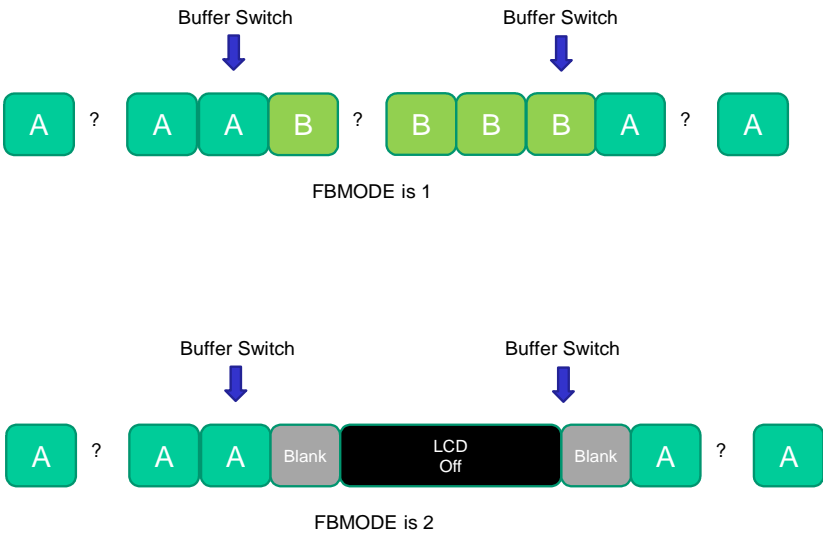


图 21-7 不同 FBMODE 下的帧缓存器操作模式

21.9.4 LCD\_SEGCTRL0 寄存器

表 21-7 LCD\_SEGCTRL0 寄存器

Bit	名称	类型	描述	默认值
31:0	SEGCTRL	R/W	每个位控制 SEG0~SEG31 的 LCD 信号使能。 位 0: SEG0 的使能控制。 位 1: SEG1 的使能控制。 .... 位 31: SEG31 的使能控制。 0: 禁用 SEG 输出。 1: 使能 SEG 输出。	0x00000000

21.9.5 LCD\_SEGCTRL1 寄存器

表 21-8 LCD\_SEGCTRL1 寄存器

Bit	名称	类型	描述	默认值
31:0	SEGCTRL	R/W	每个位控制 SEG32~SEG63 的 LCD 信号使能。 位 0: SEG32 的使能控制。 位 1: SEG33 的使能控制。 ....	0x00000000

			位 31: SEG63 的使能控制。 0: 禁用 SEG 输出。 1: 使能 SEG 输出。	
--	--	--	--	--

## 21.9.6 LCD\_SEGCTRL2 寄存器

表 21-9 LCD\_SEGCTRL2 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	SEGCTRL	R/W	每个位控制 SEG64~SEG79 的 LCD 信号使能。 位 0: SEG64 的使能控制。 位 1: SEG65 的使能控制。 ... 位 15: SEG79 的使能控制。 0: 禁用 SEG 的输出。 1: 使能 SEG 输出	0x0000

## 21.9.7 ANA\_REG6 寄存器

表 21-10 ANA\_REG6 寄存器

Bit	名称	Function	Notes
0	LCD_BMODE	LCD 偏置模式选择。	0: 1/3 Bias; 1: 1/4 Bias。
4:1	VLCD[3: 0]	LCD 驱动电压。	当 VLCD=0: Default 当 VLCD=0~5: 调整幅度=+60mV*VLCD 当 VLCD=6~15: 调整幅度=-60mV*(VLCD-5)

对于每个 COM 和 SEG，为了确保 LCD 正常工作，在使能相应引脚的 LCD 功能之前，用户应设置相应的 IO 禁止输入禁止输出。

## 第22章 SPI 控制器

### 22.1 简介

V94XX(A)中集成了 1 个串行外设接口 (SPI) 控制器, 以便与其他设备或组件通信。SPI 控制器中的设置在从浅睡眠或深睡眠模式唤醒后会被复位, 用户应在浅睡眠/深睡眠模式唤醒后手动恢复设置。SPI 主机模式支持的 SPICLK 时钟: APBCLK/2/4/8/16/32/64/128。SPI 从机模式支持的最大 SPICLK 时钟为 APBCLK/2。

### 22.2 特点

- 支持主机模式和从机模式
- 支持单字节和连续字节传输
- 支持接收溢出错误指示
- 支持发送/接收中断请求
- 主机模式下相位和极性可编程
- 可选择的数据采样时间 (时钟周期的结束或中间)
- 主机模式下 SCK 时钟频率可编程: APBClock/2/4/8/16/32/64/128
- 发送和接收方向上内置 8 个 8 位的 FIFO, 两种 FIFO 的中断触发阈值都可编程。

### 22.3 功能框图

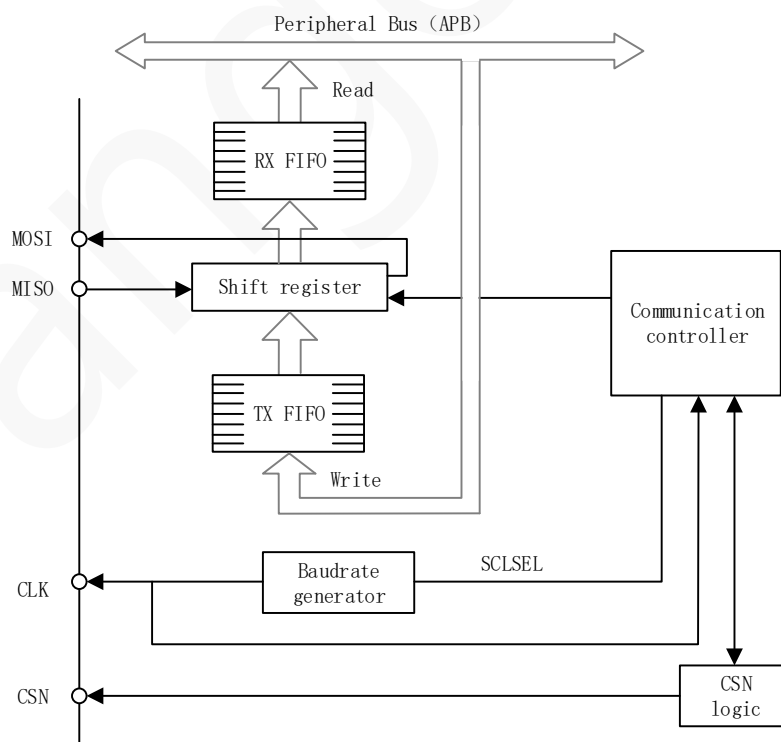


图 22-1 SPI 控制器的功能框图

通常 SPI 通过 4 个引脚与外部设备相连。

**MOSI:** 主设备输出/从设备输入。一般情况下, 该引脚用于主机模式下的数据发送和从机模式下的数据接收。

**MISO:** 主设备输入/从设备输出。一般情况下, 该引脚用于主机模式下的数据接收和从机模式下的数据发送。

CLK: SPI 串行时钟输出引脚, 时钟主机控制。

CSN: 从机片选引脚。根据 SPI 和 CSN 的设置, 该引脚可用于: 选择一个从机通信、同步数据帧。

注意: SPI 从机模式支持的最大 SPICLK 时钟为 1/2 APBCLK。

## 22.4 寄存器地址

表 22-1 SPI 控制器(SPI 基地址: 0x40015800)

名称	类型	地址	描述	默认值
SPI_CTRL	R/W	0x0000	SPI 控制寄存器	0x0000
SPI_TXSTS	R/W	0x0004	SPI 发送状态寄存器	0x8200
SPI_TXDAT	R/W	0x0008	SPI 发送 FIFO 寄存器	--
SPI_RXSTS	R/W	0x000C	SPI 接收状态寄存器	0x0000
SPI_RXDAT	R/W	0x0010	SPI 接收 FIFO 寄存器	--
SPI_MISC	R/W	0x0014	SPIMisc 控制寄存器	0x0003

## 22.5 寄存器定义

### 22.5.1 SPI\_CTRL 寄存器

SPI\_CTRL 寄存器用来控制 SPI 控制器的行为。

表 22-2 SPI\_CTRL 寄存器

Bit	名称	类型	描述	默认值
15	SPIEN	R/W	若该位置 1, SPICLK、SPIMISO、SPIMOSI 这些引脚将不能被用作普通 GPIO, 因此, 任何对 GPIO 的输入输出设置都会无效。 0: 禁用 SPI 功能 1: 启用 SPI 功能	0x0
14:12	-	-	保留	0
11	SPIRST	W	SPI 软件复位。如果该位写 1, SPI 控制器和 FIFO 指针的状态会恢复到初始状态。 0: 无效 1: 复位 SPI 控制器	0x0
10	CSGPIO	R/W	SPICS 管脚由 GPIO 或 H/W 控制。 主控模式: 0: SPICS 管脚由硬件 H/W 控制 1: SPICS 管脚由软件 GPIO 控制 从机模式: 0: SPICS 管脚由外部主机控制 1: SPICS 管脚为逻辑 0	0x0
9	SWAP	R/W	SPI MISO/MOSI 引脚互换控制寄存器。 0: 不交换 MISO/MOSI 1: 交换 MISO/MOSI	0x0

8	MOD	R/W	SPI 模式选择寄存器。 0: 主机模式 1: 从机模式	0x0
7:6	-	-	保留	0
5	SCKPHA	R/W	SPI 时钟相位, 请参考时序原理图。	0x0
4	SCKPOL	R/W	SPI 时钟极性, 请参考时序原理图。	0x0
3	-	-	保留	0
2:0	SCKSEL	R/W	主机模式时钟选择。 000: APBCLK/2 001: APBCLK/4 010: APBCLK/8 011: APBCLK/16 100: APBCLK/32 101: APBCLK/64 110: APBCLK/128 111: 保留。	0x0

## 22.5.2 SPI\_TXSTS 寄存器

SPI\_TXSTS 寄存器控制 SPI 发送 FIFO 和相关中断。

表 22-3 SPI\_TXSTS 寄存器

Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15	SPITXIF	R/C	SPI 发送中断标志。 当发送 FIFO 中数据数量低于用户设置的值时, 该位由硬件置“1”。如果 SPI_MISC 寄存器中的 SMART 值 1, 只要发送 FIFO 中数据数量高于用户设置的值, 该位就会被清零, 否则应该对该位写“1”以清除该标志。 读 0: 未发生中断 读 1: 发生中断 写 0: 无效 写 1: 清除标志	0x1
14	SPITXIEN	R/W	SPI 发送中断使能。 如果该位置为“1”, 且 SPI 中断发生时, 硬件将向 CPU 发出一个中断。如果该位清零, 该中断将被屏蔽。 0: 禁止发送中断 1: 使能发送中断	0x0
13:10	-	-	保留	0
9	TXEMPTY	R	发送 FIFO 空标志。 当发送 FIFO 为空时, 该位由硬件置 1。当发送 FIFO 不为空时, 该位由硬件清除。	0x1

			0: 发送 FIFO 不为空 1: 发送 FIFO 为空	
8	TXFUR	R	运行中从机发送 FIFO 空标志位。 当发送 FIFO 为空且外部 SPI 主机请求更多数据时，该位将被置 1，该位只在 SPI 从机模式下会置 1。	0x0
7	-	-	保留	0
6:4	TXFLEV	R/W	发送 FIFO 的中断触发阈值。 该寄存器用于指示在发生中断时，发送 FIFO 中存储的字节数。设置值越小，中断触发越少，因此可以在一个中断中写入更多数据。 000: FIFO 中的数据<1，允许写入 8 个数据。 001: FIFO 中的数据<2，允许写入 7 个数据。 010: FIFO 中的数据<3，允许写入 6 个数据。 011: FIFO 中的数据<4，允许写入 5 个数据。 100: FIFO 中的数据<5，允许写入 4 个数据。 101: FIFO 中的数据<6，允许写入 3 个数据。 110: FIFO 中的数据<7，允许写入 2 个数据。 111: FIFO 中的数据<8，允许写入 1 个数据。	0x0
3	-	-	保留	0
2:0	TXFFLAG	R	发送 FIFO 中数据个数。 该寄存器用于指示 FIFO 中的数据数量。 0000: FIFO 中没有数据或 FIFO 中有 8 个字节。 0001: FIFO 中有 1 个字节。 0010: FIFO 中有 2 个字节。 0011: FIFO 中有 3 个字节。 0100: FIFO 中有 4 个字节。 0101: FIFO 中有 5 个字节。 0110: FIFO 中有 6 个字节。 0111: FIFO 中有 7 个字节。	0x0

22.5.3 SPI\_TXDAT 寄存器

SPI\_TXDAT 寄存器用于 SPI 发送机将数据写入到 SPI 的发送 FIFO，并移出到外部主机或从机。

表 22-4 SPI\_TXDAT 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	SPITXD	W	向 SPI 发送 FIFO 写数据。	--

22.5.4 SPI\_RXSTS 寄存器

SPI\_RXSTS 寄存器用于控制 SPI 接收 FIFO 及相关中断。表 22-6 表示 SPI\_RXSTS 寄存器的 bit 定义。

表 22-5 SPI\_RXSTS 寄存器





Bit	名称	类型	描述	默认值
31:16	-	-	保留	0
15	SPIRXIF	R/C	<p>SPI 接收中断标志。</p> <p>当接收 FIFO 中数据个数高于用户设定的值时，该位由硬件置“1”。</p> <p>如果 SPI_MISC 寄存器中的 SMART 值设置为 1，只要接收 FIFO 中数据个数低于中断触发阈值，该位就会被清零；如果 SMART 值设置为 0，应该对该位写“1”以清除该标志。</p> <p>读 0：未发生中断 读 1：发生中断 写 0：无效 写 1：清除标志</p>	0x0
14	SPIRXIEN	R/W	<p>SPI 接收中断使能</p> <p>如果该位设置为“1”，且 SPI 中断发生时，硬件将向 CPU 发出一个中断。如果该位清零，该中断将被屏蔽。</p> <p>0：禁止接收中断 1：使能接收中断</p>	0x0
13:10	-	-	保留	0
9	RXFULL	R	<p>接收 FIFO 满标志。</p> <p>当接收 FIFO 已满时，该位由硬件置 1。当接收 FIFO 未滿时，硬件将清零该位。</p> <p>0：接收 FIFO 不满 1：接收 FIFO 满</p>	0x0
8	RXFOV	R	<p>接收 FIFO 溢出寄存器。</p> <p>接收 FIFO 已满并且从 SPI 数据总线上接收到更多数据时，该位将置 1。向 SPIRXIF 位写 1 会将该位清零。</p>	0x0
7	-	-	保留	0
6:4	RXFLEV	R/W	<p>接收 FIFO 的中断触发阈值。</p> <p>该寄存器表示在发生中断时接收 FIFO 中有多少字节的数据。设置值越大，中断触发越少，因此可以在一个中断中读取更多数据。</p> <p>000：FIFO 中的数据 ≥ 1，允许 1 次数据读取。 001：FIFO 中的数据 ≥ 2，允许 2 次数据读取。 010：FIFO 中的数据 ≥ 3，允许 3 次数据读取。 011：FIFO 中的数据 ≥ 4，允许 4 次数据读取。 100：FIFO 中的数据 ≥ 5，允许 5 次数据读取。 101：FIFO 中的数据 ≥ 6，允许 6 次数据读取。 110：FIFO 中的数据 ≥ 7，允许 7 次数据读取。 111：FIFO 中的数据 ≥ 8，允许 8 次数据读取。</p>	0x0
3	-	-	保留	0

2:0	RXFFLAG	R	接收 FIFO 数据个数。 该寄存器表示接收 FIFO 的数据数量。 0000: FIFO 中没有数据或 FIFO 中有 8 个字节 0001: FIFO 中有 1 个字节。 0010: FIFO 中有 2 个字节。 0011: FIFO 中有 3 个字节。 0100: FIFO 中有 4 个字节。 0101: FIFO 中有 5 个字节。 0110: FIFO 中有 6 个字节。 0111: FIFO 中有 7 个字节。	0x0
-----	---------	---	--	-----

## 22.5.5 SPI\_RXDAT 寄存器

SPI\_RXDAT 寄存器用于 SPI 接收机从接收 FIFO 中读取数据。

表 22-6 SPI\_RXDAT 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	SPIRXD	R	从 SPI 接收 FIFO 中读取数据。	--

## 22.5.6 SPI\_MISC 寄存器

SPI\_MISC 寄存器用于配置 SPI 接收发送特殊功能。

表 22-7 SPI\_MISC 寄存器

Bit	名称	类型	描述	默认值
31:10	-	-	保留	0
9	OVER	R/W	SPI FIFO 溢出控制。 该寄存器用于控制当 TX/RXFIFO 已满时，数据会被重写或跳过。 0: 将跳过对已满 FIFO 的进一步写操作。 1: 写入已满 FIFO，会覆盖 FIFO 中最先写入的数据。	0x0
8	SMART	R/W	SPI FIFO SMART 模式寄存器。 如果该位设置为“1”，用户在 FIFO 状态到达中断等级时不需要手动清除发送/接收中断标志，用户只需向发送 FIFO 写入数据或从接收 FIFO 读取数据，中断标志将自动清零。当选择 DMA 模式时，必须将该位设置为 SMART 模式，以保证 DMA 操作的正确。 0: 正常中断清零 1: 智能中断清零	0x0
7:5			保留	0
4	BSY	R	SPI 控制器忙标志。 该位用于指示 SPI 控制器是否忙。 0: 空闲	0x0

			1: 忙	
3	RFF	R	接收 FIFO 满标志。 该位指示 SPI 控制器接收 FIFO 是否为满。 0: 接收 FIFO 未满足 1: 接收 FIFO 已满足	0x0
2	RNE	R	接收 FIFO 非空标记。 该位显示接收 FIFO 是否有数据。 0: 接收 FIFO 为空 1: 接收 FIFO 非空	0x0
1	TNF	R	发送 FIFO 未满足标记。 该位显示发送 FIFO 是否未满足。 0: 发送 FIFO 满, 不能写入 1: 发送 FIFO 未满足	0x1
0	TFE	R	发送 FIFO 为空标记 该位显示发送 FIFO 是否为空。 0: 发送 FIFO 非空 1: 发送 FIFO 空	0x1

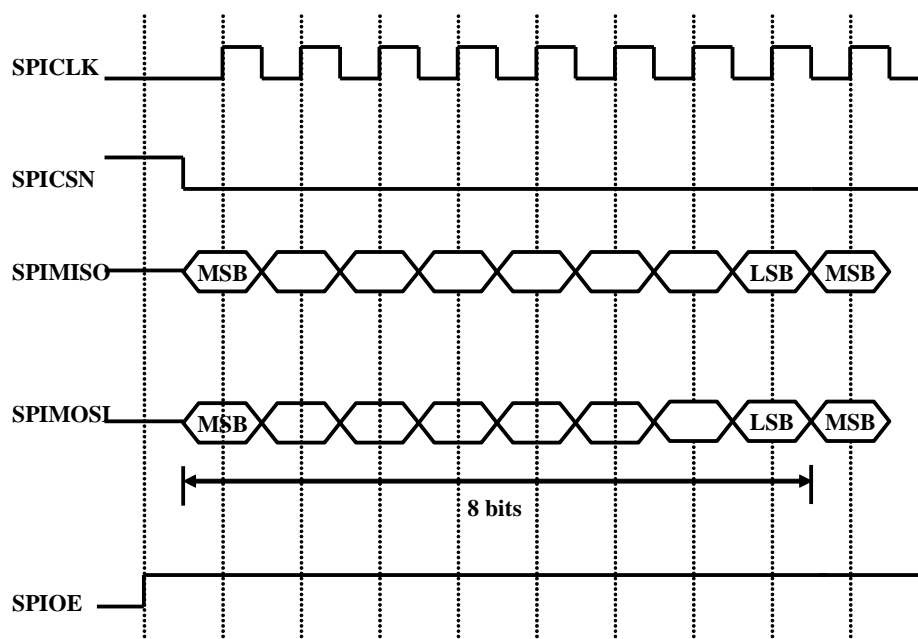
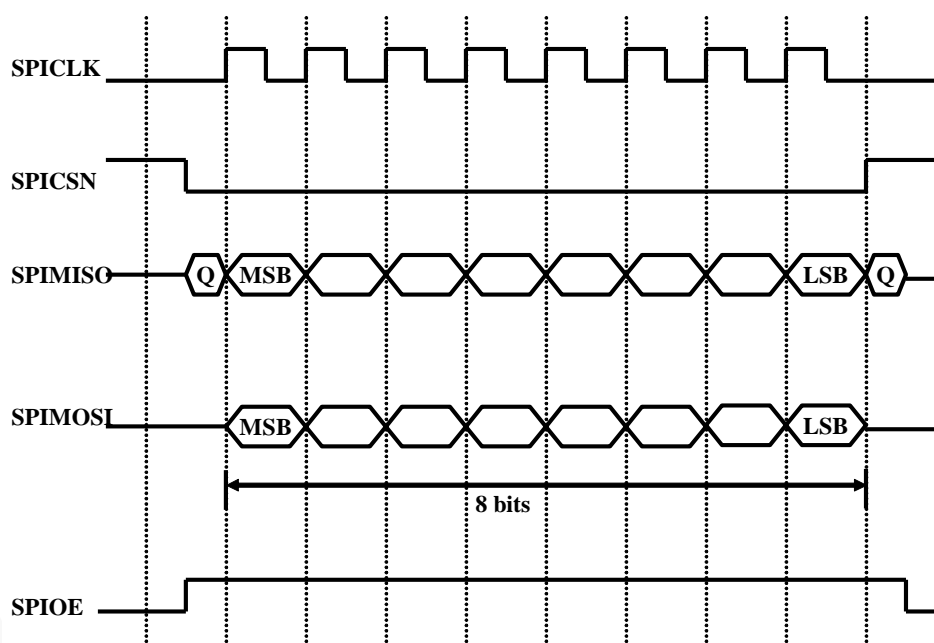
## 22.6 应用笔记

### 22.6.1 主机模式

主控模式下, 移位时钟 (SPICLK) 由 V94XX(A) 产生。两个控制位分别控制时钟相位和极性。向 SPI\_TXDAT 寄存器写入一个数据即可开始数据传输。只要 FIFO 中有数据, 在发送完一字节数据后会自动重新开始发送下一个数据。

SPI 从 SDO 引脚发送数据时, 先发高位, 再发低位。移出 8 位数据需要 8 个 SCK 周期。同时, 数据也通过从器件 SDI 引脚移入。当发送 FIFO 等级低于中断触发等级时, SPITXIF 标志位将被置 1。同时, 如果 SPITXIEN 位置 1, 将产生 SPI 中断。当接收 FIFO 等级高于中断触发等级时, SPIRXIF 标志位将被置 1。同时, 如果 SPIRXIEN 位置 1, 将产生 SPI 中断。从 SPI\_RXDAT 寄存器可以读取 SPI 数据。

下图展示 SPI 主机模式下不同操作类型配置下的不同时序方案。(极性控制位等于“1”或“0”, 相位控制位等于“1”或“0”)。

图 22-2 主机模式,  $SPO=0$ ,  $SPH=0$ 图 22-3 主机模式,  $SPO=0$ ,  $SPH=1$

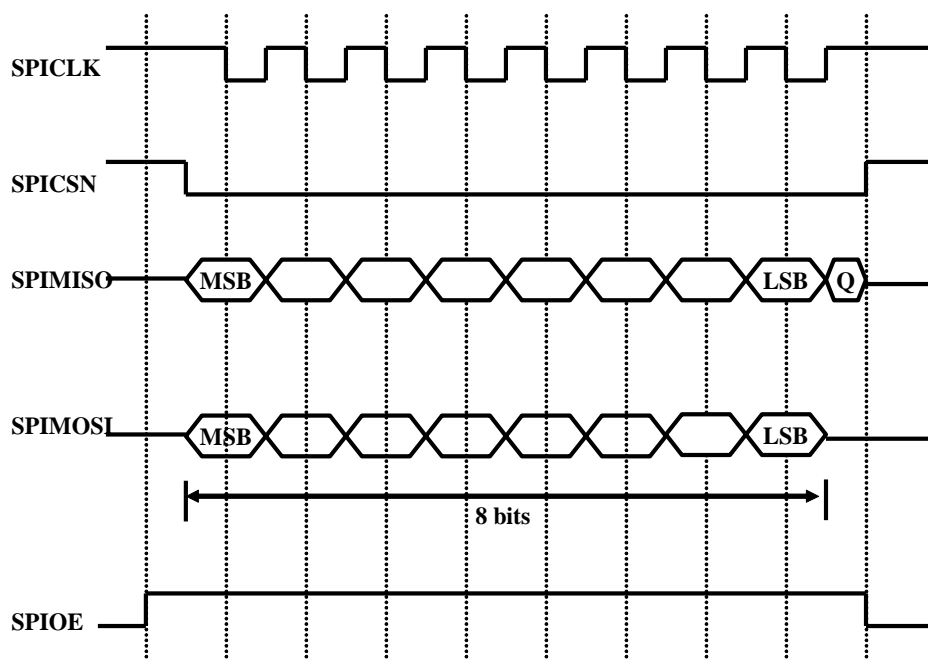


图 22-4 主机模式，SPO=1，SPH=0

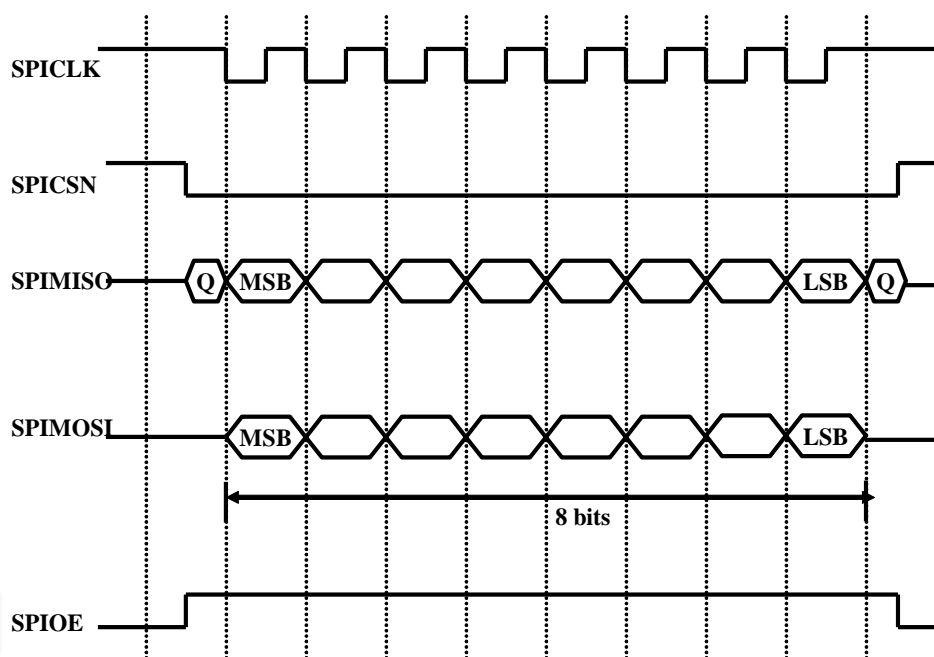


图 22-5 主机模式，SPO=1，SPH=1

## 22.6.2 从机模式

SPI 从机模式时，SPICLK 和 SPICSN 由外部主机产生。SPI 从机模式仅支持时钟极性和相位都为 0（SPO=0 和 SPH=0）。SPI\_TXDAT 和 SPI\_RXDAT 寄存器都可以用于数据发送或接收。SPI 从机模式支持的最大 SPICLK 时钟为 1/2APBCLK。当接收时钟速率较快时，建议使用 DMA 传输。

## 22.6.3 连续字节传输

连续字节传输在主机模式或从机模式都可用。发送数据时，只要 TNF 位为 1，软件就能够连续发送数据。接收数据时，软件会检查溢出标志，以监测是否由于轮询速率太低而丢失数据。

## 第23章 I<sup>2</sup>C 控制器

### 23.1 简介

I<sup>2</sup>C 子组件是 I<sup>2</sup>C 总线控制器，它提供了一个满足 Philips I<sup>2</sup>C 总线规范的接口，并支持 I<sup>2</sup>C 总线双向传输。I<sup>2</sup>C 总线使用两条线在连接到总线的设备之间传输信息：“scl”（串行时钟线）和“sda”（串行数据线）。I<sup>2</sup>C 支持单字节自动传输，支持串行传输，状态寄存器反映 I<sup>2</sup>C 总线控制器和 I<sup>2</sup>C 总线的状态。I<sup>2</sup>C 控制器中的设置在从浅睡眠或深睡眠模式唤醒后会被复位，用户应在浅睡眠/深睡眠模式唤醒后手动恢复设置。I<sup>2</sup>C 总线最大支持 fSCL= 1MHz（周期：1μs），可以通过设置定时器 3 溢出频率实现。

### 23.2 特点

- 支持主机发送、主机接收模式。
- 支持从机发送、从机接收模式。
- 总线仲裁失败监测。
- 支持中断信号。
- 应答（ACK）产生可编程。
- 主机模式时钟速率可编程。
- 输入端具有消抖电路。

### 23.3 功能框图

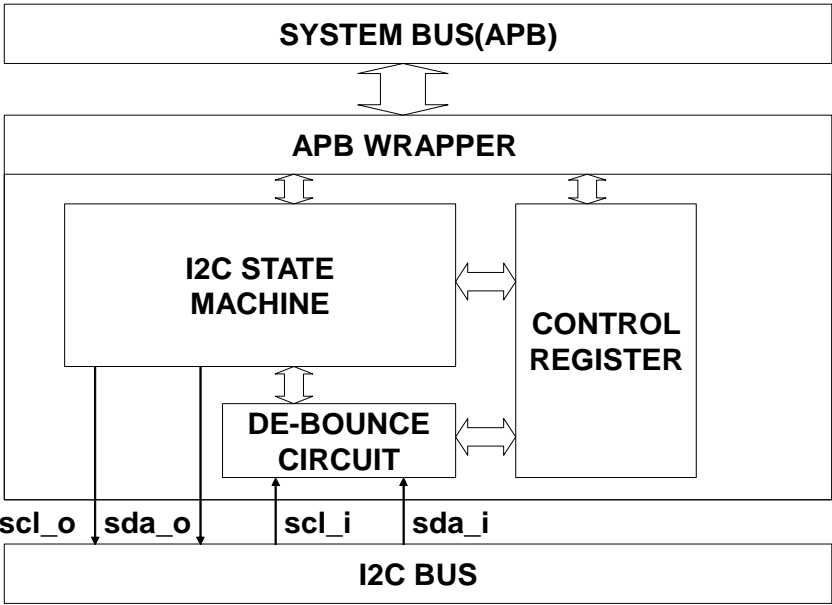


图 23-1 I<sup>2</sup>C 控制器功能框图

### 23.4 寄存器地址

表 23-1 I<sup>2</sup>C 控制器(I<sup>2</sup>C 基地址：0x40010800)

名称	类型	地址	描述	默认值
----	----	----	----	-----

I2C_DATA	R/W	0x0000	I <sup>2</sup> C 数据寄存器	0x00
I2C_ADDR	R/W	0x0004	I <sup>2</sup> C 地址寄存器	0x00
I2C_CTRL	R/W	0x0008	I <sup>2</sup> C 控制/状态寄存器	0x00
I2C_STS	R/W	0x000c	I <sup>2</sup> C 状态寄存器	0xF8
I2C_CTRL2	R/W	0x0018	I <sup>2</sup> C 中断使能寄存器	0x0

## 23.5 寄存器定义

### 23.5.1 I2C\_DATA 寄存器

表 23-2 I2C\_DATA 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	DATA	R/W	I2C_DATA 寄存器包含一个要通过 I <sup>2</sup> C 总线传输的字节或刚通过 I <sup>2</sup> C 总线接收的字节。该 8 位寄存器不处于字节移位时, CPU 可以对其进行读写操作。I2C_DATA 寄存器没有影子寄存器或双缓存, 因此用户只能在发生 I <sup>2</sup> C 中断时读取 I2C_DATA。	0x00

### 23.5.2 I2C\_ADDR 寄存器

表 23-3 I2C\_ADDR 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:1	SLA	R/W	自有 I <sup>2</sup> C 从地址(7 位)	0x00
0	GC	R/W	广播呼叫地址响应。 如果该位置 1, 则识别广播呼叫地址; 否则会忽略广播呼叫地址。	0x0

### 23.5.3 I2C\_CTRL 寄存器

表 23-4 I2C\_CTRL 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7	CR2	R/W	时钟速率控制位 2	0x0
6	EN	R/W	I <sup>2</sup> C 使能位 当 EN='0', 会将"sdao"和"sclo"输出置为 1, 芯片输出处于高阻状态。此时, "sdai"和"scli"引脚的输入信号会被忽略。 当 EN='1', I <sup>2</sup> C 组件被使能, 相应 IO 将自动设置为 I <sup>2</sup> C 功能。	0x0
5	STA	R/W	START 标志 当 STA='1'时, I <sup>2</sup> C 组件检查 I <sup>2</sup> C 总线状态, 如果总线空闲, 则产生 START 状态。	0x0
4	STO	R/W	STOP 标志	0x0



			当 STO='1' 且 I <sup>2</sup> C 接口处于主机模式时，会向 I <sup>2</sup> C 总线发送一个 STOP 状态。	
3	SI	R/C	串行中断标志  I <sup>2</sup> C 输入状态为 26 种可能输入的 25 种（F8h 除外）时，“SI”由硬件置 1。F8h 是唯一不置位“SI”的状态，其表示没有相关的可用状态信息。“SI”标志必须由软件清零。为了清除“SI”位，必须向该位写“0”。向该位写 1 不会改变“SI”的值。当发生中断时，清除 SI 表示下一个动作的触发，因此在清除 SI 之前，必须设置好必需的控制寄存器或数据。	0x0
2	AA	R/W	有效应答标志  当 AA='1' 时，将在以下情况下返回“应答”： -已接收到“自有从地址” -在 I <sup>2</sup> C 地址寄存器中的 GC 位置 1 时接收到广播呼叫地址 -当 I <sup>2</sup> C 处于主接收器模式时，已接收到数据字节 -当 I <sup>2</sup> C 处于从接收器模式时，已接收到数据字节  当 AA='0' 时，将在以下情况下返回“不应答”： -当 I <sup>2</sup> C 处于主接收器模式时，已接收到数据字节 -当 I <sup>2</sup> C 处于从接收器模式时，已接收到数据字节	0x0
1	CR1	R/W	时钟速率控制位 1	0x0
0	CR0	R/W	时钟速率控制位 0	0x0

表 23-5 SCL 频率设置显示在不同 CR2~CR0 设置下，I<sup>2</sup>C 作为主机的 SCL 时钟频率。

表 23-5 SCL 频率设置

CR2	CR1	CR0	SCL 频率(KHz)				时钟分频数
			PCLK= 6.5536MHz	PCLK= 13.1072MHz	PCLK= 19.6608MHz	PCLK= 26.2144 MHz	
0	0	0	25.6	51.2	76.8	102.4	256
0	0	1	29.2	58.4	87.6	116.8	224
0	1	0	34.1	68.2	102.3	136.4	192
0	1	1	40.9	81.9	122.9	163.8	160
1	0	0	6.8	13.7	20.5	27.3	960
1	0	1	54.6	109.2	163.8	218.4	120
1	1	0	109.2	218.4	327.7	436.9	60
1	1	1	TIMER3 溢出频率/8				

## 23.5.4 I2C\_STS 寄存器

表 23-6 I2C\_STS 寄存器

Bit	名称	类型	描述	默认值
31:8	-	-	保留	0
7:3	STS	R	I <sup>2</sup> C 状态码	0x1F
2:0		R	保留	0

表 23-7 主机发送模式下的 I<sup>2</sup>C 状态

STS	I <sup>2</sup> C 状态	应用软件响应					I <sup>2</sup> C 的下一步动作
		读/写 I <sup>2</sup> C_DATA	I <sup>2</sup> C_CTRL				
			STA	STO	SI	AA	
0x08	START 状态已发送。	写 SLA+W*	X	0	0	X	SLA+W 将发送, ACK 将被接收。
0x10	重复 START 状态发送。	写 SLA+W	X	0	0	X	SLA+W 将发送, ACK 将被接收。
		写 SLA+R*	X	0	0	X	SLA+R 将发送, I <sup>2</sup> C 将会切换到主机接收模式。
0x18	SLA+W 已发送, ACK 已接收。	写数据字节	0	0	0	X	数据字节将发送, ACK 将被接收。
		无操作	1	0	0	X	重复 START 状态将发送。
		无操作	0	1	0	X	STOP 状态将发送, STO 标志将复位。
		无操作	1	1	0	X	STOP 状态发送后, 紧跟着发送一个 START 状态, STO 标志会复位。
0x20	SLA+W 已发送, ACK 未接收到。	写数据字节	0	0	0	X	将传输数据字节, ACK 将被接收。
		无操作	1	0	0	X	重复 START 将发送。
		无操作	0	1	0	X	STOP 状态将发送, STO 标志将复位。
		无操作	1	1	0	X	STOP 状态发送后, 紧接着发送一个 START 状态, STO 标志会复位。
0x28	I <sup>2</sup> C 数据字节已发送, ACK 已接收。	写数据字节	0	0	0	X	数据字节将发送, ACK 将被接收。
		无操作	1	0	0	X	重复 START 会发送。
		无操作	0	1	0	X	STOP 状态将发送, STO 标志将复位。
		无操作	1	1	0	X	STOP 状态发送后, 紧接着发送一个 START 状态, STO 标志会复位。
0x30	I <sup>2</sup> C 数据字节已发送, ACK 未接收。	写数据字节*	0	0	0	X	数据字节将发送, ACK 位将被接收。
		无操作	1	0	0	X	重复 START 状态会发送。
		无操作	0	1	0	X	STOP 状态将发送, STO 标志将复位。
		无操作	1	1	0	X	STOP 状态发送后, 紧接着发送一个 START 状态, STO 标志将复位。

\*: SLA+W: Slave 地址[7: 1]+写操作控制位[0:0], 0 表示写。

\*: SLA+R: Slave 地址[7: 1]+读操作控制位[0:0], 1 表示读。

表 23-8 主机接收模式下的 I<sup>2</sup>C 状态

STS	I <sup>2</sup> C 状态	应用软件响应					I <sup>2</sup> C 的下一步动作
		读/写 I <sup>2</sup> C_DATA	I <sup>2</sup> C_CTRL				
			STA	STO	SI	AA	
0x08	START 状态已发送。	写 SLA+R	X	0	0	X	将发送 SLA+R，将接收 ACK。
0x10		写 SLA+R	X	0	0	X	将发送 SLA+R，将接收 ACK。

	重复 START 状态发送。	写 SLA+W	X	0	0	X	将发送 SLA+W, I <sup>2</sup> C 将切换为主机发送模式。
0x40	SLA+R 已发送, ACK 已接收。	无操作	0	0	0	0	将接收数据字节, 无 ACK 返回。
		无操作	0	0	0	1	将接收数据字节, 将返回 ACK。
0x48	SLA+R 已发送, ACK 未接收。	无操作	1	0	0	X	重复 START 状态将发送。
		无操作	0	1	0	X	STOP 状态将发送, STO 标志将重置。
		无操作	1	1	0	X	STOP 状态发送后, 紧接着发送一个 START 状态, STOP 标志将重置。
0x50	I <sup>2</sup> C 数据的数字字节已接收, 已返回 ACK。	读数据字节	0	0	0	0	将接收数据字节, 无 ACK 返回。
		读数据字节	0	0	0	1	将接收数据字节, 返回 ACK。
0x58	I <sup>2</sup> C 数据的数字字节已发送, 未返回 ACK。	读数据字节	1	0	0	X	重复 START 状态将发送。
		读数据字节	0	1	0	X	STOP 状态将发送, STO 标志将重置。
		读数据字节	1	1	0	X	STOP 状态发送后, 紧接着发送一个 START 状态, STOP 标志将重置。

表 23-9 从机接收模式下的 I<sup>2</sup>C 状态

STS	I <sup>2</sup> C 状态	应用软件响应					I <sup>2</sup> C 的下一步动作
		读/写 I <sup>2</sup> C_DATA	I <sup>2</sup> C_CTRL				
			STA	STO	SI	AA	
0x60	接 收 到 自 有 SLA+W, 已返回 ACK。	无操作	X	0	0	0	将接收数据字节, 并且不返回 ACK。
		无操作	X	0	0	1	将接收数据字节, 并且返回 ACK。
0x68	仲 裁 在 SLA+R/W 作为 主机时丢失。  已 接 收 自 有 SLA+W, ACK 已返回。	无操作	X	0	0	0	将接收数据字节, 并且不返回 ACK。
		无操作	X	0	0	1	将接收数据字节, 并且返回 ACK。
0x70	已接收到广播地 址 (00H), ACK 已返回。	无操作	X	0	0	0	将接收数据字节, 并且不返回 ACK。
		无操作	X	0	0	1	将接收数据字节, 并且返回 ACK。
0x78	仲 裁 在 SLA+R/W 作为 主机时丢失。  已接收到广播地 址 (00H), ACK 已返回。	无操作	X	0	0	0	将接收数据字节, 并且不返回 ACK。
		无操作	X	0	0	1	将接收数据字节, 并且返回 ACK。
0x80	之前在使用自己 的 SLV 地址寻 址。DATA 已收 到。ACK 返回。	读数据字节	X	0	0	0	将接收数据字节, 并且不返回 ACK。
		读数据字节	X	0	0	1	将接收数据字节, 并且返回 ACK。

0x88	之前在使用自己的 SLA 地址寻址。DATA 字节已收到。无 ACK 返回。	读数据字节	0	0	0	0	切换为无寻址 SLV 模式。没有识别到自有 SLA 或广播地址
		读数据字节	0	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。
		读数据字节	1	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
		读数据字节	1	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
0x90	之前在使用广播地址寻址。DATA 已收到。ACK 已返回。	读数据字节	X	0	0	0	将收到数据字节，无 ACK 返回。
		读数据字节	X	0	0	1	将接收数据字节，并且返回 ACK。
0x98	之前在使用广播呼叫地址寻址。DATA 已收到。无 ACK 返回。	读数据字节	0	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。
		读数据字节	0	0	0	1	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。
		读数据字节	1	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
		读数据字节	1	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
0xA0	已接收到 STOP 状态或重复 START 状态，但仍以从接收或从发送模式寻址。	无操作	0	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。
		无操作	0	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。
		无操作	1	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
		无操作	1	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。

表 23-10 从机发送模式下的 I<sup>2</sup>C 状态

STS	I <sup>2</sup> C	应用软件响应					I <sup>2</sup> C 的下一步动作
		读/写	I2C_CTRL				
			I2C_DATA	STA	STO	SI	
0xA8	已接收自有 SLA+R，已返回 ACK。	写数据字节	X	0	0	0	将发送上一个数据字节，并接收 ACK。
		写数据字节	X	0	0	1	将发送数据字节，并接收 ACK。
0xB0	仲裁在 SLA+R/W 作为主机时丢失。已接收自有	写数据字节	X	0	0	0	将发送上一个数据字节，并接收 ACK。
		写数据字节	X	0	0	1	将发送数据字节，并接收 ACK。

	SLA+R, ACK 已返回。						
0xB8	数据字节已发送, 已接收 ACK。	写数据字节	X	0	0	0	将发送上一个数据字节, 并接收 ACK。
		写数据字节	X	0	0	1	将发送数据字节, 并接收 ACK。
0xC0	数据字节已发送, 未接收 ACK。	无操作	0	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。
		无操作	0	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。
		无操作	1	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
		无操作	1	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
0xC8	上一个数据字节已发送, 并已接收 ACK。	无操作	0	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。
		无操作	0	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。
		无操作	1	0	0	0	切换为无寻址 SLV 模式。未识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。
		无操作	1	0	0	1	切换为无寻址 SLV 模式。将识别到自有 SLA 或广播地址。START 状态将在总线空闲时发送。

表 23-11 I<sup>2</sup>C 状态

STS	I <sup>2</sup> C 状态	应用软件响应					I <sup>2</sup> C 硬件下一步动作
		读/写 I <sup>2</sup> C_DATA	I <sup>2</sup> C_CTRL				
			STA	STO	SI	AA	
0x38	已接收自有 SLA+R，ACK 已返回。	写数据字节	X	0	0	0	将发送上一个数据字节，并接收 ACK。
		写数据字节	X	0	0	1	将发送数据字节，并接收 ACK。
0xF8	仲裁在 SLA+R/W 作为主机时丢失。已接收自有 SLA+R，ACK 已返回。	写数据字节	X	0	0	0	将发送上一个数据字节，并接收 ACK。
0x00	数据字节已发送。ACK 已接收。	写数据字节	X	0	0	0	将发送上一个数据字节，并接收 ACK。

## 23.5.5 I2C\_CTRL2 寄存器

表 23-12 I2C\_CTRL2 寄存器

Bit	名称	类型	描述	默认值
-----	----	----	----	-----

31:1	-	-	保留	0
0	INTEN	R/W	I <sup>2</sup> C 控制器的中断使能控制。 0: 禁止。 1: 使能。	0x0

## 23.6 应用笔记

### 23.6.1 I<sup>2</sup>C 总线协议：启动/停止的发生

启动状态可以通过 SDA 线传输一个字节的串行数据，停止状态可以终止数据传输。停止状态是 SCL 为高电平时 SDA 线的低电平向高电平跳变。开始和停止状态始终由主器件生成。当产生启动状态时，I<sup>2</sup>C 总线忙。停止状态后几个时钟，I<sup>2</sup>C 总线将再次被释放。

当主器件发起启动状态时，它应发送从器件地址以确认从器件。一字节的地址包含 7 位地址和 1 位传输方向指示符（即，写或读）组成。如果第 8 位为 0，则表示写操作（发送操作）；如果第 8 位为 1，则表示数据读取请求（接收操作）。

主机通过发送停止状态结束传输操作。如果主机想继续向 I<sup>2</sup>C 总线传输数据，则它应该重新发送一个启动状态和从机地址。通过这种方式，可以进行各种格式的读/写操作。

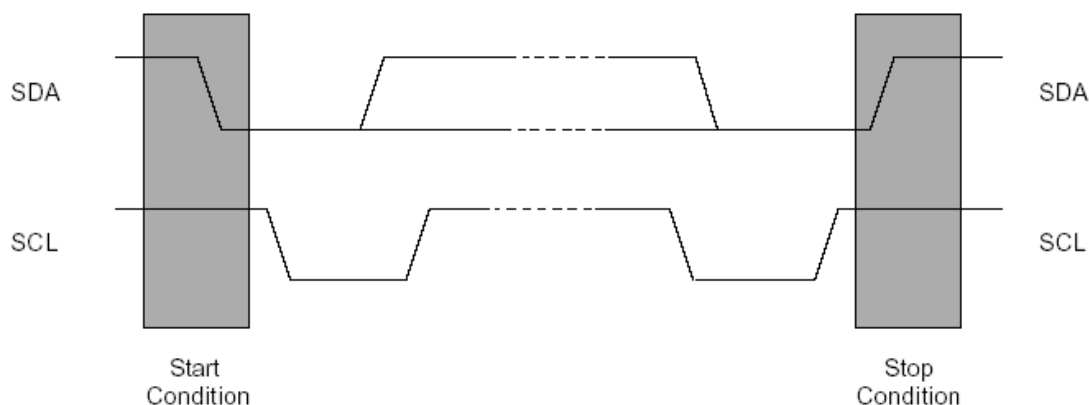


图 23-2 启动和停止条件

### 23.6.2 数据传输格式

位于 SDA 线上的每个字节的长度应为 8 位。每次允许传输的字节数量没有限制。启动状态后的第一个字节应该为地址字段。当 I<sup>2</sup>C 总线工作在主模式时，地址字段可以由主机发送。每个字节后面应跟一个确认（ACK）位。串行数据和地址总是高位（MSB）先发。

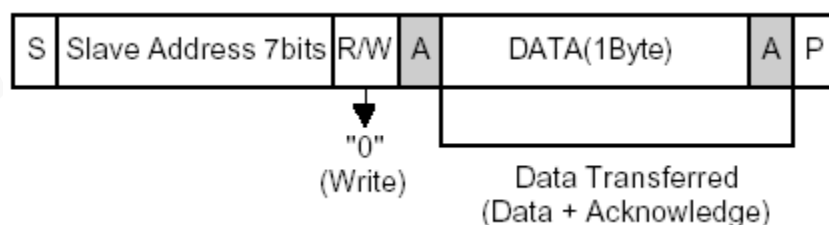


图 23-37 位地址写模式

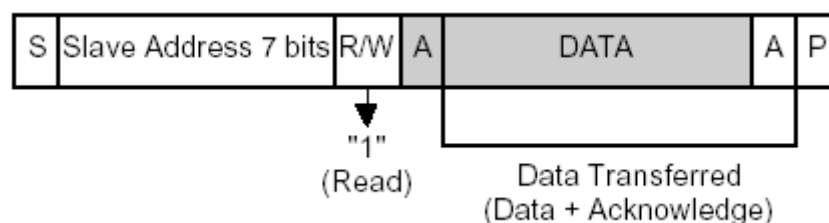




图 23-47 位地址读模式

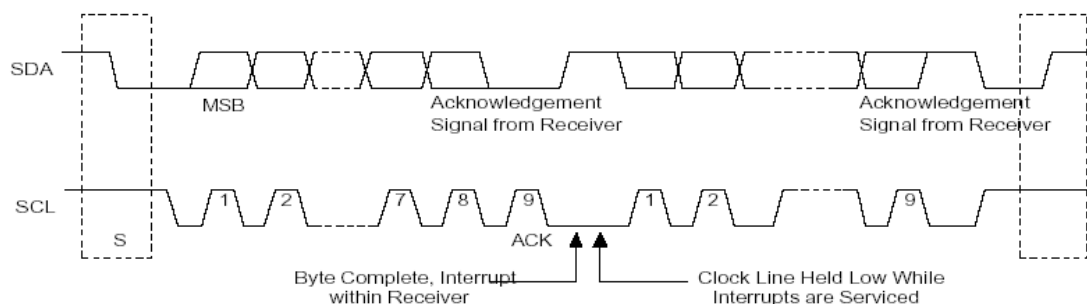


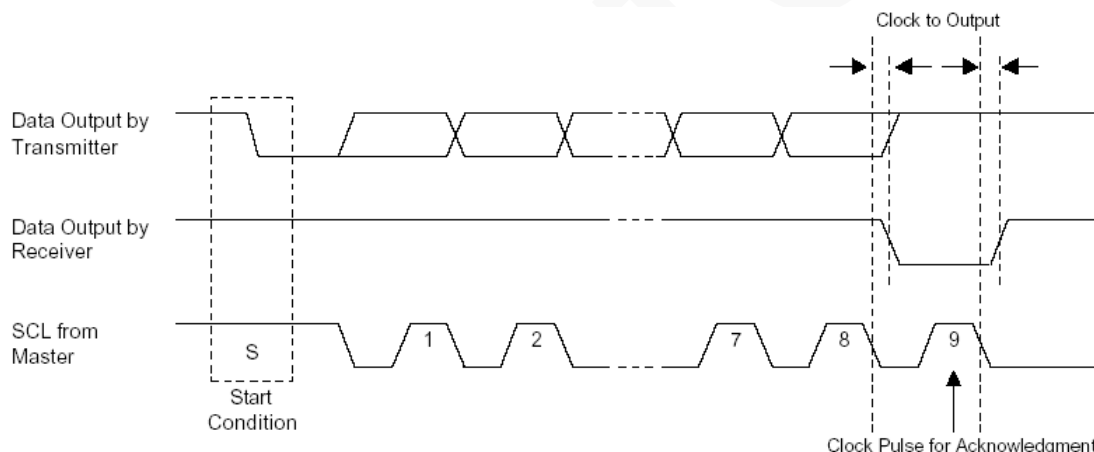
图 23-5 数据传输

### 23.6.3 ACK 信号传输

要完全完成一个字节的传输操作，接收器应向发送器发送一个 ACK 位。ACK 脉冲应在 SCL 线的第 9 个时钟周期发生。一字节数据传输需要八个时钟周期。主机应产生发送 ACK 位所需的时钟脉冲。

当接收到 ACK 时钟脉冲时，发送器应通过使 SDA 线设置为高电平来释放 SDA 线。接收器还应在 ACK 时钟脉冲期间将 SDA 线驱动为低电平，以便在第九个 SCL 脉冲的高电平期间 SDA 为低电平。

可以通过软件（AA）使能或禁止 ACK 位的发送功能。但是，SCL 的第九个时钟周期上的 ACK 脉冲需要完成一个字节的传输操作。

图 23-6 I<sup>2</sup>C 总线的应答机制

### 23.6.4 读写操作

发送模式下，数据传输后，I<sup>2</sup>C 总线接口将等待，直到挂起的中断被清除。在中断被清除之前，SCL 线将保持低电平。中断清除后，SCL 线将被释放。在 CPU 接收到中断请求后，它应该在清除挂起中断之前将新数据写入 I2C\_DATA。

接收模式下，在接收到数据后，I<sup>2</sup>C 总线接口将等待，直到挂起的中断被清除。在等待中断被清除之前，SCL 线将保持低电平。等待中断清除后，SCL 线将被释放。在 CPU 接收到中断请求后，它应该在清除挂起中断之前从 I2C\_DATA 中读取数据。

#### 总线仲裁过程

仲裁发生在 SDA 线路上，以防止两个主机之间的总线上的竞争。如果 SDA 为高电平的主机监测到另一个主机的 SDA 为低电平，则它不会启动数据传输，因为总线上的当前电平与其自身不对应。仲裁过程将延长，直到 SDA 线变为高电平。

然而，当主机 SDA 线同时为低电平时，每个主机应该评估控制权是否属于自己。为了进行评估，每个主机应该监测地址位。当每个主机产生从机地址时，它也应该监测 SDA 线上的地址位，因为 SDA 电平为低比 SDA 线为高电平的优先级更高。例如，一个主机的首地址位是低电平，而另一个主机维持高电平，在这种情况下，两个主机将监测总线上的低电平，因为即使第一主机试图在电路上保持高电平，低电平也强于高电平。当这种情况发生时，产生低电平（作为地址





tHIGH	scl/2	min(30clk)	scl/2	(4bclk)
tSU:STA	scl/4	min(15clk)	scl/2-if(scl/4<7clk)	
			scl/4-others	
tHD:DAT	6*clk	(6clk)	6*clk	(6clk)
tSU:DAT	1*clk(-forfirstbit)*	(1clk)	1*clk(-forfirstbit)*	(1clk)
	scl/2-6*clk	min(24clk)	scl/2-6*clk	min(1clk)
tSU:STO	scl/4+7*clk	min(22clk)	min(bclk+9)	
tBUF	( $\frac{3}{4}$ scl)+9*clk	min(54clk)	min( $\frac{3}{4}$ scl)	

\*-第一个字节数据的第一个位建立时间会短一点，只有一个 clk，而这种状况只会发生在写入数据寄存器时，刚好很接近内部 CSL 的上升沿。

Clk: APB 时钟周期

bclk: 定时器 3 的溢出周期

scl: I<sup>2</sup>C 时钟周期

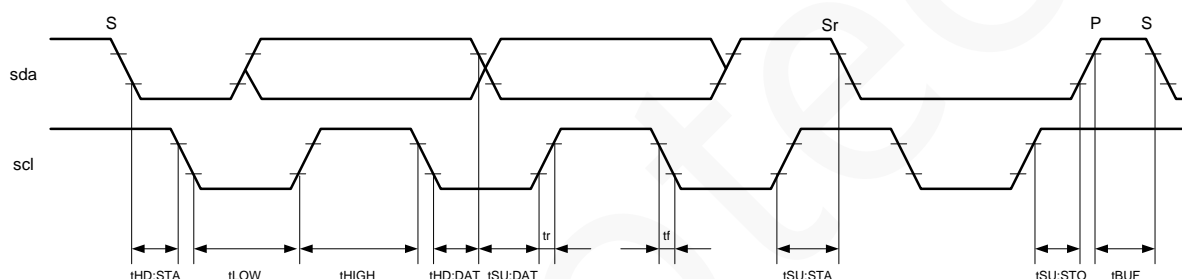


图 23-9 I<sup>2</sup>C 控制器的 AC 时序

### 23.6.6 I<sup>2</sup>C SDA 管脚死锁的解决方法

I<sup>2</sup>C 通讯过程中，当芯片异常复位（软复位、看门狗复位、外部 EXRST 管腿复位）发生时，有可能会发生 I<sup>2</sup>C 总线死锁（SDA 死锁在低电平状态）。规避方法：在开启 IIC 之前，用户增加 I<sup>2</sup>C 总线恢复时序。即每次 I<sup>2</sup>C 主设备复位后，软件模拟控制 I<sup>2</sup>C 中的 SCL 时钟线产生 9 个时钟脉冲，这样 I<sup>2</sup>C 的 SDA 管腿可以从死锁状态中恢复过来。如果当前不处于死锁状态，增加总线恢复程序也不影响 I<sup>2</sup>C 使用。

## 23.7 工作流程

### 23.7.1 主机发送模式

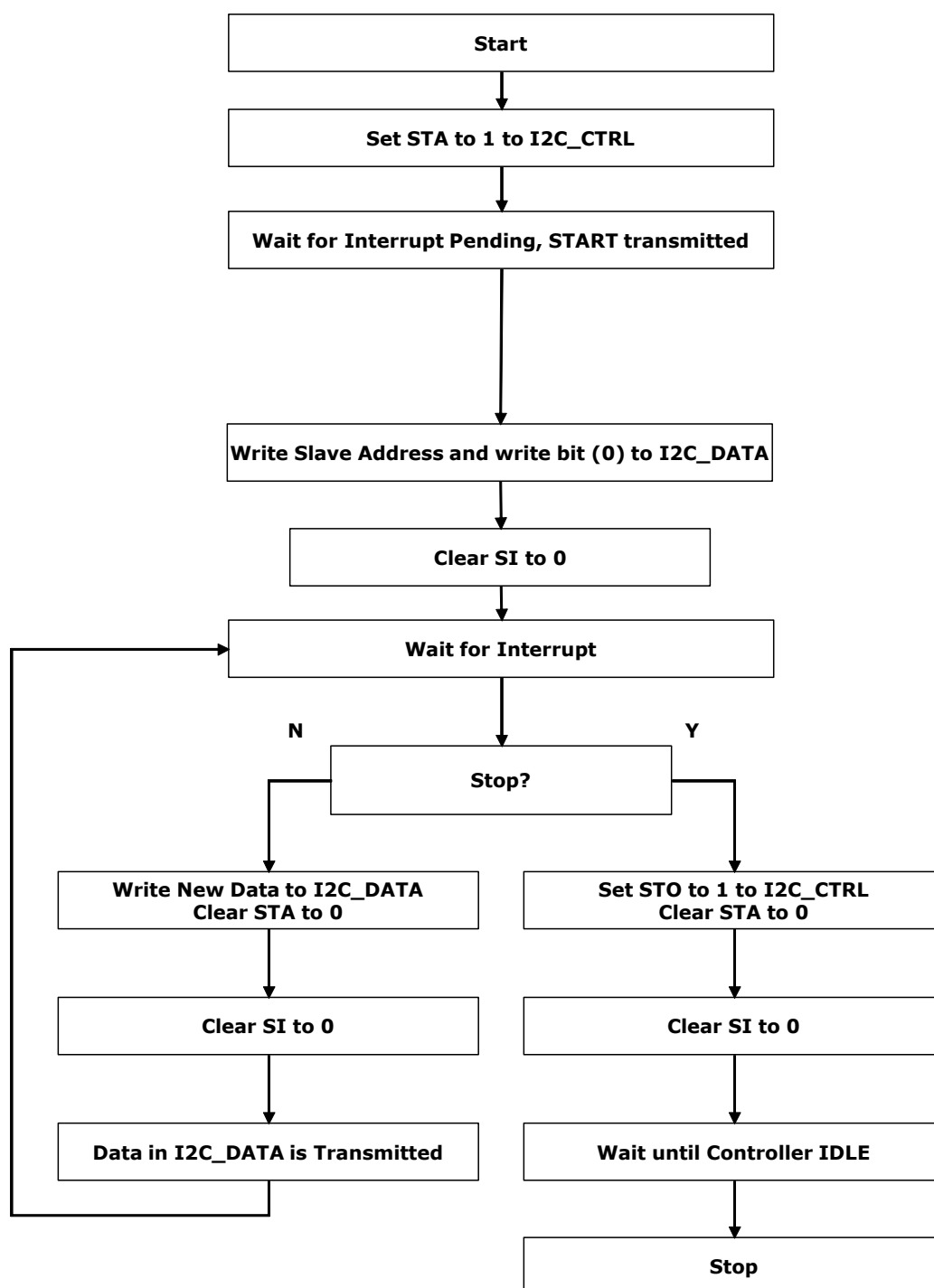


图 23-10 主机发送模式的操作流程

## 23.7.2 主机接收模式

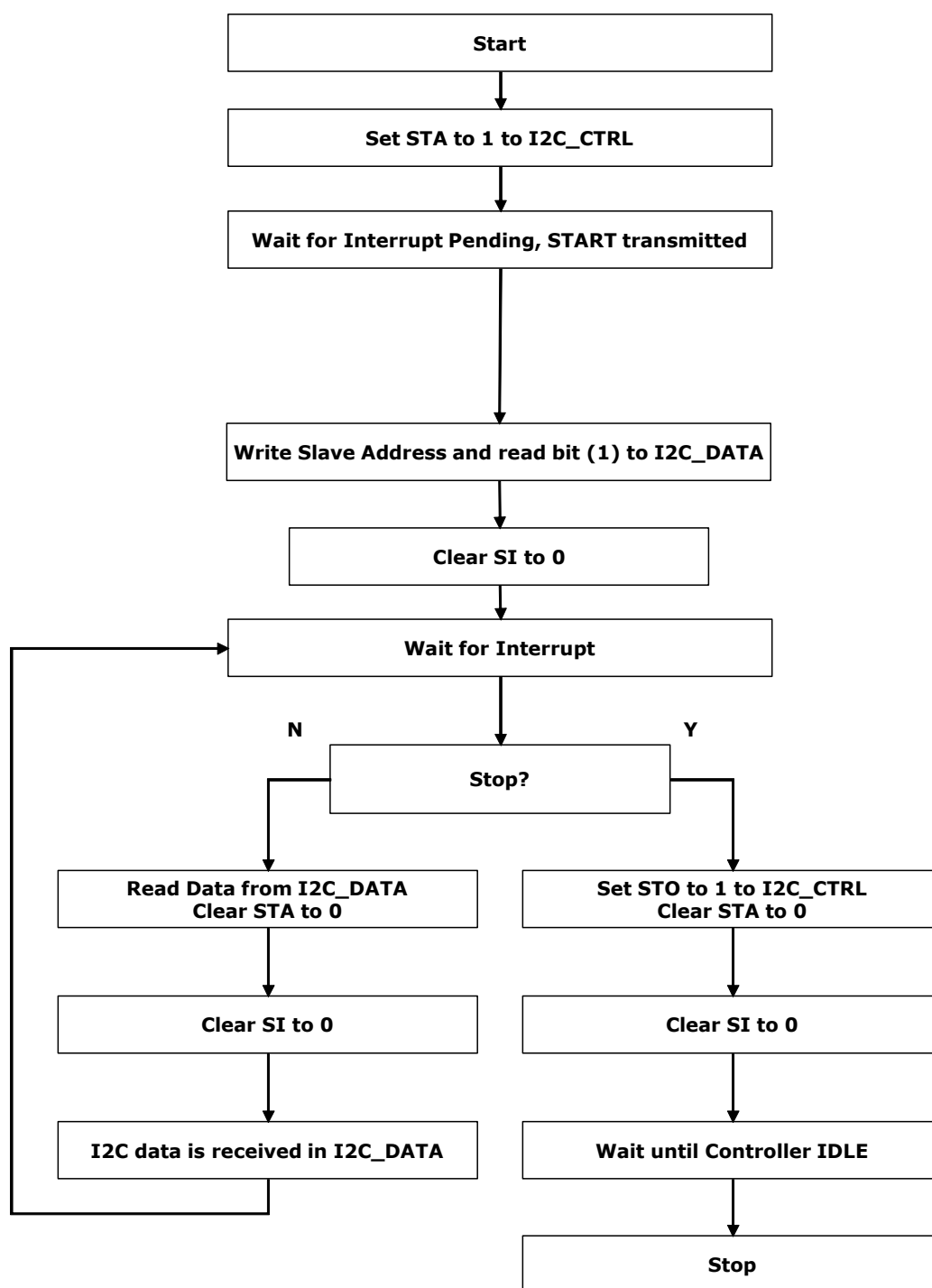


图 23-11 主机接收模式的操作流程

### 23.7.3 从机发送模式

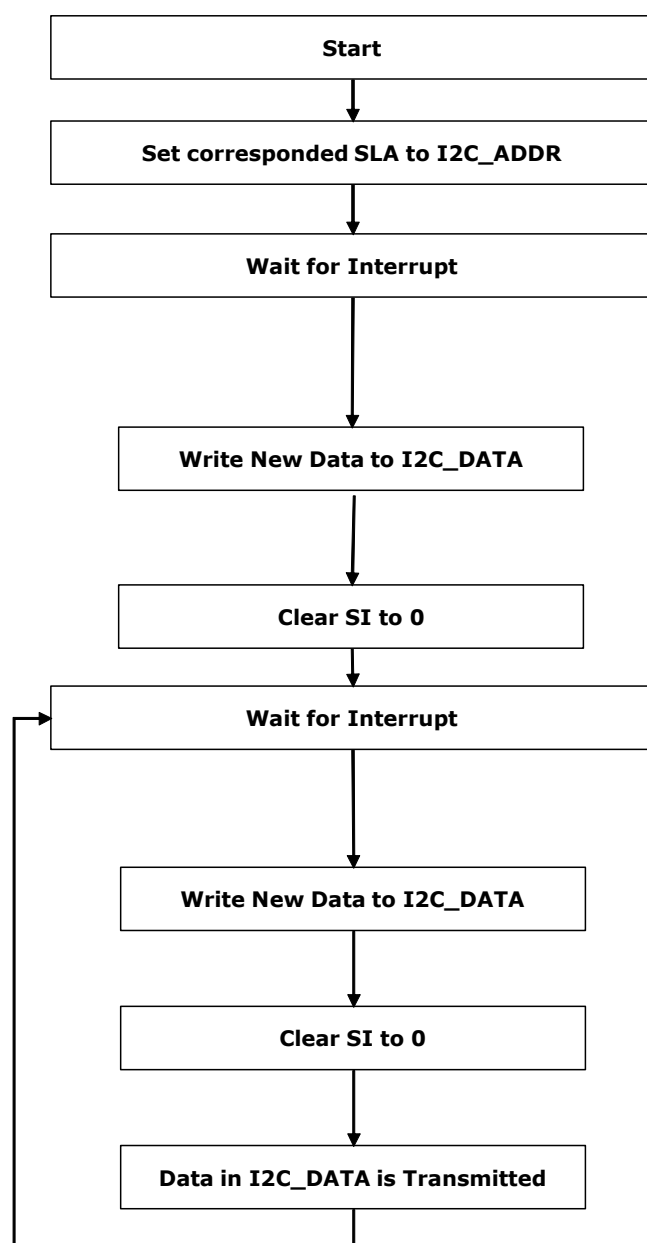


图 23-12 从机发送模式的操作流程

## 23.7.4 从机接收模式

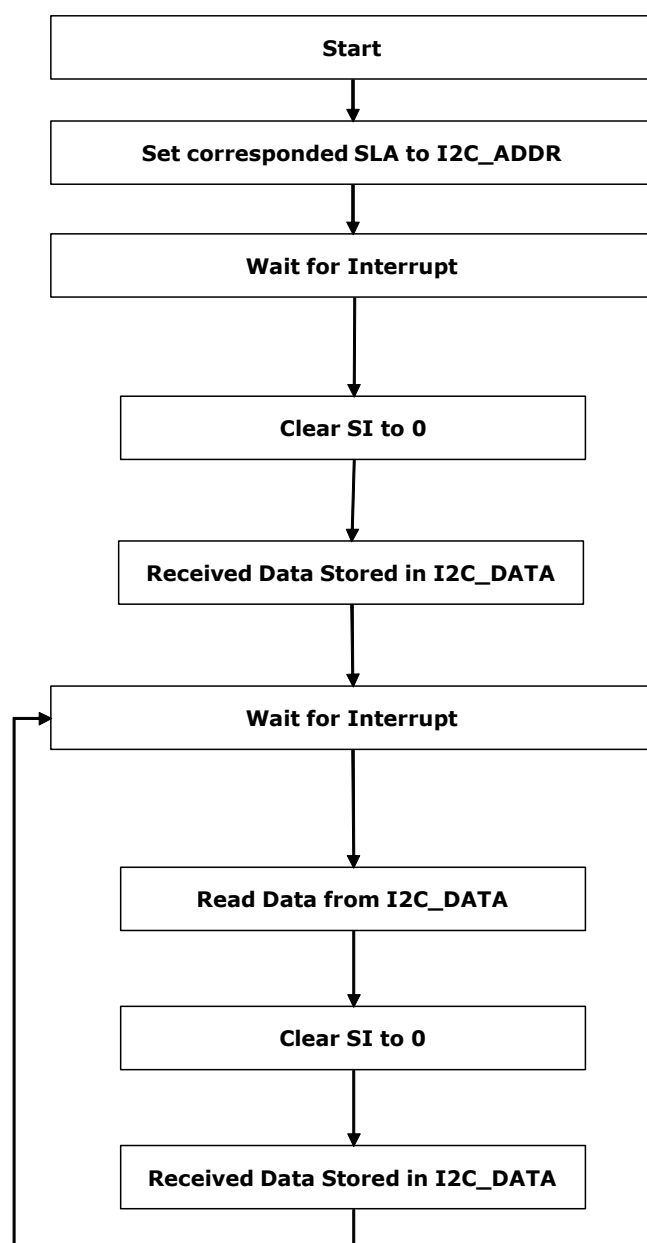


图 23-13 从机接收模式的操作流程

## 第24章 中断控制器

### 24.1 简介

任何中断都能将系统从挂起（IDLE）状态唤醒，其中的少部分可以将系统从浅睡眠（SLEEP）中唤醒，其中更少的部分可以将系统从深睡眠（DEEPSLEEP）中唤醒。

### 24.2 特点

### 24.3 中断源

表 24-1 中断源

序列	向量地址	中断序号	描述	外设事件使能位	外设事件标记位	唤醒源	
						深睡眠	浅睡眠
NMI	00000008h	-14	NMI				
HardFault	0000000Ch	-13	HardFault				
SVCall	0000002Ch	-5	SVCall				
PendSV	00000038h	-2	PendSV				
SysTick	0000003Ch	-1	SysTick				
PMU	00000040h	0	IOA0~1	PMU_CONTROL.0 And PMU_IOAWKUEN.0~1	PMU_IOAINTSTS.0~1	V	V
			计量中断 0	PMU_CONTROL.0 And PMU_IOAWKUEN.2	PMU_IOAINTSTS.2	V	V
			IOA3~9	PMU_CONTROL.0 And PMU_IOAWKUEN.3~9	PMU_IOAINTSTS.3~9	V	V
			计量中断 1	PMU_CONTROL.0 And PMU_IOAWKUEN.5	PMU_IOAINTSTS.5	V	V
			IOA11~15	PMU_CONTROL.0 And PMU_IOAWKUEN.11~15	PMU_IOAINTSTS.11~15	V	V
			32K 晶振失效	PMU_CONTROL.2	PMU_STS.0	V	V
			6M 晶振失效	PMU_CONTROL.3	PMU_STS.1		

RTC	00000044h	1	非法时间格式	RTC_INTEN.1	RTC_INTSTS.1	V	V
			多秒中断	RTC_INTEN.2	RTC_INTSTS.2	V	V
			多分钟中断	RTC_INTEN.3	RTC_INTSTS.3	V	V
			多小时中断	RTC_INTEN.4	RTC_INTSTS.4	V	V
			午夜（00: 00）中断	RTC_INTEN.5	RTC_INTSTS.5	V	V
			32K 计数器中断	RTC_INTEN.6	RTC_INTSTS.6	V	V
			对 CE 寄存器的非法写入	RTC_INTEN.8	RTC_INTSTS.8		
U32K 0\1	00000048h 0000004Ch	2\3	接收到数据	U32Kx_CTRL1.0	U32Kx_STS.0	V	V
			接收奇偶校验错误	U32Kx_CTRL1.1	U32Kx_STS.1	V	V
			接收缓存溢出	U32Kx_CTRL1.2	U32Kx_STS.2	V	V
I <sup>2</sup> C	00000050h	4	I <sup>2</sup> C 串行中断	I2C_CTRL2.0	I2C_CTRL.3		
SPI	000000ACh	27	SPI 发送	SPIx_TXSTS.14	SPIx_TXSTS.15		
			SPI 接收	SPIx_RXSTS.14	SPIx_RXSTS.15		
UART0/1/ 2/4/5	00000058h / 0000005Ch / 00000060h / 00000068h / 0000006Ch	6\7\8 \10\11	接收中断	UARTx_CTRL.3	UARTx_INTSTS.1		
			发送溢出	UARTx_CTRL.4	UARTx_INTSTS.2		
			接收溢出	UARTx_CTRL.5	UARTx_INTSTS.3		
			接收奇偶校验错误	UARTx_CTRL.7	UARTx_INTSTS.4		
			发送完成	UARTx_CTRL.8	UARTx_INTSTS.5		
ISO7816	00000070h	12	接收中断	ISO7816_CFG.5	ISO7816_INFO.5		
			发送中断	ISO7816_CFG.6	ISO7816_INFO.6		
			接收溢出	ISO7816_CFG.7	ISO7816_INFO.7		
Timer0~3	00000078h ~00000084h	14\15\16\17	定时器溢出	TMRx_CTRL.3	TMRx_INT.0		
PWM0~3	00000088h ~000094h	18\19\20\21	PWM 定时器溢出	PWMx_CTL.1	PWMx_CTL.0		
			比较 0	PWMx_CCTL0.4	PWMx_CCTL0.0		
			比较 1	PWMx_CCTL1.4	PWMx_CCTL1.0		
			比较 2	PWMx_CCTL2.4	PWMx_CCTL2.0		
DMA	00000098h	22	通道 0 包结束	DMA_IE.0	DMA_STS.4		
			通道 1 包结束	DMA_IE.1	DMA_STS.5		
			通道 2 包结束	DMA_IE.2	DMA_STS.6		

			通道 3 包结束	DMA_IE.3	DMA_STS.7		
			通道 0 帧结束	DMA_IE.4	DMA_STS.8		
			通道 1 帧结束	DMA_IE.5	DMA_STS.9		
			通道 2 帧结束	DMA_IE.6	DMA_STS.10		
			通道 3 帧结束	DMA_IE.7	DMA_STS.11		
			通道 0 数据丢弃	DMA_IE.8	DMA_STS.12		
			通道 1 数据丢弃	DMA_IE.9	DMA_STS.13		
			通道 2 数据丢弃	DMA_IE.10	DMA_STS.14		
			通道 3 数据丢弃	DMA_IE.11	DMA_STS.15		
FLASH	0000009Ch	23	校验和错误	FLASH_CTRL.2	FLASH_INT.0		
ANA	000000A0h	24	手动 ADC 转换完成	ANA_INTEN.0	ANA_INTSTS.0		
			自动 ADC 转换完成	ANA_INTEN.1	ANA_INTSTS.1		
			COMP1 的上升和下降沿	ANA_INTEN.2	ANA_INTSTS.2	V	V
			COMP2 的上升和下降沿	ANA_INTEN.3	ANA_INTSTS.3	V	V
			VDDALARM 的上升和下降沿	ANA_INTEN.7	ANA_INTSTS.7	V	V
			VDCIN 的上升和下降沿	ANA_INTEN.8	ANA_INTSTS.8	V	V
			VDDLVL 的上升和下降沿	ANA_INTEN.10	ANA_INTSTS.10	√	√
			VDCINDROP=0 同时进入了浅睡眠/深睡眠状态	ANA_INTEN.11	ANA_INTSTS.11	V	V
			ANA_REGx 错误	ANA_INTEN.12	ANA_INTSTS.12	V	V
			TADC 变化超过阈值	ANA_INTEN.13	ANA_INTSTS.13	V	V



# 第25章 MISC 控制器

## 25.1 简介

MISC 控制器用于控制 V94XX(A)的某些特殊功能。V94XX(A)内置两个 MISC 控制器，一个在核心域（MISC 控制器），在浅睡眠和深睡眠模式下会掉电。从浅睡眠或深睡眠模式唤醒后，MISC 控制器的设置会被复位，在从这两个状态唤醒后，用户应手动恢复设置。另一个（MISC2 控制器）处于保留域，会在深睡眠模式下掉电。在从深睡眠模式唤醒后，MISC2 控制器的设置会被复位，用户应在深睡眠模式唤醒后手动恢复设置。

## 25.2 特点

- 每个子模块的时钟控制。
- AHBCLK 和 APBCLK 的时钟分频器。
- FLASH 编程滴答控制
- 串口 IR 占空比控制
- SRAM 奇偶校验中断控制。

## 25.3 功能框图

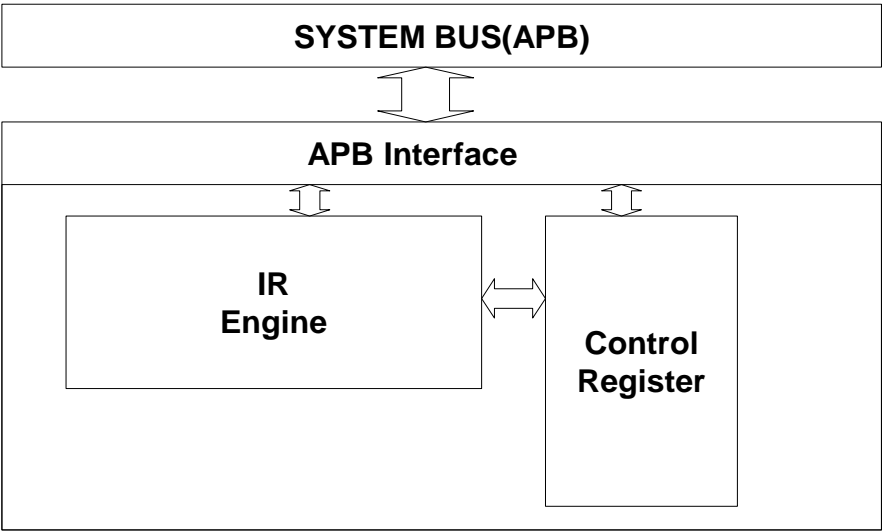


图 25-1 MISC 控制器的功能框图

## 25.4 寄存器地址

表 25-1 MISC 控制器(MISC 基地址：0x40013000)

名称	类型	地址	描述	默认值
MISC_SRAMINT	R/C	0x0000	SRAM 奇偶校验错误中断	0x00
MISC_SRAMINIT	R/W	0x0004	SRAM 初始化寄存器	0x01
MISC_PARERR	R	0x0008	SRAM 奇偶校验错误地址寄存器	0x000

MISC_IREN	R/W	0x000C	IR 控制寄存器	0x00
MISC_DUTYL	R/W	0x0010	IR 调制波低电平宽度控制寄存器	0x0000
MISC_DUTYH	R/W	0x0014	IR 调制波高电平宽度控制寄存器	0x0000
MISC_IRQLAT	R/W	0x0018	Cortex-M0 IRQ 延迟控制寄存器	0x000
MISC_HIADDR	R	0x0020	AHB 无效访问地址	--
MISC_PIADDR	R	0x0024	APB 无效访问地址	--

表 25-2 MISC2 控制器(MISC2 基地址: 0x40013E00)

名称	类型	地址	描述	默认值
MISC2_FLASHWC	R/W	0x0000	FLASH 等待周期寄存器	0x2100
MISC2_CLKSEL	R/W	0x0004	时钟选择寄存器	0x0
MISC2_CLKDIVH	R/W	0x0008	AHB 时钟分频控制寄存器	0x00
MISC2_CLKDIVP	R/W	0x000C	APB 时钟分频控制寄存器	0x01
MISC2_HCLKEN	R/W	0x0010	AHB 时钟使能控制寄存器	0x1FF
MISC2_PCLKEN	R/W	0x0014	APB 时钟使能控制寄存器	0xFFFFFFFF

## 25.5 寄存器定义

### 25.5.1 MISC\_SRAMINT 寄存器

表 25-3 MISC\_SRAMINT 寄存器

位	名称	类型	描述	默认值
31:5	-	-	保留	0
4	LOCKUP	R/C	该位表示发生了 CM0 锁定, 写 1 清除该标志	0x0
3	PIAC	R/C	该位表示进行了 APB 总线上的无效地址访问。同时, 无效访问地址会被锁存到 MISC_PIADDR。无效 APB 地址是不在 0x40010000~0x4001FFFF 区域内的地址。如果 PIACIE 为 1, 则向 CM0 发出 NMI 中断。写 1 清除该标志。	0x0
2	HIAC	R/C	该位表示进行了 AHB 总线上的无效地址访问。同时, 无效访问地址会被锁存到 MISC_HIADDR。无效 AHB 地址是 FLASH、SRAM 或 IO 区域之外的地址。如果 HIACIE 为 1, 则将向 CM0 发出 NMI 中断。写 1 清除该标志。	0x0
1	-	-	保留	0
0	PERR	R/C	该位表示在 SRAM 读取过程中发生 SRAM 奇偶校验错误。如果 PERRIE 为 1, 则将向 CM0 发出 NMI 中断。写 1 清除该标志。	0x0

### 25.5.2 MISC\_SRAMINIT 寄存器

表 25-4 MISC\_SRAMINIT 寄存器

位	名称	类型	描述	默认值
31:8	-	-	保留	0
7	LOCKIE	R/W	CM0 发生锁定时, 使能 NMI	0x0
6	PIACIE	R/W	发生 APB 无效地址访问, 使能 NMI	0x0

5	HIACIE	R/W	发生 AHB 无效地址访问，使能 NMI	0x0
4:3	-	-	保留	0
2	INIT	R/W	SRAM 初始化寄存器，将该寄存器置 1，会使用奇偶校验正确的全零值初始化 SRAM，该位将自动清零。在初始化期间，不允许访问总线上的任何主机（CPU 或 DMA），否则访问值可能是错误值	0x0
1	PERRIE	R/W	SRAM 奇偶校验错误时，使能 NMI	0x0
0	PEN	R/W	奇偶校验使能寄存器。当该位为 1 时，任何对 SRAM 的写访问，都会向奇偶校验缓存器写入校验信息。SRAM 读取时，将进行自动奇偶校验检查，并在奇偶校验失败时产生奇偶校验错误中断。该功能默认打开，因此无需在软件端进行特殊操作。但是如果 CPU 或 DMA 访问一些未初始化区域，可能会发生奇偶校验错误，因为非初始化区域中不存在奇偶校验信息。此时，用户可以使用 INIT 位自动进行 SRAM 初始化。	0x1

### 25.5.3 MISC\_PARERR 寄存器

表 25-5 MISC\_PARERR 寄存器

Bit	名称	类型	描述	默认值
31:13	-	-	保留	0
12:0	PEADDR	R	奇偶校验错误地址。该寄存器存储奇偶校验错误地址的信息，当校验错误发生时，用户可以通过该寄存器定位 SRAM 校验出现错误的地址。 SRAM 校验错误地址=0x20000000+4*PEADDR	--

### 25.5.4 MISC\_IREN 寄存器

表 25-6 MISC\_IREN 寄存器

位	名称	类型	描述	默认值
31:6			保留	0
5:0	IREN	R/W	IR 使能控制寄存器。该寄存器中的每个位对应于 1 个 UARTTX 通道。当 IREN[x] 设置为 1 时，表示 UARTTX[x] 将被 IR 脉冲调制输出。	0x0

### 25.5.5 MISC\_DUTYL 寄存器

表 25-7 MISC\_DUTYL 寄存器

位	名称	类型	描述	默认值
31:16	-	-	保留	0
15:0	DUTYL	R/W	红外载波（IR）调制波低电平宽度控制寄存器。低脉冲宽度将是 (DUTYL+1) * APBCLK 周期。	0x0000

### 25.5.6 MISC\_DUTYH 寄存器

表 25-8 MISC\_DUTYH 寄存器

位	名称	类型	描述	默认值
31:16	-	-	保留	0

15:0	DUTYH	R/W	红外载波（IR）调制波高电平宽度控制寄存器。高脉冲宽度为（DUTYH+1）*APBCLK 周期。	0x0000
------	-------	-----	--	--------

## 25.5.7 MISC\_IRQLAT 寄存器

表 25-9 MISC\_IRQLAT 寄存器

位	名称	类型	描述	默认值
31:10	-	-	保留	0
9	NOHARDFFAULT	R/W	该寄存器用于禁止 CPU 产生 HardFault 中断。 0: 当发生总线错误时, 使能 HardFault 中断。 1: 当发生总线错误时, 禁止 HardFault 中断。 当 CM0 监测发生硬件故障, 将跳转到硬件故障中断服务程序。硬件故障服务程序执行完毕后, 程序将跳到正常代码并再次执行相同的指令。因此, 如果硬故障再次发生, 它将停在这个循环中, 永远不会出来。该寄存器用于 CPU 暂时禁止硬故障产生。	0x0
8	LOCKRESET	R/W	此寄存器用于控制锁定是否发出系统复位。 0: 禁止 CM0 锁定的复位产生。 1: 使能 CM0 锁定的复位产生。	0x0
7:0	IRQLAT	R/W	该寄存器用于控制 Cortex-M0 IRQ 响应延迟。	0x00

## 25.5.8 MISC\_HIADDR 寄存器

表 25-10 MISC\_HIADDR 寄存器

位	名称	类型	描述	默认值
31: 0	HIADDR	R	AHB 总线错误地址。该寄存器存储访问 AHB 总线错误地址的信息, 当错误发生时, 用户可以通过该寄存器定位 AHB 总线访问出现错误的地址。 错误地址 = HIADDR	--

## 25.5.9 MISC\_PIADDR 寄存器

表 25-11 MISC\_PIADDR 寄存器

位	名称	类型	描述	默认值
31: 0	PIADDR	R	APB 总线错误地址。该寄存器存储访问 APB 总线错误地址的信息, 当错误发生时, 用户可以通过该寄存器定位 APB 总线访问出现错误的地址。 错误地址 = 0x40000000 + PIADDR	--

## 25.5.10 MISC2\_FLASHWC 寄存器

表 25-12 MISC2\_FLASHWC 寄存器

位	名称	类型	描述	默认值
31:14	-	-	保留	0
13:8	1USCYCLE	R/W	这个寄存器用于 FLASH 控制器以 AHBClock 计算 1μs 的滴答。该设置与 FLASH 的唤醒时间以及 FLASH 的擦除时间、	0x21

			FLASH 的编程时间有关。FLASH 唤醒时间=1 $\mu$ s 的滴答时间*10。必须满足 1 $\mu$ stick $\geq$ 1 $\mu$ s。  1 $\mu$ stick=(AHB 时钟周期)*(1USCYCLE+1)。  比如 AHB 时钟频率为 26.2144M，为了保证 1 $\mu$ s 的滴答时间，以及最小的唤醒时间，1USCYCLE 应该设置为 26。此时 FLASH 的唤醒时间为 27 / 26214400 * 10，大约为 10 $\mu$ s。  比如 AHB 时钟频率为 32.768k，为了保证 1 $\mu$ s 的滴答时间，以及最小的唤醒时间，1USCYCLE 应该设置为 0。此时 FLASH 的唤醒时间是 1 / 32768 * 10，大约为 305 $\mu$ s。	
7:0	-	-	保留。	0

## 25.5.11 MISC2\_CLKSEL 寄存器

表 25-13 MISC2\_CLKSEL 寄存器

位	名称	类型	描述	默认值
31:3	-	-	保留	0
2:0	CLKSEL	R/W	该寄存器用于控制 AHB 时钟源。 0: RCH (6.5MRC)。 1: XOH (6.5MXTAL)。 2: PLLH。 3: RTCCLK (由 PMU_CONTROL 寄存器中的 RTCCLK_SEL 控制)。 4: PLLL。  在时钟选择到一个时钟源之前，用户应首先通过设置 PMU_CONTROL 寄存器来使能相应的模块。	0x0

## 25.5.12 MISC2\_CLKDIVH 寄存器

表 25-14 MISC2\_CLKDIVH 寄存器

位	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	CLKDIVH	R/W	该寄存器用于控制 AHB 时钟分频 0: 时钟源除以 1 1: 时钟源除以 2 2: 时钟源除以 3 ... 255: 时钟源除以 256	0x00

## 25.5.13 MISC2\_CLKDIVP 寄存器

表 25-15 MISC2\_CLKDIVP 寄存器

位	名称	类型	描述	默认值
31:8	-	-	保留	0
7:0	CLKDIVP	R/W	该寄存器用于控制 APB 时钟分频	0x01

			0: AHB 时钟除以 1 1: AHB 时钟除以 2 2: AHB 时钟除以 3 ... 255: AHB 时钟除以 256	
--	--	--	---	--

## 25.5.14 MISC2\_HCLKEN 寄存器

表 25-16 MISC2\_HCLKEN 寄存器

位	名称	类型	描述	默认值
31:9	-	-	保留	0
8:0	HCLKEN	R/W	该寄存器用于控制每个 AHB 模块的时钟使能。对应的模块只有在其功能未被使用时才能被关闭。有关每个模块的详细信息，请参见下表。 0: 关闭; 1: 开启。	0x1FF

表 25-17 HCLK 时钟使能

位	模块	注意事项
0	--	禁止写 0
1	Arbiter 和 BusMatrix	禁止写 0
2	FLASH 控制器	禁止写 0
3	SRAM 控制器	禁止写 0
4	DMA 控制器	
5	GPIO 控制器	
6	LCD 控制器	
7	--	
8	CRYPT 控制器	

## 25.5.15 MISC2\_PCLKEN 寄存器

表 25-18 MISC2\_PCLKEN 寄存器

位	名称	类型	描述	默认值
31:0	PCLKEN	R/W	该寄存器用于控制每个 APB 模块的时钟使能。对应的模块只有在其功能未被使用时才能被关闭。有关每个模块的详细信息，请参见下表。 0: 关闭; 1: 开启。	0xFFFFFFFF

表 25-19 PCLK 时钟使能

Bit	模块	注意事项
0	AHB2APB Bridge	禁止写 0
1	DMA 控制器	
2	I <sup>2</sup> C	

3	保留	
4	UART0	
5	UART1	
6	UART2	
7	保留	
8	UART4	
9	UART5	
10	ISO7816	
11	保留	
12	Timer	
13	MISC	
14	MISC2	
15	PMU	
16	RTC	
17	ANA	
18	U32K0	
19	U32K1	
20	保留	
21	SPI	
31:22	保留	

## 第26章 CRYPT 控制器

### 26.1 简介

CRYPT 控制器用来给 ECC 加速。CRYPT 控制器的主要特征是实现 VLI 可变长度整数乘法、加法（带进位）、减法（带借位）和移位。它比纯软件实现的处理能力提高 4 倍。凡是用到 VLI 的加密算法都可以用该硬件加速处理，提升处理速度。CRYPT 控制器能直接访问 SRAM，而不会影响 M0 处理器核的访问。

### 26.2 特点

- 支持 32 位到 512 位的可变长度整数
- VLI 乘法
- VLI 带进位加法
- VLI 带借位减法
- VLI 逻辑右移一位
- 直接访问内部 SRAM

### 26.3 功能框图

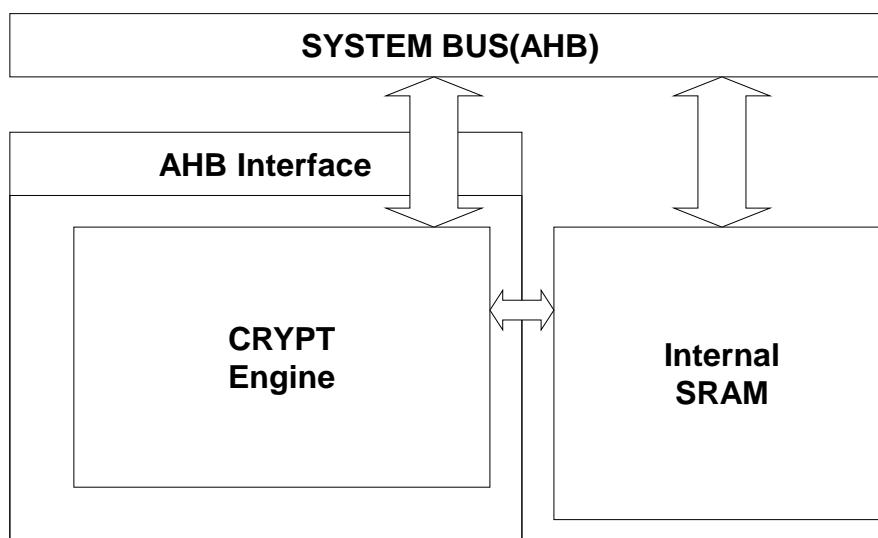


图 26-1 CRYPT 控制器的功能框图

### 26.4 寄存器地址

表 26-1 CRYPT 控制器(MISC 基地址: 0x40006000)

名称	类型	地址	描述	默认值
CRYPT_CTRL	R/W	0x0000	CRYPT 控制寄存器	0x0000
CRYPT_PTRA	R/W	0x0004	CRYPT 指针 A	0x0000



CRYPT_PTRB	R/W	0x0008	CRYPT 指针 B	0x0000
CRYPT_PTRO	R/W	0x000C	CRYPT 指针 O	0x0000
CRYPT_CARRY	R	0x0010	CRYPT 借位/进位寄存器	0x0

## 26.5 寄存器定义

### 26.5.1 CRYPT\_CTRL 寄存器

表 26-2 CRYPT\_CTRL 寄存器

位	名称	类型	描述	默认值
31:16	-	-	保留	0
15	NOSTOP	R/W	该位用来设置当 CRYPT 忙碌时，是否允许 CPU 读写 CRYPT 寄存器。 0：当 CRYPT 忙碌时禁止 CPU 访问 CRYPT 寄存器 1：当 CRYPT 忙碌时允许 CPU 访问 CRYPT 寄存器 该位仅在 CPU 要访问 CRYPT 寄存器时起作用，如果 CPU 没有访问 CRYPT 寄存器，该位是无影响的。	0x0
14:12	-	-	保留	0
11:8	LENGTH	R/W	该位用来控制当前操作数据（VLI）的长度。该长度的设置可以和 ACT 置 1 同时操作。但是当 ACT 已经是 1 时，不能改变该位的设置。 0：32 位 1：64 位 2：96 位 ..... 15：512 位	0x0
7	-	-	保留	0
6:4	MODE	R/W	该位用来控制 CRYPT 的操作模式。该位的设置与 ACT 置 1 可以同时操作。但是当 ACT 已经是 1 时，不能改变该位的设置。 0：乘法模式， $*PTRO = *PTRAx * PTRB$ 1：加法模式， $*PTRO = *PTRA + *PTRB$ 2：减法模式， $*PTRO = *PTRA - *PTRB$ 3：逻辑右移一位， $*PTRO = (*PTRA >> 1)$ 4~7：保留	0x0
3:1	-	-	保留	0
0	ACT	R/W	该位写 1 将启动 MODE 寄存器所指定的运算，在运算完成后，该位会自动清零。对该位写 0 是无效的。	0x0

### 26.5.2 CRYPT\_PTRA 寄存器

表 26-3 CRYPT\_PTRA 寄存器

位	名称	类型	描述	默认值
31:15	-	-	保留	0

14:0	PTRA	R/W	CRYPT 控制器的指针 A。PTRA 的数值是 SRAM 的一个地址，该地址中的数据用做 CRYPT 计算。由于 CRYPT 只能控制 32 位的访问，所以 PTRA[1:0]应该始终是 0。	0x0000
------	------	-----	---	--------

### 26.5.3 CRYPT\_PTRB 寄存器

表 26-4 CRYPT\_PTRB 寄存器

位	名称	类型	描述	默认值
31:15	-	-	保留	0
14:0	PTRB	R/W	CRYPT 控制器的指针 B。PTRB 的数值是 SRAM 的一个地址，该地址中的数据用做 CRYPT 计算。由于 CRYPT 只能控制 32 位的访问，所以 PTRB[1:0]应该始终是 0。	0x0000

### 26.5.4 CRYPT\_PTRO 寄存器

表 26-5 CRYPT\_PTRO 寄存器

位	名称	类型	描述	默认值
31:15	-	-	保留	0
14:0	PTRO	R/W	CRYPT 控制器的指针 O。PTRO 的数值是 SRAM 的一个地址，该地址用来存储 CRYPT 的计算结果。由于 CRYPT 只能控制 32 位的访问，所以 PTRO[1:0]应该始终是 0。	0x0000

### 26.5.5 CRYPT\_CARRY 寄存器

表 26-6 CRYPT\_CARRY 寄存器

位	名称	类型	描述	默认值
31:1	-	-	保留	0
0	CARRY	R	该位代表加法后产生了进位或者减法后产生了借位。用户在运算完成后（ACT 被清 0）直接读取该寄存器位的值。	0x0

## 26.6 使用时的注意事项

### 26.6.1 数据格式

VLI 是一个 32~512 位之间的正整数，所以所有的操作都用的是无符号数计算。下面表格中列出了 SRAM 中的数据顺序。

地址	Data
PTR+0	VLI[31:0]
PTR+4	VLI[63:32]
PTR+8	VLI[95:64]
PTR+12	VLI[127:96]
.....	.....
PTR+60	VLI[511:480]

在不同的长度模式（LENGTH 寄存器）下，只有 VLI 定义的数据长度才能作为输入数据。下表是在不同 LENGTH 设置下有效的 VLI 位数。

LENGTH	Data
0	VLI[31:0]
1	VLI[63:0]
2	VLI[95:0]
3	VLI[127:0]
.....	.....
15	VLI[511:0]

## 26.6.2 运算细节

对于每一种运算，它们都会表现出多种不同的行为，比如处理的时间周期、输出数据宽度等。下表详细列出了每一种模式的运算细节。

	MULT	ADD	SUB	RSHIFT1
Operation	*PTRO= *PTRAx*PTRB	*PTRO= *PTRA+*PTRB	*PTRO= *PTRA-*PTRB	*PTRO= (*PTRA>>1)
Outputlength	2*input length	Input length	Input length	Input length
CARRYbit	X	0: 输出没有进位 1: 输出有进位	0: *PTRA>*PTRB 1: *PTRA<*PTRB	X
LENGTH	运算周期			
0	8	6	6	6
1	17	8	8	8
2	32	12	12	10
3	53	14	14	12
4	80	18	18	14
5	113	20	20	16
6	152	24	24	18
7	197	26	26	20
8	248	30	30	22
9	305	32	32	24
10	368	36	36	26
11	437	38	38	28
12	512	42	42	30
13	593	44	44	32
14	680	48	48	34
15	773	50	50	36

默认情况下，运算过程中，任何 CPU 对 CRYPT 寄存器的访问都会挂起 CPU，直到运算结束。运算过程中不允许改变配置，所以运算过程中有高优先级事件发生时，都会被阻塞，直到 CPU 被 CRYPT 事件释放。将 NOSTOP 位置为 1 会防止这种情况的发生，但在进行 CRYPT 运算过程中需要确保 CRYPT 配置参数不变。

因为 CRYPT 只支持 SRAM 的数据操作，所以当其中一个操作数不在 SRAM 中时，运算之前，用户需要先将该数据搬移到 SRAM 中。

# 第27章 调试特点

V94XX(A)使用的是 Cortex-M0 内核，包含硬件调试模块。

## 27.1 特点

Cortex-M0 的硬件调试包含如下几个特点：

- 程序的暂停、恢复以及单步执行
- 访问处理器核寄存器和特殊功能寄存器
- 最多支持 4 个硬件断点
- 软件断点(BKPT 指令)
- 数据监测点(最多两个)
- 动态存储器访问
- 支持串行线调试 SWD 协议

## 27.2 SWD 引脚

表 27-1 SWD 引脚的特殊功能

模式	管脚 1	管脚 2
调试模式 MODE 脚是低电平	SWCLK	SWDIO
正常模式 MODE 脚是高电平	LCDSEG54/IOA0	LCDSEG53/IOA1

## 第28章 Cortex-M0 Core 简要描述

ARM Cortex-M0 处理器主要是满足超低功耗 MCU 和混合信号器件而设计的。它采用三级流水线冯·诺伊曼结构 (VonNeumann architecture)。通过简单、功能强大的指令集以及全面优化的设计 (提供包括一个单周期乘法器在内的高端处理硬件)，ARM Cortex-M0 处理器可实现极高的能效。ARM Cortex-M0 处理器采用 ARMv6-M 结构，基于 16 位的 Thumb 指令集，并包含 Thumb-2 技术。

### 28.1 CMSIS 函数说明

表 28-1 CMSIS 函数

函数	说明
<code>void NVIC_EnableIRQ(IRQn_Type IRQn);</code>	使能中断号为IRQn的中断，该功能不适用于系统异常的情况。
<code>Void NVIC_DisableIRQ(IRQn_Type IRQn);</code>	屏蔽中断号为IRQn的中断，该功能不适用于系统异常的情况。
<code>Void VIC_SetPendingIRQ(IRQn_Type IRQn);</code>	设置中断号为IRQn中断的挂起状态，该功能不适用于系统异常的情况。
<code>Void NVIC_ClearPendingIRQ(IRQn_Type IRQn);</code>	清除中断号为IRQn中断的挂起状态，该功能不适用于系统异常的情况。
<code>UInt32_t NVIC_GetPendingIRQ(IRQn_Type IRQn);</code>	获取中断号为IRQn中断的挂起状态
<code>void NVIC_SetPriority(IRQn_Type IRQn,uint32_t priority);</code>	设置中断号为IRQn中断的优先级
<code>uint32_t NVIC_GetPriority(IRQn_TypeIRQn);</code>	获取中断号为IRQn的中断优先级。
<code>Void __enable_irq(void);</code>	使能全局中断。
<code>Void __disable_irq(void);</code>	屏蔽全局中断(除了hardfault和NMI)。
<code>UInt32_t SysTick_Config(uint32_t ticks);</code>	初始化并启动SysTick计数器和它的中断。
<code>Void NVIC_SystemReset(void);</code>	M0内核软复位
<code>SCB-&gt;SCR  = SCB_SCR_SLEEPDEEP_Msk;</code> <code>void __WFI(void);</code>	等待中断 (进入浅睡眠模式)
<code>SCB-&gt;SCR &amp;=</code> <code>(uint32_t)~((uint32_t)SCB_SCR_SLEEPDEEP_Msk);</code> <code>void __WFI(void);</code>	等待中断 (进入IDLE模式)

## 第29章 电能计量 EM

V94XX(A)内置一个电能计量模块，支持多种模式的全波和基波能量计量，并支持监测多种电网事件。同时，波形数据可通过Px以SPI协议传输出去，或者通过波形缓存存在本地。

注意：该电能计量模块的所有寄存器读写访问，都需要调用公司提供的函数库实现。

- 电能计量特点：
  - 3路独立的过采样 $\Sigma/\Delta$  ADC：1路电压；1路A通道电流；另外1路B通道电流
  - 计量精度高：
    - ✧ 满足 GB/T 17215.321-2008/IEC 62053-21:2003、GB/T 17215.322-2008/IEC 62053-22:2003 和 GB/T 17215.323-2008/IEC 62053-23:2003 的要求
    - ✧ 5000:1 动态范围内，全波/基波有功能量计量误差小于 0.1%
    - ✧ 5000:1 动态范围内，全波/基波无功能量计量误差小于 0.2%
    - ✧ 5000:1 动态范围内，全波/基波电压/电流有效值误差小于 0.5%
  - 提供各种测量数据：
    - ✧ 电压/电流信号直流分量
    - ✧ 全波/基波电压/电流有效值瞬时值/平均值
    - ✧ 全波/基波有功/无功功率瞬时值/平均值
    - ✧ 全波 10 或 12 周波有效值
    - ✧ 全波视在功率瞬时值/平均值
    - ✧ 正向/反向能量，有功/无功/视在功率/电流有效值/常数值/基波通道可选
    - ✧ 频率和相位
  - 支持直流信号测量
  - 支持软件校表
  - 支持小信号加速校表
- 支持过流、过压、欠流、欠压、电压骤升、电压骤降检测
- 支持波形缓存和波形输出
- 电流输入：支持锰铜、CT、霍尔及 TMR
- 该模块工作的时钟频率为 6.5536 MHz

29.1 电能计量功能框图

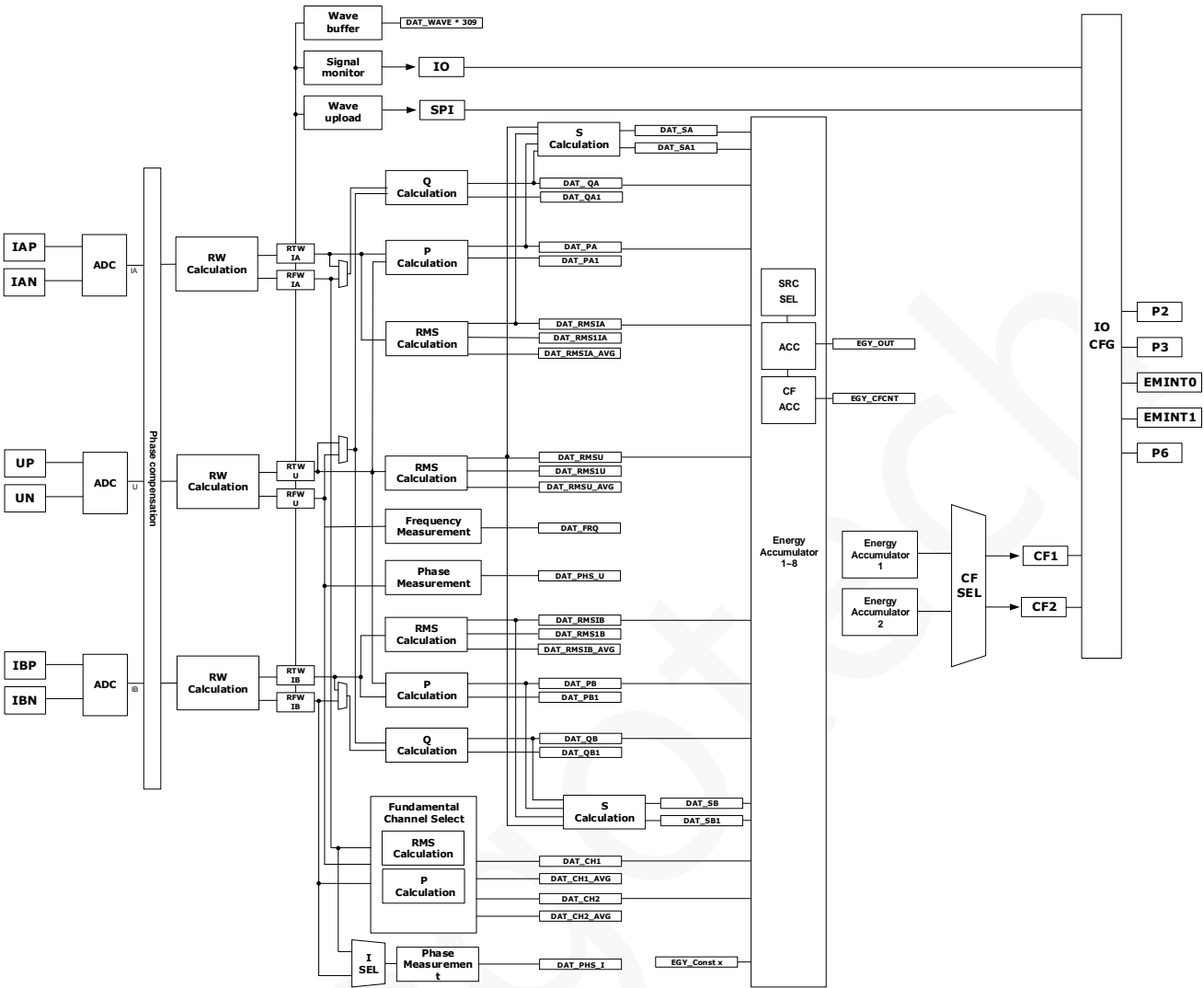


图 29-1 电能计量功能框图

29.2 寄存器列表

在发生 POR 复位、片外输入 RSTN 复位，所有寄存器被复位为默认值。以下表格中所有“默认值”均为十六进制数值。注意：该表格内的所有寄存器读写访问，都需要调用公司提供的函数库实现，寄存器读写速度与电能计量系统时钟有关。当 DSP 模式发生改变，系统时钟可能会有相应变化，需要用户注意调整寄存器读写访问速度。用户通过调用公司提供的函数库实现寄存器读写访问速度调整。读写一类寄存器时，最大速度可为系统时钟的 1/4；读写二类寄存器时，最大速度为系统时钟的 1/16。

二类寄存器地址范围为 0x11~0x38, 0x43~0x54, 0x68, 0x69。其余地址为一类寄存器地址。

29.2.1 寄存器总列表

注意：该表格内的所有寄存器读写访问，都需要调用公司提供的函数库实现。

表 29-1 寄存器总列表

寄存器	类型	地址	描述	默认值
EM_ANA_CTRL0	R/W	0x00	计量模拟控制寄存器0	0x00000000
EM_ANA_CTRL1	R/W	0x01	计量模拟控制寄存器1	0x00000000

EM_CTRL0	R/W	0x02	计量控制寄存器0	0x00000000
EM_CTRL1	R/W	0x03	计量控制寄存器1	0x00000000
EM_EGY_CTRL0	R/W	0x04	能量桶控制寄存器0	0x00000000
EM_EGY_CTRL1	R/W	0x05	能量桶控制寄存器1	0x00000000
EM_FD_CTRL	R/W	0x06	快速检测控制寄存器	0x00000000
EM_WAVE_CTRL	R/W	0x07	波形输出和缓存控制寄存器	0x00000000
EM_DAT_PA	R	0x08	A通道瞬时有功功率	--
EM_DAT_QA	R	0x09	A通道瞬时无功功率	--
EM_DAT_SA	R	0x0A	A通道瞬时视在功率	--
EM_DAT_PB	R	0x0B	B通道瞬时有功功率	--
EM_DAT_QB	R	0x0C	B通道瞬时无功功率	--
EM_DAT_SB	R	0x0D	B通道瞬时视在功率	--
EM_DAT_RMS0UA	R	0x0E	电压瞬时有有效值	--
EM_DAT_RMS0IA	R	0x0F	电流A瞬时有有效值	--
EM_DAT_RMS0IB	R	0x10	电流B瞬时有有效值	--
EM_DAT_CH1	R	0x11	基波可选通道1瞬时值	--
EM_DAT_CH2	R	0x12	基波可选通道2瞬时值	--
EM_DAT_PA1	R	0x13	A通道平均有功功率	--
EM_DAT_QA1	R	0x14	A通道平均无功功率	--
EM_DAT_SA1	R	0x15	A通道平均视在功率	--
EM_DAT_PB1	R	0x16	B通道平均有功功率	--
EM_DAT_QB1	R	0x17	B通道平均无功功率	--
EM_DAT_SB1	R	0x18	B通道平均视在功率	--
EM_DAT_RMS1U	R	0x19	电压平均有效值	--
EM_DAT_RMS1IA	R	0x1A	电流A平均有效值	--
EM_DAT_RMS1IB	R	0x1B	电流B平均有效值	--
EM_DAT_CH1_AVG	R	0x1C	基波可选通道1平均值	--
EM_DAT_CH2_AVG	R	0x1D	基波可选通道2平均值	--
EM_DAT_RMSU_AVG	R	0x1E	10或12个周波（通过电网频率选择）的电压有效值平均值	--
EM_DAT_RMSIA_AVG	R	0x1F	10或12个周波（通过电网频率选择）的电流IA有效值平均值	--
EM_DAT_RMSIB_AVG	R	0x20	10或12个周波（通过电网频率选择）的电流IB有效值平均值	--
EM_DAT_FRQ	R	0x21	电网频率值	--
EM_DAT_DCU	R	0x22	电压通道直流值	--
EM_DAT_DCIA	R	0x23	电流A通道直流值	--
EM_DAT_DCIB	R	0x24	电流B通道直流值	--
EM_CFG_CALI_PA	R/W	0x25	有功功率A比差校正	0x00000000
EM_CFG_DC_PA	R/W	0x26	有功功率A小信号校正	0x00000000



EM_CFG_CALI_QA	R/W	0x27	无功功率A比差校正	0x00000000
EM_CFG_DC_QA	R/W	0x28	无功功率A小信号校正	0x00000000
EM_CFG_CALI_PB	R/W	0x29	有功功率B比差校正	0x00000000
EM_CFG_DC_PB	R/W	0x2A	有功功率B小信号校正	0x00000000
EM_CFG_CALI_QB	R/W	0x2B	无功功率B比差校正	0x00000000
EM_CFG_DC_QB	R/W	0x2C	无功功率B小信号校正	0x00000000
EM_CFG_CALI_RMSU	R/W	0x2D	电压有效值比差校正	0x00000000
EM_CFG_RMS_DCU	R/W	0x2E	电压有效值小信号校正	0x00000000
EM_CFG_CALI_RMSIA	R/W	0x2F	电流A有效值比差校正	0x00000000
EM_CFG_RMS_DCIA	R/W	0x30	电流A有效值小信号校正	0x00000000
EM_CFG_CALI_RMSIB	R/W	0x31	电流B有效值比差校正	0x00000000
EM_CFG_RMS_DCIB	R/W	0x32	电流B有效值小信号校正	0x00000000
EM_CFG_PHC	R/W	0x33	角差校正寄存器 [10: 0]位是A通道角差校正 [26: 16]位是B通道角差校正 其范围为-766~767	0x00000000
EM_CFG_DCU	R/W	0x34	电压通道直流校正	0x00000000
EM_CFG_DCIA	R/W	0x35	电流A通道直流校正	0x00000000
EM_CFG_DCIB	R/W	0x36	电流B通道直流校正	0x00000000
EM_CFG_BPF	R/W	0x37	带通滤波器系数。该参数的设置与计量控制寄存器0（0x02，EM_CTRL0）的Bit[7:4] EM_MODE相关。 EM_MODE=0、1、2时，写入0x806764B6； EM_MODE=6、7时，写入0x80DD7A8C； EM_MODE=8时，写入0x82B465F0； 其余模式不支持频率测量，该寄存器需写0x0。	0x00000000
EM_CFG_CKSUM	R/W	0x38	校验和配置寄存器	0x00000000
EM_EGY_CRPTH	R/W	0x39	能量桶防潜阈值，当防潜能量桶累加值超过该阈值并且高速能量桶累加值未超过该阈值时，高速能量桶的累加值会被清掉。	0x00000000
EM_EGY_PWRTH	R/W	0x3A	能量桶累加阈值。由于能量桶为46Bit，实际高速能量桶累加阈值为该阈值*16384；低速能量桶累加阈值为该阈值*4。	0x00000000
EM_EGY_CONST1	R/W	0x3B	能量桶1累加常数	0x00000000
EM_EGY_OUT1L	R/W	0x3C	能量桶1累加值低位	0x00000000
EM_EGY_OUT1H	R/W	0x3D	能量桶1累加值高位 低14Bit有效	0x00000000
EM_EGY_CFCNT1	R	0x3E	能量桶1脉冲计数器	0x00000000
EM_EGY_CONST2	R/W	0x3F	能量桶2累加常数	0x00000000

EM_EGY_OUT2L	R/W	0x40	能量桶2累加值低位	0x00000000
EM_EGY_OUT2H	R/W	0x41	能量桶2累加值高位 低14Bit有效	0x00000000
EM_EGY_CFCNT2	R	0x42	能量桶2脉冲计数器	0x00000000
EM_EGY_CONST3	R/W	0x43	能量桶3累加常数	0x00000000
EM_EGY_OUT3	R/W	0x44	能量桶3累加值	0x00000000
EM_EGY_CFCNT3	R	0x45	能量桶3脉冲计数器	0x00000000
EM_EGY_CONST4	R/W	0x46	能量桶4累加常数	0x00000000
EM_EGY_OUT4	R/W	0x47	能量桶4累加值	0x00000000
EM_EGY_CFCNT4	R	0x48	能量桶4脉冲计数器	0x00000000
EM_EGY_CONST5	R/W	0x49	能量桶5累加常数	0x00000000
EM_EGY_OUT5	R/W	0x4A	能量桶5累加值	0x00000000
EM_EGY_CFCNT5	R	0x4B	能量桶5脉冲计数器	0x00000000
EM_EGY_CONST6	R/W	0x4C	能量桶6累加常数	0x00000000
EM_EGY_OUT6	R/W	0x4D	能量桶6累加值	0x00000000
EM_EGY_CFCNT6	R	0x4E	能量桶6脉冲计数器	0x00000000
EM_EGY_CONST7	R/W	0x4F	能量桶7累加常数	0x00000000
EM_EGY_OUT7	R/W	0x50	能量桶7累加值	0x00000000
EM_EGY_CFCNT7	R	0x51	能量桶7脉冲计数器	0x00000000
EM_EGY_CONST8	R/W	0x52	能量桶8累加常数	0x00000000
EM_EGY_OUT8	R/W	0x53	能量桶8累加值	0x00000000
EM_EGY_CFCNT8	R	0x54	能量桶8脉冲计数器	0x00000000
EM_OV_THL	R/W	0x55	潜动判断下限阈值	0x00000000
EM_OV_THH	R/W	0x56	潜动判断上限阈值	0x00000000
EM_SWELL_THL	R/W	0x57	电压骤升下限阈值。	0x00000000
EM_SWELL_THH	R/W	0x58	电压骤升上限阈值。	0x00000000
EM_DIP_THL	R/W	0x59	电压骤降下限阈值	0x00000000
EM_DIP_THH	R/W	0x5A	电压骤降上限阈值	0x00000000
EM_FD_OVTH	R/W	0x5B	快速检测过压阈值 位宽为30Bit	0x00000000
EM_FD_LVTH	R/W	0x5C	快速检测欠压阈值 位宽为30Bit	0x00000000
EM_FD_IA_OCTH	R/W	0x5D	快速检测电流A通道过流阈值 位宽为30Bit	0x00000000
EM_FD_IA_LCTH	R/W	0x5E	快速检测电流A通道欠流阈值 位宽为30Bit	0x00000000
EM_FD_IB_OCTH	R/W	0x5F	快速检测电流B通道过流阈值 位宽为30Bit	0x00000000
EM_FD_IB_LCTH	R/W	0x60	快速检测电流B通道欠流阈值	0x00000000

			位宽为30Bit	
EM_PHS_STT	R/W	0x61	相位测量控制位，写操作使能一次相位测量。	--
EM_PHS_U	R	0x62	电压相位值	1
EM_PHS_UN	R	0x63	电压过零点之前波形数据值	0
EM_PHS_UP	R	0x64	电压过零点之后波形数据值	0x80000000
EM_PHS_I	R	0x65	电流相位值	1
EM_PHS_IN	R	0x66	电流过零点之前波形数据值	0
EM_PHS_IP	R	0x67	电流过零点之后波形数据值	0x80000000
-	R	0x68	保留	0
EM_DAT_WAVE	R	0x69	波形数据读取，可重复读取该地址，从而获得完整波形数据。若不需要读完全部数据，可通过EM_WAVE_CTRL的Bit31复位读取地址。	0
EM_DAT_SWELL_CNT	R/C	0x6A	电压骤升时间记录，半波为单位。24Bit有效。向该寄存器写任意值，可清零该计数值。	0
EM_DAT_DIP_CNT	R/C	0x6B	电压骤降时间记录，半波为单位。24Bit有效。向该寄存器写任意值，可清零该计数值。	0
EM_SYS_INTSTS	R/C	0x72	电能计量中断状态寄存器	--
EM_SYS_INTEN	R/W	0x73	电能计量中断使能寄存器	0x00000000
EM_SYS_STS	R	0x74	电能计量系统状态寄存器	--
EM_SYS_MISC	R/W	0x75	电能计量配置寄存器	--
EM_SYS_IOCFG0	R/W	0x7D	P2, P3输出配置寄存器	0
EM_SYS_IOCFG1	R/W	0x7E	EMINT0, EMINT1, P6输出配置寄存器	0

## 29.2.2 模拟控制寄存器

在发生 POR 复位、片外输入 RSTN 复位时，所有模拟控制寄存器被复位为默认值。以下表格中所有“默认值”均为十六进制数值。

模拟控制寄存器的地址范围为 0x00~0x01，均可读可写。

所有模拟控制寄存器均需要参与参数配置自检校验。

### 29.2.2.1 EM\_ANA\_CTRL0 寄存器

表 29-2 模拟控制寄存器 0 (0x00, EM\_ANA\_CTRL0)

0x00, R/W, 模拟控制寄存器 0, EM_ANA_CTRL0			
位		默认值	功能说明
Bit[31:12]	保留	0	为保证系统正常工作，必须写入默认值。
Bit11	SHORT_IA	0	电流 IA 通道 ADC 输入短路。 0: 短路 1: 正常

Bit10	SHORT_V	0	电压 U 通道 ADC 输入短路。 0: 短路 1: 正常
Bit[9:8]	IT	0	模拟全局偏置电流调整。 01: 1; 00: -33%; 11: -66%; 10: -75%。
Bit[7:0]	保留	0	为保证系统正常工作, 必须写入默认值。

### 29.2.2.2 EM\_ANA\_CTRL1 寄存器

表 29-3 模拟控制寄存器 1 (0x01, EM\_ANA\_CTRL1)

0x01, R/W, 模拟控制寄存器 1, EM_ANA_CTRL1			
位		默认值	功能说明
Bit[31:30]	ADCKSEL<1:0>	0	ADC 时钟频率选择。 00 对应 819.2KHz。 00: ×1; 01: ×2; 10: ×1/4; 11: ×1/2
Bit[29:15]	保留	0	为保证系统正常工作, 必须写入默认值。
Bit[14:12]	GIB<2:0>	0	B 路电流 ADC 增益。 000: 4; 001: 1; 010: 32; 011: 16; 100/101/110/111: 禁止 正常工作下, GIB<2:0>建议设置为 000。
Bit11	GU	0	电压 ADC 增益。 0: 8; 1: 4 正常工作下, 该 Bit 建议设置为 0。
Bit[10:8]	GIA<2: 0>	0	A 路电流 ADC 增益。 000: 32; 001: 16; 010: 4; 011: 1; 100~111: 禁止 正常工作下, GIA<2:0>建议设置为 000。
Bit[7:5]	保留	0	为保证系统正常工作, 必须写入默认值。
Bit[4:3]	RESTL<1:0>	0	EMBG 温度系数粗调。 00: 0ppm; 01: -58ppm; 10: +111ppm; 11: +56ppm
Bit[2:0]	REST<2:0>	0	EMBG 温度系数细调。 000: 0ppm; 001: +7ppm; 010: +14ppm; 011: +28ppm; 100: -32ppm; 101: -21ppm; 110: -14ppm; 111: -7ppm

## 29.2.3 系统配置和状态寄存器

发生上电 POR 复位、片外输入 RSTN 复位时，系统配置寄存器被复位为默认值。下表中的“默认值”均为十六进制数值。

### 29.2.3.1 EM\_SYS\_INTSTS 寄存器

表 29-4 系统中断状态寄存器（0x72，EM\_SYS\_INTSTS）

0x72，中断状态寄存器，EM_SYS_INTSTS				
位		R/W	默认值	功能说明
Bit[31:30]	保留	-	-	-
Bit29	EGY2OV	R/C	0	高速能量桶 2 的溢出标志位。 读 0：能量溢出未发生。 读 1：能量溢出发生。 写 0：无影响。 写 1：清该 bit。
Bit28	EGY1OV	R/C	0	高速能量桶 1 的溢出标志位。 读 0：能量溢出未发生。 读 1：能量溢出发生。 写 0：无影响。 写 1：清该 bit。
Bit27	UDIP	R/C	0	电压骤降标志位。 读 0：电压骤降未发生。 读 1：电压骤降发生。 写 0：无影响。 写 1：清该 Bit。
Bit26	USWELL	R/C	0	电压骤升标志位。 读 0：电压骤升未发生。 读 1：电压骤升发生。 写 0：无影响。 写 1：清该 Bit。
Bit25	IBLC	R/C	0	IB 欠流标志位。 读 0：IB 欠流未发生。 读 1：IB 欠流发生。 写 0：无影响。 写 1：清该 Bit。
Bit24	IBOC	R/C	0	IB 过流标志位。 读 0：IB 过流未发生。 读 1：IB 过流发生。 写 0：无影响。 写 1：清该 Bit。
Bit23	IALC	R/C	0	IA 欠流标志位。 读 0：IA 欠流未发生。 读 1：IA 欠流发生。 写 0：无影响。 写 1：清该 Bit。
Bit22	IAOC	R/C	0	IA 过流标志位。 读 0：IA 过流未发生。 读 1：IA 过流发生。 写 0：无影响。 写 1：清该 Bit。
Bit21	ULV	R/C	0	U 通道欠压标志位。 读 0：U 通道欠压未发生。

				读 1: U 通道欠压发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit20	UOV	R/C	0	U 通道过压标志位。 读 0: U 通道过压未发生。 读 1: U 通道过压发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit[19:16]	保留	-	-	-
Bit15	WAVEOUT_FINISH	R/C	0	波形输出完成标志位。 读 0: 波形输出未完成。 读 1: 波形输出已完成。 写 0: 无影响。 写 1: 清该 Bit。
Bit14	CKERR	R/C	0	校验和错误事件标志位。校验和从正确到错误, 该 Bit 会置 1。 读 0: 校验和错误事件未发生。 读 1: 校验和错误事件发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit13	HSE_FAIL	R/C	0	电能计量模块的 6.5M 高频时钟缺失事件标志位。 读 0: PM 的 6.5M 高频时钟缺失事件未发生。 读 1: PM 的 6.5M 高频时钟缺失事件发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit12	PMREF_ERR	R/C	0	电能计量模块的 Reference 错误标志位。 读 0: Reference 错误事件未发生。 读 1: Reference 错误事件发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit11	BIST_ERR	R/C	0	电能计量模块的 BIST 错误标志位。 读 0: BIST 错误事件未发生。 读 1: BIST 错误事件发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit10	ISIGN	R/C	0	电流过零点标志位, 可通过计量控制寄存器 1 (0x03, EM_CTRL1) 的 Bit20 选择过零点通道为 IA 或者 IB 通道, 可通过计量控制寄存器 1 (0x03, EM_CTRL1) 的 Bit19~Bit18 选择过零点方向。 读 0: 电流过零点未发生。 读 1: 电流过零点发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit9	USIGN	R/C	0	电压过零点标志位, 可通过计量控制寄存器 1 (0x03, EM_CTRL1) 的 Bit19~Bit18 选择过零点方向。 读 0: 电压过零点未发生。 读 1: 电压过零点发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit8	WAVE_OVERFLOW	R/C	0	波形缓存溢出标志位。 读 0: 波形缓存未溢出。 读 1: 波形缓存已溢出。 写 0: 无影响。 写 1: 清该 Bit。
Bit7	WAVE_STORE	R/C	0	波形缓存完成标志位。

				读 0: 波形缓存未完成。 读 1: 波形缓存已完成。 写 0: 无影响。 写 1: 清该 Bit。
Bit6	WAVE_UPD	R/C	0	波形数据更新标志位。 读 0: 未更新。 读 1: 已更新。 写 0: 无影响。 写 1: 清该 Bit。
Bit5	CURRMS_UPD	R/C	0	瞬时有有效值数据更新标志位。 读 0: 未更新。 读 1: 已更新。 写 0: 无影响。 写 1: 清该 Bit。
Bit4	AVGRMS_UPD	R/C	0	平均有效值数据更新标志位。 读 0: 未更新。 读 1: 已更新。 写 0: 无影响。 写 1: 清该 Bit。
Bit3	CURPOWER_UPD	R/C	0	瞬时功率数据更新标志位。 读 0: 未更新。 读 1: 已更新。 写 0: 无影响。 写 1: 清该 Bit。
Bit2	AVGPOWER_UPD	R/C	0	平均功率数据更新标志位。 读 0: 未更新。 读 1: 已更新。 写 0: 无影响。 写 1: 清该 Bit。
Bit1	INTVDDPDN	R/C	0	电能计量模块的掉电事件发生标志位。当电源输入 (VDD) 低于 2.65V ( $\pm 6\%$ ), 发生掉电。 读 0: 掉电未发生。 读 1: 掉电发生。 写 0: 无影响。 写 1: 清该 Bit。
Bit0	INTUPHSDONE	R/C	0	相位测量完成标志位。这个 Bit 在相位测量开始时, 会自动清零。 读 0: 相位测量未完成。 读 1: 相位测量已完成。 写 0: 无影响。 写 1: 清该 Bit。

### 29.2.3.2 EM\_SYS\_INTEN 寄存器

表 29-5 电能计量系统中断使能寄存器描述 (0x73, EM\_SYS\_INTEN)

0x73, 电能计量系统中断使能寄存器, EM_SYS_INTEN				
位		R/W	默认值	功能说明
Bit[31:30]	保留		0	为保证系统正常工作, 必须写入默认值。
Bit29	EGY2OV	R/W	0	高速能量桶 2 的溢出中断使能。 0: 关闭中断; 1: 使能中断
Bit28	EGY1OV	R/W	0	高速能量桶 1 的溢出中断使能。 0: 关闭中断; 1: 使能中断



Bit27	UDIP	R/W	0	电压骤降中断。 0: 关闭中断; 1: 使能中断
Bit26	USWELL	R/W	0	电压骤升中断。 0: 关闭中断; 1: 使能中断
Bit25	IBLC	R/W	0	IB 欠流中断。 0: 关闭中断; 1: 使能中断
Bit24	IBOC	R/W	0	IB 过流中断。 0: 关闭中断; 1: 使能中断
Bit23	IALC	R/W	0	IA 欠流中断。 0: 关闭中断; 1: 使能中断
Bit22	IAOC	R/W	0	IA 过流中断。 0: 关闭中断; 1: 使能中断
Bit21	ULV	R/W	0	U 通道欠压中断。 0: 关闭中断; 1: 使能中断
Bit20	UOV	R/W	0	U 通道过压中断。 0: 关闭中断; 1: 使能中断
Bit[19:16]	保留		0	为保证系统正常工作, 必须写入默认值。
Bit15	WAVEOUT_FINISH	R/W	0	波形输出完成中断。 0: 关闭中断; 1: 使能中断
Bit14	CKERR	R/W	0	校验和错误中断。 0: 关闭中断; 1: 使能中断
Bit13	HSE_FAIL	R/W	0	电能计量模块的 6.5M 高频时钟缺失中断。 0: 关闭中断; 1: 使能中断
Bit12	PMREF_ERR	R/W	0	电能计量模块的 Reference 错误中断。 0: 关闭中断; 1: 使能中断
Bit11	BIST_ERR	R/W	0	BIST 错误中断。 0: 关闭中断; 1: 使能中断
Bit10	ISIGN	R/W	0	电流过零点中断。 0: 关闭中断; 1: 使能中断
Bit9	USIGN	R/W	0	电压过零点中断。 0: 关闭中断; 1: 使能中断
Bit8	WAVE_OVERFLOW	R/W	0	波形缓存溢出中断。 0: 关闭中断; 1: 使能中断
Bit7	WAVE_STORE	R/W	0	波形缓存完成中断。 0: 关闭中断; 1: 使能中断
Bit6	WAVE_UPD	R/W	0	波形数据更新中断。 0: 关闭中断; 1: 使能中断



Bit5	CURRMS_UPD	R/W	0	瞬时有效值数据更新中断。 0: 关闭中断; 1: 使能中断
Bit4	AVGRMS_UPD	R/W	0	平均有效值数据更新中断。 0: 关闭中断; 1: 使能中断
Bit3	CURPOWER_UPD	R/W	0	瞬时功率数据更新中断。 0: 关闭中断; 1: 使能中断
Bit2	AVGPOWER_UPD	R/W	0	平均功率数据更新中断。 0: 关闭中断; 1: 使能中断
Bit1	INTPDN	R/W	0	电能计量模块掉电事件中断。 0: 关闭中断; 1: 使能中断
Bit0	INTUPHSDONE	R/W	0	相位测量完成中断。 0: 关闭中断; 1: 使能中断

### 29.2.3.3 EM\_SYS\_STS 寄存器

表 29-6 系统状态寄存器描述 (0x74, EM\_SYS\_STS)

0x74, 系统状态寄存器, EM_SYS_STS				
位		R/W	默认值	功能说明
Bit31	保留	-	-	-
Bit30	UDIP	R	0	电压骤降状态位。 1: 电压处于骤降状态。 0: 电压未处于骤降状态。
Bit29	USWELL	R	0	电压骤升状态位。 1: 电压处于骤升状态。 0: 电压未处于骤升状态。
Bit28	IBLC	R	0	IB 欠流中断状态位 1: IB 处于欠流状态。 0: IB 未处于欠流状态。
Bit27	IBOC	R	0	IB 过流中断状态位。 1: IB 处于过流状态。 0: IB 未处于过流状态。
Bit26	IALC	R	0	IA 欠流中断状态位。 1: IA 处于欠流状态。 0: IA 未处于欠流状态。
Bit25	IAOC	R	0	IA 过流中断状态位。 1: IA 处于过流状态。 0: IA 未处于过流状态。

Bit24	ULV	R	0	U 通道欠压中断状态位。 1: U 通道处于欠压状态。 0: U 通道未处于欠压状态。
Bit23	UOV	R	0	U 过压状态位。 1: U 通道处于过压状态。 0: U 通道未处于过压状态。
Bit[22:20]	RST_SOURCE	R	-	电能计量模块的当前复位源状态位。 1: 上电复位。 2: 外部复位。 其它: 保留。
Bit19	CRP_OUT2	R	0	能量桶 2 潜动状态位。 0: 启动状态; 1: 潜动状态
Bit18	CRP_OUT1	R	0	能量桶 1 潜动状态位。 0: 启动状态; 1: 潜动状态
Bit17	SBCREEP	R	-	B 通道视在功率潜动状态位。 0: 启动状态; 1: 潜动状态
Bit16	QBCREEP	R	-	B 通道无功功率潜动状态位。 0: 启动状态; 1: 潜动状态
Bit15	PBCREEP	R	-	B 通道有功功率潜动状态位。 0: 启动状态; 1: 潜动状态
Bit14	SACREEP	R	-	A 通道视在功率潜动状态位。 0: 启动状态; 1: 潜动状态
Bit13	QACREEP	R	-	A 通道无功功率潜动状态位。 0: 启动状态; 1: 潜动状态
Bit12	PACREEP	R	-	A 通道有功功率潜动状态位。 0: 启动状态; 1: 潜动状态
Bit11	QBSIGN	R	0	B 通道无功功率符号位。在潜动状态时, 不刷新该标志位。 0: 正; 1: 负

Bit10	PBSIGN	R	0	B 通道有功功率符号位。在潜动状态时，不刷新该标志位。 0：正； 1：负
Bit9	QASIGN	R	0	A 通道无功功率符号位。在潜动状态时，不刷新该标志位。 0：正； 1：负
Bit8	PASIGN	R	0	A 通道有功功率符号位。在潜动状态时，不刷新该标志位。 0：正； 1：负
Bit7	保留			
Bit6	HSEFAIL	R	0	当前电能计量模块的 6.5M 高速时钟状态位。 0：当前时钟正常； 1：当前时钟异常
Bit5	BIST_ERR	R	0	电能计量模块的 SRAM BIST 状态位。 0：SRAM BIST 正常； 1：SRAM BIST 错误
Bit4	RAMINITIAL	R	0	电能计量模块的 RAM 初始化完成状态位。 0：未完成； 1：已完成
Bit3	PHSDONE	R	0	电能计量模块的相位测量是否完成状态位。 0：未完成； 1：已完成
Bit2	VDDPDN	R	0	电能计量模块的当前电压状态位。当电源输入（VDD）低于 2.65V（±6%），发生掉电。 0：VDD ≥ 2.65V（±6%）； 1：VDD < 2.65V（±6%）
Bit1	PMREFLK	R	0	电能计量模块的 reference 漏电状态位。 0：reference 电路正常 1：reference 电路下降大于 2.5%
Bit0	CHECKSUM	R	1	电能计量模块的校验和状态位。参与校验和计算的数据有： 地址 0x0~0x7，0x25~0x3A，0x55~0x60。 0：校验和正确 1：校验和不正确

### 29.2.3.4 EM\_SYS\_MISC 寄存器

表 29-7 系统配置寄存器（0x75，EM\_SYS\_MISC）

0x75，系统配置寄存器，EM_SYS_MISC			
位	R/W	默认值	功能说明

Bit[31:25]	保留	-	0	为保证系统正常工作，必须写入默认值。
Bit[24:16]	WAVE_STORE_CNT	R	0	波形已缓存深度记录。
Bit[15:6]	保留	-	0	为保证系统正常工作，必须写入默认值。
Bit5	BIST_EM_EGY_EN	R/W	0	<p>RAM 自检错误时强制关闭能量桶和 CF 输出。</p> <p>0: 使能; 1: 关闭</p> <p>如果使能该功能, 当发生 RAM 自检错误时会强制关闭能量桶和 CF 输出。</p> <p>能量桶累加器和 CF 计数器控制位 (EM_CTRL1 寄存器的 Bit[15]和 Bit[7:6]) 不会发生变化。</p> <p>虽然寄存器值没有发生改变, 但是状态恢复正常后, 能量桶和 CF 输出功能不会自动恢复, 需要用户手动开启。</p>
Bit4	CK_EM_EGY_EN	R/W	0	<p>校验和错误时强制关闭能量桶和 CF 输出。</p> <p>0: 使能; 1: 关闭</p> <p>如果使能该功能, 当发生校验和错误时会强制关闭能量桶和 CF 输出。</p> <p>能量桶累加器和 CF 计数器控制位 (EM_CTRL1 寄存器的 Bit[15]和 Bit[7:6]) 不会发生变化。</p> <p>虽然寄存器值没有发生改变, 但是状态恢复正常后, 能量桶和 CF 输出功能不会自动恢复, 需要用户手动开启。</p>
Bit3	PD_EM_EGY_EN	R/W	0	<p>掉电时强制关闭能量桶和 CF 输出。</p> <p>0: 使能; 1: 关闭</p> <p>如果使能该功能, 当发生掉电时会强制关闭能量桶和 CF 输出。</p> <p>能量桶累加器和 CF 计数器控制位 (EM_CTRL1 寄存器的 Bit[15]和 Bit[7:6]) 不会发生变化。</p> <p>虽然寄存器值没有发生改变, 但是状态恢复正常后, 能量桶和 CF 输出功能不会自动恢复, 需要用户手动开启。</p>
Bit2	INTPOL	R/W	0	<p>中断引脚输出反向。</p> <p>0: 中断引脚高电平有效, 默认低电平。 1: 中断引脚低电平有效, 默认高电平。</p>
Bit[1:0]	保留		0	为保证系统正常工作，必须写入默认值。

### 29.2.3.5 EM\_SYS\_IOCFCGX 寄存器

表 29-8 Px 口的 IO 配置寄存器 0 (0x7D, EM\_SYS\_IOCFCG0)

0x7D, Px 口的 IO 配置寄存器 0, EM_SYS_IOCFCG0				
位		R/W	默认值	功能说明
Bit[31:24]	P3CFG	R/W	0	配置同 P2CFG
Bit[23:16]	P2CFG	R/W	0	<p>Bit7~Bit6:</p> <p>00: 如下表所示, Bit5~Bit0 可配置为不同组合中断</p>

				10: CF1 输出, Bit5~Bit0 可配置任意值 01: CF2 输出, Bit5~Bit0 可配置任意值 11: 保留
Bit[15:8]	保留		0	为保证系统正常工作, 必须写入默认值。
Bit[7:0]	保留		0	为保证系统正常工作, 必须写入默认值。

表 29-9 PxCFG Bit5~Bit0 说明

Bit5~Bit3	Bit2~Bit0	功能
0	0	高阻态
0	1	电流过零点中断
0	2	电压过零点中断
0	3	电流过零点输出方波
0	4	电压过零点输出方波
0	5	高速能量桶 1 溢出中断
0	6	高速能量桶 2 溢出中断
0	7	1 类中断
1	0	波形刷新中断
1	1	瞬时有效值刷新中断
1	2	平均有效值刷新中断
1	3	瞬时功率刷新中断
1	4	平均功率刷新中断
1	5	波形缓存完成中断
1	6	波形缓存地址溢出中断
1	7	波形输出完成中断
2	0	电流 IB 通道欠流中断
2	1	电流 IB 通道过流中断
2	2	电流 IA 通道欠流中断
2	3	电流 IA 通道过流中断
2	4	电压通道欠压中断
2	5	电压通道过压中断
2	6	电压骤降中断
2	7	电压骤升中断
3	0	电能计量模块 Reference 错误中断
3	1	电能计量模块的 6.5M 高频时钟缺失中断
3	2	保留
3	3	保留
3	4	掉电中断
3	5	参数自检错误中断
3	6	相位测试完成中断

3	7	RAM 自检错误中断
4	0	1 类中断
4	1	2 类中断
4	2	3 类中断
4	其他	4 类中断
5	0	3 类中断
5	1	1 类中断和 2 类中断
5	2	1 类中断和 3 类中断
5	3	1 类中断和 4 类中断
5	4	2 类中断和 3 类中断
5	5	2 类中断和 4 类中断
5	6	3 类中断和 4 类中断
5	7	所有中断
6	0	1 类中断、2 类中断和 3 类中断
6	1	1 类中断、2 类中断和 4 类中断
6	2	1 类中断、3 类中断和 4 类中断
6	3	2 类中断、3 类中断和 4 类中断
6	其他	所有中断
7	1	波形输出 SPI 片选 SPI_CSN
7	2	波形输出 SPI 时钟 SPI_CLK
7	4	波形输出 SPI 数据 SPI_MOSI
7	其他	禁止输出

其中，IO 口在不配置时（即全 0），输出高阻。

1 类中断：电流过零点中断、电压过零点中断、高速能量桶 1/2 溢出中断。

2 类中断：波形刷新中断、瞬时有效值刷新中断、平均有效值刷新中断、瞬时功率值刷新中断、平均功率值刷新中断、波形缓存完成中断、波形缓存溢出中断、波形输出完成中断。

3 类中断：电流 IB 通道欠流中断、电流 IB 通道过流中断、电流 IA 通道欠流中断、电流 IA 通道过流中断、电压通道欠压中断、电压通道过压中断、电压骤降中断、电压骤升中断。

4 类中断：参数自检错误中断、相位测量完成中断、掉电中断、Reference 错误中断、电能计量模块的 6.5M 高频时钟缺失中断、RAM 自检错误中断。

表 29-10 Px 口的 IO 配置寄存器 1 (0x7E, EM\_SYS\_IOCFCG1)

0x7E, Px 口的 IO 配置寄存器 1, EM_SYS_IOCFCG1				
位		R/W	默认值	功能说明
Bit[31:24]	保留		0	为保证系统正常工作，必须写入默认值。
Bit[23:16]	P6CFG	R/W	0	配置同 P2CFG
Bit[15:8]	EMINT1CFG	R/W	0	配置同 P2CFG
Bit[7:0]	EMINT0CFG	R/W	0	配置同 P2CFG

## 29.2.4 计量控制寄存器

发生上电复位（POR）、片外输入 RSTN 复位时，所有计量控制寄存器均会被复位为默认值。下表中的“默认值”均为十六进制数值。

所有的计量控制寄存器均需要参与参数配置自检校验。

### 29.2.4.1 EM\_CTRL0 寄存器

表 29-11 计量控制寄存器 0 (0x02, EM\_CTRL0)

0x02, 计量控制寄存器 0, EM_CTRL0				
位		R/W	默认值	功能说明
Bit31	CURDAT_RATE	R/W	0	瞬时功率值、有效值刷新时间加倍（具体时间与系统时钟的准确度有关）。 0: 瞬时功率值刷新时间 20ms，瞬时有效值刷新时间 10ms。 1: 瞬时功率值刷新时间 40ms，瞬时有效值刷新时间 20ms。
Bit30	保留		0	为保证系统正常工作，必须写入默认值。
Bit[29:28]	FRQ_SEL	R/W	0	EM_DAT_FRQ 寄存器值来源（具体时间与系统时钟的准确度有关）。 00: 16 个周波的电网频率测试值的累加值(默认是 320ms 刷新)。 01: 1 个周波的电网频率测试值(默认是 20ms 刷新)。 10: 64 个周波的电网频率测试值的累加值(默认是 1280ms 刷新)。 11: 保留。
Bit27	DC_METER_MODE	R/W	0	直流计量模式开关。 0: 关闭; 1: 打开
Bit[26:25]	保留		0	为保证系统正常工作，必须写入默认值。
Bit24	S_MODE	R/W	0	视在功率计算源选择。 0: 通过有效值计算; 1: 通过功率值计算
Bit[23:20]	CFG_CHANNEL	R/W	0	可配置基波通道选择。 0: 通道 1, 基波有功 A; 通道 2, 基波有功 B。 1: 通道 1, 基波有功 A; 通道 2, 基波电压有效值。 2: 通道 1, 基波有功 A; 通道 2, 基波电流 A 有效值。 3: 通道 1, 基波有功 A; 通道 2, 基波电流 B 有效值。 4: 通道 1, 基波有功 B; 通道 2, 基波电压有效值。 5: 通道 1, 基波有功 B; 通道 2, 基波电流 A 有效值。 6: 通道 1, 基波有功 B; 通道 2, 基波电流 B 有效值。 7: 通道 1, 基波电压有效值; 通道 2, 基波电流 A 有效值。 8: 通道 1, 基波电压有效值; 通道 2, 基波电流 B 有效值。 9: 通道 1, 基波电流 A 有效值; 通道 2, 基波电流 B 有效值。 10~15: 与配置 0 保持一致。
Bit19	QB_MODE	R/W	0	无功功率 B 模式选择。 0: 全波无功; 1: 基波无功
Bit18	QA_MODE	R/W	0	无功功率 A 模式选择。 0: 全波无功; 1: 基波无功



Bit17	PQ_HPFSEL	R/W	0	计算全波功率的数据是否经过高通滤波器。 0: 经过; 1: 不经过
Bit16	FUND_HPFSEL	R/W	0	计算基波的数据是否经过高通滤波器。 0: 经过; 1: 不经过
Bit15	RMSU_HPFSEL	R/W	0	计算全波电压有效值的数据是否经过高通滤波器。 0: 经过; 1: 不经过
Bit14	RMSIA_HPFSEL	R/W	0	计算全波电流 IA 有效值的数据是否经过高通滤波器。 0: 经过; 1: 不经过
Bit13	RMSIB_HPFSEL	R/W	0	计算全波电流 IB 有效值的数据是否经过高通滤波器。 0: 经过; 1: 不经过
Bit12	RESPONSE_TIME	R/W	0	计量数据响应时间。 0: 正常; 1: 2 倍速
Bit[11:10]	AVGRMS_RATE	R/W	0	平均有效值刷新时间（具体时间与系统时钟的准确度有关）。 00: 40ms; 01: 80ms; 10: 320ms; 11: 640ms
Bit[9:8]	AVGPQ_RATE	R/W	0	平均功率刷新时间（具体时间与系统时钟的准确度有关）。 00: 80ms; 01: 160ms; 10: 320ms; 11: 640ms
Bit[7:4]	EM_MODE	R/W	0	DSP 工作模式。 0: 系统时钟 6.5536MHz, DSP 每周波采样点数 128。 1: 系统时钟 6.5536MHz, DSP 每周波采样点数 64。 2: 系统时钟 6.5536MHz, DSP 每周波采样点数 32。 3, 4, 5: 保留 6: 系统时钟 3.2768MHz, DSP 每周波采样点数 64。 7: 系统时钟 3.2768MHz, DSP 每周波采样点数 32。 8: 系统时钟 819.2KHz, DSP 每周波采样点数 32。（此时只支持最多两路瞬时电流有效值计算） 9: 系统时钟 409.6KHz, DSP 每周波采样点数 16。（此时只支持最多两路瞬时电流有效值计算） 10~15: 与模式 0 保持一致。
Bit3	保留		0	为保证系统正常工作，必须写入默认值。
Bit2	ADCUEN	R/W	0	电压通道开关（包括 ADC 和 DSP）。
Bit1	ADCIBEN	R/W	0	电流 B 通道开关（包括 ADC 和 DSP）。
Bit0	ADCIAEN	R/W	0	电流 A 通道开关（包括 ADC 和 DSP）。

## 29.2.4.2 EM\_CTRL1 寄存器

表 29-12 计量控制寄存器 1（0x03, EM\_CTRL1）

0x03, R/W, 计量控制寄存器 1, EM_CTRL1				
位		R/W	默认值	功能说明
Bit[31:24]	保留		0	为保证系统正常工作，必须写入默认值。



Bit23	EM_EGY_CLK_SEL	R/W	0	能量桶时钟选择位。时钟切换需要稳定时间，约为 107μs，请在稳定前关闭 CF。 0: 204.8KHz（计量模块系统时钟内部分频）； 1: 32768Hz（X32KIN 管脚输入）
Bit22	LCF_ACC	R/W	0	累加周期与系统时钟的准确度有关。 0: 关闭。 当 EM_CTRL0 的 Bit31 配 0，能量桶 3, 4, 5, 6, 7, 8 累加周期为 20ms； 当 EM_CTRL0 的 Bit31 配 1，能量桶 3, 4, 5, 6, 7, 8 累加周期为 40ms。 1: 打开。 当 EM_CTRL0 的 Bit31 配 0，能量桶 3, 4, 5 累加周期为 10ms，能量桶 6, 7, 8 不累加； 当 EM_CTRL0 的 Bit31 配 1，能量桶 3, 4, 5 累加周期为 20ms，能量桶 6, 7, 8 不累加。
Bit21	PGA_U	R/W	0	电压通道数字 PGA： 0: X1； 1: X4
Bit20	PHSI_SEL	R/W	0	电流过零点输入源选择： 0: 电流 IA 通道； 1: 电流 IB 通道
Bit[19:18]	SIGN_SEL	R/W	0	过零点事件检测方式选择。 0: 负向过零点（信号从正信号变成负信号认为发生一次过零点事件）。 1: 正向过零点（信号从负信号变成正信号认为发生一次过零点事件）。 2: 正向和负向过零点。 3: 关闭过零点检测功能。
Bit[17:16]	保留		0	为保证系统正常工作，必须写入默认值。
Bit15	EM_EGY_LC_EN	R/W	0	低速能量桶和 CF 计数器开关。低速能量桶累加速度为 50Hz。 0: 关闭； 1: 开启
Bit14	CF2_INV	R/W	0	CF2 极性控制位。 0: 原始极性；1: 反向极性
Bit13	CF2_EN	R/W	0	CF2 输出使能位。 0: 关闭； 1: 开启
Bit12	CF2_SEL	R/W	0	CF2 的输入源选择 0: 来源于能量桶 1。 1: 来源于能量桶 2。
Bit11	保留		0	为保证系统正常工作，必须写入默认值。
Bit10	CF1_INV	R/W	0	CF1 极性控制位。 0: 原始极性。 1: 反向极性。
Bit9	CF1_EN	R/W	0	CF1 输出使能位。 0: 关闭；1: 开启
Bit8	CF1_SEL	R/W	0	CF1 的输入源选择 0: 来源于能量桶 1。 1: 来源于能量桶 2。
Bit7	CALCEN2	R/W	0	能量桶 2 的能量桶累加器和 CF 计数器 2 开关。能量桶 2 的累加速度 204.8KHz。 0: 关闭；1: 开启
Bit6	CALCEN1	R/W	0	能量桶 1 的能量桶累加器和 CF 计数器 1 开关。能量桶 1 的累加速度 204.8KHz。 0: 关闭； 1: 开启

Bit[5:4]	CF_PULSE	R/W	0	CF 脉冲宽度选择（具体时间与系统时钟的准确度有关）。 0: 80ms; 1: 40ms; 2: 20ms; 3: 10ms
Bit[3:2]	CF_FAST_EN	R/W	0	CF 脉冲加速产生。 0: 正常; 1: 4 倍速; 2: 8 倍速; 3: 16 倍速
Bit1	PWR_CRP_EN	R/W	0	功率潜动判断，潜动判断采用连续 3 次平均值与阈值比较，低于下限阈值，则认为此时处于潜动状态。 0: 关闭潜动判断; 1: 使能潜动判断
Bit0	EGY_CRP_EN	R/W	0	高速能量桶防潜开关。 0: 关闭潜动判断; 1: 使能潜动判断

### 29.2.4.3 EM\_EGY\_CTRL0 寄存器

表 29-13 能量桶控制寄存器 0 (0x04, EM\_EGY\_CTRL0)

0x04, R/W, 能量桶控制寄存器 0, EM_EGY_CTRL0				
位		R/W	默认值	功能说明
Bit[31:30]	INMODE4	R/W	0	能量桶 4 累加模式。 0: 功率累加。 1: 电流有效值累加。 2: 常数累加。 3: 可配置基波通道累加。
Bit29	A_SEL4	R/W	0	能量桶 4 A 通道累加开关。 0: 关闭; 1: 打开
Bit28	B_SEL4	R/W	0	能量桶 4 B 通道累加开关。 0: 关闭; 1: 打开
Bit[27:26]	TYPE_SEL4	R/W	0	能量桶 4 多通道累加模式下，累加源选择。 功率累加时： 0、3: 有功功率累加。 1: 无功功率累加。 2: 视在功率累加。 有效值累加时： 高位为 0 表示累加和，高位为 1 表示累加差。
Bit[25:24]	PROCMODE4	R/W	0	对于每一个送入能量桶 4 累加的信号的 kind 选择。 0: 能量桶只累加正数。 1: 能量桶只累加负数（此时，实际累加值为原始值转换的正数）。 2: 能量桶累加原始值。 3: 能量桶累加绝对值
Bit[23:22]	INMODE3	R/W	0	能量桶 3 累加模式。 0: 功率累加。 1: 电流有效值累加。 2: 常数累加。 3: 可配置基波通道累加。
Bit21	A_SEL3	R/W	0	能量桶 3 A 通道累加开关。 0: 关闭; 1: 打开

Bit20	B_SEL3	R/W	0	能量桶 3 B 通道累加开关。 0: 关闭; 1: 打开
Bit[19:18]	TYPE_SEL3	R/W	0	能量桶 3 多通道累加模式下, 累加源选择。 功率累加时: 0、3: 有功功率累加。 1: 无功功率累加。 2: 视在功率累加。 有效值累加时: 高位为 0 表示累加和, 高位为 1 表示累加差。
Bit[17:16]	PROCMODE3	R/W	0	对于每一个送入能量桶 3 累加的信号的种类选择。 0: 能量桶只累加正数。 1: 能量桶只累加负数 (此时, 实际累加值为原始值转换的正数)。 2: 能量桶累加原始值。 3: 能量桶累加绝对值。
Bit[15:14]	INMODE2	R/W	0	能量桶 2 累加模式。 0: 功率累加。 1: 电流有效值累加。 2: 常数累加。 3: 可配置基波通道累加。
Bit13	A_SEL2	R/W	0	能量桶 2 A 通道累加开关。 0: 关闭; 1: 打开
Bit12	B_SEL2	R/W	0	能量桶 2 B 通道累加开关。 0: 关闭; 1: 打开
Bit[11:10]	TYPE_SEL2	R/W	0	能量桶 2 多通道累加模式下, 累加源选择。 功率累加时: 0、3: 有功功率累加。 1: 无功功率累加。 2: 视在功率累加。 有效值累加时: 高位为 0 表示累加和, 高位为 1 表示累加差。
Bit[9:8]	PROCMODE2	R/W	0	对于每一个送入能量桶 2 累加的信号的种类选择。 0: 能量桶只累加正数。 1: 能量桶只累加负数 (此时, 实际累加值为原始值转换的正数)。 2: 能量桶累加原始值。 3: 能量桶累加绝对值。
Bit[7:6]	INMODE1	R/W	0	能量桶 1 累加模式。 0: 功率累加。 1: 电流有效值累加。 2: 常数累加。 3: 可配置基波通道累加。
Bit5	A_SEL1	R/W	0	能量桶 1 A 通道累加开关。 0: 关闭; 1: 打开
Bit4	B_SEL1	R/W	0	能量桶 1 B 通道累加开关。 0: 关闭; 1: 打开
Bit[3:2]	TYPE_SEL1	R/W	0	能量桶 1 多通道累加模式下, 累加源选择。 功率累加时: 0、3: 有功功率累加。 1: 无功功率累加。 2: 视在功率累加。 有效值累加时: 高位为 0 表示累加和, 高位为 1 表示累加差。

Bit[1:0]	PROCMODE1	R/W	0	<p>对于每一个送入能量桶 1 累加的信号的种类选择。</p> <p>0: 能量桶只累加正数。</p> <p>1: 能量桶只累加负数（此时，实际累加值为原始值转换的正数）。</p> <p>2: 能量桶累加原始值。</p> <p>3: 能量桶累加绝对值。</p>
----------	-----------	-----	---	--

## 29.2.4.4 EM\_EGY\_CTRL1 寄存器

表 29-14 能量桶控制寄存器 1 (0x05, EM\_EGY\_CTRL1)

0x05, R/W, 能量桶控制寄存器 1, EM_EGY_CTRL1				
位		R/W	默认值	功能说明
Bit[31:30]	INMODE8	R/W	0	<p>能量桶 8 累加模式。</p> <p>0: 功率累加。</p> <p>1: 电流有效值累加。</p> <p>2: 常数累加。</p> <p>3: 可配置基波通道累加。</p>
Bit29	A_SEL8	R/W	0	<p>能量桶 8 A 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开</p>
Bit28	B_SEL8	R/W	0	<p>能量桶 8 B 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开</p>
Bit[27:26]	TYPE_SEL8	R/W	0	<p>能量桶 8 多通道累加模式下，累加源选择。</p> <p>功率累加时：</p> <p>0、3: 有功功率累加。</p> <p>1: 无功功率累加。</p> <p>2: 视在功率累加。</p> <p>有效值累加时：</p> <p>高位为 0 表示累加和，高位为 1 表示累加差。</p>
Bit[25:24]	PROCMODE8	R/W	0	<p>对于每一个送入能量桶 8 累加的信号的种类选择。</p> <p>0: 能量桶只累加正数。</p> <p>1: 能量桶只累加负数（此时，实际累加值为原始值转换的正数）。</p> <p>2: 能量桶累加原始值。</p> <p>3: 能量桶累加绝对值。</p>
Bit[23:22]	INMODE7	R/W	0	<p>能量桶 7 累加模式。</p> <p>0: 功率累加。</p> <p>1: 电流有效值累加。</p> <p>2: 常数累加。</p> <p>3: 可配置基波通道累加。</p>
Bit21	A_SEL7	R/W	0	<p>能量桶 7 A 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开</p>
Bit20	B_SEL7	R/W	0	<p>能量桶 7 B 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开。</p>
Bit[19:18]	TYPE_SEL7	R/W	0	<p>能量桶 7 多通道累加模式下，累加源选择。</p> <p>功率累加时：</p> <p>0、3: 有功功率累加。</p> <p>1: 无功功率累加。</p> <p>2: 视在功率累加。</p> <p>有效值累加时：</p> <p>高位为 0 表示累加和，高位为 1 表示累加差。</p>

Bit[17:16]	PROCMODE7	R/W	0	<p>对于每一个送入能量桶 7 累加的信号的种类选择。</p> <p>0: 能量桶只累加正数。</p> <p>1: 能量桶只累加负数（此时，实际累加值为原始值转换的正数）。</p> <p>2: 能量桶累加原始值。</p> <p>3: 能量桶累加绝对值。</p>
Bit[15:14]	INMODE6	R/W	0	<p>能量桶 6 累加模式。</p> <p>0: 功率累加。</p> <p>1: 电流有效值累加。</p> <p>2: 常数累加。</p> <p>3: 可配置基波通道累加。</p>
Bit13	A_SEL6	R/W	0	<p>能量桶 6 A 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开</p>
Bit12	B_SEL6	R/W	0	<p>能量桶 6 B 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开</p>
Bit[11:10]	TYPE_SEL6	R/W	0	<p>能量桶 6 多通道累加模式下，累加源选择。</p> <p>功率累加时：</p> <p>0、3: 有功功率累加。</p> <p>1: 无功功率累加。</p> <p>2: 视在功率累加。</p> <p>有效值累加时：</p> <p>高位为 0 表示累加和，高位为 1 表示累加差。</p>
Bit[9:8]	PROCMODE6	R/W	0	<p>对于每一个送入能量桶 6 累加的信号的种类选择。</p> <p>0: 能量桶只累加正数。</p> <p>1: 能量桶只累加负数（此时，实际累加值为原始值转换的正数）。</p> <p>2: 能量桶累加原始值。</p> <p>3: 能量桶累加绝对值。</p>
Bit[7:6]	INMODE5	R/W	0	<p>能量桶 5 累加模式。</p> <p>0: 功率累加。</p> <p>1: 电流有效值累加。</p> <p>2: 常数累加。</p> <p>3: 可配置基波通道累加。</p>
Bit5	A_SEL5	R/W	0	<p>能量桶 5 A 通道累加开关。</p> <p>0: 关闭；1: 打开</p>
Bit4	B_SEL5	R/W	0	<p>能量桶 5 B 通道累加开关。</p> <p>0: 关闭；</p> <p>1: 打开</p>
Bit[3:2]	TYPE_SEL5	R/W	0	<p>能量桶 5 多通道累加模式下，累加源选择。</p> <p>功率累加时：</p> <p>0、3: 有功功率累加。</p> <p>1: 无功功率累加。</p> <p>2: 视在功率累加。</p> <p>有效值累加时：</p> <p>高位为 0 表示累加和，高位为 1 表示累加差。</p>
Bit[1:0]	PROCMODE5	R/W	0	<p>对于每一个送入能量桶 5 累加的信号的种类选择。</p> <p>0: 能量桶只累加正数。</p> <p>1: 能量桶只累加负数（此时，实际累加值为原始值转换的正数）。</p> <p>2: 能量桶累加原始值。</p> <p>3: 能量桶累加绝对值。</p>

## 29.2.4.5 EM\_FD\_CTRL 寄存器

表 29-15 快速检测控制寄存器 (0x06, EM\_FD\_CTRL)

0x06, R/W, 快速检测控制寄存器, EM_FD_CTRL				
位		R/W	默认值	功能说明
Bit[31:30]	IPERIOD	R/W	0	过流或欠流检测时间长度选择。当某个半周波内的采样点数中发生过流或欠流的采样点数大于等于 ITH 值, 则认为这个半周波是有效半周波。如果连续有效半周波数量达到 IPERIOD 设定的值, 则认为事件发生。 0: 半个周波。 1: 1 个周波。 2: 2 个周波。 3: 4 个周波。
Bit[29:24]	ITH	R/W	0	半周波判断为有效半周波的阈值。 0: 1 个采样点。 1: 2 个采样点。 ... 63: 64 个采样点。
Bit[23:22]	UPERIOD	R/W	0	配置情况与 IPERIOD 一致。 0: 半个周波。 1: 1 个周波。 2: 2 个周波。 3: 4 个周波。
Bit[21:16]	UTH	R/W	0	配置情况与 ITH 一致。 0: 1 个采样点。 1: 2 个采样点。 ... 63: 64 个采样点。
Bit[15:10]	保留		0	为保证系统正常工作, 必须写入默认值。
Bit9	IBLCSEL	R/W	0	IB 欠流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit8	IBOCSEL	R/W	0	IB 过流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit7	IALCSEL	R/W	0	IA 欠流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit6	IAOCSEL	R/W	0	IA 过流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit5	ULVSEL	R/W	0	U 通道欠压检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit4	UOVSEL	R/W	0	U 通道过压检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit3	保留		0	为保证系统正常工作, 必须写入默认值。
Bit2	FDIBEN	R/W	0	IB 过流欠流检测的开关。 0: 关闭; 1: 开启
Bit1	FDIAEN	R/W	0	IA 过流欠流检测的开关。 0: 关闭; 1: 开启



Bit0	FDUEN	R/W	0	U 通道过压欠压检测的开关。 0: 关闭; 1: 开启
------	-------	-----	---	-----------------------------------

## 29.2.4.6 EM\_WAVE\_CTRL 寄存器

表 29-16 波形输出和缓存控制寄存器 (0x07, EM\_WAVE\_CTRL)

0x07, R/W, 波形输出和缓存控制寄存器, EM_WAVE_CTRL				
位		R/W	默认值	功能说明
Bit31	WAVE_ADDR_CLR	R/W	0	复位读取波形缓存时的地址, 写 1 复位。
Bit[30:29]	WAVE_MEM_MODE	R/W	0	波形缓存工作模式: 0: 手动开始存, 存满停, 此时为单次存储。 1: 手动开始存, 事件触发停或手动停, 此时为循环存储。 2: 事件触发开始存, 存满停, 此时为单次存储。 3: 不开启。
Bit28	WAVE_MEM_EN	R/W	0	波形缓存使能。 波形缓存工作模式为 0 时, 写 1 使能一次波形缓存, 立即开始存波形, 写 0 无效果。 波形缓存工作模式为 1 时, 写 1 使能一次波形缓存, 立即开始存波形, 写 0 波形缓存立即停止存波形。 波形缓存工作模式为 2 时, 写 1 使能一次波形缓存, 等待事件触发后开启存数据, 写 0 无效果。
Bit27	U_DIP_TRIG	R/W	0	波形缓存和输出电压骤降事件触发开关。 0: 关闭; 1: 打开
Bit26	U_SWELL_TRIG	R/W	0	波形缓存和输出电压骤升事件触发开关。 0: 关闭; 1: 打开
Bit25	IB_LC_TRIG	R/W	0	波形缓存和输出 IB 欠流事件触发开关。 0: 关闭; 1: 打开
Bit24	IB_OC_TRIG	R/W	0	波形缓存和输出 IB 过流事件触发开关。 0: 关闭; 1: 打开
Bit23	IA_LC_TRIG	R/W	0	波形缓存和输出 IA 欠流事件触发开关。 0: 关闭; 1: 打开
Bit22	IA_OC_TRIG	R/W	0	波形缓存和输出 IA 过流事件触发开关。 0: 关闭; 1: 打开
Bit21	U_LV_TRIG	R/W	0	波形缓存和输出欠压事件触发开关。 0: 关闭; 1: 打开
Bit20	U_OV_TRIG	R/W	0	波形缓存和输出过压事件触发开关。 0: 关闭; 1: 打开
Bit[19:16]	WAVE_LENGTH	R/W	0	输出波形长度选择。 0: 1 个周波。

				1: 2 个周波。 2: 3 个周波。 ..... 15: 16 个周波。
Bit15	保留		0	为保证系统正常工作，必须写入默认值。
Bit14	WAVE_U_HPF_SEL	R/W	0	波形缓存和输出的电压波形是否经过高通滤波器选择。 0: 不经过; 1: 经过
Bit13	WAVE_IA_HPF_SEL	R/W	0	波形缓存和输出的电流 IA 波形是否经过高通滤波器选择。 0: 不经过; 1: 经过
Bit12	WAVE_IB_HPF_SEL	R/W	0	波形缓存和输出的电流 IB 波形是否经过高通滤波器选择。 0: 不经过; 1: 经过
Bit11	保留		0	为保证系统正常工作，必须写入默认值。
Bit10	WAVE_U_SEL	R/W	0	波形缓存和输出的 U 通道开关，该 Bit 决定是否存储或发送 U 通道数据。 0: 否; 1: 是
Bit9	WAVE_IA_SEL	R/W	0	波形缓存和输出的 IA 通道开关，该 Bit 决定是否存储或发送 IA 通道数据。 0: 否; 1: 是
Bit8	WAVE_IB_SEL	R/W	0	波形缓存和输出的 IB 通道开关，该 Bit 决定是否存储或发送 IB 通道数据（若波形缓存配置 Bit8, Bit9, Bit10 同时使能，该 Bit 位失效）。 0: 否; 1: 是
Bit7	SP_CHECK	R/W	0	波形输出的奇偶校验选择。 0: 奇校验; 1: 偶校验
Bit6	SPI_POL	R/W	0	波形输出的 SPI 极性选择。 0: 负; 1: 正
Bit5	SPI_PHA	R/W	0	波形输出的 SPI 相位选择。 0: 负; 1: 正
Bit[4:3]	WAVE_OUT_EN	R/W	0	波形输出功能的开关。 0: 无效; 1: 手动发 2: 手动停 3: 保留
Bit[2:0]	WAVE_OUT_MODE	R/W	0	波形输出的工作模式: 0: 手动开始发，发满 WAVE_LENGTH 个周波停。 1: 手动开始发，事件触发停或手动停。 2: 事件触发开始发，手动停。 3: 事件触发开始发，发满 WAVE_LENGTH 个周波停。 4~7: 手动开始发，手动停。



在使能波形输出前，必须至少打开一路波形缓存和输出的通道。

29.2.5 计量数据寄存器

发生上电复位（POR）、片外输入 RSTN 复位时，所有计量数据寄存器被复位。

29.2.5.1 直流分量寄存器

表 29-17 直流分量寄存器

地址	寄存器	R/W	数据格式	说明	
0x22	EM_DAT_DCU	R	32-Bit 补码	电压通道直流值	默认配置下刷新 40ms，稳定 120ms（具体时间与系统时钟的准确度有关）。
0x23	EM_DAT_DCIA	R	32-Bit 补码	电流 A 通道直流值	
0x24	EM_DAT_DCIB	R	32-Bit 补码	电流 B 通道直流值	

29.2.5.2 有效值寄存器

表 29-18 电压/电流/测量信号（M）有效值寄存器

地址	寄存器	R/W	数据格式	说明	
0x0E	EM_DAT_RMS0U	R	32-Bit 补码	电压瞬时有效值	默认配置下更新时间为 10ms，稳定时间为 30ms（具体时间与系统时钟的准确度有关）。
0x0F	EM_DAT_RMS0IA	R	32-Bit 补码	电流 A 瞬时有效值	
0x10	EM_DAT_RMS0IB	R	32-Bit 补码	电流 B 瞬时有效值	
0x19	EM_DAT_RMS1U	R	32-Bit 补码	电压平均有效值	默认配置下更新时间为 40ms，稳定时间为 120ms（具体时间与系统时钟的准确度有关）。
0x1A	EM_DAT_RMS1IA	R	32-Bit 补码	电流 A 平均有效值	
0x1B	EM_DAT_RMS1IB	R	32-Bit 补码	电流 B 平均有效值	
0x1E	EM_DAT_RMSU_AVG	R	32-Bit 补码	10 或 12 个周波（通过电网频率选择）的电压有效值平均值	
0x1F	EM_DAT_RMSIA_AVG	R	32-Bit 补码	10 或 12 个周波（通过电网频率选择）的电流 IA 有效值平均值	
0x20	EM_DAT_RMSIB_AVG	R	32-Bit 补码	10 或 12 个周波（通过电网频率选择）的电流 IB 有效值平均值	

29.2.5.3 有功/无功功率寄存器

表 29-19 有功/无功功率寄存器

地址	寄存器	R/W	数据格式	说明	
0x08	EM_DAT_PA	R	32-Bit 补码	A 通道瞬时有功功率	默认配置下更新时间为 20ms，稳定时间为 60ms
0x09	EM_DAT_QA	R	32-Bit 补码	A 通道瞬时无功功率	

地址	寄存器	R/W	数据格式	说明	
0x0A	EM_DAT_SA	R	32-Bit 补码	A 通道瞬时视在功率	(具体时间与系统时钟的准确度有关)。
0x0B	EM_DAT_PB	R	32-Bit 补码	B 通道瞬时有功功率	
0x0C	EM_DAT_QB	R	32-Bit 补码	B 通道瞬时无功功率	
0x0D	EM_DAT_SB	R	32-Bit 补码	B 通道瞬时视在功率	
0x13	EM_DAT_PA1	R	32-Bit 补码	A 通道平均有功功率	默认配置下更新时间为 80ms，稳定时间为 240ms (具体时间与系统时钟的准确度有关)。
0x14	EM_DAT_QA1	R	32-Bit 补码	A 通道平均无功功率	
0x15	EM_DAT_SA1	R	32-Bit 补码	A 通道平均视在功率	
0x16	EM_DAT_PB1	R	32-Bit 补码	B 通道平均有功功率	
0x17	EM_DAT_QB1	R	32-Bit 补码	B 通道平均无功功率	
0x18	EM_DAT_SB1	R	32-Bit 补码	B 通道平均视在功率	

### 29.2.5.4 基波通道数据寄存器

表 29-20 基波通道瞬时值寄存器

地址	寄存器	R/W	数据格式	说明
0x11	EM_DAT_CH1	R	32-Bit 补码	基波可选通道1瞬时值
0x12	EM_DAT_CH2	R	32-Bit 补码	基波可选通道 2 瞬时值

表 29-21 基波通道平均值寄存器

地址	寄存器	R/W	数据格式	说明
0x1C	EM_DAT_CH1_AVG	R	32-Bit 补码	基波可选通道1平均值
0x1D	EM_DAT_CH2_AVG	R	32-Bit 补码	基波可选通道 2 平均值

### 29.2.5.5 电网频率寄存器

表 29-22 电网频率寄存器 (0x21, EM\_DAT\_FRQ)

地址	寄存器	R/W	数据格式	说明
0x21	EM_DAT_FRQ	R	32-Bit 补码	电网频率，与 FRQ_SEL 有关。 默认配置下，更新时间为 320ms，稳定时间为 640ms (具体时间与系统时钟的准确度有关)。

### 29.2.6 能量桶寄存器

表 29-23 能量桶寄存器

地址	寄存器	R/W	数据格式	说明
0x39	EM_EGY_CRPTH	R/W	32-Bit 无符号数	能量桶防潜阈值。当防潜能量桶累加值超过 EM_EGY_CRPTH 并且高速能量桶累加值未超过 EM_EGY_PWRTH 时，高速能量桶的累加值会被清掉。

0x3A	EM_EGY_PWRTH	R/W	32-Bit 无符号数	能量桶累加阈值。由于能量桶为46Bit，实际高速能量桶累加阈值为该阈值*16384；低速能量桶累加阈值为该阈值*4。
0x3B	EM_EGY_CONST1	R/W	32-Bit 无符号数	能量桶 1 累加常数
0x3C	EM_EGY_OUT1L	R/W	32-Bit 无符号数	能量桶 1 累加值低位
0x3D	EM_EGY_OUT1H	R/W	32-Bit 无符号数	能量桶1累加值高位 低 14Bit 有效
0x3E	EM_EGY_CFCNT1	R	32-Bit 无符号数	能量桶 1 脉冲计数器
0x3F	EM_EGY_CONST2	R/W	32-Bit 无符号数	能量桶 2 累加常数
0x40	EM_EGY_OUT2L	R/W	32-Bit 无符号数	能量桶 2 累加值低位
0x41	EM_EGY_OUT2H	R/W	32-Bit 无符号数	能量桶2累加值高位 低 14Bit 有效
0x42	EM_EGY_CFCNT2	R	32-Bit 无符号数	能量桶 2 脉冲计数器
0x43	EM_EGY_CONST3	R/W	32-Bit 无符号数	能量桶 3 累加常数
0x44	EM_EGY_OUT3	R/W	32-Bit 无符号数	能量桶 3 累加值
0x45	EM_EGY_CFCNT3	R	32-Bit 无符号数	能量桶 3 脉冲计数器
0x46	EM_EGY_CONST4	R/W	32-Bit 无符号数	能量桶 4 累加常数
0x47	EM_EGY_OUT4	R/W	32-Bit 无符号数	能量桶 4 累加值
0x48	EM_EGY_CFCNT4	R	32-Bit 无符号数	能量桶 4 脉冲计数器
0x49	EM_EGY_CONST5	R/W	32-Bit 无符号数	能量桶 5 累加常数
0x4A	EM_EGY_OUT5	R/W	32-Bit 无符号数	能量桶 5 累加值
0x4B	EM_EGY_CFCNT5	R	32-Bit 无符号数	能量桶 5 脉冲计数器
0x4C	EM_EGY_CONST6	R/W	32-Bit 无符号数	能量桶 6 累加常数
0x4D	EM_EGY_OUT6	R/W	32-Bit 无符号数	能量桶 6 累加值
0x4E	EM_EGY_CFCNT6	R	32-Bit 无符号数	能量桶 6 脉冲计数器
0x4F	EM_EGY_CONST7	R/W	32-Bit 无符号数	能量桶 7 累加常数
0x50	EM_EGY_OUT7	R/W	32-Bit 无符号数	能量桶 7 累加值
0x51	EM_EGY_CFCNT7	R	32-Bit 无符号数	能量桶 7 脉冲计数器
0x52	EM_EGY_CONST8	R/W	32-Bit 无符号数	能量桶 8 累加常数
0x53	EM_EGY_OUT8	R/W	32-Bit 无符号数	能量桶 8 累加值
0x54	EM_EGY_CFCNT8	R	32-Bit 无符号数	能量桶 8 脉冲计数器

## 29.2.7 相位测量寄存器

表 29-24 相位测量寄存器

地址	寄存器	R/W	数据格式	说明
0x61	EM_PHS_STT	R/W	32-Bit 无符号数	相位测量控制位，写操作使能一次相位测量。
0x62	EM_PHS_U	R	32-Bit 无符号数	电压相位值
0x63	EM_PHS_UN	R	32-Bit 无符号数	电压过零点之前波形数据值
0x64	EM_PHS_UP	R	32-Bit 无符号数	电压过零点之后波形数据值
0x65	EM_PHS_I	R	32-Bit 无符号数	电流相位值
0x66	EM_PHS_IN	R	32-Bit 无符号数	电流过零点之前波形数据值
0x67	EM_PHS_IP	R	32-Bit 无符号数	电流过零点之后波形数据值

## 29.2.8 功率潜动阈值寄存器

表 29-25 功率潜动阈值寄存器

地址	寄存器	R/W	数据格式	说明
0x55	EM_OV_THL	R/W	32-Bit 补码	A 通道和 B 通道的瞬时有功功率/无功功率/视在功率的潜动判断下限阈值
0x56	EM_OV_THH	R/W	32-Bit 补码	A 通道和 B 通道的瞬时有功功率/无功功率/视在功率的潜动判断上限阈值

## 29.2.9 电压骤升骤降寄存器

表 29-26 电压骤升骤降阈值寄存器

地址	寄存器	R/W	数据格式	说明
0x57	EM_SWELL_THL	R/W	32-Bit 补码	电压骤升下限阈值。
0x58	EM_SWELL_THH	R/W	32-Bit 补码	电压骤升上限阈值。
0x59	EM_DIP_THL	R/W	32-Bit 补码	电压骤降下限阈值
0x5A	EM_DIP_THH	R/W	32-Bit 补码	电压骤降上限阈值
0x6A	EM_DAT_SWELL_CNT	R/C	32-Bit 补码	电压骤升时间记录，半波为单位。24Bit 有效。向该寄存器写任意值，可清零该计数值。
0x6B	EM_DAT_DIP_CNT	R/C	32-Bit 补码	电压骤降时间记录，半波为单位。24Bit 有效。向该寄存器写任意值，可清零该计数值。

## 29.2.10 快速检测寄存器

表 29-27 快速检测阈值寄存器

地址	寄存器	R/W	数据格式	说明
0x5B	EM_FD_OVTH	R/W	30-Bit 补码	快速检测过压阈值。位宽为30Bit。
0x5C	EM_FD_LVTH	R/W	30-Bit 补码	快速检测欠压阈值。位宽为30Bit。
0x5D	EM_FD_IA_OCTH	R/W	30-Bit 补码	快速检测电流A通道过流阈值。位宽为30Bit。
0x5E	EM_FD_IA_LCTH	R/W	30-Bit 补码	快速检测电流A通道欠流阈值。位宽为30Bit。
0x5F	EM_FD_IB_OCTH	R/W	30-Bit 补码	快速检测电流B通道过流阈值。位宽为30Bit。
0x60	EM_FD_IB_LCTH	R/W	30-Bit 补码	快速检测电流B通道欠流阈值。位宽为30Bit。

## 29.2.11 波形数据寄存器

表 29-28 波形数据寄存器（0x69，EM\_DAT\_WAVE）

地址	寄存器	R/W	数据格式	说明
0x69	EM_DAT_WAVE	R	32-Bit 补码	波形数据读取，可重复读取该地址，从而获得完整波形数据。若不需要读完全部数据，可通过 波形输出和缓存控制寄存器（0x07，EM_WAVE_CTRL）的 Bit31 复位读取地址。

## 29.2.12 校表参数寄存器

发生上电复位（POR）、片外输入 RSTN 复位时，所有设置校表参数的寄存器均被复位为默认值。下表中的“默认值”为十六进制数值。所有校表参数寄存器均需参与参数配置自检校验。

### 29.2.12.1 预设直流偏置值寄存器

表 29-29 预设直流偏置值寄存器

地址	寄存器	默认值	R/W	数据格式	说明
0x34	EM_CFG_DCU	0	R/W	32-Bit 补码	电压通道直流校正
0x35	EM_CFG_DCIA	0	R/W	32-Bit 补码	电流 A 通道直流校正
0x36	EM_CFG_DCIB	0	R/W	32-Bit 补码	电流 B 通道直流校正
所有校表参数寄存器均需参与参数配置自检校验。					

### 29.2.12.2 有效值校正寄存器

表 29-30 电压/电流/测量值有效值校正寄存器

地址	寄存器	默认值	R/W	数据格式	说明
0x2D	EM_CFG_CALI_RMSU	0	R/W	32-Bit 补码	电压有效值比差校正
0x2E	EM_CFG_RMS_DCU	0	R/W	32-Bit 补码	电压有效值小信号校正
0x2F	EM_CFG_CALI_RMSIA	0	R/W	32-Bit 补码	电流 A 有效值比差校正
0x30	EM_CFG_RMS_DCIA	0	R/W	32-Bit 补码	电流 A 有效值小信号校正

0x31	EM_CFG_CALI_RMSIB	0	R/W	32-Bit 补码	电流 B 有效值比差校正
0x32	EM_CFG_RMS_DCIB	0	R/W	32-Bit 补码	电流 B 有效值小信号校正

### 29.2.12.3 功率校正寄存器

表 29-31 全波有功/无功功率校正寄存器

地址	寄存器	默认值	R/W	数据格式	说明
0x25	EM_CFG_CALI_PA	0	R/W	32-Bit 补码	有功功率 A 比差校正
0x26	EM_CFG_DC_PA	0	R/W	32-Bit 补码	有功功率 A 小信号校正
0x27	EM_CFG_CALI_QA	0	R/W	32-Bit 补码	无功功率 A 比差校正
0x28	EM_CFG_DC_QA	0	R/W	32-Bit 补码	无功功率 A 小信号校正
0x29	EM_CFG_CALI_PB	0	R/W	32-Bit 补码	有功功率 B 比差校正
0x2A	EM_CFG_DC_PB	0	R/W	32-Bit 补码	有功功率 B 小信号校正
0x2B	EM_CFG_CALI_QB	0	R/W	32-Bit 补码	无功功率 B 比差校正
0x2C	EM_CFG_DC_QB	0	R/W	32-Bit 补码	无功功率 B 小信号校正

### 29.2.12.4 门限值寄存器

表 29-32 门限值寄存器

地址	寄存器	默认值	R/W	数据格式	说明
0x39	EM_EGY_CRPTH	0	R/W	32-Bit 补码	<p>能量桶防潜阈值。当防潜能量桶累加值超过 EM_EGY_CRPTH 并且高速能量桶累加值未超过 EM_EGY_PWRTH 时，高速能量桶的累加值会被清掉。</p> <p>能量桶 1 和能量桶 2 分别有一个潜动能累加寄存器。当使能防潜功能后，该潜动能累加值固定为 1。其累加速率与能量桶 1 和能量桶 2 累加速率相等。</p> <p>用户应分别在启动/潜动判断门限值寄存器（EM_EGY_CRPTH）和能量累加门限值寄存器（EM_EGY_PWRTH）设置各自的门限值。如果潜动能累加寄存器的累加值先达到 EM_EGY_CRPTH 值时，能量累加寄存器被清空，系统进入潜动状态。当能量累加寄存器的累加值先达到 EM_EGY_PWRTH 值时，潜动能累加寄存器被清空，系统进入启动状态，正常工作。</p> <p>寄存器 EM_EGY_CRPTH 的实际位宽为 32 位，但在参与启动/潜动的判断运算时，寄存器的内容会被自动在低位补 4 个 0，扩展到 36 位再参与运算。</p>
0x3A	EM_EGY_PWRTH	0	R/W	32-Bit 补码	能量桶累加阈值。由于能量桶为 46Bit，实际高速能量桶累加阈值为该阈值*16384；低速能量桶累加阈值为该阈值*4。
0x55	EM_OV_THL	0	R/W	32-Bit 补码	A 通道和 B 通道的瞬时有功功率/无功功率/视在功率的潜动判断下限阈值。该寄存器需参与参数配置自检校验。



地址	寄存器	默认值	R/W	数据格式	说明
0x56	EM_OV_THH	0	R/W	32-Bit 补码	A 通道和 B 通道的瞬时有功功率/无功功率/视在功率的潜动判断上限阈值。

## 29.2.12.5 角差校正寄存器

表 29-33 角差校正寄存器（0x33，EM\_CFG\_PHC）

地址	寄存器	默认值	R/W	数据格式	说明
0x33	EM_CFG_PHC	0	R/W	32-Bit 补码	角差校正寄存器。该寄存器需要参与参数配置自检校验。 [10:0]位是A通道角差校正值。 [26:16]位是B通道角差校正值。 其范围为-766~767。

## 29.2.12.6 带通滤波器系数寄存器

表 29-34 带通滤波器系数寄存器（0x37，EM\_CFG\_BPF）

地址	寄存器	默认值	R/W	数据格式	说明
0x37	EM_CFG_BPF	0	R/W	32-Bit 补码	带通滤波器系数。该参数的设置与计量控制寄存器0（0x02，EM_CTRL0）的EM_MODE（Bit[7:4]）相关。 EM_MODE=0、1、2时，写入0x806764B6； EM_MODE=6、7时，写入0x80DD7A8C； EM_MODE=8时，写入0x82B465F0； 其余模式不支持频率测量，该寄存器需写 0x0。

## 29.2.13 校验和寄存器

表 29-35 校验和寄存器（0x38，EM\_CFG\_CKSUM）

地址	寄存器	默认值	R/W	数据格式	说明
0x38	EM_CFG_CKSUM	0	R/W	32-Bit 补码	该寄存器需要参与参数配置自检校验。该寄存器与地址 0x0~0x7，0x25~0x3A，0x55~0x60 寄存器一起参与参数配置自检校验。如果上述所有寄存器值的累加和为 0xFFFFFFFF 时，参数配置自检通过。 为保证参数配置自校验成功，该寄存器应写入 0xFFFFFFFF 与其它参与校验的寄存器配置值和的差值。

表 29-36 参与参数自检寄存器列表

寄存器	类型	地址	描述	默认值
EM_ANA_CTRL0	R/W	0x00	电能计量模拟控制寄存器 0	0x00000000
EM_ANA_CTRL1	R/W	0x01	电能计量模拟控制寄存器 1	0x00000000

EM_CTRL0	R/W	0x02	计量控制寄存器 0	0x00000000
EM_CTRL1	R/W	0x03	计量控制寄存器 1	0x00000000
EM_EGY_CTRL0	R/W	0x04	能量桶控制寄存器 0	0x00000000
EM_EGY_CTRL1	R/W	0x05	能量桶控制寄存器 1	0x00000000
EM_FD_CTRL	R/W	0x06	快速检测控制寄存器	0x00000000
EM_WAVE_CTRL	R/W	0x07	波形输出和缓存控制寄存器	0x00000000
EM_CFG_CALI_PA	R/W	0x25	有功功率 A 比差校正	0x00000000
EM_CFG_DC_PA	R/W	0x26	有功功率 A 小信号校正	0x00000000
EM_CFG_CALI_QA	R/W	0x27	无功功率 A 比差校正	0x00000000
EM_CFG_DC_QA	R/W	0x28	无功功率 A 小信号校正	0x00000000
EM_CFG_CALI_PB	R/W	0x29	有功功率 B 比差校正	0x00000000
EM_CFG_DC_PB	R/W	0x2A	有功功率 B 小信号校正	0x00000000
EM_CFG_CALI_QB	R/W	0x2B	无功功率 B 比差校正	0x00000000
EM_CFG_DC_QB	R/W	0x2C	无功功率 B 小信号校正	0x00000000
EM_CFG_CALI_RMSU	R/W	0x2D	电压有效值比差校正	0x00000000
EM_CFG_RMS_DCU	R/W	0x2E	电压有效值小信号校正	0x00000000
EM_CFG_CALI_RMSIA	R/W	0x2F	电流 A 有效值比差校正	0x00000000
EM_CFG_RMS_DCIA	R/W	0x30	电流 A 有效值小信号校正	0x00000000
EM_CFG_CALI_RMSIB	R/W	0x31	电流 B 有效值比差校正	0x00000000
EM_CFG_RMS_DCIB	R/W	0x32	电流 B 有效值小信号校正	0x00000000
EM_CFG_PHC	R/W	0x33	角差校正寄存器。 [10: 0]位是 A 通道角差校正值。 [26: 16]位是 B 通道角差校正值。 其范围为-766~767。	0x000
EM_CFG_DCU	R/W	0x34	电压通道直流校正值	0x00000000
EM_CFG_DCIA	R/W	0x35	电流 A 通道直流校正值	0x00000000
EM_CFG_DCIB	R/W	0x36	电流 B 通道直流校正值	0x00000000
EM_CFG_BPF	R/W	0x37	带通滤波器系数。该参数的设置与计量控制寄存器0（0x02，EM_CTRL0）的EM_MODE（Bit[7:4]）相关。 EM_MODE=0、1、2时，写入0x806764B6； EM_MODE=6、7时，写入0x80DD7A8C； EM_MODE=8时，写入0x82B465F0； 其余模式不支持频率测量，该寄存器需写0x0。	0x00000000
EM_CFG_CKSUM	R/W	0x38	电能计量校验和配置寄存器	0x00000000



EM_EGY_CRPTH	R/W	0x39	能量桶防潜阈值。当防潜能量桶累加值超过 EM_EGY_CRPTH 并且高速能量桶累加值未超过 EM_EGY_PWRTH 时，高速能量桶的累加值会被清掉。	0x00000000
EM_EGY_PWRTH	R/W	0x3A	能量桶累加阈值。由于能量桶为 46Bit，实际高速能量桶累加阈值为该阈值*16384；低速能量桶累加阈值为该阈值*4。	0x00000000
EM_OV_THL	R/W	0x55	功率潜动判断下限阈值	0x00000000
EM_OV_THH	R/W	0x56	功率潜动判断上限阈值	0x00000000
EM_SWELL_THL	R/W	0x57	电压骤升下限阈值。	0x00000000
EM_SWELL_THH	R/W	0x58	电压骤升上限阈值。	0x00000000
EM_DIP_THL	R/W	0x59	电压骤降下限阈值	0x00000000
EM_DIP_THH	R/W	0x5A	电压骤降上限阈值	0x00000000
EM_FD_OVTH	R/W	0x5B	快速检测过压阈值 位宽为 30Bit。	0x00000000
EM_FD_LVTH	R/W	0x5C	快速检测欠压阈值。 位宽为 30Bit。	0x00000000
EM_FD_IA_OCTH	R/W	0x5D	快速检测电流 A 通道过流阈值。 位宽为 30Bit。	0x00000000
EM_FD_IA_LCTH	R/W	0x5E	快速检测电流 A 通道欠流阈值。 位宽为 30Bit。	0x00000000
EM_FD_IB_OCTH	R/W	0x5F	快速检测电流 B 通道过流阈值。 位宽为 30Bit。	0x00000000
EM_FD_IB_LCTH	R/W	0x60	快速检测电流 B 通道欠流阈值。 位宽为 30Bit。	0x00000000

## 29.3 电能计量模块电压基准电路（简称 EMBGP）

电压基准电路（Bandgap）输出一个随温度变化较小的（典型温度漂移 10ppm/°C）约 1.21V 的基准电压，为电能计量 ADC 提供基准电压和偏置电流。Bandgap 电路默认开启。

该电路负载电流约 0.09mA（典型）。

用户可以通过配置模拟控制寄存器 1（0x01, EM\_ANA\_CTRL1）中的 RESTL<1:0>（Bit[4:3]）和 REST<2:0>（Bit[2:0]）对 Bandgap 电路的温度系数进行调整，以便与系统（如片外元器件等）带来的温度系数相抵消。具体步骤如下：

- 1) 先确定现有温度系数寄存器的配置，假设当前细调寄存器 REST<2: 0>='010'，粗调 RESTL<1: 0>='00'，那么查表知道目前对应的温度系数调节幅度为+14ppm；
- 2) 实验确定电表的高低温误差。例如 20℃时误差为 0，高温 80℃误差为 0.6%，低温-40℃为-0.4%。那么需要调节的误差幅度是 $(0.6\% - (-0.4\%))/2$ ，为 0.5%，调节方向以高温为基准，应该将高温误差往负调。对应 ppm 是 $-0.5\%/(80-20)=-5000/60=-83\text{ppm}$ ；
- 3) 误差和 Reference 的变化幅度是负两倍关系，因此对应要将 PMREF 往正方向调节 41.5ppm。考虑到现有配置已经是+14ppm，那么最终要调+55.5ppm。查表知道当 RESTL<1:0>='11'时，对应 56ppm，REST<2:0>='000'时，对应 0ppm，组合起来等于 56ppm。

注意：对 Reference 温度参数的调节会影响到基本误差，因此，客户每设计一款新的产品时，首先确认 Reference 的温度参数，再对电能表进行误差校正。

当 Bandgap 温度系数的调整幅度为 x 时，电表计量误差的温度系数调整幅度为-2x。

表 29-37 Bandgap 电路相关寄存器

寄存器	位	默认值	说明
EM_ANA_CTRL1	Bit[4:3] RESTL<1:0>	0	电压基准（Bandgap）电路的温度系数粗调节。 00: 0ppm; 01: -58ppm; 10: +111ppm; 11: +56ppm; 正常计量时，为了获取最佳的计量性能和温度性能，用户必须根据计算结果进行调整。
	Bit[2:0] REST<2:0>	0	电压基准（Bandgap）电路的温度系数微调节。 000: 0ppm; 001: +7ppm; 010: +14ppm; 011: +28ppm; 100: -32ppm; 101: -21ppm; 110: -14ppm; 111: -7ppm; 正常计量时，为了获取最佳的计量性能和温度性能，用户必须根据计算结果进行调整。

29.4电能计量时钟

- 在 V94XX(A)中，电能计量时钟系统包括 2 个时钟源，包括：
- PCLK 时钟经过开关和分频，提供 6.5MHz 时钟（EM\_PCLKDIV）：经 SDIV 分频后（SDIV 通过 EM\_MODE（Bit<7:4>，计量控制寄存器 0（0x02，EM\_CTRL0））控制），供所有数字模块（除能量桶外）使用；经 ADIV 分频后（ADIV 通过 ADCKSEL（Bit<31:30>，模拟控制寄存器 1（0x01，EM\_ANA\_CTRL1））控制），供 ADC 使用。
  - 内置高频 RC 振荡电路提供的 6.5MHz（批量芯片之间偏差在±20%以内，单个芯片-40~85 摄氏度范围内偏差在±5%以内）RC 时钟（EM\_RCH）：当 EM\_PCLKDIV 缺失时，该时钟可供所有数字模块使用。
  - 外部低频时钟（32768Hz），通过 X32KIN 管脚输入，该时钟提供能量桶低速累加使用。能量桶时钟通过 EM\_EGY\_CLK\_SEL（Bit23，计量控制寄存器 1（0x03，EM\_CTRL1））控制。

上述两个时钟产生电路之间的关系如下图所示：

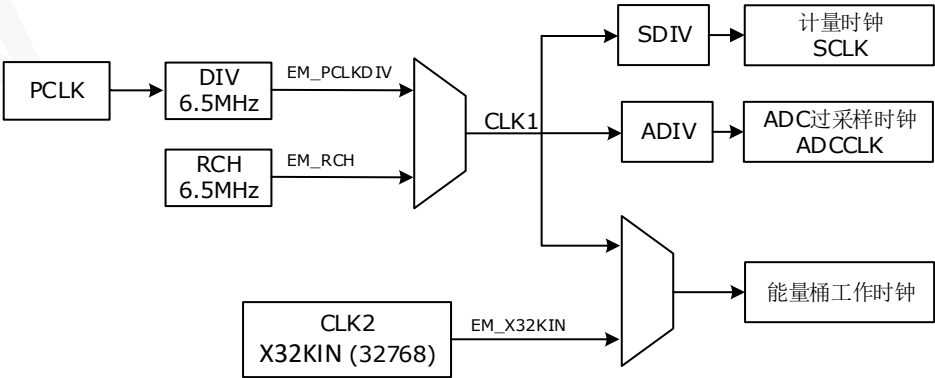


图 29-2 电能计量时钟系统

表 29-38 时钟产生电路相关寄存器

寄存器	位	默认值	说明
模拟控制寄存器 1 (0x01, EM_ANA_CTRL1)	Bit[31:30] ADCKSEL<1:0>	0	ADC时钟频率选择。00对应默认频率819.2KHz。 00: $\times 1$ ; 01: $\times 2$ ; 10: $\times 1/4$ ; 11: $\times 1/2$
计量控制寄存器 0 (0x02, EM_CTRL0)	Bit[7:4] EM_MODE<3:0>	0	DSP工作模式 0: 系统时钟6.5536MHz, DSP每周波采样点数128。 1: 系统时钟6.5536MHz, DSP每周波采样点数64。 2: 系统时钟6.5536MHz, DSP每周波采样点数32。 3, 4, 5: 保留。 6: 系统时钟3.2768MHz, DSP每周波采样点数64。 7: 系统时钟3.2768MHz, DSP每周波采样点数32。 8: 系统时钟819.2KHz, DSP每周波采样点数32。(此时只支持最多两路瞬时电流有效值计算)。 9: 系统时钟409.6KHz, DSP每周波采样点数16。(此时只支持最多两路瞬时电流有效值计算)。 其余: 与模式0保持一致。
计量控制寄存器 1 (0x03, EM_CTRL1)	Bit23 EM_EGY_CLK_SEL	0	能量桶时钟选择位。时钟切换需要稳定时间, 约为107 $\mu$ s, 请在稳定前关闭CF。 0: 204.8KHz; 1: 32768Hz

### 29.4.1 EM\_PCLKDIV 时钟

用户可通过对 PCLK 时钟分频产生 6.5536MHz 时钟, 为 V94XX(A)电能计量模块提供 CLK1 时钟, 供计量 VMA、ADC 和能量桶等使用。如果用户没有使能 CLK1 时钟, 此时支持两种模式应用: 能量桶功能不使用; 选择使用 X32KIN 引脚灌入 32768Hz 频率时钟作为能量桶时钟。

### 29.4.2 EM\_RCH 时钟

电能计量模块内置一个高频 RC 振荡器, 产生一个 6.5 MHz (批量芯片之间偏差在 $\pm 20\%$ 以内, 单个芯片-40~85 摄氏度范围内偏差在 $\pm 5\%$ 以内) 的 RC 时钟。

在计量模式下, 当 EM\_PCLKDIV 停止工作时, 该电路自动开启, 为 V94XX(A)电能计量模块提供 CLK1 时钟, 供计量 VMA、ADC 和能量桶等使用。

发生 POR 复位、片外输入 RSTN 复位时, 该电路和 Bandgap 电路自动打开。

### 29.4.3 外部输入低频时钟

外部可以向 V94XX(A)的 X32KIN 引脚灌入 32768Hz 频率时钟, 为 V94XX(A)电能计量模块提供 CLK2 时钟, 供能量桶低速累加使用。

## 29.5 模拟信号输入

V94XX(A)支持 2 种电流信号输入。其中, 电流传感器 (CT, Current Transformer) 输入电流采用双端完全差动输入方式, 共需要占用 4 个端口, 接线方式如下图所示 (以 AGND 为共模电压, 共模电压也可以为其他形式)。IAP、IAN 是电流通道 IA 的差分输入引脚; IBP、IBN 是电流通道 IB 的差分输入引脚。

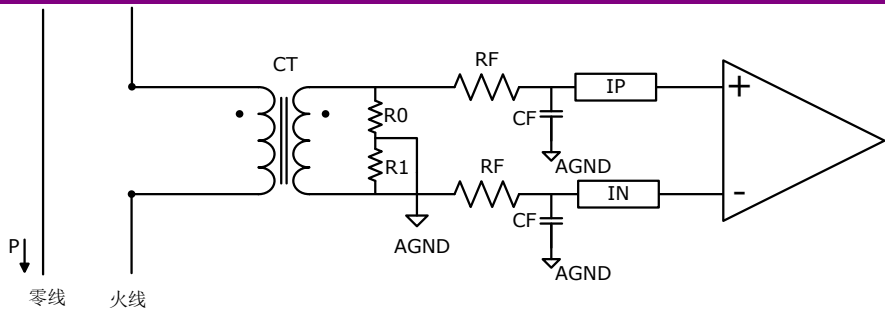


图 29-3 采用 CT 输入电流

也可以采用锰铜电阻分流网络输入电流，采用 AGND 接地。

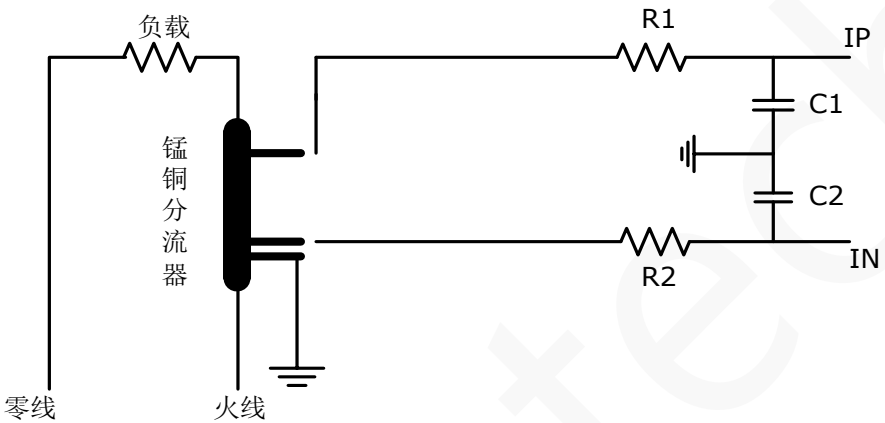


图 29-4 采用锰铜电阻分流网络输入电流

V94XX(A)支持 1 路电压信号输入。电压采用伪差分输入方式，相对于 UN 接地，UP 为正端，共需要 2 个端口，下面为电压通道的两种典型接法，分别采用电压互感器方式和电阻分压方式。

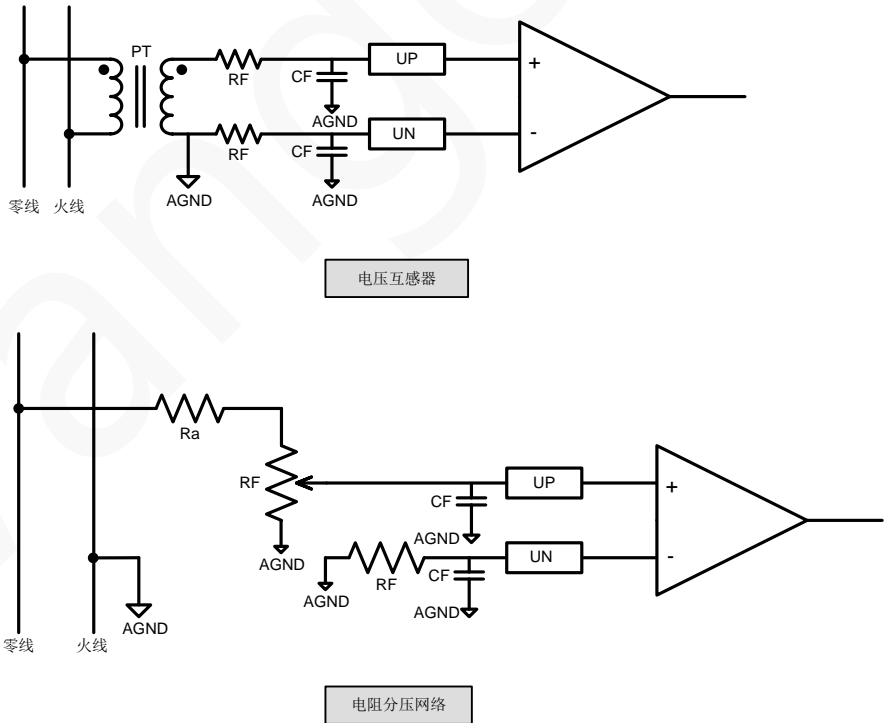


图 29-5 电压输入方式

V94XX(A)三个差分输入通道的电压输入范围是 $\pm 200\text{mV}$ （幅值），模数转换器（以下简称 ADC）的满量程输入范围是 $\pm 1.1\text{V}$ ，所以，片外输入电压与模拟增益（PGA）的乘积不能超过 $\pm 1.1\text{V}$ 。电流通道 IA 和 IB 的信号输入来自锰铜或电流传感器（CT），电压通道的信号输入来自电阻分压或电压传感器（PT），为了使传感器输出信号与 ADC 之间的量程相匹配，用户可通过设置模拟控制寄存器 1（0x01，EM\_ANA\_CTRL1）对电流和电压通道进行模拟增益配置。

表 29-39 电压/电流通道模拟增益配置

寄存器	位	默认值	说明
模拟控制寄存器 1 (0x01, EM_ANA_CTRL1)	Bit[14:12] GIB<2:0>	0	B路电流ADC增益。 000: 4; 001: 1; 010: 32; 011: 16; 100/101/110/111: 禁止 正常工作下, GIB<2:0>建议设置为000。
	Bit11 GU	0	电压ADC增益。 0: 8; 1: 4 正常工作下, 该Bit建议设置为0。
	Bit[10:8] GIA<2: 0>	0	A路电流ADC增益。 000: 32; 001: 16; 010: 4; 011: 1; 100~111: 禁止 正常工作下, GIA<2:0>建议设置为000。

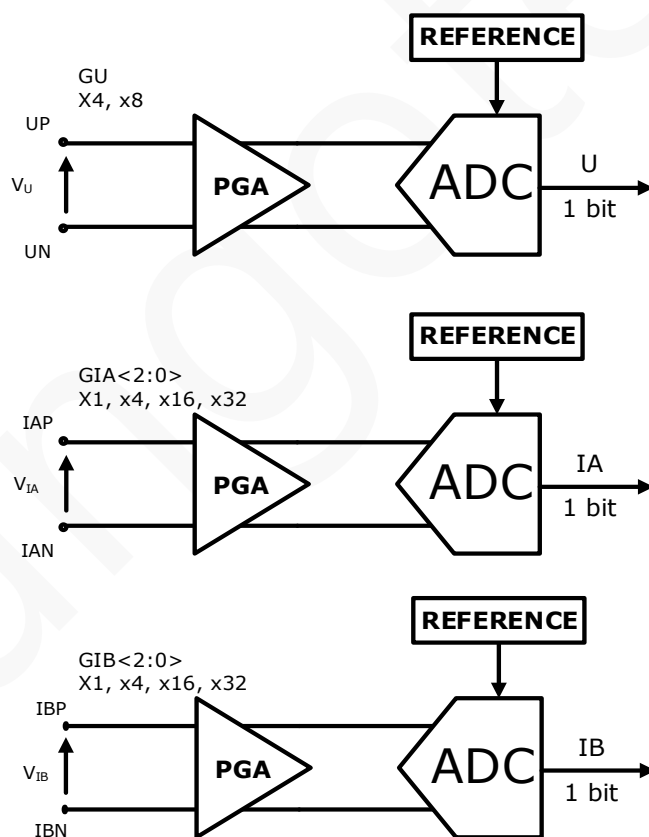


图 29-6 电流/电压通道模拟增益配置

## 29.6 模拟数字转换

V94XX(A)中的电压/电流通道 ADC 都采用的是二阶的 $\Sigma/\Delta$  ADC 结构, 其满量程输入范围是 $\pm 1100\text{mV}$ 。用户可以通过 ADCUEN、ADCIBEN、ADCIAEN (计量控制寄存器 0 (0x02, EM\_CTRL0)) 开启或关闭各路通道的 ADC。需要注意的是, 各通道 ADC 和数字处理是使用同一个控制位, 所以是同时开启和关闭的。

表 29-40 开关电压/电流通道的 ADC

寄存器	位	默认值	说明
计量控制寄存器 0 (0x02, EM_CTRL0)	Bit2 ADCUEN	0	电压通道开关 (包括ADC和DSP)。
	Bit1 ADCIBEN	0	电流B通道开关 (包括ADC和DSP)。
	Bit0 ADCIAEN	0	电流A通道开关 (包括ADC和DSP)。

## 29.7 角差校正

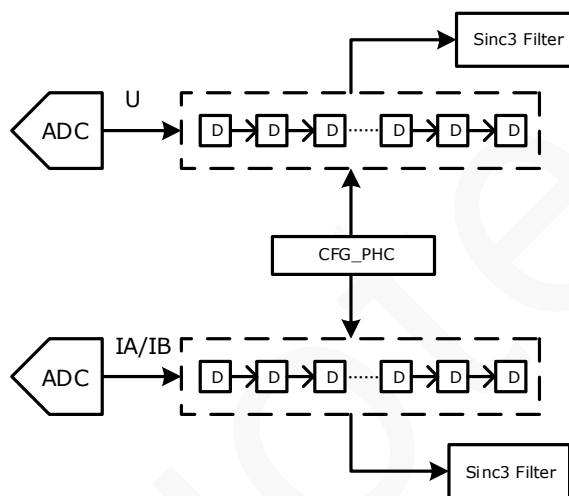


图 29-7 角差校正原理图

角差校正的原理是在电压和电流通道上引入固定长度的延迟链，根据电压和电流之间的相位超前或滞后关系，选择电压被延迟或电流被延迟，因此角差校正的最小分辨率就是延迟链上两个相邻单元之间的相位差，而总的角差校正角度是最小分辨率与延迟链的长度之积。

角差校正默认关闭。用户可通过配置角差校正寄存器（0x33, EM\_CFG\_PHC）开启角差校正功能，只要该寄存器值非零开启角差校正自动开启。

在 V94XX(A)中，当角差校正电路的采样频率（ $f_{smpl}$ ）为 3.2768MHz 时，角差校正的分辨率是 0.0055°/lsb，总校正量为 ±4.21875°。角差校正电路的采样频率（ $f_{smpl}$ ）由 EM\_MODE <3:0> 位（计量控制寄存器 0 (0x02, EM\_CTRL0)）的配置决定。

表 29-41 不同  $f_{smpl}$  下的角差分辨率和校正范围

EM_MODE	$f_{smpl}$	calibration_accuracy (degree)	Calibration_range (degree)
0x00, 0x01, 0x02	6.5536MHz	0.005493164	±4.21875
0x06, 0x07	3.2768MHz	0.010986328	±8.4375
0x08	819.2KHz	0.021972656	±16.855

IA 角差校正值  $phc\_ia$  = 待校正角度/calibration\_acuracy

IB 角差校正值  $phc\_ib$  = 待校正角度/calibration\_acuracy

IA 角差校正值和 IB 角差校正值以补码形式组合后写入角差校正寄存器（0x33, EM\_CFG\_PHC）。



表 29-42 角差校正配置

寄存器	位	默认值	说明
角差校正寄存器 (0x33, EM_CFG_PHC)	Bit[26:16] CFG_PHCB	0	角差校正寄存器。该寄存器需要参与参数配置自检校验。 B通道角差校正。 其范围为-766~767。
	Bit[10:0] CFG_PHCA	0	角差校正寄存器。该寄存器需要参与参数配置自检校验。 A通道角差校正。 其范围为-766~767。

29.8原始波形数据产生

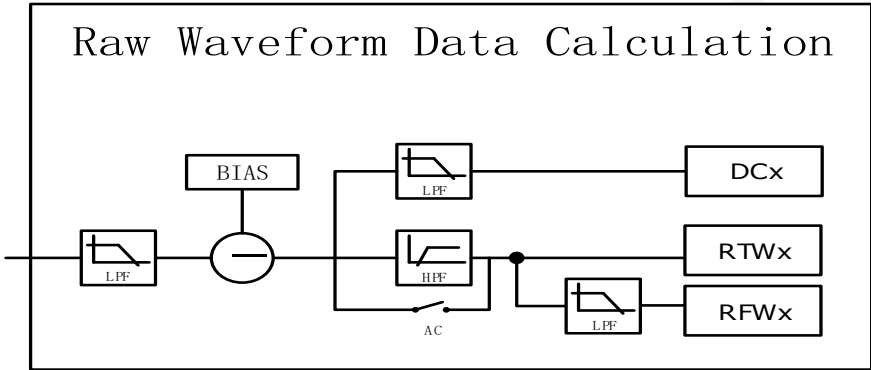


图 29-8 原始波形数据产生原理图

过采样  $\Sigma/\Delta$  ADC 输出 1bit 码流伴有大量的高频噪声，通常用低通滤波器[CIC 抽取滤波器（Decimation Filter）]来抑制该噪声，并将 ADC 采样频率降低 256 倍。

每个 ADC 通道可以产生直流原始瞬时数据和全波交流原始瞬时数据和基波交流原始瞬时数据。

当配置 DC\_METER\_MODE（计量控制寄存器 0（0x02，EM\_CTRL0））为 1 的时候，可以支持直流计量。  
用户可通过 ADCUEN、ADCIBEN、ADCIAEN（计量控制寄存器 0（0x02，EM\_CTRL0））开启或关闭各路通道的 ADC。需要注意的是，各通道 ADC 和数字处理是使用同一个控制位，所以是同时开启和关闭的。

表 29-43 开关电压/电流通道的 ADC

寄存器	位	默认值	说明
计量控制寄存器 0（0x02，EM_CTRL0）	Bit27 DC_METER_MODE	0	直流计量模式开关。 0：关闭； 1：打开
	Bit2 ADCUEN	0	电压通道开关（包括 ADC 和 DSP）。
	Bit1 ADCIBEN	0	电流 B 通道开关（包括 ADC 和 DSP）。
	Bit0 ADCIAEN	0	电流 A 通道开关（包括 ADC 和 DSP）。

经 CIC 滤波器处理的信号，再经过高通滤波器，滤除传感器和 ADC 可能引入的直流分量。在 V94XX(A)中，高通滤波器可被旁路。当用户想要更快的响应时间的时候，可以旁路高通滤波器。

用户可通过 BGP\_U（计量控制寄存器 1（0x03，EM\_CTRL1））对电压信号进行数字增益配置，从而增大小信号输入下对截断噪声的抑制能力。

表 29-44 计量通道信号数字增益配置

寄存器	位	默认值	说明
计量控制寄存器 1 (0x03, EM_CTRL1)	Bit21 PGA_U	0	电压通道数字 PGA: 0: X1; 1: X4

原始波形数据可以用于有效值计算、功率计算、线电压频率测试、相位测试、波形存储、波形数据输出、电信号监测等。

29.9有效值计算和校正

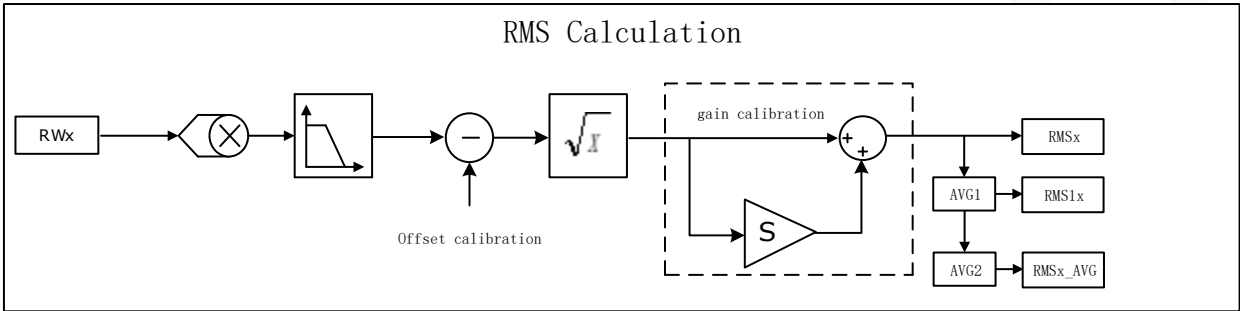


图 29-9 电流/电压有效值计算信号处理

电流/电压有效值计算的信号流如上图所示。来自电流/电压信号原始波形信号首先自相关相乘，其乘积含有 2 次谐波，经过低通滤波器后可滤除谐波和噪声引起的纹波，再经过 **Offset** 校正去除噪声后，得到电流/电压的平方，然后再做开平方运算，得到 32 位有效值数据，该数据经过比差校正后，存于瞬时电压/电流有效值寄存器。

用户可在有效值比差寄存器中设置电压/电流有效值的比差校正值。比差校正后的有效值数据存于电压/电流有效值寄存器，该数据会进行平均，平均后的值存于电压/电流有效值平均值寄存器。提供 10 个周波或 12 个周波的平均有效值用于电压闪变检测。上述所有寄存器均为 32 位补码数据。

该瞬时电流有效值可以用于后续的能量累加，瞬时电压有效值可以用于后续的电压骤升骤降。

29.10 有功功率计算和校正

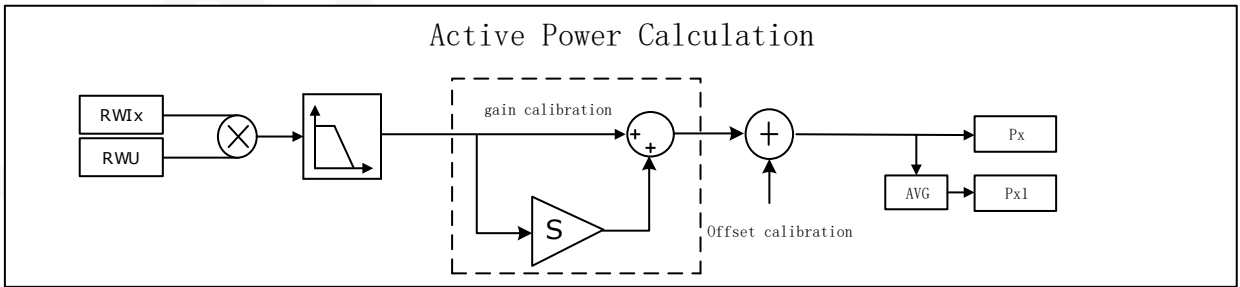


图 29-10 有功功率计算信号处理

电流和电压原始波形信号相乘，其乘积经过低通滤波器后可滤除谐波和噪声引起的纹波，得到 32 位有功功率数据，该数据经过 **offset** 校正后，再经过比差校正，存于瞬时 **PA** 或 **PB** 功率寄存器，可以参与后续的能量累加。

比差校正后的有功功率数据会进行平均，存于 **PA1** 或 **PB1** 功率平均值寄存器。



29.11 无功功率计算和校正

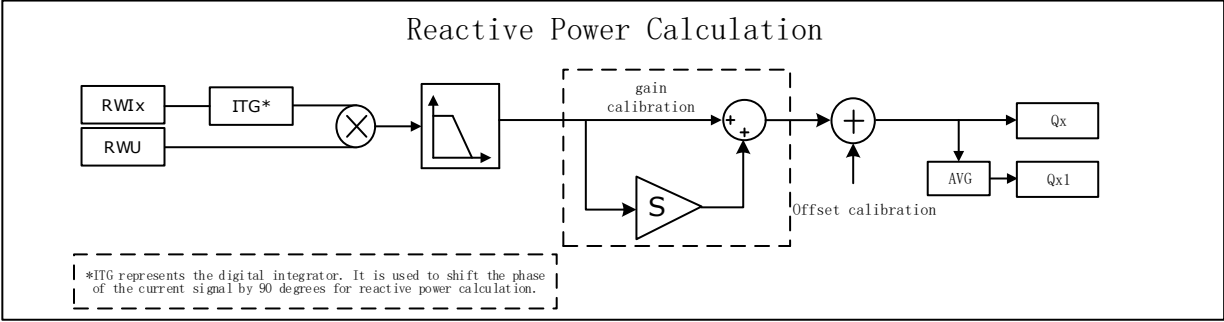


图 29-11 无功功率计算信号处理

电流原始波形信号经过希尔伯特滤波器滤波调整 90°相位后和电压原始波形信号相乘，其乘积经过低通滤波器后可滤除谐波和噪声引起的纹波，得到 32 位无功功率数据，该数据经过 offset 校正后，再经过比差校正，存于瞬时 QA 或 QB 功率寄存器，可以参与后续的能量累加。

无功功率来源是全波数据和基波数据两种，通过 QB\_MODE 和 QA\_MODE 选择（计量控制寄存器 0（0x02，EM\_CTRL0））。

表 29-45 无功功率模式选择

寄存器	位	默认值	说明
计量控制寄存器 0（0x02，EM_CTRL0）	Bit19 QB_MODE	0	无功功率 B 模式选择。 0：全波无功； 1：基波无功
	Bit18 QA_MODE	0	无功功率 A 模式选择。 0：全波无功； 1：基波无功

比差校正后的无功功率数据会进行平均，存于 QA1 或 QB1 功率平均值寄存器。

29.12 视在功率计算

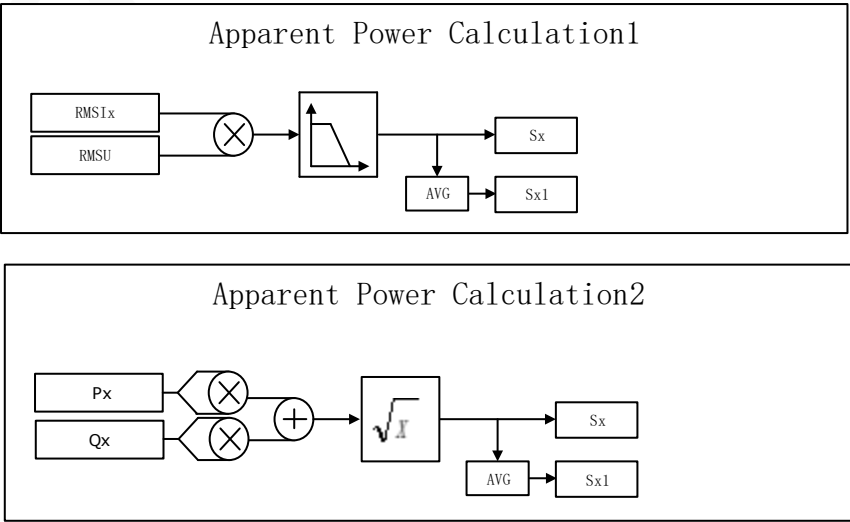


图 29-12 视在功率计算信号处理

视在功率有两种计算方法：通过有效值计算，通过功率值计算。



瞬时电压/电流有效值用于视在功率计算：

$$S = I_{rms} \times U_{rms}$$

其中，

S 为视在功率；

I<sub>rms</sub> 为瞬时电流有效值；

U<sub>rms</sub> 为瞬时电压有效值。

瞬时有功功率和无功功率用于视在功率计算：

$$S = \sqrt{P^2 + Q^2}$$

其中，

S 为视在功率；

P 为瞬时有功功率值；

Q 为瞬时无功功率值。

计算得到的瞬时视在功率，存于瞬时视在功率寄存器 S 中，可以参与后续的能量累加。瞬时视在功率经过求平均运算，得到平均视在功率，存于平均视在功率寄存器 S1 中。上述寄存器为 32 位补码数据。

视在功率计算方法，通过 S\_MODE 选择（计量控制寄存器 0（0x02，EM\_CTRL0））。

表 29-46 视在功率计算源选择

寄存器	位	默认值	说明
计量控制寄存器 0（0x02，EM_CTRL0）	Bit24 S_MODE	0	视在功率计算源选择。 0：通过有效值计算；1：通过功率值计算

29.13 功率启动潜动判断

有功功率、无功功率、视在功率支持启动潜动功能。相关寄存器见功率潜动阈值寄存器。功率潜动潜动判断采用连续 3 次平均值与阈值比较，低于下限阈值，则认为此时处于潜动状态；高于上限阈值时，进入启动状态。该功能默认关闭，可以通过配置 PWR\_CRP\_EN（计量控制寄存器 1（0x03，EM\_CTRL1））开启。

当 A 通道和 B 通道的瞬时有功功率/无功功率/视在功率增大，当连续三次高于上限阈值时，进入启动状态。当 A 通道和 B 通道的瞬时有功功率/无功功率/视在功率减少，当连续三次低于下限阈值时，进入潜动状态。

用户可通过系统状态寄存器描述（0x74，EM\_SYS\_STS）的 Bit17~Bit12 查看瞬时有功功率/无功功率/视在功率是否处于潜动状态。

表 29-47 功率启动潜动开关和状态

寄存器	位	默认值	说明
计量控制寄存器 1（0x03，EM_CTRL1）	Bit1 PWR_CRP_EN	0	功率潜动判断使能。 0：关闭潜动判断； 1：使能潜动判断
	Bit17 SBCREEP	0	B 通道视在功率潜动状态位。 0：启动状态； 1：潜动状态

寄存器	位	默认值	说明
系统状态寄存器描述（0x74，EM_SYS_STS）	Bit16 QBCREEP	0	B 通道无功功率潜动状态位。 0：启动状态； 1：潜动状态
	Bit15 PBCREEP	0	B 通道有功功率潜动状态位。 0：启动状态； 1：潜动状态
	Bit14 SACREEP	0	A 通道视在功率潜动状态位。 0：启动状态； 1：潜动状态
	Bit13 QACREEP	0	A 通道无功功率潜动状态位。 0：启动状态； 1：潜动状态
	Bit12 PACREEP	0	A 通道有功功率潜动状态位。 0：启动状态； 1：潜动状态

表 29-48 功率潜动阈值寄存器

地址	寄存器	R/W	数据格式	说明
0x55	EM_OV_THL	R/W	32-Bit 补码	A 通道和 B 通道的瞬时有功功率/无功功率/视在功率的潜动判断下限阈值
0x56	EM_OV_THH	R/W	32-Bit 补码	A 通道和 B 通道的瞬时有功功率/无功功率/视在功率的潜动判断上限阈值

29.14线电压频率测量

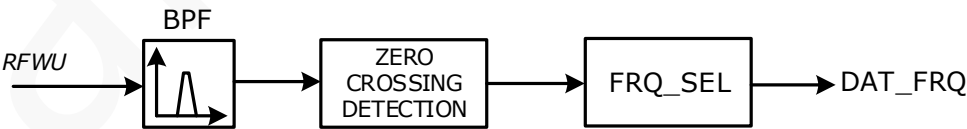


图 29-13 线电压频率测量原理

V94XX(A)支持线电压频率测量。其实现原理是电压原始基波波形信号经过一个中心频率为 50Hz 的带通滤波器（该滤波器在 150Hz 有 25dB 衰减）进行滤波，然后对滤波输出的信号做过零点检测，经过 16 个信号周期平均后输出频率测量结果。线电压频率测量的分辨率可达到 0.05Hz/lsb，频率的测量范围是 35Hz~75Hz。

频率测量值存于电网频率寄存器（0x21，EM\_DAT\_FRQ）。用户需要根据不同的 EM\_MODE（计量控制寄存器 0（0x02，EM\_CTRL0））配置，配置带通滤波器系数，具体见带通滤波器系数寄存器（0x37，EM\_CFG\_BPF）说明。

用户通过计量控制寄存器 0（0x02，EM\_CTRL0）的 FRQ\_SEL 配置频率测试累加周波个数以及 EM\_MODE 配置频率常数。

表 29-49 测试累加周波个数选择

寄存器	位	默认值	说明
计量控制寄存器 0 (0x02, EM_CTRL0)	Bit[29:28] FRQ_SEL	0	EM_DAT_FRQ 寄存器值来源 (具体时间与系统时钟的准确度有关)。 00: 16 个周波的电网频率测试值的累加值(默认是 320ms 刷新)。 01: 1 个周波的电网频率测试值(默认是 20ms 刷新)。 10: 64 个周波的电网频率测试值的累加值(默认是 1280ms 刷新)。 11: 保留。

线电压频率值直接从电网频率寄存器 (0x21, EM\_DAT\_FRQ) 中读出, 一般无需校准。根据线电压频率, 用户可直接计算得到信号频率:

$$f = \text{wave\_cnt} * \text{freq\_const} / \text{DSP\_DAT\_FRQ}$$

其中,

- f: 信号频率, Hz;
- wave\_cnt: 周波个数;
- freq\_const: 频率常数, Hz;
- EM\_DAT\_FRQ: 频率值寄存器的值 (以十进制计)。频率值寄存器的值为 16 位无符号正数。

表 29-50 频率寄存器来源 FRQ\_SEL 描述

FRQ_SEL	wave_cnt
0	16
1	1
2	64

表 29-51 频率常量 Frequency constant 描述

EM_MODE	freq_const
0x08	1600
0x06, 0x07	3200
0x0, 0x01, 0x02	6400

在信号频率增大的时候, 频率值寄存器输出值线性减小; 在信号频率减小的时候, 频率值寄存器输出值线性增大。

29.15 相位测量

V94XX(A)支持电压相位和一路电流相位测量功能。相关寄存器见表 29-24 相位测量寄存器。工作原理: 通过往寄存器 EM\_PHS\_STT (0x61) 写 1, 即启动相位测量命令, V94XX(A)开始以一定频率进行计数, 直到判断该信号过零点事件发生, 停止计数, 将此计数值写入相位寄存器, 并且记录过零点前后两个信号采样值以便于用户做插值运算, 获得更高精度的相位值。

计数频率与 EM\_MODE 有关。EM\_MODE 为 0, 1, 2 时, 用于相位测试的计数频率是 6.4KHz; EM\_MODE 为 6, 7 时, 用于相位测试的计数频率是 3.2KHz。

电流相位测量一次只支持 IA 通道或 IB 通道, 可配置。该通道选择与电流过零点输入源一致。电流过零点输入源通过 PHSI\_SEL ( 计量控制寄存器 1 (0x03, EM\_CTRL1)) 选择 IA 通道或 IB 通道。过零点事件检测方式可以通过 SIGN\_SEL ( 计量控制寄存器 1 (0x03, EM\_CTRL1)) 选择负向过零点或正向过零点。

表 29-52 过零点选择



寄存器	位	默认值	说明
计量控制寄存器 1 (0x03, EM_CTRL1)	Bit20 PHSI_SEL	0	电流过零点输入源选择: 0: 电流 IA 通道; 1: 电流 IB 通道
	Bit[19:18] SIGN_SEL	0	过零点事件检测方式选择。 0: 负向过零点 (信号从正信号变成负信号认为发生一次过零点事件)。 1: 正向过零点 (信号从负信号变成正信号认为发生一次过零点事件)。 2: 正向和负向过零点。 3: 关闭过零点检测功能。

## 29.16 波形输出和缓存

V94XX(A)波形数据可通过Px传输出去, 或者通过波形缓存存在本地。触发方式支持命令触发和事件以通过 波形输出和缓存控制寄存器 (0x07, EM\_WAVE\_CTRL) 进行主动波形输出和缓存的相关设置。

表 29-53 波形输出和缓存控制寄存器

0x07, R/W, 波形输出和缓存控制寄存器, EM_WAVE_CTRL				
位		R/W	默认值	功能说明
Bit31	WAVE_ADDR_CLR	R/W	0	复位读取波形缓存时的地址, 写 1 复位。
Bit[30:29]	WAVE_MEM_MODE	R/W	0	波形缓存工作模式: 0: 手动开始存, 存满停, 此时为单次存储。 1: 手动开始存, 事件触发停或手动停, 此时为循环存储。 2: 事件触发开始存, 存满停, 此时为单次存储。 3: 不开启。
Bit28	WAVE_MEM_EN	R/W	0	波形缓存使能。 波形缓存工作模式为 0 时, 写 1 使能一次波形缓存, 立即开始存波形, 写 0 无效果。 波形缓存工作模式为 1 时, 写 1 使能一次波形缓存, 立即开始存波形, 写 0 波形缓存立即停止存波形。 波形缓存工作模式为 2 时, 写 1 使能一次波形缓存, 等待事件触发后开启存数据, 写 0 无效果。
Bit27	U_DIP_TRIG	R/W	0	波形缓存和输出电压骤降事件触发开关。 0: 关闭; 1: 打开
Bit26	U_SWELL_TRIG	R/W	0	波形缓存和输出电压骤升事件触发开关。 0: 关闭; 1: 打开
Bit25	IB_LC_TRIG	R/W	0	波形缓存和输出 IB 欠流事件触发开关。 0: 关闭; 1: 打开
Bit24	IB_OC_TRIG	R/W	0	波形缓存和输出 IB 过流事件触发开关。 0: 关闭; 1: 打开
Bit23	IA_LC_TRIG	R/W	0	波形缓存和输出 IA 欠流事件触发开关。 0: 关闭; 1: 打开

Bit22	IA_OC_TRIG	R/W	0	波形缓存和输出 IA 过流事件触发开关。 0: 关闭; 1: 打开
Bit21	U_LV_TRIG	R/W	0	波形缓存和输出欠压事件触发开关。 0: 关闭; 1: 打开
Bit20	U_OV_TRIG	R/W	0	波形缓存和输出过压事件触发开关。 0: 关闭; 1: 打开
Bit[19:16]	WAVE_LENGTH	R/W	0	输出波形长度选择。 0: 1 个周波。 1: 2 个周波。 2: 3 个周波。 ..... 15: 16 个周波。
Bit15	保留		0	为保证系统正常工作, 必须写入默认值。
Bit14	WAVE_U_HPF_SEL	R/W	0	波形缓存和输出的电压波形是否经过高通滤波器选择。 0: 不经过; 1: 经过
Bit13	WAVE_IA_HPF_SEL	R/W	0	波形缓存和输出的电流 IA 波形是否经过高通滤波器选择。 0: 不经过; 1: 经过
Bit12	WAVE_IB_HPF_SEL	R/W	0	波形缓存和输出的电流 IB 波形是否经过高通滤波器选择。 0: 不经过; 1: 经过
Bit11	保留		0	为保证系统正常工作, 必须写入默认值。
Bit10	WAVE_U_SEL	R/W	0	波形缓存和输出的 U 通道开关, 该 Bit 决定是否存储或发送 U 通道数据。 0: 否; 1: 是
Bit9	WAVE_IA_SEL	R/W	0	波形缓存和输出的 IA 通道开关, 该 Bit 决定是否存储或发送 IA 通道数据。 0: 否; 1: 是
Bit8	WAVE_IB_SEL	R/W	0	波形缓存和输出的 IB 通道开关, 该 Bit 决定是否存储或发送 IB 通道数据 (若波形缓存配置 Bit8, Bit9, Bit10 同时使能, 该 Bit 位失效)。 0: 否; 1: 是
Bit7	SP_CHECK	R/W	0	波形输出奇偶校验选择。 0: 奇校验; 1: 偶校验
Bit6	SPI_POL	R/W	0	波形输出的 SPI 极性选择。 0: 负; 1: 正
Bit5	SPI_PHA	R/W	0	波形输出的 SPI 相位选择。 0: 负; 1: 正



Bit[4:3]	WAVE_OUT_EN	R/W	0	波形输出的开关。 0：无效； 1：手动发 2：手动停 3：保留
Bit[2:0]	WAVE_OUT_MODE	R/W	0	波形输出的工作模式： 0：手动开始发，发满 WAVE_LENGTH 个周波停。 1：手动开始发，事件触发停或手动停。 2：事件触发开始发，手动停。 3：事件触发开始发，发满 WAVE_LENGTH 个周波停。 4~7：手动开始发，手动停。 在使能传输前，必须至少打开一路波形缓存和输出的通道。

29.16.1 波形输出

V94XX(A)支持通过 SPI 接口主机模式向外部 MCU 发送最大 3 路原始波形数据。用户可以通过 波形输出和缓存控制寄存器（0x07，EM\_WAVE\_CTRL）进行主动波形输出的相关设置，通过 Px 口的 IO 配置寄存器 0（0x7D，EM\_SYS\_IOCFIG0）和 Px 口的 IO 配置寄存器 1（0x7E，EM\_SYS\_IOCFIG1）配置波形输出 IO 口 P2、P3、P6。

波形输出数据的每周波采样点数与计量控制寄存器 0（0x02，EM\_CTRL0）的 EM\_MODE（Bit[7:4]）相关；通道数量与 波形输出和缓存控制寄存器（0x07，EM\_WAVE\_CTRL）的 Bit10~8 相关。其关系如下表所示：

表 29-54 波形输出配置

EM_MODE 配置	EM_WAVE_CTRL Bit10~8(2 进制)配置	通道数量	每周波采样点数	SPI 时钟频率
0 或 10~15	111	3	128	819.2KHz
0 或 10~15	011 或 101 或 110	2	128	819.2KHz
0 或 10~15	001 或 010 或 100	1	128	409.6KHz
1 或 6	111	3	64	409.6KHz
1 或 6	011 或 101 或 110	2	64	409.6KHz
1 或 6	001 或 010 或 100	1	64	204.8KHz
2 或 7	001~111	1、2、3	32	204.8KHz

V94XX(A)可通过 Px 口以 SPI 协议向外围器件传输信号的原始波形。SPI 极性和相位可配置，当极性为 0、相位为 0 的时候传输时序如下图：

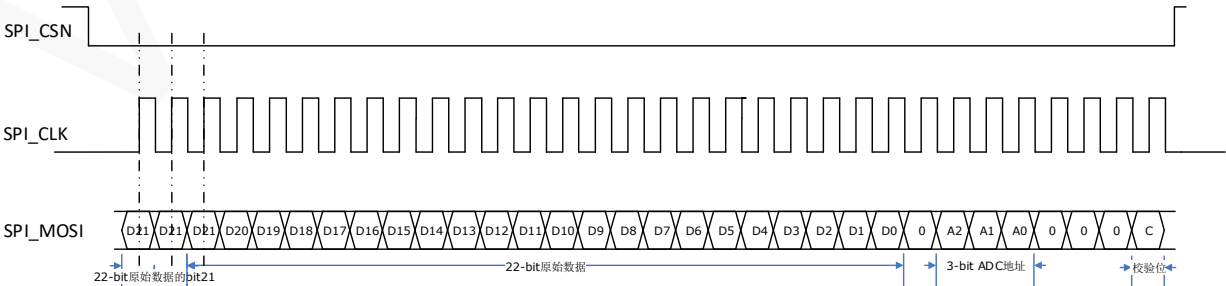


图 29-14 波形输出 SPI 传输时序

传输方式：一次完成 32-Bit 数据的传送，每次传输的数据帧格式如下表所示：

表 29-55 主动波形输出数据格式

Bit	内容
31	同 Bit29 值
30	同 Bit29 值
29:8	各通道 ADC 信号原始波形数据，22-Bit
7	0
6	指示当前波形数据是否来自电压通道 0: 否; 1: 是
5	指示当前波形数据是否来自 IA 通道 0: 否; 1: 是
4	指示当前波形数据是否来自 IB 通道 0: 否; 1: 是
3:1	000
0	奇偶校验位，校验范围为前 31 位

## 29.16.2 波形缓存

波形缓存功能开启之后，波形数据存储于 RAM 中，支持单通道波形数据存储和双通道波形数据同时存储模式。如果同时使能三个通道的波形缓存，IB 通道无效。用户可以通过 波形输出和缓存控制寄存器（0x07，EM\_WAVE\_CTRL）配置波形缓存相关配置以及开启和结束条件选择。波形缓存配置完成后，用户可以通过系统中断状态寄存器（0x72，EM\_SYS\_INTSTS）的 WAVE\_STORE 查看波形缓存是否完成。完成后，用户可以通过重复读取波形数据寄存器（0x69，EM\_DAT\_WAVE）获取波形缓存数据，每次最多可读取 309 个数据。

表 29-56 波形缓存数据格式

通道	高 16Bit	低 16Bit
IA	IADATA <sub>2n+1</sub>	IADATA <sub>2n</sub>
IB	IBDATA <sub>2n+1</sub>	IBDATA <sub>2n</sub>
U	UDATA <sub>2n+1</sub>	UDATA <sub>2n</sub>
IA+IB	IBDATA <sub>n</sub>	IADATA <sub>n</sub>
IA+U	IADATA <sub>n</sub>	UDATA <sub>n</sub>
IB+U	IBDATA <sub>n</sub>	UDATA <sub>n</sub>
IA+IB+U（此时 IB 自动失效）	IADATA <sub>n</sub>	UDATA <sub>n</sub>

其中：n 值范围：0~308

## 29.17 电信号监测

### 29.17.1 过零点检测

V94XX(A)支持电压通道过零点检测和电流通道（可通过计量控制寄存器 1（0x03，EM\_CTRL1）的 Bit20 选择过零点通道为 IA 或者 IB 通道）过零点检测。可通过计量控制寄存器 1（0x03，EM\_CTRL1）的 Bit19~Bit18 选择过零点方向。当电压/电流通道信号发生过零点事件时，系统中断状态寄存器（0x72，EM\_SYS\_INTSTS）电压过零点标志位 USIGN/电流过零点标志位 ISIGN 会置 1。用户需要写 1 清零。

表 29-57 过零点选择



寄存器	位	默认值	说明
计量控制寄存器 1 (0x03, EM_CTRL1)	Bit20 PHSI_SEL	0	电流过零点输入源选择: 0: 电流 IA 通道; 1: 电流 IB 通道
	Bit[19:18] SIGN_SEL	0	过零点事件检测方式选择。 0: 负向过零点 (信号从正信号变成负信号认为发生一次过零点事件)。 1: 正向过零点 (信号从负信号变成正信号认为发生一次过零点事件)。 2: 正向和负向过零点。 3: 关闭过零点检测功能。

使能电压/电流过零点中断输出时, 即系统中断使能寄存器 (0x73, SYS\_INTEN) 的 USIGN/ISIGN 置 1, 通过配置 Px 口的 IO 配置寄存器 0 (0x7D, EM\_SYS\_IOCFIG0) 或 Px 口的 IO 配置寄存器 1 (0x7E, EM\_SYS\_IOCFIG1) 配置电压/电流过零点中断输出。引脚 Px 的输出电平自动根据电压过零点标志位 USIGN/电流过零点标志位 ISIGN 翻转。

过零点检测精度与采样点数选择有关, 如果是一个周波 256 点, 则精度为 360/256。

通过配置 Px 口的 IO 配置寄存器 0 (0x7D, EM\_SYS\_IOCFIG0) 或 Px 口的 IO 配置寄存器 1 (0x7E, EM\_SYS\_IOCFIG1) 配置电压/电流过零点输出方波。引脚 Py 的输出电平自动根据电压/电流过零点状态实时翻转。每发生一次过零点事件, IO 口翻转一次。

下图配置为使能过零点中断输出, 过零点检测方式选择为负向过零点时, USIGN/ISIGN 标志位、过零点中断输出以及过零点输出方波波形。

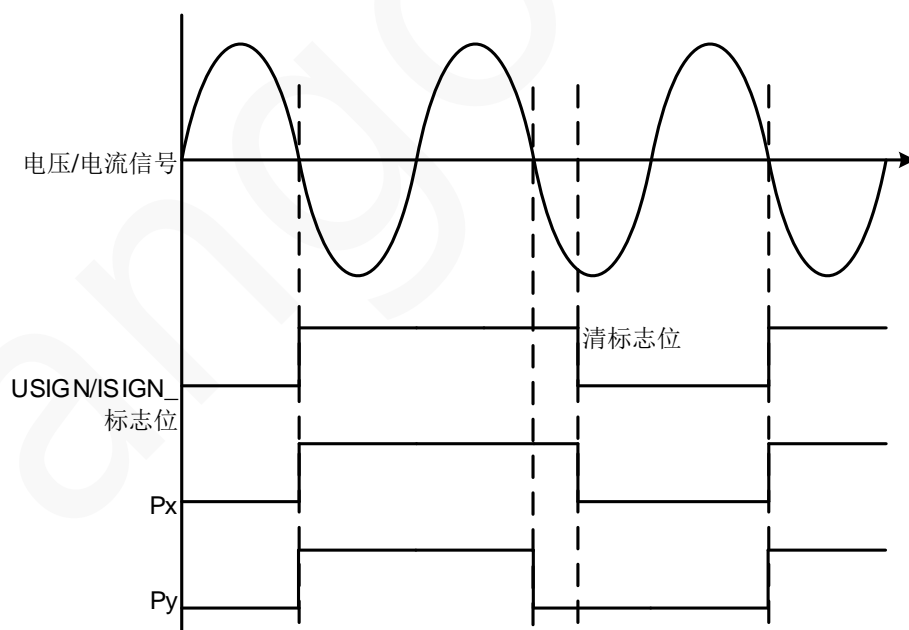


图 29-15 电压/电流过零点输出图

## 29.17.2 电压骤升骤降

V94XX(A)可以通过编程的方式来指示电压骤升骤降。

表 29-58 电压骤升骤降寄存器

地址	寄存器	R/W	数据格式	说明
0x57	EM_SWELL_THL	R/W	32-Bit 补码	电压骤升下限阈值。
0x58	EM_SWELL_THH	R/W	32-Bit 补码	电压骤升上限阈值。
0x59	EM_DIP_THL	R/W	32-Bit 补码	电压骤降下限阈值
0x5A	EM_DIP_THH	R/W	32-Bit 补码	电压骤降上限阈值
0x6A	EM_DAT_SWELL_CNT	R/C	32-Bit 补码	电压骤升时间记录，半波为单位。24Bit 有效。向该寄存器写任意值，可清零该计数值。
0x6B	EM_DAT_DIP_CNT	R/C	32-Bit 补码	电压骤降时间记录，半波为单位。24Bit 有效。向该寄存器写任意值，可清零该计数值。

当电压瞬时有效值高于电压骤升阈值上限时，电压骤升状态位（系统状态寄存器描述（0x74，EM\_SYS\_STS）的 USWELL）置为 1。同时电压骤升标志位（系统中断状态寄存器（0x72，EM\_SYS\_INTSTS）的 USWELL）置为 1，该标志位写 1 清零。

当电压瞬时有效值低于电压骤升阈值下限时，电压骤升状态位（系统状态寄存器描述（0x74，EM\_SYS\_STS）的 USWELL）恢复为 0。

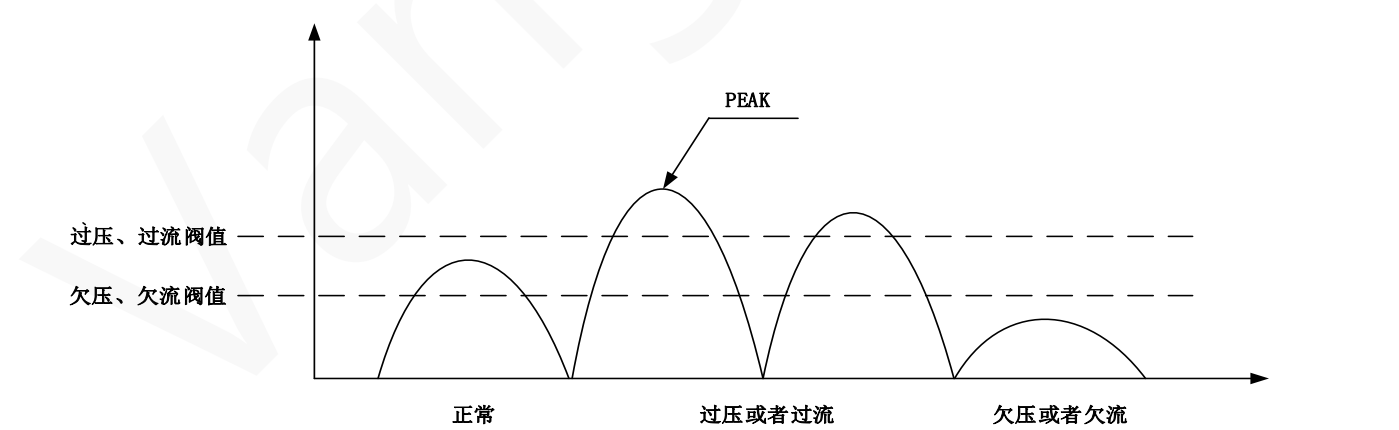
当电压瞬时有效值低于电压骤降阈值下限时，电压骤降状态位（系统状态寄存器描述（0x74，EM\_SYS\_STS）的 UDIP）置为 1。同时电压骤降标志位（系统中断状态寄存器（0x72，EM\_SYS\_INTSTS）的 UDIP）置为 1，该标志位写 1 清零。

当电压瞬时有效值高于电压骤降阈值上限时，电压骤降状态位（系统状态寄存器描述（0x74，EM\_SYS\_STS）的 UDIP）恢复为 0。

同时可以通过读寄存器 EM\_DAT\_SWELL\_CNT/EM\_DAT\_DIP\_CNT 获取电压骤升/骤降时间记录，半波为单位。24Bit 有效。向该寄存器写任意值，可清零该计数值。

电压骤升中断和电压骤降中断标志可以通过配置 IO 口输出，具体见表 29-8 Px 口的 IO 配置寄存器 0（0x7D，EM\_SYS\_IOCFG0）和表 29-10 Px 口的 IO 配置寄存器 1（0x7E，EM\_SYS\_IOCFG1）。

29.17.3 过压欠压和过流欠流



V94XX(A)可以通过编程的方式来指示 U 通道过压欠压、IA 通道过流欠流以及 IB 通道过流欠流。相关寄存器见表 29-27 快速检测阈值寄存器。用户可以通过快速检测控制寄存器（0x06，EM\_FD\_CTRL）的 FDUEN、FDIAEN、FDIBEN 分别开启 U 通道过压欠压、IA 通道过流欠流以及 IB 通道过流欠流检测。检测源支持高通滤波器和不过高通滤波器两种。过压或欠压/过流或欠流检测时间长度可选择。具体说明如下：

- 1) 支持对三路 ADC 进行波形监测。
- 2) 每一路 ADC 波形监测具有两个阈值：上限阈值（过压、过流）；下限阈值（欠压、欠流）
- 3) 超上限阈值采样点数：比如设置值为 4，表示如果半个周波采样点中有大于 4 个点超过阈值，则认为该半周波波形超限。

4) 超上限阈值半周波数: 比如设置为 2, 表示如果连续有两个半周波都超限, 则认为过压或者过流事件发生。

5) 低于下限阈值采样点数: 比如设置为 4, 如果发现半个周波采样点中小于或等于 4 个点高于下限阈值, 则认为该半波低于下限。

6) 低于下限值半波数: 比如设置为 2, 表示如果连续有 1 个周波都低于下限, 则认为欠压或者欠流事件发生。

7) 当事件发生时, 产生标志位, 用户可以通过事件相应状态位 (系统状态寄存器描述 (0x74, EM\_SYS\_STS)) 和标志位 (系统中断状态寄存器 (0x72, EM\_SYS\_INTSTS)) 查看。

8) 相应事件标志可以通过配置中断使能以及 IO 口输出。

9) 响应时间: 当 IPERIOD 设置为 0 时, 开启或关闭高通滤波器的情况下, 响应时间: 10ms (与系统时钟的准确度有关), 即当输入信号超过阈值时, 半个周波之后能输出事件中断。

表 29-59 过压欠压和过流欠流控制

0x06, R/W, 快速检测控制寄存器, EM_FD_CTRL				
位		R/W	默认值	功能说明
Bit[31:30]	IPERIOD	R/W	0	过流或欠流检测时间长度选择。当某个半周波内的采样点数中发生过流或欠流的采样点数大于等于 ITH 值, 则认为这个周波是有效周波。如果连续有效周波数量达到 IPERIOD 设定的值, 则认为事件发生。 0: 半个周波。 1: 1 个周波。 2: 2 个周波。 3: 4 个周波。
Bit[29:24]	ITH	R/W	0	半周波判断为有效半周波的阈值。 0: 1 个采样点。 1: 2 个采样点。 ... 63: 64 个采样点。
Bit[23:22]	UPERIOD	R/W	0	配置情况与 IPERIOD 一致。 0: 半个周波。 1: 1 个周波。 2: 2 个周波。 3: 4 个周波。
Bit[21:16]	UTH	R/W	0	配置情况与 ITH 一致。 0: 1 个采样点。 1: 2 个采样点。 ... 63: 64 个采样点。
Bit[15:10]	保留		0	为保证系统正常工作, 必须写入默认值。
Bit9	IBLCSEL	R/W	0	IB 欠流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit8	IBOCSEL	R/W	0	IB 过流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit7	IALCSEL	R/W	0	IA 欠流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit6	IAOCSEL	R/W	0	IA 过流检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit5	ULVSEL	R/W	0	U 通道欠压检测源是否经过高通滤波器。 0: 不经过; 1: 经过

Bit4	UOVSEL	R/W	0	U 通道过压检测源是否经过高通滤波器。 0: 不经过; 1: 经过
Bit3	保留		0	为保证系统正常工作, 必须写入默认值。
Bit2	FDIBEN	R/W	0	IB 过流欠流检测的开关。 0: 关闭; 1: 开启
Bit1	FDIAEN	R/W	0	IA 过流欠流检测的开关。 0: 关闭; 1: 开启
Bit0	FDUEN	R/W	0	U 通道过压欠压检测的开关。 0: 关闭; 1: 开启

表 29-60 快速检测阈值寄存器

地址	寄存器	R/W	数据格式	说明
0x5B	EM_FD_OVTH	R/W	30-Bit 补码	快速检测过压阈值。位宽为30Bit。
0x5C	EM_FD_LVTH	R/W	30-Bit 补码	快速检测欠压阈值。位宽为30Bit。
0x5D	EM_FD_IA_OCTH	R/W	30-Bit 补码	快速检测电流A通道过流阈值。位宽为30Bit。
0x5E	EM_FD_IA_LCTH	R/W	30-Bit 补码	快速检测电流A通道欠流阈值。位宽为30Bit。
0x5F	EM_FD_IB_OCTH	R/W	30-Bit 补码	快速检测电流B通道过流阈值。位宽为30Bit。
0x60	EM_FD_IB_LCTH	R/W	30-Bit 补码	快速检测电流B通道欠流阈值。位宽为30Bit。

## 29.18 能量桶

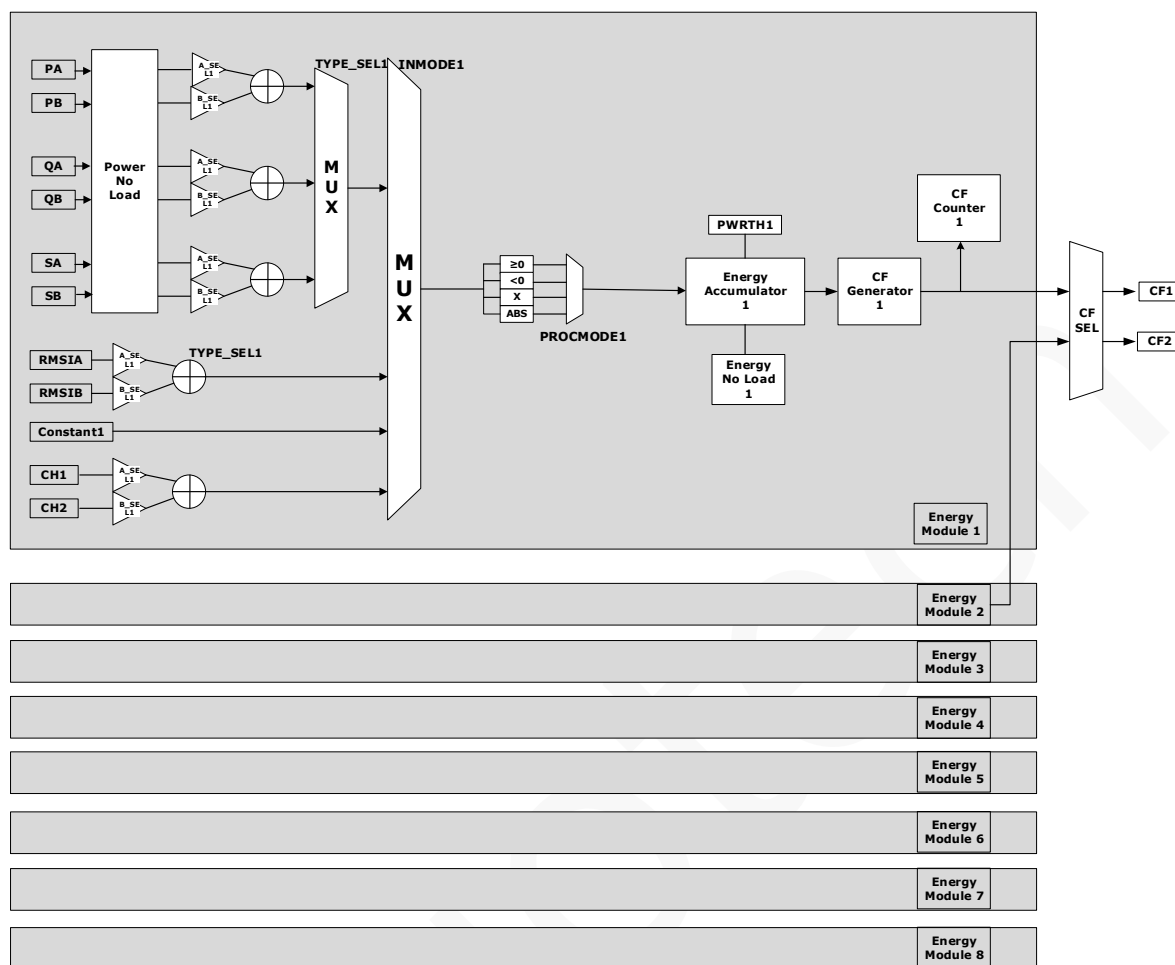


图 29-16 能量桶功能框图

V94XX(A)有 8 个能量桶，包括 2 个高速能量桶和 6 个低速能量桶。每个能量桶有 4 种累加模式：功率累加、电流有效值累加、常数累加和可配置基波通道累加。

功率累加，通过 A 通道累加开关（A\_SEL）和 B 通道累加开关（B\_SEL）控制，可以实现仅累加 A 通道功率、仅累加 B 通道功率、累加 A 和 B 两通道功率和。功率可以通过 TYPE\_SEL 选择有功功率、无功功率和视在功率三种。

电流有效值累加，通过 A 通道累加开关（A\_SEL）和 B 通道累加开关（B\_SEL）控制，可以实现仅累加 IA 通道有效值、仅累加 IB 通道有效值、累加 IA 和 IB 两通道有效值。其中，累加 IA 和 IB 两通道有效值可以通过 TYPE\_SEL 选择累加两通道的累加和或累加差（RMSIA+RMSIB 或 RMSIA-RMSIB）。

常数累加不受 A/B 通道累加开关影响。

可配置基波通道累加，通过 A 通道累加开关（A\_SEL）和 B 通道累加开关（B\_SEL）控制，可以实现仅累加基波通道 1 数据、仅累加基波通道 2 数据、累加基波通道 1 和 2 数据累加和。

对于每一个送入能量桶累加的数据运算的种类有四种，以有累加两个通道有功功率为例，运算方法如下所示。

0：能量桶只累加正数。只累加  $PA+PB>0$  的数据。如果  $PA+PB<0$ ，不累加。

1：能量桶只累加负数（此时，实际累加值为原始值转换的正数）。只累加  $PA+PB<0$  的数据（ $\text{abs}(PA+PB)$ ）。如果  $PA+PB>0$ ，不累加。

2：能量桶累加原始值。累加  $PA+PB$ 。

3：能量桶累加绝对值。累加  $\text{abs}(PA+PB)$ 。

当使能量累加和能量脉冲计数时，V94XX(A)根据能量桶配置将待累加数据以一定频率累加到能量累加寄存器，当能量累加寄存器的值超过相对应的能量脉冲门限值时，在对应的能量脉冲计数器中加 1，并从能量累加寄存器累积数据中减去一个能量脉冲门限值。

表 29-61 能量桶模式配置

寄存器	位	默认值	说明
EM_EGY_CTRL0 寄存器 EM_EGY_CTRL1 寄存器	INMODEx[1:0]	0	能量桶 x 累加模式。 0: 功率累加。 1: 电流有效值累加。 2: 常数累加。 3: 可配置基波通道累加。
	A_SELx	0	能量桶 x A 通道累加开关。 0: 关闭; 1: 打开
	B_SELx	0	能量桶 x B 通道累加开关。 0: 关闭; 1: 打开
	TYPE_SELx	0	能量桶 x 多通道累加模式下, 累加源选择。 功率累加时: 0、3: 有功功率累加。 1: 无功功率累加。 2: 视在功率累加。 有效值累加时: 高位为 0 表示累加和, 高位为 1 表示累加差。
	PROCMODEx	0	对于每一个送入能量桶 x 累加的信号的种类选择。 0: 能量桶只累加正数。 1: 能量桶只累加负数 (此时, 实际累加值为原始值转换的正数)。 2: 能量桶累加原始值。 3: 能量桶累加绝对值。

表 29-62 能量桶寄存器

地址	寄存器	R/W	数据格式	说明
0x39	EM_EGY_CRPTH	R/W	32-Bit 无符号数	能量桶防潜阈值。当防潜能量桶累加值超过 EM_EGY_CRPTH 并且高速能量桶累加值未超过 EM_EGY_PWRTH 时, 高速能量桶的累加值会被清除。
0x3A	EM_EGY_PWRTH	R/W	32-Bit 无符号数	能量桶累加阈值。由于能量桶为 46Bit, 实际高速能量桶累加阈值为该阈值*16384; 低速能量桶累加阈值为该阈值*4。
0x3B	EGY_CONST1	R/W	32-Bit 无符号数	能量桶1累加常数
0x3C	EGY_OUT1L	R/W	32-Bit 无符号数	能量桶1累加值低位
0x3D	EGY_OUT1H	R/W	32-Bit 无符号数	能量桶1累加值高位 低14Bit有效
0x3E	EGY_CFCNT1	R	32-Bit 无符号数	能量桶1脉冲计数器
0x3F	EGY_CONST2	R/W	32-Bit 无符号数	能量桶2累加常数
0x40	EGY_OUT2L	R/W	32-Bit 无符号数	能量桶2累加值低位



0x41	EGY_OUT2H	R/W	32-Bit 无符号数	能量桶2累加值高位 低14Bit有效
0x42	EGY_CFCNT2	R	32-Bit 无符号数	能量桶2脉冲计数器
0x43	EGY_CONST3	R/W	32-Bit 无符号数	能量桶3累加常数
0x44	EGY_OUT3	R/W	32-Bit 无符号数	能量桶3累加值
0x45	EGY_CFCNT3	R	32-Bit 无符号数	能量桶3脉冲计数器
0x46	EGY_CONST4	R/W	32-Bit 无符号数	能量桶4累加常数
0x47	EGY_OUT4	R/W	32-Bit 无符号数	能量桶4累加值
0x48	EGY_CFCNT4	R	32-Bit 无符号数	能量桶4脉冲计数器
0x49	EGY_CONST5	R/W	32-Bit 无符号数	能量桶5累加常数
0x4A	EGY_OUT5	R/W	32-Bit 无符号数	能量桶5累加值
0x4B	EGY_CFCNT5	R	32-Bit 无符号数	能量桶5脉冲计数器
0x4C	EGY_CONST6	R/W	32-Bit 无符号数	能量桶6累加常数
0x4D	EGY_OUT6	R/W	32-Bit 无符号数	能量桶6累加值
0x4E	EGY_CFCNT6	R	32-Bit 无符号数	能量桶6脉冲计数器
0x4F	EGY_CONST7	R/W	32-Bit 无符号数	能量桶7累加常数
0x50	EGY_OUT7	R/W	32-Bit 无符号数	能量桶7累加值
0x51	EGY_CFCNT7	R	32-Bit 无符号数	能量桶7脉冲计数器
0x52	EGY_CONST8	R/W	32-Bit 无符号数	能量桶8累加常数
0x53	EGY_OUT8	R/W	32-Bit 无符号数	能量桶8累加值
0x54	EGY_CFCNT8	R	32-Bit 无符号数	能量桶8脉冲计数器

## 29.18.1 高速能量桶

开启高速能量桶 1 需要配置 EM\_CTRL1 中的 CALCEN1 (Bit[6])，开启高速能量桶 2 需要配置 EM\_CTRL1 中的 CALCEN2 (bit[7])。高速能量桶默认累加速度是 204.8KHz，也可以通过能量桶时钟 (EM\_CTRL1 中的 bit23) 选择 32768 Hz。

高速能量桶支持 CF 输出。CF 输出通过配置 EM\_SYS\_IOCFCGX 寄存器选择 IO 口输出。

## 29.18.2 低速能量桶

6 个低速能量桶开关是 EM\_CTRL1 中的 DGY\_LC\_EN (bit[15]) 控制，默认累加速度是 50Hz。

低速能量桶累加速度受 EM\_CTRL0 中的 CURDAT\_RATE (bit[31]) 和 EM\_CTRL1 中的 LCF\_ACC (bit[22]) 控制。

- CURDAT\_RATE=0 时：

LCF\_ACC=0，能量桶 3、4、5、6、7、8 累加周期是 20 ms (与系统时钟的准确度有关)；

LCF\_ACC=1，能量桶 3、4、5 累加周期是 10ms (与系统时钟的准确度有关)，能量桶 6、7、8 不累加。

- CURDAT\_RATE=1 时：

LCF\_ACC=0，能量桶 3、4、5、6、7、8 累加周期是 40 ms (与系统时钟的准确度有关)；

LCF\_ACC=1，能量桶 3、4、5 累加周期是 20 ms (与系统时钟的准确度有关)，能量桶 6、7、8 不累加。

## 29.18.3 能量桶启动潜动判断

能量桶 1 和能量桶 2 分别有一个潜动能量累加寄存器。当能量桶启动潜动使能后，该潜动能量累加寄存器的输入固定为 1。其累加速率与能量桶 1 和能量桶 2 累加速率相等。

用户应分别在启动/潜动判断门限值寄存器 (EM\_EGY\_CRPTH) 和能量累加门限值寄存器 (EM\_EGY\_PWRTH) 设置各自的门限值。如果潜动能量桶累加寄存器的累加值先达到 EM\_EGY\_CRPTH 值时，能量累加寄存器被清空，系统进入潜动状态。当能量累加寄存器的累加值先达到 EM\_EGY\_PWRTH 值时，潜动能量累加寄存器被清空，系统进入启动状态，正常工作。

寄存器 EM\_EGY\_CRPTH 的实际位宽为 32 位，但在参与启动/潜动的判断运算时，寄存器的内容会被自动在低位补 4 个 0，扩展到 36 位后再参与运算。

用户可通过 EM\_SYS\_STS (BIT19/BIT18) 判断是否处于潜动状态。

表 29-63 能量启动潜动开关和状态

寄存器	位	默认值	说明
计量控制寄存器 1 (0x03, EM_CTRL1)	Bit0 EGY_CRP_EN	0	高速能量桶防潜开关。 0: 关闭潜动判断; 1: 使能潜动判断
系统状态寄存器描述 (0x74, EM_SYS_STS)	Bit19 CRP_OUT2	0	能量桶 2 潜动状态位。 0: 启动状态; 1: 潜动状态
	Bit18 CRP_OUT1	0	能量桶 1 潜动状态位。 0: 启动状态; 1: 潜动状态



## 29.19 CF 输出

V94XX(A)支持 2 路 CF。CF 输出通过配置 EM\_SYS\_IOCFCGX 寄存器选择 IO 口输出。

当能量桶 1 或能量桶 2 使能能量累加和能量脉冲计数时，V94XX(A)根据能量桶配置将待累加数据以一定频率累加到能量累加寄存器，当能量累加寄存器的值超过相对应的能量脉冲门限值时，在对应的能量脉冲计数器中加 1，并从能量累加寄存器累积数据中减去一个能量脉冲门限值，如果使能了 CF 输出，此时会产生一个能量脉冲。

CF 支持来源选择。可以选择来源于能量桶 1 或者能量桶 2。

CF 支持极性选择、脉冲宽度选择、加速小信号校表。

详细信息参考计量控制寄存器 1（0x03，EM\_CTRL1）中关于 CF 的描述。

表 29-64 CF 模式配置

寄存器	位	默认值	说明
计量控制寄存器 1 (0x03, EM_CTRL1)	Bit14 CF2_INV	0	CF2 极性控制位。 0: 原始极性; 1: 反向极性
	Bit13 CF2_EN	0	CF2 输出使能位。 0: 关闭; 1: 开启
	Bit12 CF2_SEL	0	CF2 的输入源选择 0: 来源于能量桶 1。 1: 来源于能量桶 2。
	Bit10 CF1_INV	0	CF1 极性控制位。 0: 原始极性。 1: 反向极性。
	Bit9 CF1_EN	0	CF1 输出使能位。 0: 关闭; 1: 开启
	Bit8 CF1_SEL	0	CF1 的输入源选择 0: 来源于能量桶 1。 1: 来源于能量桶 2。
	Bit[5:4] CF_PULSE	0	CF 脉冲宽度选择（具体时间与系统时钟的准确度有关）。 0: 80ms; 1: 40ms; 2: 20ms; 3: 10ms
	Bit[3:2] CF_FAST_EN	0	CF 脉冲加速产生。 0: 正常; 1: 4 倍速; 2: 8 倍速; 3: 16 倍速

29.20 信号输出口

V94XX(A)最多有 5 个信号输出口，5 个信号输出口可以配置为 CF 输出、波形输出接口、过零点方波以及四大种类中断输出。信号输出口可以设置为输出某个单独信号，也可以设置为输出某几类中断信号。具体见 EM\_SYS\_IOCFCGX 寄存器。

- 1 类中断（过零点中断输出）：电流过零点中断、电压过零点中断。
- 2 类中断（数据更新指示）：波形刷新中断、瞬时有效值刷新中断、平均有效值刷新中断、瞬时功率值刷新中断、平均功率值刷新中断、波形缓存完成中断、波形缓存溢出中断、波形输出完成中断。
- 3 类中断（电信号监测输出）：电流 IB 通道欠流中断、电流 IB 通道过流中断、电流 IA 通道欠流中断、电流 IA 通道过流中断、电压通道欠压中断、电压通道过压中断、电压骤降中断、电压骤升中断。
- 4 类中断（系统状态报警）：参数自检错误中断、相位测量完成中断、掉电中断、电能计量模块的 Reference 错误中断、电能计量模块的 6.5M 高频时钟缺失中断、RAM 自检错误中断。

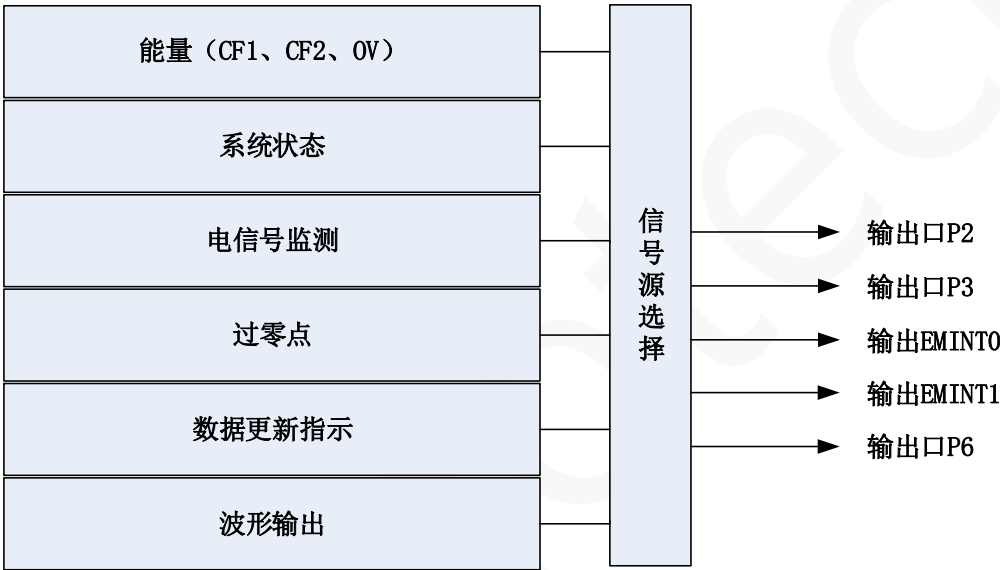


图 29-17 信号输出口功能框图

说明：

- 1) 当输出口作为 CF 输出时，支持设置输出脉冲的宽度，选择输出极性。当输出周期小于两倍 CF 输出脉冲宽度时，CF 按照占空比 50%输出。比如，输出宽度为 80ms，输出周期小于 160ms 时，则按照占空比 50%输出。
- 2) 作为中断事件输出口时，输出口默认为低电平输出，如果事件发生，则输出高电平，直到用户清除事件标志位，输出状态才恢复默认电平。
- 3) 当作为过零点方波输出时，输出口默认为低电平输出，如果配置为正向过零点，则当信号从负信号过渡到正信号时，IO 口翻转。
- 4) 当作为波形输出口时，使用 SPI 协议，需要用户选择 SPI\_CLK、SPI\_MOSI 和 SPI\_CSN 口。

表 29-65 Px 口的 IO 配置寄存器 0 (0x7D, EM\_SYS\_IOCFCG0)

0x7D, Px 口的 IO 配置寄存器 0, EM_SYS_IOCFCG0				
位		R/W	默认值	功能说明
Bit[31:24]	P3CFG	R/W	0	配置同 P2CFG
Bit[23:16]	P2CFG	R/W	0	Bit7~Bit6: 00: 如下表所示, Bit5~Bit0 可配置为不同组合中断 10: CF1 输出, Bit5~Bit0 可配置任意值 01: CF2 输出, Bit5~Bit0 可配置任意值

				11: 保留
Bit[15:8]	保留		0	为保证系统正常工作, 必须写入默认值。
Bit[7:0]	保留		0	为保证系统正常工作, 必须写入默认值。

表 29-66 PxCFG Bit5~Bit0 说明

Bit5~Bit3	Bit2~Bit0	功能
0	0	高阻态
0	1	电流过零点中断
0	2	电压过零点中断
0	3	电流过零点输出方波
0	4	电压过零点输出方波
0	5	高速能量桶 1 溢出中断
0	6	高速能量桶 2 溢出中断
0	7	1 类中断
1	0	波形刷新中断
1	1	瞬时有效值刷新中断
1	2	平均有效值刷新中断
1	3	瞬时功率刷新中断
1	4	平均功率刷新中断
1	5	波形缓存完成中断
1	6	波形缓存地址溢出中断
1	7	波形输出完成中断
2	0	电流 IB 通道欠流中断
2	1	电流 IB 通道过流中断
2	2	电流 IA 通道欠流中断
2	3	电流 IA 通道过流中断
2	4	电压通道欠压中断
2	5	电压通道过压中断
2	6	电压骤降中断
2	7	电压骤升中断
3	0	电能计量模块 Reference 错误中断
3	1	电能计量模块的 6.5M 高频时钟缺失中断
3	2	保留
3	3	保留
3	4	掉电中断
3	5	参数自检错误中断
3	6	相位测试完成中断
3	7	RAM 自检错误中断
4	0	1 类中断

4	1	2 类中断
4	2	3 类中断
4	其他	4 类中断
5	0	3 类中断
5	1	1 类中断和 2 类中断
5	2	1 类中断和 3 类中断
5	3	1 类中断和 4 类中断
5	4	2 类中断和 3 类中断
5	5	2 类中断和 4 类中断
5	6	3 类中断和 4 类中断
5	7	所有中断
6	0	1 类中断、2 类中断和 3 类中断
6	1	1 类中断、2 类中断和 4 类中断
6	2	1 类中断、3 类中断和 4 类中断
6	3	2 类中断、3 类中断和 4 类中断
6	其他	所有中断
7	1	波形输出 SPI 片选 SPI_CSN
7	2	波形输出 SPI 时钟 SPI_CLK
7	4	波形输出 SPI 数据 SPI_MOSI
7	其他	禁止输出

其中，IO 口在不配置时（即全 0），输出高阻。

1 类中断：电流过零点中断、电压过零点中断、高速能量桶 1/2 溢出中断。

2 类中断：波形刷新中断、瞬时有效值刷新中断、平均有效值刷新中断、瞬时功率值刷新中断、平均功率值刷新中断、波形缓存完成中断、波形缓存溢出中断、波形输出完成中断。

3 类中断：电流 IB 通道欠流中断、电流 IB 通道过流中断、电流 IA 通道欠流中断、电流 IA 通道过流中断、电压通道欠压中断、电压通道过压中断、电压骤降中断、电压骤升中断。

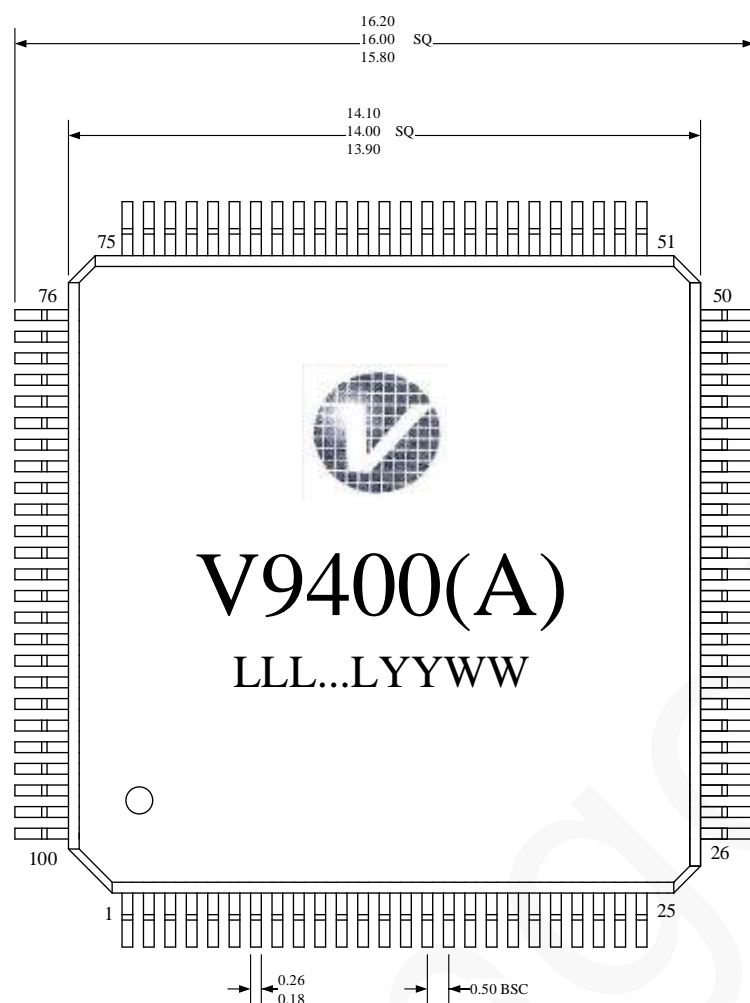
4 类中断：参数自检错误中断、相位测量完成中断、掉电中断、Reference 错误中断、电能计量模块的 6.5M 高频时钟缺失中断、RAM 自检错误中断。

表 29-67 Px 口的 IO 配置寄存器 1 (0x7E, EM\_SYS\_IOCFCG1)

0x7E, Px 口的 IO 配置寄存器 1, EM_SYS_IOCFCG1				
位		R/W	默认值	功能说明
Bit[31:24]	保留		0	为保证系统正常工作，必须写入默认值。
Bit[23:16]	P6CFG	R/W	0	配置同 P2CFG
Bit[15:8]	EMINT1	R/W	0	配置同 P2CFG
Bit[7:0]	EMINT0	R/W	0	配置同 P2CFG

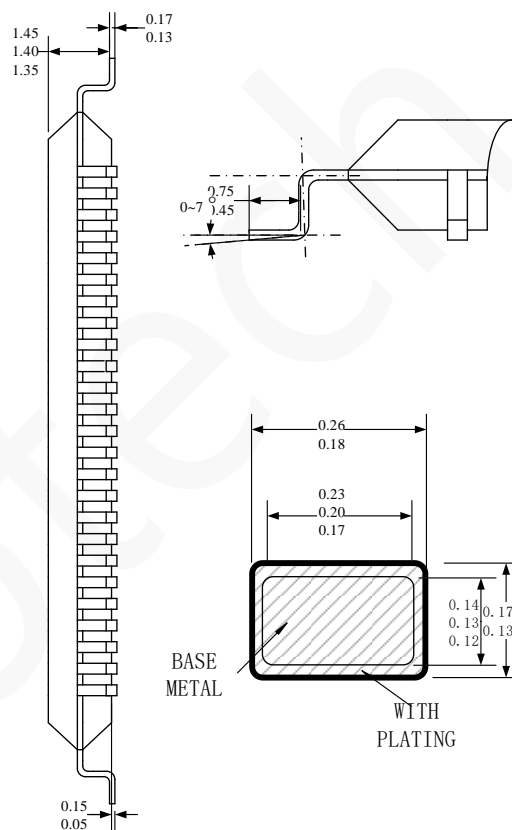
## 第30章 封装尺寸图

### 30.1 V9400(A)封装尺寸图



100Pin-LQFP

Unit: mm

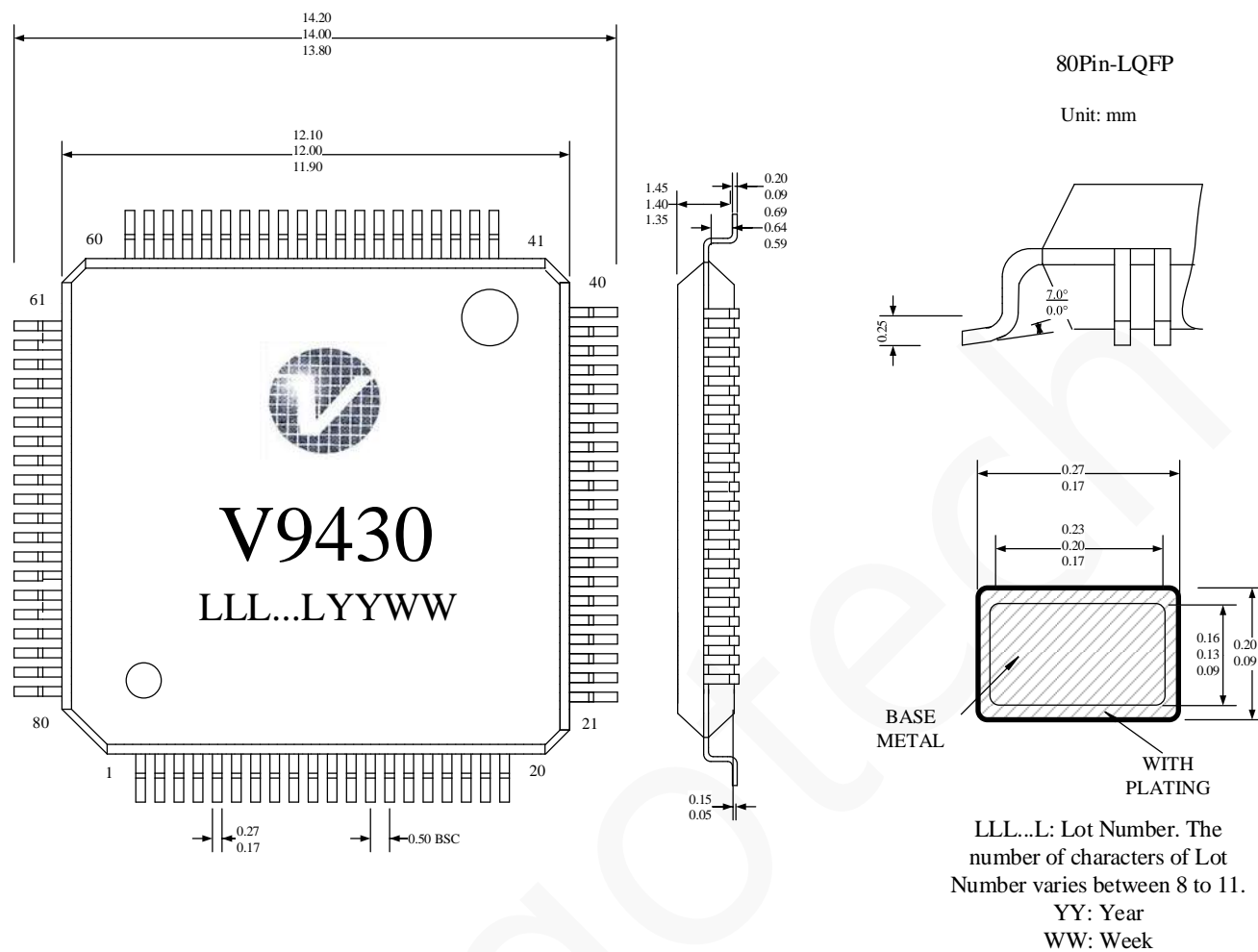


LLL...L: Lot Number. The number of characters of Lot Number varies between 8 to 11.

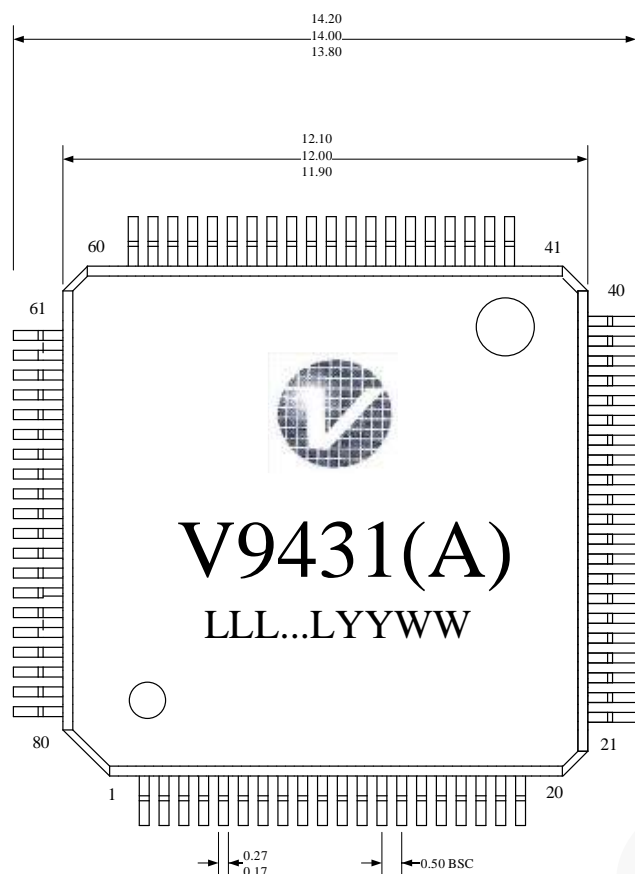
YY: Year

WW: Week

## 30.2 V9430 封装尺寸图

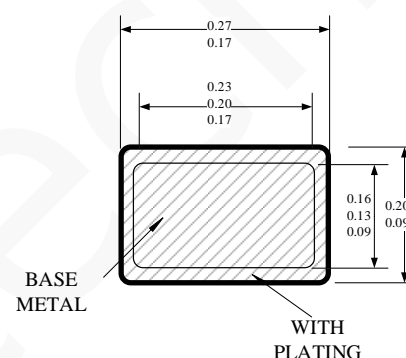
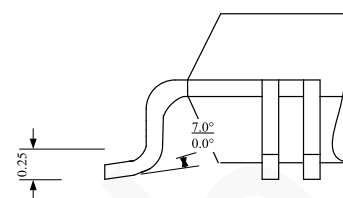


## 30.3 V9431(A)封装尺寸图



80Pin-LQFP

Unit: mm

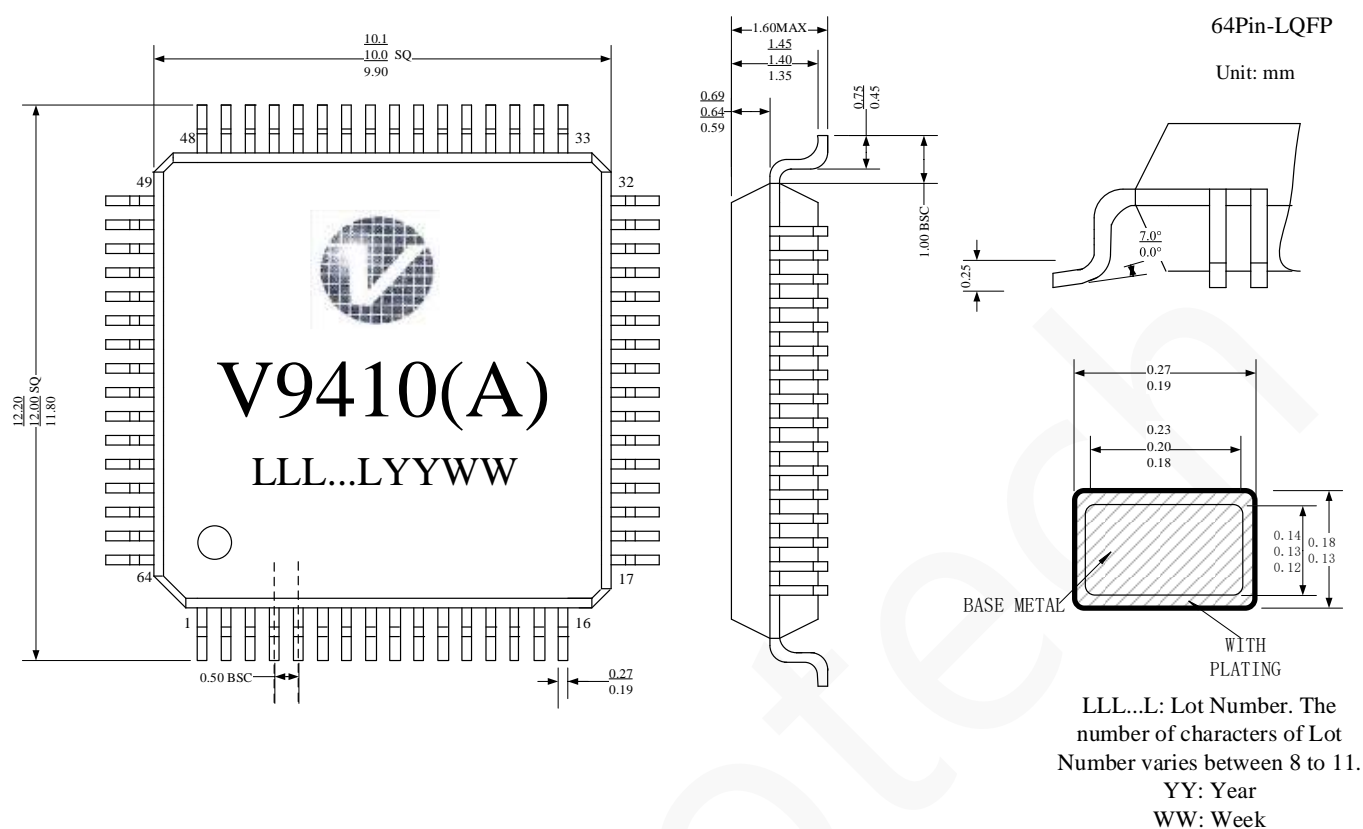


LLL...L: Lot Number. The number of characters of Lot Number varies between 8 to 11.

YY: Year

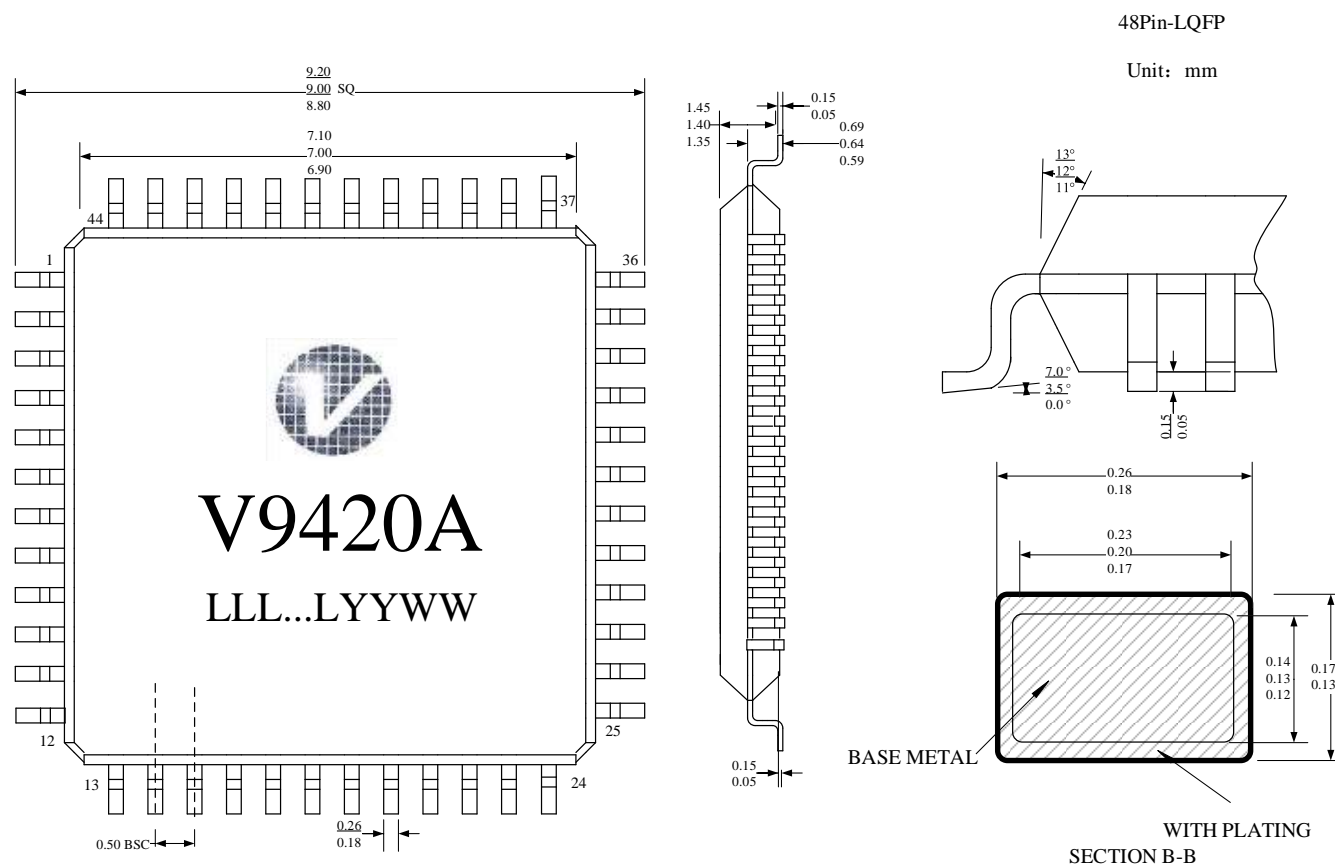
WW: Week

## 30.4 V9410(A) 封装尺寸图





## 30.5 V9420A 封装尺寸图



# 图索引

图 3-1 V94XX(A)系统功能框图 .....	41
图 3-2 V94XX(A)电源系统功能框图 .....	42
图 3-3 V94XX(A)时钟功能框图 .....	43
图 4-1 V94XX(A)内存映射.....	44
图 5-1 V94XX(A)电源系统功能框图 .....	62
图 5-2 DVCCLD0 输出电压与负载电流之间的关系 .....	67
图 5-3 VDCIN 管脚输入信号与位 VDCINDROP 的关系 .....	67
图 6-1 PMU 功能框图 .....	70
图 6-2 片外复位的去毛刺时序 .....	74
图 6-3 V94XX(A)工作模式转换 .....	78
图 6-4 外部唤醒示意图 .....	80
图 6-5 进入深睡眠的流程.....	81
图 6-6 进入浅睡眠的流程.....	82
图 6-7 进入 IDLE 的流程 .....	83
图 7-1 V94XX(A)时钟系统功能框图 .....	85
图 8-1 模拟控制器的功能框图 .....	92
图 9-1 ADC 功能框图 .....	108
图 10-1 比较器的迟滞窗口 .....	116
图 10-2 比较器功能框图.....	116
图 11-1 模拟控制器功能框图 .....	121
图 13-1 RTC 控制器功能框图 .....	126
图 13-2 RTC 手动温度补偿流程图 .....	139
图 14-1 FLASH 控制器的功能框图.....	140
图 15-1 GPIO 控制器的功能框图 .....	146
图 16-1 DMA 控制器功能框图 .....	156
图 17-1 UART 控制器的功能框图 .....	163
图 18-1 UART32K 控制器的功能框图.....	168
图 19-1 ISO7816 控制器功能框图 .....	172
图 20-1 定时器功能框图.....	176
图 20-2 PWM 定时器功能框图.....	177
图 20-3 PWM 定时器模式 .....	181
图 20-4 连续模式示例 .....	181
图 20-5 向上计数模式下的 PWM 输出 .....	183
图 20-6 连续计数模式下的 PWM 输出 .....	183
图 20-7 向上/向下计数模式下的 PWM 输出 .....	184
图 21-1 LCD 控制器的功能框图 .....	186
图 21-2 LCD 控制器结构图.....	187

图 21-3 LCD 波形[1/4 Duty, 1/3 Bias]	188
图 21-4 LCD 波形[1/6 Duty, 1/3 Bias]	189
图 21-5 LCD 波形[1/8 Duty, 1/3 Bias]	190
图 21-6 LCD 波形[1/8 Duty, 1/4 Bias]	191
图 21-7 不同 FBMODE 下的帧缓存器操作模式	195
图 22-1 SPI 控制器的功能框图	197
图 22-2 主机模式, SPO=0, SPH=0	204
图 22-3 主机模式, SPO=0, SPH=1	204
图 22-4 主机模式, SPO=1, SPH=0	205
图 22-5 主机模式, SPO=1, SPH=1	205
图 23-1 I <sup>2</sup> C 控制器功能框图	207
图 23-2 启动和停止条件	214
图 23-37 位地址写模式	214
图 23-47 位地址读模式	215
图 23-5 数据传输	215
图 23-6 I <sup>2</sup> C 总线的应答机制	215
图 23-7 I <sup>2</sup> C 总线连接例	216
图 23-8 SCL 和 SDA 时序	216
图 23-9 I <sup>2</sup> C 控制器的 AC 时序	217
图 23-10 主机发送模式的操作流程	218
图 23-11 主机接收模式的操作流程	219
图 23-12 从机发送模式的操作流程	220
图 23-13 从机接收模式的操作流程	221
图 25-1 MISC 控制器的功能框图	225
图 26-1 CRYPT 控制器的功能框图	232
图 29-1 电能计量功能框图	239
图 29-2 电能计量时钟系统	274
图 29-3 采用 CT 输入电流	276
图 29-4 采用锰铜电阻分流网络输入电流	276
图 29-5 电压输入方式	276
图 29-6 电流/电压通道模拟增益配置	277
图 29-7 角差校正原理图	278
图 29-8 原始波形数据产生原理图	279
图 29-9 电流/电压有效值计算信号处理	280
图 29-10 有功功率计算信号处理	280
图 29-11 无功功率计算信号处理	281
图 29-12 视在功率计算信号处理	281
图 29-13 线电压频率测量原理	283
图 29-14 波形输出 SPI 传输时序	287
图 29-15 电压/电流过零点输出图	289

图 29-16 能量桶功能框图 .....	293
图 29-17 信号输出口功能框图 .....	298

# 表索引

表 1-1 绝对最大额定值 .....	17
表 1-2 供电电压 .....	17
表 1-3 电源管脚的驱动条件 .....	17
表 1-4 常用工作条件 .....	17
表 1-5 各个模块的功耗 .....	18
表 1-6 电源和复位控制模块特性 .....	19
表 1-7 电压回差特性 .....	19
表 1-8 GPIO 特性 .....	19
表 1-9 通用 ADC 特性 .....	20
表 1-10 比较器特性 .....	20
表 1-11 时钟和锁相环特性 .....	21
表 1-12 FLASH 和 SRAM 特性 .....	21
表 1-13 晶振的 ESR 特性 .....	22
表 1-14 电能计量相关特性 .....	22
表 1-15 时钟输出稳定时间及唤醒时间 .....	23
表 1-16 ADC 转换时间（CIC 滤波器降采样率与 ADC 时钟频率） .....	24
表 1-17 ADC 转换时间（CIC 滤波器忽略采样点数） .....	24
表 1-18 TinyADC 转换时间 .....	25
表 1-19 芯片可靠性分析结果 .....	25
表 2-1 V94XX(A)管脚描述 .....	31
表 4-1 PMU 寄存器地址(PMU 基地址: 0x40014000) .....	45
表 4-2 PMU 寄存器地址(PMU 掉电保持 RAM 基地址: 0x40014400) .....	45
表 4-3 模拟控制寄存器地址(ANA 基地址: 0x40014200) .....	45
表 4-4 RTC 控制寄存器地址(RTC 基地址: 0x40014800) .....	46
表 4-5 FLASH 控制寄存器地址(FLASH 控制寄存器基地址: 0x00000000) .....	47
表 4-6 GPIO 控制寄存器地址(GPIO 基地址: 0x40000000) .....	48
表 4-7 DMA 控制寄存器地址(DMA 基地址: 0x40010000) .....	49
表 4-8 UART 寄存器地址(UART 基地址: 0x40011800) .....	50
表 4-9 UART32K 寄存器地址(UART32K 基地址: 0x40014100) .....	51
表 4-10 ISO7816 寄存器地址(ISO7816 基地址: 0x40012000) .....	51
表 4-11 TIMER 寄存器地址(32 位 TIMER 基地址: 0x40012800) .....	51
表 4-12 PWM TIMER 寄存器地址(16 位 PWM TIMER 基地址: 0x40012900) .....	52
表 4-13 LCD 寄存器地址(LCD 基地址: 0x40002000) .....	53
表 4-14 SPI 寄存器地址(SPI 基地址: 0x40015800) .....	54
表 4-15 I <sup>2</sup> C 寄存器地址(I <sup>2</sup> C 基地址: 0x40010800) .....	54
表 4-16 MISC 寄存器地址(MISC 基地址: 0x40013000) .....	55
表 4-17 MISC2 寄存器地址(MISC2 基地址: 0x40013E00) .....	55

表 4-18 CRYPT 寄存器地址(CRYPT 基地址: 0x40006000) .....	55
表 4-19 Info 信息寄存器 .....	56
表 5-1 ANA 寄存器地址(ANA 基地址: 0x40014200) .....	63
表 5-2 ANA_REG5 寄存器 .....	63
表 5-3 ANA_REG6 寄存器 .....	63
表 5-4 ANA_REG7 寄存器 .....	63
表 5-5 ANA_REG8 寄存器 .....	64
表 5-6 ANA_REG9 寄存器 .....	64
表 5-7 ANA_REGA 寄存器 .....	64
表 5-8 ANA_REGF 寄存器 .....	64
表 5-9 ANA_CTRL 寄存器 .....	64
表 5-10 ANA_CMPOUT 寄存器 .....	65
表 5-11 ANA_INTSTS 寄存器 .....	65
表 5-12 ANA_INTEN 寄存器 .....	66
表 6-1 PMU 工作模式寄存器地址(PMU 基地址: 0x40014000).....	70
表 6-2 PMU RAM Retention 寄存器地址(PMU 掉电保持 RAM 基地址: 0x40014400).....	70
表 6-3 ANA 控制器寄存器地址(基地址: 0x40014200) .....	70
表 6-4 PMU_DSLEEPEN 寄存器 .....	71
表 6-5 PMU_DSLEPPASS 寄存器 .....	71
表 6-6 PMU_CONTROL 寄存器 .....	71
表 6-7 PMU_STS 寄存器 .....	72
表 6-8 ANA_CTRL 寄存器 .....	72
表 6-9 各个模块在不同复位及睡眠唤醒后的复位情况 .....	73
表 6-10 在不同模式下的各个时钟的工作状态 .....	74
表 6-11 中断源 .....	75
表 6-12 每一种工作模式下的唤醒源 .....	78
表 7-1 不同工作模式下的时钟模块开关 .....	84
表 7-2 ANA 控制寄存器(ANA 基地址: 0x40014200) .....	85
表 7-3 PMU 寄存器地址(PMU 基地址: 0x40014000).....	86
表 7-4 MISC2 控制寄存器(MISC2 基地址: 0x40013E00).....	86
表 7-5 ANA_REG3 控制寄存器 .....	86
表 7-6 ANA_REG4 控制寄存器 .....	86
表 7-7 ANA_REG9 控制寄存器 .....	87
表 7-8 ANA_REGB 控制寄存器 .....	87
表 7-9 ANA_REGC 控制寄存器 .....	87
表 7-10 ANA_CMPOUT 寄存器 .....	88
表 7-11 PMU_CONTROL 寄存器 .....	88
表 7-12 PMU_STS 寄存器 .....	88
表 7-13 MISC2_CLKSEL 寄存器 .....	89
表 7-14 MISC2_CLKDIVH 寄存器 .....	89

表 7-15 MISC2_CLKDIVP 寄存器 .....	89
表 7-16 MISC2_HCLKEN 寄存器 .....	90
表 7-17 HCLK 时钟使能 .....	90
表 7-18 MISC2_PCLKEN 寄存器 .....	90
表 7-19 各个模块的 PCLK 时钟使能 .....	90
表 8-1 ANA 控制寄存器(ANA 基地址: 0x40014200) .....	92
表 8-2 ANA_REGx .....	93
表 8-3 ANA_REG0 各个 bit 的功能描述 .....	95
表 8-4 ANA_REG1 各个 bit 的功能描述 .....	95
表 8-5 ANA_REG2 各个 bit 的功能描述 .....	96
表 8-6 ANA_REG3 各个 bit 的功能描述 .....	96
表 8-7 ANA_REG4 各个 bit 的功能描述 .....	97
表 8-8 ANA_REG5 各个 bit 的功能描述 .....	97
表 8-9 ANA_REG6 各个 bit 的功能描述 .....	97
表 8-10 ANA_REG7 各个 bit 的功能描述 .....	97
表 8-11 ANA_REG8 各个 bit 的功能描述 .....	98
表 8-12 ANA_REG9 各个 bit 的功能描述 .....	98
表 8-13 ANA_REGA 各个 bit 的功能描述 .....	99
表 8-14 ANA_REGB 各个 bit 的功能描述 .....	99
表 8-15 ANA_REGC 各个 bit 的功能描述 .....	99
表 8-16 ANA_REGD 各个 bit 的功能描述 .....	100
表 8-17 ANA_REGD 各个 bit 的功能描述 .....	100
表 8-18 ANA_REGF 各个 bit 的功能描述 .....	100
表 8-19 ANA_CTRL 寄存器 .....	100
表 8-20 ANA_CMPOUT 寄存器 .....	102
表 8-21 ANA_INTSTS 寄存器 .....	102
表 8-22 ANA_INTEN 寄存器 .....	104
表 8-23 ANA_ADCCTRL 寄存器 .....	105
表 8-24 ANA_ADCDATAx 寄存器 .....	107
表 8-25 ANA_CMPCNTx 寄存器 .....	107
表 8-26 ANA_MISC 寄存器 .....	107
表 9-1 ANA 控制寄存器(ANA 基地址: 0x40014200) .....	108
表 9-2 ANA_REG1 寄存器 .....	109
表 9-3 ANA_REG3 各个 bit 的功能描述 .....	109
表 9-4 ANA_INTSTS 寄存器 .....	110
表 9-5 ANA_INTEN 寄存器 .....	110
表 9-6 ANA_ADCCTRL 寄存器 .....	110
表 9-7 ADC 采样通道 .....	112
表 9-8 ANA_ADCDATAx 寄存器 .....	113
表 9-9 电压测量量程与公式 .....	113

表 10-1 ANA 控制寄存器(ANA 基地址: 0x40014200) .....	117
表 10-2 ANA_REG2 寄存器 .....	117
表 10-3 ANA_REG3 寄存器 .....	117
表 10-4 ANA_REG5 寄存器 .....	118
表 10-5 ANA_CTRL 寄存器 .....	118
表 10-6 ANA_CMPOUT 寄存器 .....	119
表 10-7 ANA_INTSTS 寄存器 .....	119
表 10-8 ANA_INTEN 寄存器 .....	119
表 10-9 ANA_CMPCNTx 寄存器 .....	120
表 11-1 ANA 控制器(ANA 基地址: 0x40014200) .....	121
表 11-2 ANA_REGF 寄存器 .....	122
表 11-3 有关 TADC 的 ANA_CMPOUT 寄存器 .....	122
表 11-4 有关 TADC 的 ANA_INTSTS 寄存器 .....	122
表 11-5 ANA_INTEN 寄存器 forTADC .....	123
表 11-6 ANA_MISC 寄存器 .....	123
表 12-1 看门狗在不同模式下状态(MODE=1) .....	124
表 12-2 PMU_WDT 控制器(PMU 基地址: 0x40014000) .....	124
表 12-3 PMU_WDTPASS 寄存器 .....	124
表 12-4 PMU_WDTEN 寄存器 .....	125
表 12-5 PMU_WDTCLR 寄存器 .....	125
表 13-1 RTC 控制器(RTC 基地址: 0x40014800) .....	127
表 13-2 RTC_SEC/MIN/DAY/WEEK/MONTH/YEAR 寄存器 .....	128
表 13-3 RTC_WKUSEC 寄存器 .....	129
表 13-4 RTC_WKUMIN 寄存器 .....	129
表 13-5 RTC_WKU 时寄存器 .....	129
表 13-6 RTC_WKUCNT 寄存器 .....	130
表 13-7 RTC_CAL 寄存器 .....	130
表 13-8 RTC_DIV 寄存器 .....	130
表 13-9 RTC_CTL 寄存器 .....	131
表 13-10 RTC_PWD 寄存器 .....	131
表 13-11 RTC_CE 寄存器 .....	131
表 13-12 RTC_LOAD 寄存器 .....	132
表 13-13 RTC_INTSTS 寄存器 .....	132
表 13-14 RTC_INTEN 寄存器 .....	132
表 13-15 RTC_PSCA 寄存器 .....	133
表 13-16 RTC_ACTI 寄存器 .....	133
表 13-17 RTC_ACF200 寄存器 .....	133
表 13-18 RTC_ACP0 寄存器 .....	134
表 13-19 RTC_ACP1 寄存器 .....	134
表 13-20 RTC_ACP2 寄存器 .....	134



表 13-21 RTC_ACP3 寄存器 .....	134
表 13-22 RTC_ACP4 寄存器 .....	134
表 13-23 RTC_ACP5 寄存器 .....	135
表 13-24 RTC_ACP6 寄存器 .....	135
表 13-25 RTC_ACP7 寄存器 .....	135
表 13-26 RTC_ACKx 寄存器 .....	135
表 13-27 RTC_ACKTEMP 寄存器 .....	135
表 13-28 RTC_WKUCNTR 寄存器 .....	136
表 13-29 Info 信息寄存器(与 RTC 温度补偿有关) .....	136
表 14-1 FLASH 控制器地址(FLASH 控制器基地址: 0x00000000) .....	141
表 14-2 MISC2 控制器 for FLASH(MISC2 基地址: 0x40013E00) .....	141
表 14-3 FLASH_PASS 寄存器 .....	141
表 14-4 FLASH_CTRL 寄存器 .....	142
表 14-5 FLASH_PGADDR 寄存器 .....	142
表 14-6 PGADDR 及写数据寄存器 .....	142
表 14-7 FLASH_PGDATA 寄存器 .....	142
表 14-8 FLASH_PGBx 寄存器 .....	143
表 14-9 FLASH_PGHwX 寄存器 .....	143
表 14-10 FLASH_SERASE 寄存器 .....	143
表 14-11 FLASH_CERASE 寄存器 .....	143
表 14-12 FLASH_DSTB 寄存器 .....	144
表 14-13 FLASH_INT 寄存器 .....	144
表 14-14 FLASH_CSSADDR 寄存器 .....	144
表 14-15 FLASH_CSEADDR 寄存器 .....	144
表 14-16 FLASH_CSVALUE 寄存器 .....	144
表 14-17 FLASH_CSCVALUE 寄存器 .....	145
表 14-18 FLASH_STS 寄存器 .....	145
表 14-19 MISC2_FLASHWC 寄存器 .....	145
表 15-1 PMU_IOA 控制器(PMU 基地址: 0x40014000) .....	146
表 15-2 GPIO 控制器(GPIO 基地址: 0x40000000) .....	147
表 15-3 PMU_IOAOEN 寄存器 .....	148
表 15-4 PMU_IOAIE 寄存器 .....	148
表 15-5 PMU_IOADAT 寄存器 .....	148
表 15-6 PMU_IOAATT 寄存器 .....	148
表 15-7 不同设置下的 IO 状态 .....	148
表 15-8 PMU_IOAWKUEN 寄存器 .....	149
表 15-9 IO 唤醒模式 .....	149
表 15-10 PMU_IOASTS 寄存器 .....	149
表 15-11 PMU_IOAINTSTS 寄存器 .....	150
表 15-12 PMU_IOASEL 寄存器 .....	150

表 15-13 PMU_IOANODEGL 寄存器 .....	150
表 15-14 IOX_OEN 寄存器 .....	150
表 15-15 IOX_IE 寄存器 .....	151
表 15-16 IOX_DAT 寄存器 .....	151
表 15-17 IOX_ATT 寄存器 .....	151
表 15-18 IOX_STS 寄存器 .....	151
表 15-19 IOB_SEL 寄存器 .....	152
表 15-20 IOE_SEL 寄存器 .....	152
表 15-21 IO_MISC 寄存器 .....	152
表 15-22 特殊功能 IOA .....	153
表 15-23 特殊功能 IOB .....	153
表 15-24 特殊功能 IOC .....	154
表 15-25 特殊功能 IOD .....	154
表 15-26 特殊功能 IOE .....	155
表 15-27 特殊功能 IOF .....	155
表 16-1 DMA 控制器(DMA 基地址: 0x40010000) .....	156
表 16-2 DMA_IE 寄存器 .....	157
表 16-3 DMA_STS 寄存器 .....	158
表 16-4 DMA_CxCTL 寄存器 .....	159
表 16-5 DMA 请求源选择 .....	160
表 16-6 DMA_CxSRC 寄存器 .....	161
表 16-7 DMA_CxDST 寄存器 .....	161
表 16-8 DMA_CxLEN 寄存器 .....	161
表 16-9 DMA_AESCTL 寄存器 .....	161
表 16-10 DMA_AESKEYx 寄存器 .....	162
表 17-1 UART 控制器(UART 基地址: 0x40011800) .....	163
表 17-2 MISC 控制器(MISC 基地址: 0x40013000) .....	164
表 17-3 UARTx_数据寄存器 .....	165
表 17-4 UARTx_STATE 寄存器 .....	165
表 17-5 UARTx_CTRL 寄存器 .....	165
表 17-6 UARTx_INTSTS 寄存器 .....	166
表 17-7 UARTx_BAUDDIV 寄存器 .....	166
表 17-8 UARTx_CTRL2 寄存器 .....	166
表 17-9 MISC_IREN 寄存器 .....	167
表 17-10 MISC_DUTYL 寄存器 .....	167
表 17-11 MISC_DUTYH 寄存器 .....	167
表 18-1 UART32K 控制器(UART32K0 基地址: 0x40014100) .....	168
表 18-2 U32Kx_CTRL0 寄存器 .....	169
表 18-3 U32Kx_CTRL1 寄存器 .....	170
表 18-4 U32Kx_PHASE 寄存器 .....	170

表 18-5 U32Kx_数据寄存器 .....	170
表 18-6 U32Kx_STS 寄存器 .....	171
表 19-1 ISO7816 控制器地址(ISO7816 基地址: 0x40012000) .....	172
表 19-2 ISO7816_BAUDDIVL 寄存器 .....	173
表 19-3 ISO7816_BAUDDIVH 寄存器 .....	173
表 19-4 ISO7816_数据寄存器 .....	173
表 19-5 ISO7816_INFO 寄存器 .....	173
表 19-6 ISO7816_CFG 寄存器 .....	174
表 19-7 ISO7816_CLK 寄存器 .....	174
表 19-8 ISO7816 数据包格式 .....	175
表 20-1 32 位 TIMER 控制器(32 位 TIMER 基地址: 0x40012800) .....	177
表 20-2 16 位 PWM TIMER 控制器(16bPWM TIMER 基地址: 0x40012900) .....	178
表 20-3 TMRx_CTRL 寄存器 .....	179
表 20-4 TMRx_VALUE 寄存器 .....	179
表 20-5 TMRx_RELOAD 寄存器 .....	179
表 20-6 TMRx_INT 寄存器 .....	179
表 20-7 PWMx_CTL 寄存器 .....	180
表 20-8 PWMx_TAR 寄存器 .....	182
表 20-9 PWMx_CCTLx 寄存器 .....	182
表 20-10 PWMx_CCRx 寄存器 .....	184
表 20-11 PWM_O_SEL 寄存器 .....	184
表 21-1 与 LCD 相关的 ANA 控制器(ANA 基地址: 0x40014200) .....	192
表 21-2 LCD 控制器(LCD 基地址: 0x40002000) .....	192
表 21-3 LCD_FBx 寄存器 .....	193
表 21-4 LCD_FBx 寄存器的数据排布 .....	193
表 21-5 LCD_CTRL 寄存器 .....	193
表 21-6 LCD_CTRL2 寄存器 .....	194
表 21-7 LCD_SEGCTRL0 寄存器 .....	195
表 21-8 LCD_SEGCTRL1 寄存器 .....	195
表 21-9 LCD_SEGCTRL2 寄存器 .....	196
表 21-10 ANA_REG6 寄存器 .....	196
表 22-1 SPI 控制器(SPI 基地址: 0x40015800) .....	198
表 22-2 SPI_CTRL 寄存器 .....	198
表 22-3 SPI_TXSTS 寄存器 .....	199
表 22-4 SPI_TXDAT 寄存器 .....	200
表 22-5 SPI_RXSTS 寄存器 .....	200
表 22-6 SPI_RXDAT 寄存器 .....	202
表 22-7 SPI_MISC 寄存器 .....	202
表 23-1 I <sup>2</sup> C 控制器(I <sup>2</sup> C 基地址: 0x40010800) .....	207
表 23-2 I2C_DATA 寄存器 .....	208

表 23-3 I2C_ADDR 寄存器.....	208
表 23-4 I2C_CTRL 寄存器.....	208
表 23-5 SCL 频率设置 .....	209
表 23-6 I2C_STS 寄存器 .....	209
表 23-7 主机发送模式下的 I <sup>2</sup> C 状态.....	210
表 23-8 主机接收模式下的 I <sup>2</sup> C 状态.....	210
表 23-9 从机接收模式下的 I <sup>2</sup> C 状态.....	211
表 23-10 从机发送模式下的 I <sup>2</sup> C 状态 .....	212
表 23-11 I <sup>2</sup> C 状态 .....	213
表 23-12 I2C_CTRL2 寄存器 .....	213
表 23-13 I <sup>2</sup> C AC 特性 .....	216
表 24-1 中断源.....	222
表 25-1 MISC 控制器(MISC 基地址: 0x40013000).....	225
表 25-2 MISC2 控制器(MISC2 基地址: 0x40013E00) .....	226
表 25-3 MISC_SRAMINT 寄存器 .....	226
表 25-4 MISC_SRAMINIT 寄存器 .....	226
表 25-5 MISC_PARERR 寄存器 .....	227
表 25-6 MISC_IREN 寄存器 .....	227
表 25-7 MISC_DUTYL 寄存器 .....	227
表 25-8 MISC_DUTYH 寄存器.....	227
表 25-9 MISC_IRQLAT 寄存器 .....	228
表 25-10 MISC_HIADDR 寄存器 .....	228
表 25-11 MISC_PIADDR 寄存器.....	228
表 25-12 MISC2_FLASHWC 寄存器 .....	228
表 25-13 MISC2_CLKSEL 寄存器 .....	229
表 25-14 MISC2_CLKDIVH 寄存器 .....	229
表 25-15 MISC2_CLKDIVP 寄存器.....	229
表 25-16 MISC2_HCLKEN 寄存器.....	230
表 25-17 HCLK 时钟使能 .....	230
表 25-18 MISC2_PCLKEN 寄存器 .....	230
表 25-19 PCLK 时钟使能.....	230
表 26-1 CRYPT 控制器(MISC 基地址: 0x40006000) .....	232
表 26-2 CRYPT_CTRL 寄存器.....	233
表 26-3 CRYPT_PTRA 寄存器.....	233
表 26-4 CRYPT_PTRB 寄存器.....	234
表 26-5 CRYPT_PTRO 寄存器 .....	234
表 26-6 CRYPT_CARRY 寄存器 .....	234
表 27-1 SWD 引脚的特殊功能.....	236
表 28-1 CMSIS 函数 .....	237
表 29-1 寄存器总列表.....	239

表 29-2 模拟控制寄存器 0 (0x00, EM_ANA_CTRL0) .....	243
表 29-3 模拟控制寄存器 1 (0x01, EM_ANA_CTRL1) .....	244
表 29-4 系统中断状态寄存器 (0x72, EM_SYS_INTSTS) .....	245
表 29-5 电能计量系统中断使能寄存器描述 (0x73, EM_SYS_INTEN) .....	247
表 29-6 系统状态寄存器描述 (0x74, EM_SYS_STS) .....	249
表 29-7 系统配置寄存器 (0x75, EM_SYS_MISC) .....	251
表 29-8 Px 口的 IO 配置寄存器 0 (0x7D, EM_SYS_IOCFOG0) .....	252
表 29-9 PxCFG Bit5~Bit0 说明 .....	253
表 29-10 Px 口的 IO 配置寄存器 1 (0x7E, EM_SYS_IOCFOG1) .....	254
表 29-11 计量控制寄存器 0 (0x02, EM_CTRL0) .....	255
表 29-12 计量控制寄存器 1 (0x03, EM_CTRL1) .....	256
表 29-13 能量桶控制寄存器 0 (0x04, EM_EGY_CTRL0) .....	258
表 29-14 能量桶控制寄存器 1 (0x05, EM_EGY_CTRL1) .....	260
表 29-15 快速检测控制寄存器 (0x06, EM_FD_CTRL) .....	262
表 29-16 波形输出和缓存控制寄存器 (0x07, EM_WAVE_CTRL) .....	263
表 29-17 直流分量寄存器 .....	265
表 29-18 电压/电流/测量信号 (M) 有效值寄存器 .....	265
表 29-19 有功/无功功率寄存器 .....	265
表 29-20 基波通道瞬时值寄存器 .....	266
表 29-21 基波通道平均值寄存器 .....	266
表 29-22 电网频率寄存器 (0x21, EM_DAT_FRQ) .....	266
表 29-23 能量桶寄存器 .....	266
表 29-24 相位测量寄存器 .....	268
表 29-25 功率潜动阈值寄存器 .....	268
表 29-26 电压骤升骤降阈值寄存器 .....	268
表 29-27 快速检测阈值寄存器 .....	269
表 29-28 波形数据寄存器 (0x69, EM_DAT_WAVE) .....	269
表 29-29 预设直流偏置值寄存器 .....	269
表 29-30 电压/电流/测量值有效值校正寄存器 .....	269
表 29-31 全波有功/无功功率校正寄存器 .....	270
表 29-32 门限值寄存器 .....	270
表 29-33 角差校正寄存器 (0x33, EM_CFG_PHC) .....	271
表 29-34 带通滤波器系数寄存器 (0x37, EM_CFG_BPF) .....	271
表 29-35 校验和寄存器 (0x38, EM_CFG_CKSUM) .....	271
表 29-36 参与参数自检寄存器列表 .....	271
表 29-37 Bandgap 电路相关寄存器 .....	274
表 29-38 时钟产生电路相关寄存器 .....	275
表 29-39 电压/电流通道模拟增益配置 .....	276
表 29-40 开关电压/电流通道 ADC .....	278
表 29-41 不同 fsmpl 下的角差分辨率和校正范围 .....	278

表 29-42 角差校正配置 .....	279
表 29-43 开关电压/电流通道 ADC .....	279
表 29-44 计量通道信号数字增益配置 .....	280
表 29-45 无功功率模式选择 .....	281
表 29-46 视在功率计算源选择 .....	282
表 29-47 功率启动潜动开关和状态 .....	282
表 29-48 功率潜动阈值寄存器 .....	283
表 29-49 测试累加周波个数选择 .....	283
表 29-50 频率寄存器来源 FRQ_SEL 描述 .....	284
表 29-51 频率常量 Frequency constant 描述 .....	284
表 29-52 过零点选择 .....	284
表 29-53 波形输出和缓存控制寄存器 .....	285
表 29-54 波形输出配置 .....	287
表 29-55 主动波形输出数据格式 .....	287
表 29-56 波形缓存数据格式 .....	288
表 29-57 过零点选择 .....	288
表 29-58 电压骤升骤降寄存器 .....	290
表 29-59 过压欠压和过流欠流控制 .....	291
表 29-60 快速检测阈值寄存器 .....	292
表 29-61 能量桶模式配置 .....	294
表 29-62 能量桶寄存器 .....	294
表 29-63 能量启动潜动开关和状态 .....	296
表 29-64 CF 模式配置 .....	297
表 29-65 Px 口的 IO 配置寄存器 0 (0x7D, EM_SYS_IOCFCG0) .....	298
表 29-66 PxCFG Bit5~Bit0 说明 .....	299
表 29-67 Px 口的 IO 配置寄存器 1 (0x7E, EM_SYS_IOCFCG1) .....	300

# 版本更新说明

时间	版本	说明
2020.9.18	V1.0	初版发布。
2021.6.24	V1.1	修改 V94XX(A) 概述，增加电能计量模块描述。
2021.7.6	V1.2	V9430 UART 资源修正为 3 组。
2022.3.10	V1.3	<p>增加 ADC 和计量工作电压</p> <p>去除罗氏线圈描述，包括寄存器等</p> <p>修改 EM_SYS_INTSTS 和 EM_SYS_INTEN 寄存器，增加高速能量桶溢出中断</p> <p>修改 EM_WAVE_CTRL 寄存器 DMA_MODE 控制位说明，增加说明：在使能 DMA 传输前，必须至少打开一路波形缓存和输出的通道。</p> <p>修改 EM_WAVE_CTRL 寄存器 bit3 和 bit4 描述，即波形输出的 DMA 通道手动开关描述</p> <p>删除 EM_CTRL0 bit3 功能，即删除电网频率选择</p> <p>修改 UPERIOD 和 IPERIOD 功能描述，从半周波单位改为周波单位</p> <p>修改 n 个 RCL 时钟间隔描述为确定时间，算法为最长时间间隔 x1.5</p> <p>修改 EM_SYS_MISC 寄存器中的强制关闭能量桶和 CF 输出功能描述</p>
2022.9.2	V1.4	<p>增加 V9400(A)、V9410(A)、V9431(A)、V9420A 型号</p> <p>删除 SEG34、SEG35、SEG58</p>