

V9811 是一款高性能的单相电能计量 SoC 芯片，集成模拟前端、电能计量模块、增强型 8052 内核、RTC、WDT、Flash、SRAM 和 LCD 驱动等功能模块，可为单相多功能电能表提供单芯片解决方案。

特点

- 3.3V 或 5V 电源供电，宽电压输入范围：2.5V~5.5V
- 基准电压：1.185V (20ppm/°C)
- 支持灵活的防窃电应用
- 休眠时，芯片典型工作电流：11 μ A
- 64-LQFP 封装
- 工作温度：-40~+85°C
- 储存温度：-40~+125°C
- 计量特点：
 - 4 路独立的过采样 Σ/Δ ADC：一路电压、两路电流和一路多功能测量通道
 - 计量精度：
 - 满足 GB/T 17215.321-2008、GB/T 17215.322-2008 和 GB/T 17215.323-2008 的要求
 - 在电流 2000:1 动态范围内，有功计量误差小于 0.1%
 - 在电流 1000:1 动态范围内，无功计量误差小于 0.1%
 - 在电流 1000:1 动态范围内，电压/电流有效值计量误差小于 0.5%
 - 提供各种测量数据：
 - 3 路电压/电流信号原始波形和直流分量
 - 瞬时/秒平均有功/无功功率、秒平均视在功率、正反向有功/无功电能
 - 瞬时/秒平均电压/电流有效值
 - 线电压频率
 - 温度，测量精度 $\pm 1^\circ\text{C}$
 - 电池电压/系统电压/外部电压
 - 支持同一路电流进行有功与无功电能计量，或者两路电流同时进行有功电能计量
 - 支持电流有效值电能计量、常数计量和 1/4 降频计量
 - 提供四个可配置的 CF 脉冲输出
 - 支持软件校表：
 - 支持角差校正，角差校正范围 $\pm 1.4^\circ$
 - 支持有效值/功率比差校正，并支持有功/无功功率二次补偿（offset 校正）
 - 支持小电流加速校表

- 支持电压过零点中断
- 电流输入：普通 CT 和锰铜分流器
- 其它特点：
 - 高性能 8 位 8052 兼容 MCU 内核，工作频率可编程，运算能力最高可达 26MHz/6.5mips
 - 集成起振电路和 PLL，片外仅需一个 32768Hz 晶体，支持晶体停振时的起振复位功能
 - 64kB FLASH 存储器，具有写保护和加密功能，支持 ISP 和 IAP
 - 4kB 外部 SRAM 存储器
 - 高性能 RTC，支持温度补偿，实时秒脉冲校正，全温度范围内误差小于 5ppm
 - 4 路全速 UART，其中一路支持红外通信
 - 最多 43 个可配置的 GPIO，提供端口中断
 - 最多 10 个硬件定时器。其中，16-bit TimerA，支持多个比较/捕获功能，支持多个时序控制和多个 PWM 输出，具有广泛的中断功能
 - 4×24/6×22 段 LCD 驱动电路，支持多种扫描频率；LCD 驱动电压在 2.7V~3.3V 范围内可调，调整步长 100mV
 - 时钟独立的硬件看门狗电路，增强系统的稳定性
 - JTAG 实时调试系统
 - 提供多种工作模式以及休眠唤醒方法，满足不同功耗需求

声明

杭州万工科技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> 获取最新信息。

目录

特点	1
声明	3
目录	4
引脚分布图	8
模拟性能参数	12
计量性能参数	13
极限参数	14
时序参数	15
功能框图	16
第 1 章 特殊功能寄存器 (SFR)	17
第 2 章 复位	23
2.1 复位等级 3	25
2.2 复位等级 2	25
2.2.1 恢复供电复位	25
2.2.2 IO 休眠唤醒复位	26
2.2.3 RTC 休眠唤醒复位	26
2.3 复位等级 1	26
2.3.1 片外输入复位 RSTN	26
2.3.2 上电复位 (POR)	26
2.3.3 WDT 溢出复位	26
第 3 章 8052 MCU 结构	27
3.1 IRAM	27
3.2 数据存储器空间	28
3.3 程序存储空间	29
3.4 调试模式、ISP 和程序加密	30
3.4.1 调试模式	30
3.4.2 ISP	31
3.4.3 程序加密	31
3.5 MCU	31
3.5.1 指令集	31
3.5.2 可编程的 MOVX 访问周期	36
3.5.3 双数据指针	36
第 4 章 模拟控制寄存器	37
第 5 章 时钟	44
5.1 时钟控制电路	44
5.2 RC 时钟产生电路	45
5.3 OSC 时钟产生电路	45
5.4 PLL 时钟产生电路	46
5.5 切换 PLL 时钟与 OSC 时钟	47
5.5.1 MCU 和电能计量时钟源普通切换流程	47
5.6 MCU 时钟源快速切换流程	49

5.6.2	MCU 时钟源切换普通操作方式 V.S 快速操作方式.....	50
第 6 章	功耗.....	52
6.1	OSC 状态.....	54
6.2	工作状态.....	54
6.3	休眠状态.....	55
6.3.1	休眠/唤醒 - 使用普通操作方式.....	55
6.3.2	休眠/唤醒 - 使用快速操作方式.....	56
6.3.3	休眠功耗.....	56
第 7 章	电源系统.....	57
7.1	3.3V 稳压电路 (LDO33).....	57
7.2	数字电源电路.....	58
7.3	掉电监测电路.....	58
7.4	上电复位和 RSTN 复位.....	58
7.5	电池供电.....	59
第 8 章	电压基准电路 (BANDGAP).....	60
第 9 章	电能计量.....	61
9.1	寄存器列表和使用说明.....	61
9.1.1	访问计量控制/数据寄存器.....	61
9.1.2	计量控制寄存器.....	62
9.1.3	计量数据寄存器.....	66
9.2	电能计量模块结构和原理.....	72
9.3	模拟信号输入.....	73
9.4	模拟/数字转换.....	75
9.5	角差校正.....	76
9.6	电流计量通道选择.....	77
9.7	数字信号输入.....	77
9.8	有效值计算和校正.....	79
9.9	视在功率计算.....	80
9.10	有功/无功功率计算和校正.....	80
9.10.1	有功功率计算.....	80
9.10.2	无功功率计算.....	81
9.10.3	功率校正.....	82
9.11	能量累加和 CF 脉冲输出.....	82
9.11.1	能量累加和脉冲产生.....	82
9.11.2	CF 脉冲输出.....	83
9.12	起动/潜动判断.....	84
9.13	线电压频率测量.....	85
9.14	M 通道测量.....	86
9.14.1	M 通道结构.....	86
9.14.2	电池电压和外部信号测量.....	88
9.15	计量模块初始化.....	89
9.16	校表.....	90
9.16.1	相关寄存器的计算公式.....	90
9.16.2	有功能量校正.....	92
9.16.3	电流有效值校正.....	92
9.16.4	电压有效值校正.....	93
9.16.5	角差校正.....	93
9.16.6	潜动.....	93
9.16.7	校表举例.....	94

第 10 章	中断控制系统.....	98
10.1	中断控制 SFR.....	98
10.1.1	IE SFR (0XA8)	98
10.1.2	IP SFR (0XB8)	98
10.1.3	EIE SFR (0XE8)	99
10.1.4	EIP SFR (0XF8)	99
10.1.5	EXIF SFR (0X91)	100
10.1.6	EICON SFR (0XD8)	100
10.2	中断处理	101
10.3	中断扩展	102
10.3.1	中断向量 8.....	102
10.3.2	中断向量 9.....	103
10.3.3	中断向量 10.....	105
10.3.4	中断向量 11.....	106
10.4	中断资源列表	109
第 11 章	TIMERS/UART.....	113
11.1	定时器/计数器	113
11.1.1	TIMERA	113
11.1.2	TIMER0/TIMER1/TIMER2.....	121
11.2	UART.....	128
11.2.1	UART1	128
11.2.2	扩展外设 UART	129
11.2.3	UART 的模式选择	132
第 12 章	LCD.....	135
12.1	时序和驱动电路.....	135
12.2	LCD 驱动电压.....	138
第 13 章	GPIO	139
13.1	P0.....	139
13.2	P1.....	140
13.3	P2.....	142
13.4	P3.....	144
13.5	P4.....	146
13.6	P5.....	147
13.7	P6.....	148
13.8	P7.....	149
13.9	P8.....	150
13.10	P9.....	151
第 14 章	WDT.....	154
第 15 章	RTC.....	155
15.1	RTC 相关寄存器	155
15.2	读写 RTC	157
15.2.1	写 RTC	157
15.2.2	读 RTC 数据	158
15.3	RTC 计时	158
15.4	RTC 中断	158
15.4.1	RTC 非法数据中断	158
15.4.2	RTC 秒中断.....	158
15.5	高频秒脉冲校正.....	158
15.6	RTC 校正	159

15.6.1 利用高频秒脉冲校正 RTC 159

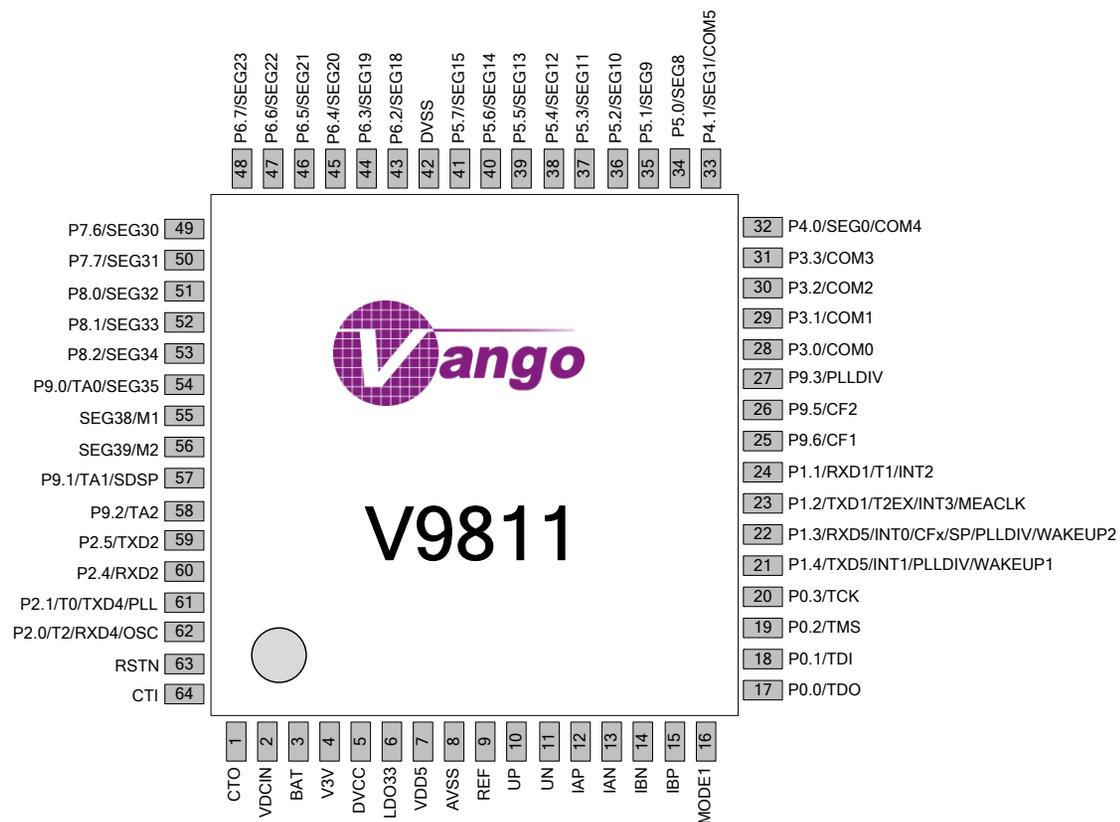
15.6.2 温度补偿校正 RTC 160

图索引 162

表索引 164

版本更新说明 170

引脚分布图



引脚编号	名称	输入/输出	功能说明
1	CTO	输出	32768Hz 晶体振荡输出。芯片集成起振电路，片外仅需连接一个 32768Hz 的晶体
2	VDCIN	输入	掉电监测输入。高于 1.1V 表明当前供电正常，低于 1.0V 表示当前发生了掉电。
3	BAT	输入	测量系统电池电压输入
4	V3V	输入	3.3V/5V 系统切换控制。1, 3.3V 供电系统；0, 5V 供电系统。



引脚编号	名称	输入/输出	功能说明
5	DVCC	输入/输出	数字电源电压输出，为数字电路、PLL 时钟产生电路、RTC 和晶体起振电路供电。应连接一个由 10 μ F 和 0.1 μ F 去耦电容组成的并联电路后，再接到数字地。
6	LDO33	输入/输出	模拟 3.3V 电压输出，为模拟电路和 GPIO 端口供电。外部应连接一个由 10 μ F 和 0.1 μ F 去耦电容组成的并联电路后，再接地。
7	VDD5	输入	5V 电源输入。外部应连接一个 10 μ F 去耦电容，然后接地。
8	AVSS	输入/输出	模拟地
9	REF	输入/输出	基准电压输出。外部应先连接一个 1 μ F 的去耦电容，然后再接地。
10	UP	输入	电压信号 P 端输入
11	UN	输入	电压信号 N 端输入
12	IAP	输入	电流通道 A 电流信号 P 端输入
13	IAN	输入	电流通道 A 电流信号 N 端输入
14	IBN	输入	电流通道 B 电流信号 N 端输入
15	IBP	输入	电流通道 B 电流信号 P 端输入
16	MODE1	输入	芯片工作模式选择：0，调试模式；1，正常运行模式
17	P0.0/TDO	输入/输出	普通 GPIO 口/JTAG 口，用于测试数据输出
18	P0.1/TDI	输入/输出	普通 GPIO 口/JTAG 口，用于测试数据输入
19	P0.2/TMS	输入/输出	普通 GPIO 口/JTAG 口，用于测试模式选择
20	P0.3/TCK	输入/输出	普通 GPIO 口/JTAG 口，用于测试时钟输入
21	P1.4/TXD5/INT1/ PLLDIV/WAKEUP1	输入/输出	普通 GPIO 口/UART5 发送数据输出/端口中断 1/高频分频后的系统时钟输出，可用于高频秒脉冲输出/IO 休眠唤醒。其中，作端口中断和 IO 休眠唤醒用时，下降沿有效。
22	P1.3/RXD5/INT0/CFx/SP/ PLLDIV/WAKEUP	输入/输出	普通 GPIO 口/UART5 接收数据输入/端口中断 0/可配置的 CF 脉冲输出/RTC 秒脉冲输出，每 30s 输出一个脉冲/高频分频后的系统时钟输出，可用于高频秒脉冲输出/IO 休眠唤醒。其中，作端口中断和 IO 休眠唤醒用时，下降沿有效。当该引脚被用于可配置的 CF 脉冲输出时，用户可配置寄存器 P13FS (0x28C7)。
23	P1.2/TXD1/T2EX/INT3/ME ACLK	输入/输出	普通 GPIO 口/UART1 发送数据输出/Timer2 捕获/重载触发信号输入/端口中断 3/计量时钟输

引脚编号	名称	输入/输出	功能说明
			出。其中，作端口中断用时，下降沿有效。
24	P1.1/RXD1/T1/INT2	输入/输出	普通 GPIO 口/UART1 接收数据输入/Timer1 外部输入/端口中断 2/RC 时钟输出。其中，作端口中断用时，下降沿有效。
25	P9.6/CF1	输入/输出	快速 GPIO 口/E1 通量 CF 脉冲输出
26	P9.5/CF2	输入/输出	快速 GPIO 口/E2 能量 CF 脉冲输出
27	P9.3/PLLDIV	输入/输出	快速 GPIO 口/高频分频后的系统时钟输出，可用于高频秒脉冲输出
28~31	P3.0~P3.3/COM0~COM3	输入/输出	普通 GPIO 口/COM 口
32	P4.0/SEG0/COM4	输入/输出	普通 GPIO 口/LCD 的 SEG 输出引脚/COM 口 当 LCDTYPE=0 (bit4, LCDCtrl, 0x2C1E) 时，该引脚复用为普通 GPIO 口和 SEG0 口； 当 LCDTYPE=1 (bit4, LCDCtrl, 0x2C1E) 时，该引脚复用为普通 GPIO 口和 COM4 口。
33	P4.1/SEG1/COM5	输入/输出	普通 GPIO 口/LCD 的 SEG 输出引脚/COM 口 当 LCDTYPE=0 (bit4, LCDCtrl, 0x2C1E) 时，该引脚复用为普通 GPIO 口和 SEG1 口； 当 LCDTYPE=1 (bit4, LCDCtrl, 0x2C1E) 时，该引脚复用为普通 GPIO 口和 COM5 口。
34~41	P5.0~P5.7/SEG8~SEG15	输入/输出	普通 GPIO 口/LCD 的 SEG 输出引脚
42	DVSS	输入/输出	数字地
43~48	P6.2~P6.8/SEG18~SEG23	输入/输出	普通 GPIO 口/LCD 的 SEG 输出引脚
49~53	P7.6~P8.2/SEG30~SEG34	输入/输出	普通 GPIO 口/LCD 的 SEG 输出引脚
54	P9.0/SEG35/TA0	输入/输出	快速 GPIO 口/LCD 的 SEG 输出引脚/TimerA 比较/捕获模块 0 的输入或输出
55~56	SEG38~SEG39/M1~M2	输入/输出	LCD 的 SEG 输出引脚/M 通道信号输入，用于测量系统电池电压或外部电压信号
57	P9.1/ TA1/SDSP	输入/输出	快速 GPIO 口/TimerA 比较/捕获模块 1 的输入或输出/标准秒信号输入 当 P9.1 作标准秒信号输入时，P9.1 输入使能。
58	P9.2/ TA2	输入/输出	快速 GPIO 口/TimerA 比较/捕获模块 2 的输入或输出
59	P2.5/TXD2	输入/输出	GPIO 口/UART2 发送数据输出。该引脚作 TXD2 用时，可以选择发送带有 38kHz 载波的信号。
60	P2.4/RXD2	输入/输出	GPIO 口/UART2 接收数据输入。

引脚编号	名称	输入/输出	功能说明
			该引脚作 RXD2 用时，如用作红外通信时，需要外接红外解调器件。
61	P2.1/TXD4/T0/PLL	输入/输出	GPIO 口/UART4 发送数据输出/Timer0 外部输入/PLL 时钟输出
62	P2.0/RXD4/T2/OSC	输入/输出	GPIO 口/UART4 接收数据输入/Timer2 外部输入/OSC 时钟输出
63	RSTn	输入	复位输入，低电平有效。必须保持低电平 5ms 以上才有效
64	CTI	输入	32768Hz 晶体振荡输入。芯片内有起振电路，片外只需要一个 32768Hz 的晶体

模拟性能参数

参数	最小	典型	最大	单位	说明
模拟输入信号					
最大输入信号	-200		+200	mV	
信号带宽 (-3dB)		1.6		KHz	
ADC 性能					
直流失调			10	mV	
有效位数		20		bit	不包含符号位
片上基准电压					
偏差	-20		20	mV	
电源抑制比		80		dB	
温度系数		20		ppm/°C	
输出电压		1.185		V	
POR 性能					
检测门限 (DVCC)	1.7	1.8	1.9	V	
VDCIN					
输入电压	0		VDD	V	
输入阻抗		1.5		MΩ	
低电压检测门限 (PWRDN)		1.0		V	
低电压检测门限 (PWRUP)		1.1		V	

计量性能参数

参数	典型	单位	说明
通道间相差			
PF=0.8 容性	±0.05	度	
PF=0.5 感性	±0.05	度	
有功计量误差	0.1	%	动态范围 2000:1 @ 25°C
有功计量带宽	1.6	kHz	
无功计量误差	0.1	%	动态范围 1000:1 @ 25°C
无功计量带宽	1.6		
有功秒平均功率误差	0.1	%	动态范围 2000:1 @ 25°C
视在功率误差	0.5		动态范围 1000:1 @ 25°C
电压有效值误差	0.5	%	动态范围 1000:1 @ 25°C
电压有效值带宽	1.6	kHz	
电流有效值误差	0.5	%	动态范围 1000:1 @ 25°C
电流有效值带宽	1.6	kHz	
CF 输出特性			
最大输出频率	6.4	kHz	
占空比	50%		当脉宽低于 80ms 时
最大脉宽	80	ms	

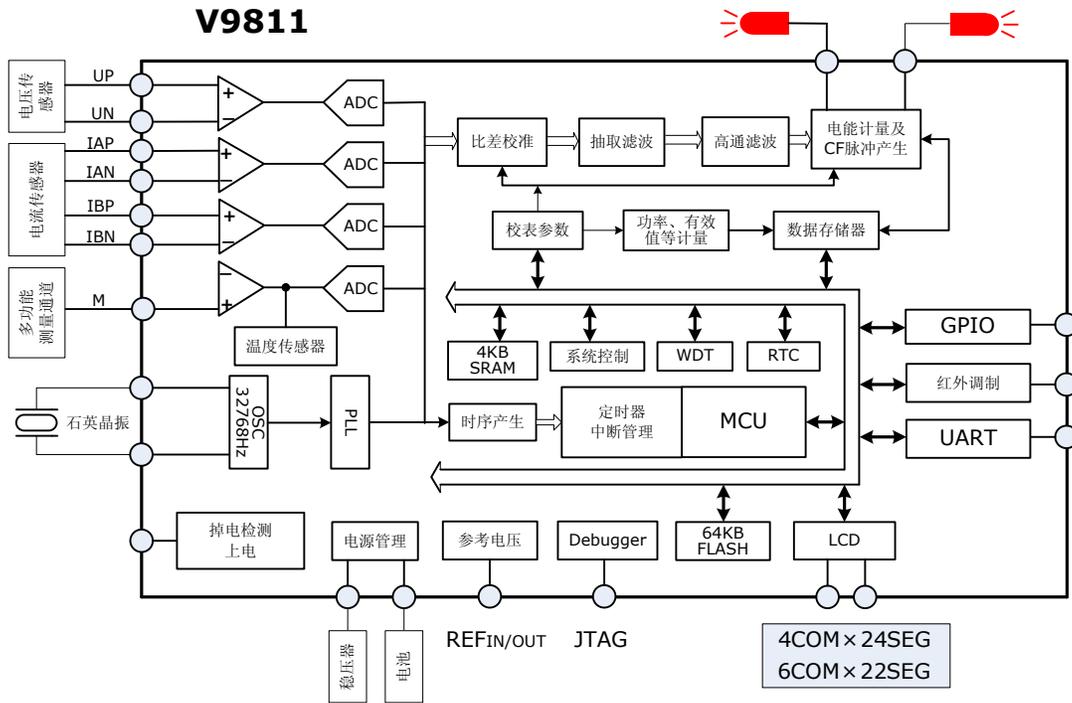
极限参数

参数	最小	典型	最大	单位	说明
数字 IO, 输出					
输出高电平, V_{OH}	2.4			V	12mA 电流短时间内不会对芯片造成损坏; 长时间大于 10mA 的电流会对芯片造成损坏。 LDO33 如果是由 3.3V-LDO 供电, IO 上消耗的总电流不能大于 LDO33 的最大驱动能力。
I_{SOURCE}		10	12	mA	
输出低电平 V_{OL}			0.4	V	
I_{SINK}		10	12	mA	
数字 IO, 输入					
输入高电平, V_{INH}	2.0			V	
输出低电平, V_{INL}			0.4	V	
电源输入					
VDD5	3	5.0	9.0	V	
LDO33 输出					
电压		3.3		V	仅适用于 5V 供电系统。
电流			30	mA	
数字电源输出					
电压	2.1	2.5	2.8	V	
电流			35	mA	
功耗					
电压 ADC		0.62		mA	$f_{MCU}=13.1072MHz$, $f_{电能计量}=3276.8kHz$, $f_{ADC}=819.2kHz$ 。
电流 ADC		0.66		mA	
全速工作		12.8		mA	$f_{MCU}=13.1072MHz$, $f_{电能计量}=3276.8kHz$, $f_{ADC}=819.2kHz$, 开启 4 路 ADC 通道。
休眠		11		μA	晶振电路和 RTC 正常工作, 系统监控电路正常工作, RAM 保持数据, 电压监测电路、稳压电路以及复位电路正常工作
工作温度	-40		+85	$^{\circ}C$	
储存温度	-40		+125	$^{\circ}C$	

时序参数

参数	最小	典型	最大	单位	说明
MCU 时钟频率	819.2	13107.2	26214.4	kHz	开启 PLL, 且使用 PLL 时钟
电能计量时钟频率	819.2	6553.6	6553.6	kHz	
晶振频率		32768		Hz	晶振输出 OSC 时钟

功能框图



第1章 特殊功能寄存器 (SFR)

表 1-1 特殊功能寄存器 (SFR) 列表

地址	寄存器	Bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0
80h	SysCtrl	MEAFRQ	FWC	FSC	PMG	LCDG	SLEEP1	SLEEP0	MCUFRQ
81h	SP	-	-	-	-	-	-	-	-
82h	DPL0	-	-	-	-	-	-	-	-
83h	DPH0	-	-	-	-	-	-	-	-
84h	DPL1	-	-	-	-	-	-	-	-
85h	DPH1	-	-	-	-	-	-	-	-
86h	DPS	0	0	0	0	0	0	0	SEL
87h	PCON	SMOD0	-	1	1	GF1	GF0	STOP	IDLE
88h	TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
89h	TMOD	GATE	C/T	M1	M0	GATE	C/T	M1	M0
8Ah	TL0	-	-	-	-	-	-	-	-
8Bh	TL1	-	-	-	-	-	-	-	-
8Ch	TH0	-	-	-	-	-	-	-	-
8Dh	TH1	-	-	-	-	-	-	-	-
8Eh	CKCON	-	-	T2M	T1M	T0M	MD2	MD1	MD0
8Fh	SPC_FNC	0	0	0	0	0	0	0	WRS
90h	RTCPEN	-	-	-	-	-	-	-	-
91h	EXIF	IE5	IE4	IE3	IE2	1	0	0	0
92h	Reserved	-	-	-	-	-	-	-	-
93h	RTCYC	Y80	Y40	Y20	Y10	Y8	Y4	Y2	Y1

地址	寄存器	Bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0
94h	RTCCH	-	-	C13	C12	C11	C10	C9	C8
95h	RTCCL	C7	C6	C5	C4	C3	C2	C1	C0
96h	INTRTC	-	-	-	-	-	RTC2	RTC1	RTC0
97h	RTCPWD	-	-	-	-	-	-	-	WE
98h	Reserved	-	-	-	-	-	-	-	-
99h	Reserved	-	-	-	-	-	-	-	-
9Ah	RTCSC	-	S40	S20	S10	S8	S4	S2	S1
9Bh	RTCMiC	-	M40	M20	M10	M8	M4	M2	M1
9Ch	RTCHC	-	-	H20	H10	H8	H4	H2	H1
9Dh	RTCDC	-	-	D20	D10	D8	D4	D2	D1
9Eh	RTCWC	-	-	-	-	W8	W4	W2	W1
9Fh	RTCMoC	-	-	-	Mo10	Mo8	Mo4	Mo2	Mo1
A0h	Reserved	-	-	-	-	-	-	-	-
A1h	Systate	-	-	POR	WDT	IO	RTC	PWRDN	PWRUP
A2h	Version	0	0	1	1	0	0	1	0
A3h	PLLLCK	-	-	-	-	-	-	-	PLLLCK
A4h	P9OE	-	P96OEN	P95OEN		P93OEN	P92OEN	P91OEN	P90OEN
A5h	P9IE	-	P96INEN	P95INEN		P93INEN	P92INEN	P91INEN	P90INEN
A6h	P9OD	-	-	-	-	-	-	-	-
A7h	P9ID	-	-	-	-	-	-	-	-
A8h	IE	EA	ES1	ET2	ES0	ET1	EX1	ET0	EX0
A9h	Reserved	-	-	-	-	-	-	-	-
AAh	Reserved	-	-	-	-	-	-	-	-
ABh	Reserved	-	-	-	-	-	-	-	-

地址	寄存器	Bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0
ACh	Reserved	-	-	-	-	-	-	-	-
ADh	P9FS	-	P96FNC	P95FNC		P93FNC	P92FNC	P91FNC	P90FNC
A Eh	Reserved	-	-	-	-	-	-	-	-
B8h	IP	1	PS1	PT2	PS0	PT1	PX1	PT0	PX0
C0h	SCON1	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1
C1h	SBUF1	-	-	-	-	-	-	-	-
C8h	T2CON	TF2	EXF2	-	-	EXEN2	TR2	C/T2	CP/RL2
C9h	Reserved	-	-	-	-	-	-	-	-
CAh	RCAP2L	-	-	-	-	-	-	-	-
CBh	RCAP2H	-	-	-	-	-	-	-	-
CCh	TL2	-	-	-	-	-	-	-	-
CDh	TH2	-	-	-	-	-	-	-	-
CEh	WDTEN	-	-	-	-	-	-	-	-
CFh	WDTCLR	-	-	-	-	-	-	-	-
D0h	PSW	CY	AC	-	RS1	RS0	OV	-	P
D8h	EICON	SMOD1	1	-	-	PFI	0	0	0
D9h	Reserved	-	-	-	-	-	-	-	-
DAh	RDRTC	-	-	-	-	-	-	-	-
DBh	DIVTHH	DIV23	DIV22	DIV21	DIV20	DIV19	DIV18	DIV17	DIV16
DCh	DIVTHM	DIV15	DIV14	DIV13	DIV12	DIV11	DIV10	DIV9	DIV8
DDh	DIVTHL	DIV7	DIV6	DIV5	DIV4	DIV3	DIV2	DIV1	DIV0
DEh	PLLCNT	-	-	-	-	-	-	STT1	STT0
E0h	ACC	-	-	-	-	-	-	-	-
E8h	EIE	1	1	1	EWDI	EX5	EX4	EX3	EX2

地址	寄存器	Bit 7	bit 6	bit 5	bit 4	bit 3	Bit 2	bit 1	bit 0
F0h	B	-	-	-	-	-	-	-	-
F8h	EIP	1	1	1	PWDI	PX5	PX4	PX3	PX2

阴影区域为系统 SFR。

表 1-2 时钟切换控制寄存器 (SysCtrl SFR, 0x80)

0x80, R/W, 时钟切换控制寄存器, SysCtrl SFR				
位		默认值	功能	说明
bit7	MEAFRQ	0	选择电能计量时钟源	0, OSC 时钟; 1, PLL 时钟。 配置这个位切换电能计量时钟源; 读取该位的值以显示当前电能计量时钟源。
bit6	FWC	0	快速唤醒控制位	0, 关闭; 1, 开启 当 FSC=0 时, FWC 的配置才生效。 仅当 FWC=0 时, 用户才能配置 PLL 输出频率。
bit5	FSC	0	快速休眠控制位	0, 关闭; 1, 开启 向 FSC 位写入 1, 系统立即进入浅睡眠状态。
bit4	PMG	0	开关电能计量时钟 (时钟 2)	0, 开启; 1, 关闭。 当时钟 1 和时钟 2 的时钟源为 PLL 输出时, 时钟 3 才可被关闭。
bit3	LCDG	0	开关 LCD 时序产生电路的时钟 (时钟 3)	0, 开启; 1, 关闭。 当时钟 1 和时钟 2 的时钟源为 PLL 输出时, 时钟 3 才可被关闭。
bit[2:1]	SLEEP1/SLEEPO	0	开启或者关闭时钟 1。	11/01, 系统进入浅睡眠; 10, 系统进入深睡眠
bit0	MCUFRQ	0	选择 MCU 时钟源 (时钟 1)。	0, OSC 时钟; 1, PLL 时钟。配置这个位切换 MCU 时钟源; 读取该位的值以显示当前 MCU 的时钟源。

表 1-3 时钟周期控制寄存器 (CKCON SFR, 0x8E)

0x8E, R/W, 时钟周期控制寄存器, CKCON SFR				
位		默认值	功能	说明
Bit[7:6]	保留	0		
Bit5	T2M	0	Timer2 时钟源选择	0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源。
Bit4	T1M	0	Timer1 时钟源选择	0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源。
Bit3	T0M	0	Timer0 时钟源选择	0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源。
Bit[2:0]	MD2/MD1/MD0	1	MOVX 访问周期的扩展值	详见“可编程的 MOVX 访问周期”

表 1-4 PLL 状态寄存器 (PLLLCK SFR, 0xA3)

0xA3, R, PLL 状态寄存器, PLLLCK SFR				
位		默认值	功能	说明
bit[7:1]	保留	0		
bit0	PLLLCK	0	判断 PLL 是否锁定	1, 已锁定; 0, 未锁定

表 1-5 系统状态寄存器 (Systate SFR, 0xA1)

0xA1, R, 系统状态寄存器, Sysstate SFR				
位		默认值	功能	说明
bit[7:6]	保留	0		
bit5	POR	0	判断复位原因	1, 表示上次的复位是由 POR、片外输入事件 (RSTn) 或 WDT 溢出复位引起。 当其它复位发生时, 此标志位清零
bit4	保留	0		
bit3	IO	0	判断复位原因	1, 表示上次的复位是在浅睡眠或者深睡眠状态下, 由 IO 事件引起。
bit2	RTC	0	判断复位原因	1, 表示上次的复位是在浅睡眠状态下, 由 RTC 的定时唤醒引起。

0xA1, R, 系统状态寄存器, Systate SFR

位		默认值	功能	说明
bit1	PWRDN	0	检测 VDCIN 引脚输入电平, 典型阈值 1.05V	<p>PWRDN 指示灯默认开启。</p> <p>输入小于 1.0V 时, PWRDN 位为 1, 表示发生了掉电;</p> <p>输入大于 1.1V 时, PWRDN 位为 0, 表示没有发生掉电。如果使能了 PWRDN 中断, 掉电事件会向 MCU 提起中断。</p>
bit0	PWRUP	0	检测 VDCIN 引脚输入电平, 典型阈值 1.05V	<p>输入大于 1.1V 时, PWRUP 位为 1, 表示正常供电, 或者从电池供电状态切换到了正常供电状态;</p> <p>输入小于 1.0V 时, PWRUP 位为 0, 表示当前是电池供电, 或者从正常供电状态切换到了电池供电状态。</p>

注: 以上所有复位标志位互斥, 不会同时置 1。

第2章 复位

芯片的复位电路可以分为三个复位等级，不同的复位等级有着不同的复位范围，如表 2-1 所示。系统状态寄存器（Systate SFR，0xA1）中所有的复位标志位互斥。

- 复位等级 3: 包括调试复位，可以复位 CPU 及其外设和部分电能计量模块电路(包括地址为 0x2878 ~ 0x287C 的计量控制寄存器、读写缓存寄存器、功率/有效值数据寄存器、信号波形数据寄存器、功率/有效值比差寄存器、功率二次补偿值寄存器、频率值寄存器和 M 通道数据寄存器等)；
- 复位等级 2: 包括恢复供电复位、IO 休眠唤醒复位和 RTC 休眠唤醒复位，除了复位等级 3 的复位范围外，还可以复位时钟切换控制寄存器（SysCtrl SFR，0x80）、FLASH 控制寄存器和 WDT；
- 复位等级 1: 包括片外输入复位（RSTn）、上电复位（POR）和 WDT 溢出复位，除了复位等级 2 的复位范围外，还可以复位 LCD、GPIO、系统状态寄存器（Systate SFR，0xA1）、模拟控制寄存器和电能计量模块所有的寄存器。在系统状态寄存器（Systate SFR，0xA1）中 Bit5 置标志位。

RTC 校正寄存器和 RTC 计时寄存器、IRAM 和 XRAM 内部存储的数据都不受任何复位影响。在 MODE1 引脚输入低电平时，POR 以及 WDT 溢出复位被屏蔽，并且 WDT 溢出复位不在系统状态寄存器（Systate SFR，0xA1）中置标志位。

表 2-1 复位范围

电路		是否被复位?		
		等级 1	等级 2	等级 3
CPU		√	√	√
中断控制系统		√	√	√
定时器 (Timer)		√	√	√
UART		√	√	√
电 能 计 量 模 块	计量控制寄存器	√ 复位所有计量控制寄存器	√ 仅 复 位 0x2878 ~ 0x287C 的计量控制寄存器	√ 仅 复 位 0x2878 ~ 0x287C 的计量控制寄存器
	读写缓存寄存器	√	√	√
	功率/有效值数据寄存器	√	√	√
	信号波形数据寄存器	√	√	√
	功率/有效值比差寄存器	√	√	√
	功率二次补偿值寄存器	√	√	√
	频率值寄存器	√	√	√
	M 通道数据寄存器	√	√	√
	带通滤波器系数寄存器	√	X	X
	能量桶/脉冲计数器	√	X	X
门限值/常数功率值寄存器	√	X	X	

电路	是否被复位?		
	等级 1	等级 2	等级 3
时钟切换控制寄存器 (SysCtrl SFR, 0x80)	√	√	X
FLASH 控制寄存器	√	√	X
WDT	√	√	X
LCD	√	X	X
GPIO	√	X	X
系统状态寄存器 (Systate SFR, 0xA1)	√	X	X
模拟控制寄存器	√	X	X
RTC 其它寄存器	√	X	X
RTC 校正寄存器	X	X	X
RTC 计时寄存器	X	X	X
IRAM	X	X	X
XRAM 内部存储的数据	X	X	X

复位电路的框图如图 2-1 所示。

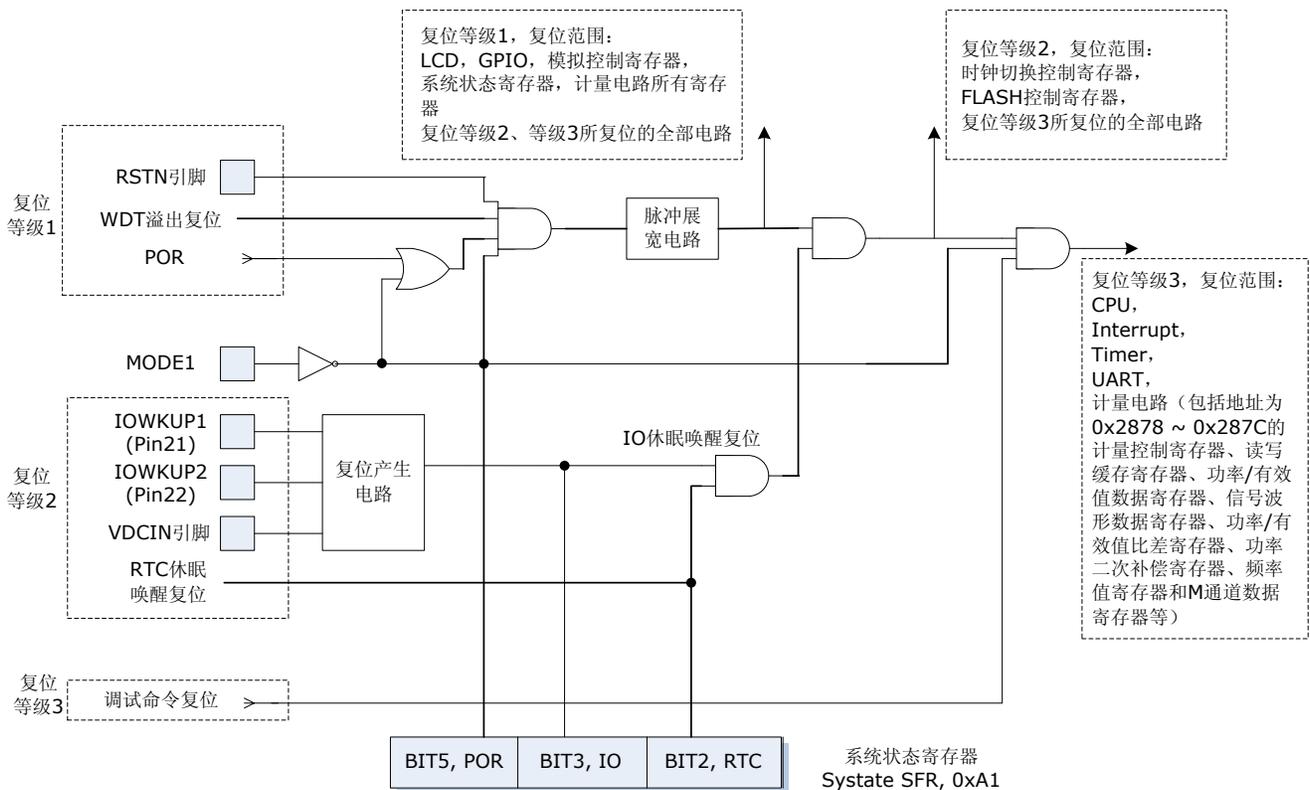


图 2-1 复位电路框图

2.1 复位等级 3

复位等级 3 包括调试复位。复位范围为 CPU、中断、定时器、UART、计量电路的部分寄存器（包括地址为 0x2878 ~ 0x287C 的计量控制寄存器、读写缓存寄存器、功率/有效值数据寄存器、功率/有效值比差寄存器、功率二次补偿值寄存器、频率值寄存器和 M 通道数据寄存器等）。

片上实时调试器接受复位调试命令后，产生调试复位。使用提供的调试开发环境进行调试的时候，最初进入调试界面，以及点击“复位”图标，都会使片上实时调试器接收到调试命令，产生调试复位。在使用提供的 ISP 编程器进行编程的时候，编程器也会发出调试复位命令。

2.2 复位等级 2

复位等级 2 包括恢复供电复位、IO 休眠唤醒复位和 RTC 休眠唤醒复位。除了复位等级 3 的复位范围外，还可以复位时钟切换控制寄存器（SysCtrl SFR, 0x80）、FLASH 控制寄存器和 WDT，详见表 2-1 复位范围。

2.2.1 恢复供电复位

在 VDCIN 引脚输入电平从低于 1.0V 到高于 1.1V 的过程中；或者等级 1 的复位发生后，VDCIN 引脚电平高于 1.1V，芯片内部的复位管理电路会产生恢复供电复位。持续 8 个 OSC 时钟周期。

2.2.2 IO 休眠唤醒复位

在进入浅睡眠或深睡眠状态前，将引脚 IOWKUP1 (Pin21) 或 IOWKUP2 (Pin22) 设置为输入，在浅睡眠或者深睡眠状态下，当其中一个 IO 端口上发生高电平到低电平的跳变（跳变的高低电平需各自持续至少 4 个 OSC 时钟周期）时，芯片内部的复位管理电路就会产生 IO 唤醒复位，持续 8 个 OSC 时钟周期。两个 IO 休眠唤醒输入相互独立，互不影响。IO 休眠唤醒复位会在系统状态寄存器 (Systate SFR, 0xA1) 的 Bit3 置标志位（见图 2-1 复位电路框图），供程序查询。

VDCIN 引脚仅在休眠的情况下导致唤醒复位，且高电平有效。

2.2.3 RTC 休眠唤醒复位

在浅睡眠状态下，RTC 根据设置的休眠唤醒间隔定时输出复位信号。持续 8 个 OSC 时钟周期。RTC 休眠唤醒复位会在系统状态寄存器 (Systate SFR, 0xA1) 的 Bit2 置标志位（见图 2-1 复位电路框图），供程序查询。

2.3 复位等级 1

复位等级 1 包括 WDT 溢出复位、片外输入的 RSTn 复位和模拟电路产生的 POR 复位。复位范围为芯片上除了 RTC 校正寄存器和 RTC 计时寄存器之外的全部电路，详见表 2-1 复位范围。在系统状态寄存器中 Bit5 置标志位。

2.3.1 片外输入复位 RSTN

RSTn 输入使用 RC 振荡器时钟进行滤波，以防静电干扰。当引脚 RSTn 上输入低电平保持 5ms 以上时，片外输入复位有效。

2.3.2 上电复位 (POR)

上电复位 (POR) 电路监测数字电源电路输出的电平 (DVCC)。当数字电源电路的输出电压大于 1.8V 时，复位信号释放；低于 1.8V 的时候，复位输出信号一直有效。阈值电平 1.8V 是一个典型值，不同芯片之间存在差别。POR 同样会被芯片内部电路展宽。

在调试模式下 (MODE1 引脚输入低电平)，POR 被屏蔽。

2.3.3 WDT 溢出复位

WDT 溢出时产生 WDT 溢出复位。复位持续时间 240 μ s。WDT 溢出复位会在系统状态寄存器 (Systate SFR, 0xA1) 的 Bit5 置标志位（见图 2-1 复位电路框图），供程序查询。

在调试模式下 (MODE1 引脚输入低电平)，WDT 溢出复位被屏蔽，且不会在 Sysstate SFR 中置标志位。

第3章 8052 MCU 结构

V9811 芯片内部存储器包括三个模块：

- 64kB Flash 存储器
- 256-Byte 的内部 SRAM (IRAM)
- 4kB 的外部 SRAM (XRAM)

IRAM 和 SFR 映射到同一地址空间。XRAM 以及外设寄存器地址映射到数据存储器空间。FLASH 映射到程序存储空间。

3.1 IRAM

IRAM 是 CPU 的内部 RAM 存储器，共 256 字节。实现 CPU 的堆栈、工作寄存器组以及通用片内存储器（高 128 字节，只能间接寻址）。IRAM 中保存的数据在 DVCC 输出电压大于 1.7V 的条件下不会丢失，也不受复位的影响。地址范围 0x00 ~ 0xFF。

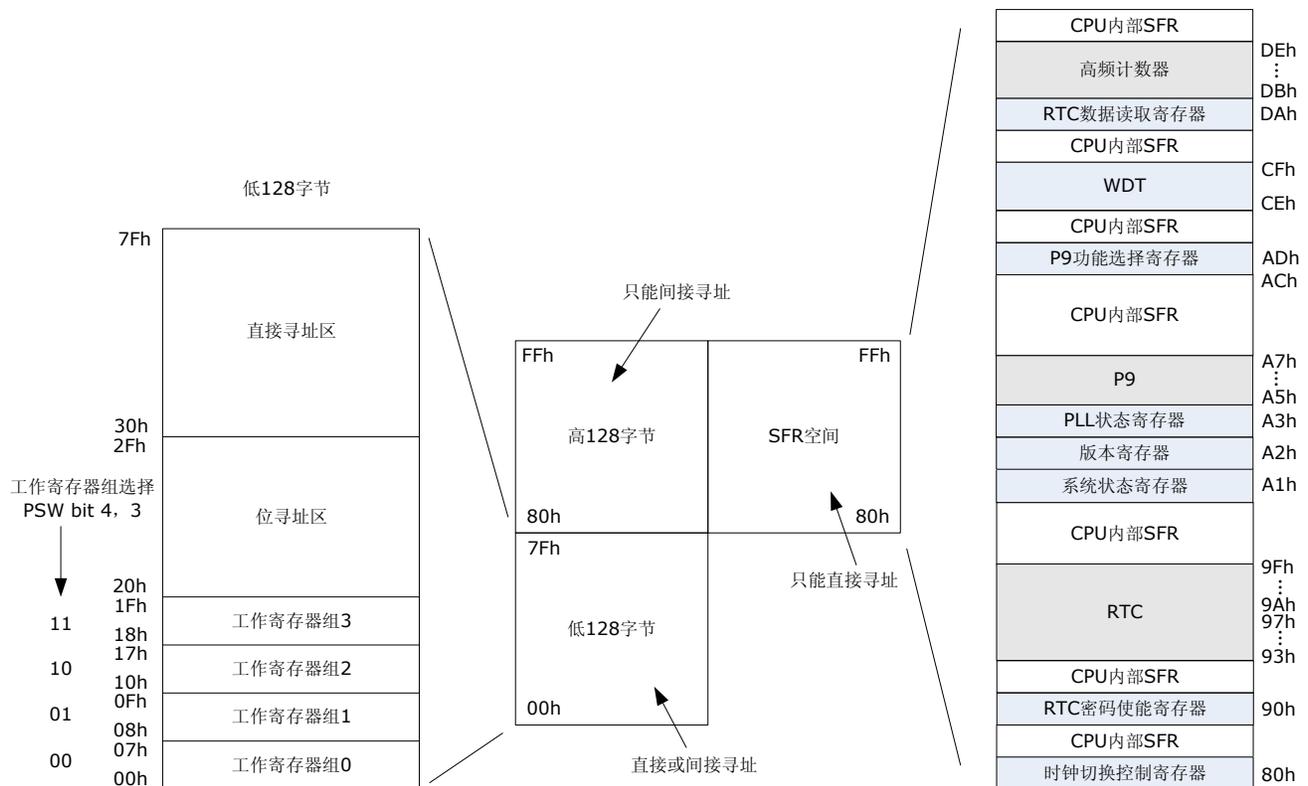


图 3-1 IRAM 和 SFR

低 128 字节 IRAM 的空间组织结构如图 3-1 所示。最低端的 32 个字节地址 00h ~ 1Fh 形成了四组分别由 8 个寄存器（R0 到 R7）构成的工作寄存器组。通过配置程序状态字寄存器（PSW SFR, 0xD0）的 BIT4（RS1）和 Bit3（RS0）两个位，选择当前使用的是哪个工作寄存器组，详见表 3-1。地址 20h ~ 2Fh 共 16 个字节形成了可位寻址存储区域，位地址从 00h 到 7Fh。所有的低 128 字节 IRAM 都可以利用直接寻址或者间接寻址的方式进行访问。

表 3-1 寄存器组选择

寄存器	位		默认值	说明
0xD0, 程序状态字寄存器, PSW SFR	Bit[4:3]	RS1/RS0	0	寄存器组选择位。默认为 0。 00, 工作寄存器组 0, 地址: 00h~07h; 01, 工作寄存器组 1, 地址: 08h~0Fh; 10, 工作寄存器组 2, 地址: 10h~17h; 11, 工作寄存器组 3, 地址: 18h~1Fh

高 128 字节 IRAM 与 SFR 寄存器共享一段相同的地址范围 (80h 到 FFh)。但是, 上述两者真正的地址空间是相互独立的, 并且利用不同的寻址方式进行访问: SFR 可直接寻址访问; 高 128 字节 IRAM 只能间接寻址访问。

3.2 数据存储器空间

4096 字节的 XRAM 以及外设寄存器地址都映射到数据存储器空间 (图 3-2)。

XRAM 的地址范围为 0000h~ 0FFFh。其中, 0000h ~ 0F7Fh 可以被自由访问; 0F80h ~ 0FFFh 的写操作受写保护寄存器控制, 在对 0F80h ~ 0FFFh 区域进行写操作之前, 必须先访问“XRAM 高 128 字节写入保护寄存器 (XRamPwd, 0x28A0)”, 取消写入保护。写保护取消后, 只要没有恢复写保护, 都可以对这个区域进行写入操作。XRAM 中保存的数据在 DVCC 输出电压大于 1.7V 的条件下不会丢失, 也不受复位的影响。

表 3-2 XRAM 高 128 字节写入保护寄存器 (XRamPwd, 0x28A0)

0x28A0, R/W, XRAM 高 128 字节写入保护寄存器, XRamPwd									
功能	值	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
取消写入保护	0x33	0	0	1	1	0	0	1	1
开启写入保护	0x32	0	0	1	1	0	0	1	0
默认值	0x00	0	0	0	0	0	0	0	0

注: 只能读出 bit0 的值

数据存储空间中, 除 XRAM 外, 所有外设寄存器 (包括 XRAM 写保护寄存器) 均可被复位。其中, LCD/GPIO/模拟控制寄存器/部分电能计量模块寄存器只会被等级 1 的复位为默认值, 其他寄存器会被等级 1/2/3 的复位为默认值。

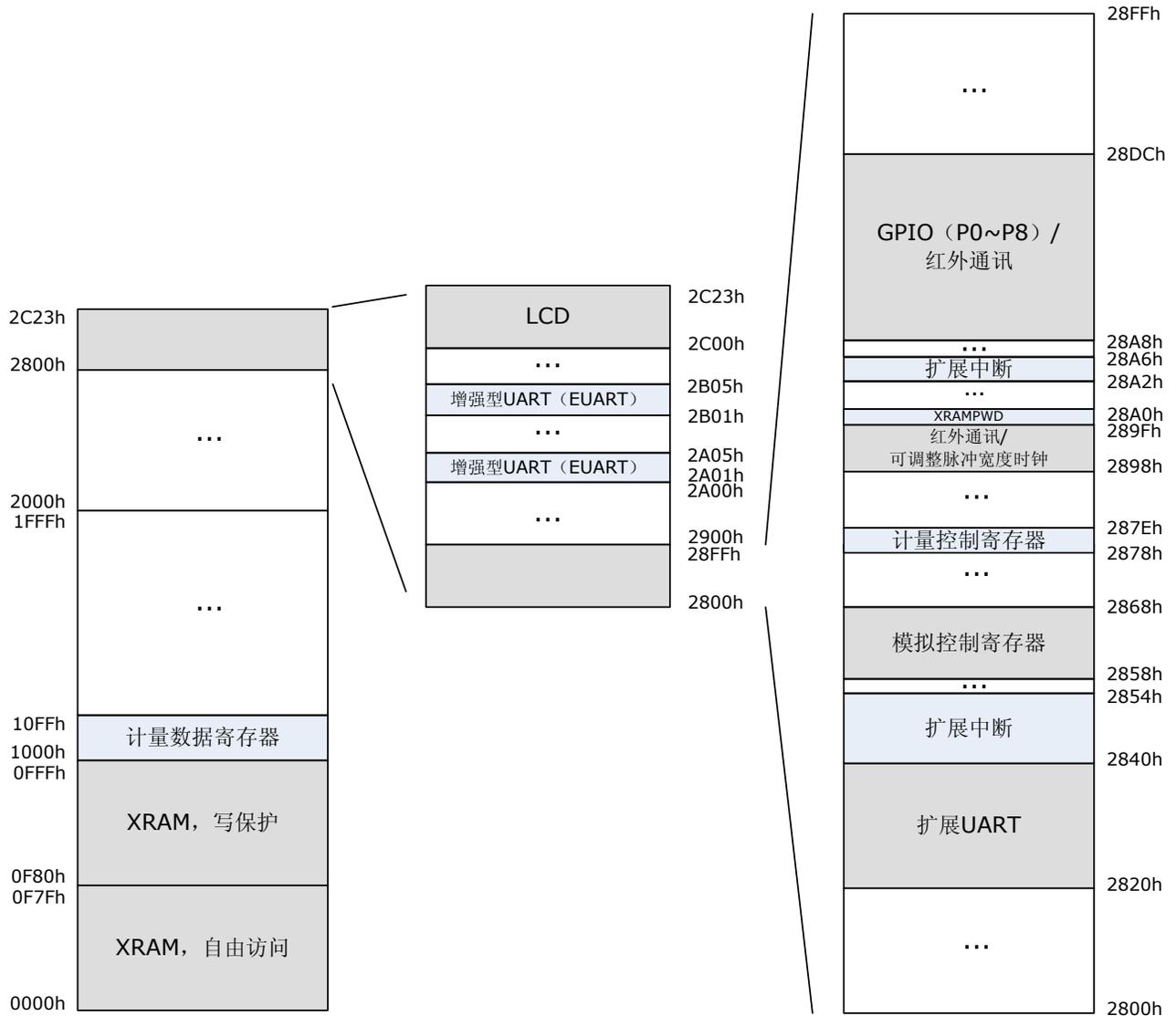


图 3-2 数据存储空间分配图

3.3 程序存储空间

64kB 的 FLASH 以及 FLASH 控制寄存器和程序加密字节都映射到程序存储空间。



图 3-3 程序存储空间分配图

FLASH 控制寄存器（0x0401 和 0x0402）控制着 FLASH 的编程模式和功耗模式等。向 FLASH 控制寄存器 0x0402 写入 0x86，FLASH 的功耗会降低，但不会降低性能。

```
unsigned char xdata *p;
SPC_FNC = 0x01;
p = 0x0402;
*p = 0x86;
SPC_FNC = 0x00;
```

3.4 调试模式、ISP 和程序加密

3.4.1 调试模式

当 MODE1 引脚输入低电平时，芯片处于调试模式。在调试模式下，系统无法进入休眠，POR/BOR 和 WDT 溢出复位被屏蔽，且 WDT 溢出复位不会在系统状态寄存器（Systate SFR, 0xA1）中置标志位；IO 的 P0 端口复用为 JTAG 端口，通过 JTAG 端口可以访问片内硬件的实时调试电路，进行各种调试以及 ISP 编程操作。

当处于休眠状态下时，系统进入调试模式会产生一个恢复供电复位信号。

在访问片内硬件的实时调试电路时，输入 0x22 调试命令可以将 TCK 的速度限制从 400Kbps 提高到 3.2Mbps，输入 0x23 调试命令恢复速度限制。系统复位后，默认速度限制为 400Kbps。JTAG 端口（P0）不能接电容，否则会导致程序烧写失败。

3.4.2 ISP

使用提供的 DLL，在 Keil μ Vision IDE 环境下进行调试和 ISP 下载操作。使用提供的编程器进行独立 ISP 下载操作。调试时，请注释掉源程序中关于从 PLL 切换到 OSC 时钟，进行休眠等语句。

3.4.3 程序加密

FLASH 存储空间中 0400h 字节的 bit0 位是程序加密位。

JTAG 对 FLASH 存储空间 0000h ~ FFFFh 的全擦除和写操作都不受程序加密位的影响。

当程序加密位为 0 时，JTAG 对 FLASH 存储空间 0000h ~ FFFFh (0kB ~ 64kB) 的读操作和页擦除操作被禁止。

当程序加密位为 1 时，JTAG 对 FLASH 存储空间 0000h~FFFFh 可以进行无限制访问。

ISP 操作对 FLASH 不同空间的访问限制，详见表 3-3。

表 3-3 ISP 操作对 FLASH 不同空间的访问限制

访问方式	地址空间	加密位为 0				加密位为 1			
		全擦除	写操作	读操作	页擦除	全擦除	写操作	读操作	页擦除
ISP 操作 通过 JTAG 访问	0000h~FFFFh	√	√	X	X	√	√	√	√

ISP 操作结束后，需要对芯片进行硬件复位（RSTn 引脚低电平有效）或者重新上电操作，以保证 ISP 读操作加密保护功能可靠生效。

3.5 MCU

3.5.1 指令集

指令集与业界标准的 8051 指令二进制代码兼容，指令集的功能也完全一样。不过，每个指令周期所需的时钟周期个数以及每个指令周期的执行时序与标准 8051 指令集的时序有所差异。一个指令周期包含 4 个时钟周期。

表 3-4 指令集

符号	功能
A	累加器
Rn	寄存器 R0 到 R7
Direct	直接寻址的内部寄存器地址
@Ri	由 R0 或者 R1 作为指针所指向的内部寄存器（MOVX 指令除外）
Rel	补码表示的偏移地址
Bit	直接寻址的位地址
#data	8 位常数
#data 16	16 位常数

addr 16	16 位目标地址			
addr 11	11 位目标地址			
指令	功能描述	字节数	指令周期	Hex 格式代码
算术指令				
ADD A, Rn	把 Rn 的值加到 A 上	1	1	28 - 2F
ADD A, direct	把直接寻址的内部寄存器地址中的内容加到 A 上	2	2	25
ADD A, @Ri	把 Ri 寻址的内部寄存器的值加到 A 上	1	1	26 - 27
ADD A, #data	把立即数的值加到 A 上	2	2	24
ADDC A, Rn	连同进位一起把 Rn 的值加到 A 上	1	1	38 - 3F
ADDC A, direct	连同进位一起把直接寻址的内部寄存器地址中的内容加到 A 上	2	2	35
ADDC A, @Ri	连同进位一起把 Ri 寻址的内部寄存器的值加到 A 上	1	1	36 - 37
ADDC A, #data	连同进位一起把立即数加到 A 上	2	2	34
SUBB A, Rn	连同借位一起从 A 中减去寄存器的值	1	1	98 - 9F
SUBB A, direct	连同借位一起从 A 中减去直接寻址的内部寄存器地址中的内容	2	2	95
SUBB A, @Ri	连同借位一起从 A 中减去 Ri 寻址的内部寄存器的值	1	1	96 - 97
SUBB A, #data	连同借位一起从 A 中减去立即数的值	2	2	94
INC A	A 的值自加 1	1	1	04
INC Rn	Rn 的内容自加 1	1	1	08 - 0F
INC direct	直接寻址的内部寄存器地址字节内容自加 1	2	2	05
INC @Ri	Ri 寻址的内部寄存器的值自加 1	1	1	06 - 07
DEC A	A 的值自减 1	1	1	14
DEC Rn	Rn 的内容自减 1	1	1	18 - 1F
DEC direct	直接寻址的内部寄存器地址字节内容自减 1	2	2	15
DEC @Ri	Ri 寻址的内部寄存器内容自减 1	1	1	16 - 17
INC DPTR	DPTR 数据指针自加 1	1	3	A3
MUL AB	A 乘 B	1	5	A4
DIV AB	除法运算, A 是被除数, B 是除数	1	5	84
DA A	对 A 进行十进制调整	1	1	D4
逻辑运算指令				
ANL A, Rn	Rn 对 A 进行逻辑与运算	1	1	58 - 5F
ANL A, direct	直接寻址的内部寄存器地址字节内容对 A 进行逻辑与运算	2	2	35

ANL A, @Ri	Ri 寻址的内部寄存器内容对 A 进行逻辑与运算	1	1	56 - 57
ANL A, #data	立即数对 A 进行逻辑与运算	2	2	54
ANL direct, A	A 对直接寻址的内部寄存器地址字节内容进行逻辑与运算	2	2	52
ANL direct, #data	立即数对直接寻址的内部寄存器地址字节内容进行逻辑与运算	3	3	53
ORL A, Rn	Rn 对 A 进行逻辑或操作	1	1	48 - 4F
ORL A, direct	直接寻址的内部寄存器地址字节内容对 A 进行逻辑或操作	2	2	45
ORL A, @Ri	Ri 寻址的内部寄存器对 A 进行逻辑或操作	1	1	46 - 47
ORL A, #data	立即数对 A 进行逻辑或操作	2	2	44
ORL direct, A	A 对直接寻址的内部寄存器地址字节内容进行逻辑或运算	2	2	42
ORL direct, #data	立即数对直接寻址的内部寄存器地址字节内容进行逻辑或运算	3	3	43
XRL A, Rn	Rn 对 A 进行逻辑异或操作	1	1	68 - 6F
XRL A, direct	直接寻址的内部寄存器地址字节内容对 A 进行逻辑异或操作	2	2	65
XRL A, @Ri	Ri 寻址的内部寄存器对 A 进行逻辑异或操作	1	1	66 - 67
XRL A, #data	立即数对 A 进行逻辑异或操作	2	2	64
XRL direct, A	A 对直接寻址的内部寄存器地址字节内容进行逻辑异或操作	2	2	62
XRL direct, #data	立即数对直接寻址的内部寄存器地址字节内容进行逻辑异或操作	3	3	63
CLR A	清空 A 的值	1	1	E4
CPL A	对 A 的值取反	1	1	F4
SWAP A	对 A 的高 4 位和低 4 位进行互换	1	1	C4
RL A	A 向左进行逻辑移位	1	1	23
RLC A	A 通过进位 C 向左进行逻辑移位	1	1	33
RR A	A 向右进行逻辑移位	1	1	03
RRC A	A 通过进位 C 向右进行移位	1	1	13
数据传输指令				
MOV A, Rn	把 Rn 的值传输给 A	1	1	E8 - EF
MOV A, direct	把直接寻址的内部寄存器地址字节内容传输给 A	2	2	E5
MOV A, @Ri	把 Ri 寻址的内部寄存器的值传输给 A	1	1	E6 - E7
MOV A, #data	把立即数传输给 A	2	2	74
MOV Rn, A	把 A 传输给 Rn 寄存器	1	1	F8 - FF
MOV Rn, direct	把直接寻址的内部寄存器地址字节传输给 Rn 寄存器	2	2	A8 - AF
MOV Rn, #data	把立即数传输给 Rn 寄存器	2	2	78 - 7F

MOV direct, A	把 A 传输给直接寻址的内部寄存器地址字节	2	2	F5
MOV direct, Rn	把 Rn 的内容传输给直接寻址的内部寄存器地址字节	2	2	88 - 8F
MOV direct, direct	把直接寻址的内部寄存器地址字节内容传输给另外一个直接寻址的内部寄存器地址字节	3	3	85
MOV direct, @Ri	把 Ri 寻址的内部寄存器的值传输给直接寻址的内部寄存器地址字节	2	2	86 - 87
MOV direct, #data	把立即数送到直接寻址的内部寄存器地址字节	3	3	75
MOV @Ri, A	把 A 的值传输到 Ri 寻址的内部寄存器	1	1	F6 - F7
MOV @Ri, direct	把直接寻址的内部寄存器地址字节的值传输给 Ri 寻址的内部寄存器	2	2	A6 - A7
MOV @Ri, #data	把立即数送到 Ri 寻址的内部寄存器	2	2	76 - 77
MOV DPTR, #data	把立即数送到数据指针寄存器中	3	3	90
MOVC A, @A+DPTR	把 DPTR+A 寻址的程序指令字节传输到 A 中	1	3	93
MOVC A, @A+PC	把 PC+A 寻址的程序指令字节传输到 A 中	1	3	83
MOVX A, @Ri	把 Ri 寻址的外部数据空间数据（8 位地址）传输到 A	1	2 9*	E2 - E3
MOVX A, @DPTR	把 DPTR 寻址的外部数据空间数据（16 位地址）传输到 A	1	2 9*	E0
MOVX @Ri, A	把 A 的内容传输到 Ri 寻址的外部数据空间字节（8 位地址）	1	2 9*	F2 - F3
MOVX @DPTR, A	把 A 的内容传输到 DPTR 寻址的外部数据空间字节（16 位地址）	1	2 9*	F0
PUSH direct	把直接寻址的内部寄存器地址字节内容压入堆栈顶端	2	2	C0
POP direct	把堆栈顶端字节出栈，送入直接寻址的内部寄存器地址字节	2	2	D0
XCH A, Rn	A 和 Rn 寄存器内容互换	1	1	C8 - CF
XCH A, direct	A 和直接寻址的内部寄存器地址字节内容互换	2	2	C5
XCH A, @Ri	A 和 Ri 寻址的内部寄存器内容互换	1	1	C6 - C7
XCHD A, @Ri	A 和 Ri 寻址的内部寄存器交换低半字节的内容	1	1	D6 - D7

* 通过寄存器可以选择指令周期个数。

位操作指令				
CLR C	清除进位	1	1	C3
CLR bit	清除直接寻址位	2	2	C2
SETB C	置进位	1	1	D3

SETB bit	置直接寻址位	2	2	D2
CPL C	进位取反	1	1	B3
CPL bit	直接寻址位取反	2	2	B2
ANL C, bit	直接寻址位对进位作与操作	2	2	82
ANL C, /bit	直接寻址位取反后对进位作与操作	2	2	B0
ORL C, bit	直接寻址位对进位作或运算	2	2	72
ORL C, /bit	直接寻址位取反后对进位作或运算	2	2	A0
MOV C, bit	把直接寻址位传输到进位	2	2	A2
MOV bit, C	把进位传输到直接寻址位	2	2	92
分支跳转指令				
ACALL addr 11	子程序的绝对调用	2	3	11 - F1
LCALL addr 16	子程序的长调用	3	4	12
RET	从子程序中返回	1	4	22
RETI	从中断中返回	1	4	32
AJMP addr 11	无条件绝对跳转	2	3	01 - E1
LJMP addr 16	无条件长跳转	3	4	02
SJMP rel	短跳转到相对地址	2	3	80
JC rel	在进位 carry = 1 的情况下跳转到相对地址	2	3	40
JNC rel	在进位 carry = 0 的情况下跳转到相对地址	2	3	50
JB bit, rel	在直接寻址位等于 1 的情况下跳转到相对地址	3	4	20
JNB bit, rel	在直接寻址位等于 0 的情况下跳转到相对地址	3	4	30
JBC bit, rel	在直接寻址位等于 1 的情况下跳转到相对地址，并清除进位	3	4	10
JMP @A + DPTR	跳转到 A + DPTR 间接寻址的地址	1	3	73
JZ rel	在累加器 ACC 等于 0 的情况下跳转到相对地址	2	3	60
JNZ rel	在累加器 ACC 不等于 0 的情况下跳转到相对地址	2	3	70
CJNE A, direct, rel	比较 A 和直接寻址字节内容，在不相等的情况下跳转到相对地址	3	4	B5
CJNE A, #d, rel	比较 A 和立即数的值，在不相等的情况下跳转到相对地址	3	4	B4
CJNE Rn, #d, rel	比较 Rn 寄存器和立即数的值，在不相等的条件下跳转到相对地址	3	4	B8 - BF
CJNE @Ri, #d, rel	比较 Ri 简介寻址的内部寄存器和立即数的值，在不相等的条件下跳转到相对地址	3	4	B6 - B7
DJNZ Rn, rel	Rn 寄存器递减，如果不等于 0 则跳转到相对地址	2	3	D8 - DF
DJNZ direct, rel	直接寻址字节的内容递减，如果不等于 0 则跳转到相对地址	3	4	D5

其他指令

NOP	空操作	1	1	00
-----	-----	---	---	----

还有一个保留的指令代码（A5）与 NOP 指令的作用相同。

3.5.2 可编程的 MOVX 访问周期

应用程序利用存储器访问周期可编程这个特点，可以调整访问数据存储器或者寄存器的速度。CPU 可以在最少两个指令周期内完成执行 MOVX 指令，但是某些情况下可能不需要这么快。CPU 中的时钟周期控制寄存器（CKCON SFR, 0x8E）中的最低 3 位控制着 MOVX 访问周期的扩展值。MD2 ~ 0 是 CKCON SFR 的低 3 位（CKCON.2 ~ 0），可以在 0 到 7 之间设置存储器访问周期的扩展值。扩展值为 0，表明在存储器访问周期中不增加任何扩展的周期，所以 MOVX 指令的执行依然仅需要两个指令周期。扩展值为 7，表明在存储器访问周期中增加了额外的 7 个指令周期，所以 MOVX 指令的执行需要 9 个指令周期。在程序的控制下，扩展周期值可以随时动态地进行改变。扩展周期值会影响读/写控制信号脉冲的宽度以及所有其他有关的时序。扩展周期值越大，读/写控制信号脉冲的宽度就越宽。在 CPU 复位后，扩展周期的默认值是 1，即执行 MOVX 指令需要 3 个指令周期。

表 3-5 可编程的 MOVX 访问周期

设置存储器访问周期扩展值			存储器访问周期	读/写脉冲宽度（时钟周期个数）
MD2	MD1	MD0		
0	0	0	2	2
0	0	1	3（默认值）	4
0	1	0	4	8
0	1	1	5	12
1	0	0	6	16
1	0	1	7	20
1	1	0	8	24
1	1	1	9	28

3.5.3 双数据指针

在进行大量的块数据移动时，使用双数据指针可以大幅提高效率。

CPU 中标准数据指针为 DPTR0，SFR 地址是 0x82（DPL0 SFR）和 0x83（DPH0 SFR）。第二个数据指针为 DPTR1，SFR 地址是 0x84（DPL1 SFR）和 0x85（DPH1 SFR）。在 DPTR 选择寄存器（DPS SFR, 0x86）中，通过设置 Bit 0（SEL 位）选择当前有效的数据指针。当 SEL = 0 时，程序所使用的 DPTR 是 DPL0 SFR（0x82）和 DPH0 SFR（0x83）。当 SEL = 1 时，程序所使用的 DPTR 是 DPL1 SFR（0x84）和 DPH1 SFR（0x85）。除 Bit0（SEL 位）外，DPS SFR 中的其他位都没有被使用。

所有与 DPTR 有关的指令都使用当前被选中的数据指针。切换 SEL 位的值就可以切换当前有效的数据指针。实现这个目的最快的方法就是使用递增指令（INC DPS），从一个源地址切换到一个目标地址只需要一条指令。当进行块数据移动时，不必先保存源地址和目标地址，从而节省了程序代码的数量。

第4章 模拟控制寄存器

当发生 POR 复位、RSTn 引脚低电平有效，或 WDT 溢出复位（详见复位等级 1）时，所有模拟控制寄存器被复位。

模拟控制寄存器的地址范围为 0x2858~0x2868，均可读可写。其中，为保证系统正常工作，地址为 0x2865 的寄存器的值必须保证为 0。

表 4-1 ADC 控制寄存器 0 (CtrlADC0, 0x2858)

0x2858, R/W, ADC 控制寄存器 0, CtrlADC0				
位		默认值	功能	说明
bit7	ADCGM	0	测量通道 (M) ADC 的模拟增益控制 (电容)	0, ×1; 1, ×2
bit6	ADCGU	0	电压通道 (U) ADC 的模拟增益控制 (电容)	0, ×1; 1, ×2
bit[5:3]	ADCGB<2:0>	0	电流通道 B (IB) ADC 的模拟增益控制 (电容)	000, ×1; 001, ×4; 010, ×8; 011, ×16; 111, ×40
bit[2:0]	ADCGA<2:0>	0	电流通道 A (IA) ADC 的模拟增益控制 (电容)	000, ×1; 001, ×4; 010, ×8; 011, ×16; 111, ×40

表 4-2 ADC 控制寄存器 1 (CtrlADC1, 0x2859)

0x2859, R/W, ADC 控制寄存器 1, CtrlADC1				
位		默认值	功能	说明
bit[7:6]	ADIT2M<1:0>	0	M 通道 ADC 第二级偏置电流调整	00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5
bit5	REFGIT	0	ADC 参考产生偏置电流调整	0, ×1; 1, ×1.5
bit4	REFBIT	0	ADC 参考 Buffer 偏置电流调整	0, ×1; 1, ×1.33
bit[3:2]	ADIT2<1:0>	0	ADC 第二级偏置电流调整	00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5
bit[1:0]	ADIT1<1:0>	0	ADC 第一级偏置电流调整	00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5

表 4-3 ADC 控制寄存器 2 (CtrlADC2, 0x285A)

位		默认值	功能	说明
bit[7:5]	保留	0		必须保证为默认值
bit4	DITAMP	0	电压通道放大器偏置电流调整	0, ×1; 1, ×0.5
bit3	ADRSTM	0	M 通道调制器积分器复位	1, 使能。默认为 0。
bit2	ADRSTU	0	电压通道调制器积分器复位	1, 使能。默认为 0。
bit1	ADRSTB	0	通道 IB 调制器积分器复位	1, 使能。默认为 0。
bit0	ADRSTA	0	通道 IA 调制器积分器复位	1, 使能。默认为 0。

表 4-4 ADC 控制寄存器 3 (CtrlADC3, 0x285B)

位		默认值	功能	说明
bit7	CLKOSEL	0	ADC 时钟反相	1, 使能。默认为 0。
bit6	ADCLKEXT	0	选择 ADC 时钟来源	1, OSC 时钟; 0, PLL 时钟
bit[5:4]	保留	0		必须保证为默认值。
bit3	SHORTV2	0	电压 ADC 内部短路 2	1, 使能。默认为 0。
bit2	SHORTV	0	电压 ADC 内部短路 1	1, 使能。默认为 0。
bit1	SHORTI2	0	电流 ADC 内部短路 2	1, 使能。默认为 0。
bit0	SHORTI	0	电流 ADC 内部短路 1	1, 使能。默认为 0。

表 4-5 电池放电控制寄存器 (CtrlBAT, 0x285C)

0x285C, R/W, 电池放电控制寄存器, CtrlBAT				
位		默认值	功能	说明
bit[7:1]	保留	0		必须保证为默认值。
bit0	BATDISC	0	使能系统供电电池放电。	1, 使能。默认为 0。

表 4-6 ADC 控制寄存器 4 (CtrlADC4, 0x285D)

0x285D, R/W, ADC 控制寄存器 4, CtrlADC4				
位		默认值	功能	说明
bit[7:2]	保留	0		必须保证为默认值。
bit[1:0]	TRB<1:0>	0	ADC 内部偏置调整	1, 使能。默认为 0。

表 4-7 LCD 驱动电压控制寄存器 (CtrlLCDV, 0x285E)

0x285E, R/W, LCD 驱动电压控制寄存器, CtrlLCDV				
位		默认值	功能	说明
bit[7:3]	保留	0		必须保证为默认值。
bit[2:0]	LCDV<2:0>	0	LCD 驱动电压调整	000, 3.3V; 001, 3.2V; 010, 3.1V; 011, 3.0V; 100, 2.9V; 101, 2.8V; 110, 2.7V; 111, 2.7V

当 V9811 由 5V 主电源供电时 (引脚 V3V 上的电平为低), LDO33 输出电压为 3.3V, 此时 LCD 的驱动电压为配置值。当 V9811 由 3.3V 主电源 (引脚 V3V 上的电平为高) 或电池供电时, 实际的 LCD 驱动电压为上述配置值与系数 K 的乘积, K 为 LDO33 输出电压与 3.3V 的比值。

表 4-8 晶振控制寄存器 0 (CtrlCry0, 0x285F)

0x285F, R/W, 晶振控制寄存器 0, CtrlCry0				
位		默认值	功能	说明
bit[7:0]	CAPSEL<7:0>	0	晶体起振频率调节	默认为 0。

表 4-9 晶振控制寄存器 1 (CtrlCry1, 0x2860)

0x2860, R/W, 晶振控制寄存器 1, CtrlCry1				
位		默认值	功能	说明
bit[7:6]	保留	0		必须保证为默认值
bit5	XEASYN	0	MOS 电阻大小调节	1, 2 倍电阻。默认为 0。
bit4	CSEL	0	晶体起振电容调节	1, 增加 2.35pF 电容。默认为 0。
bit[3:2]	保留	0		必须保证为默认值
bit[1:0]	XTRSEL<1:0>	0	起振电路电阻调节	使用 RTC 时, 必须配置为 0b11, 此时必须开启停振检测功能

表 4-10 晶振控制寄存器 2 (CtrlCry2, 0x2861)

0x2861, R/W, 晶振控制寄存器 2, CtrlCry2				
位		默认值	功能	说明
bit[7:6]	保留	0		必须保证为默认值。
bit5	XRESETEN	0	起振电路复位使能位	1, 使能。默认为 0。
bit4	保留	0		
bit[3:0]	保留	0		必须保证为默认值。

表 4-11 BGP 控制寄存器 (CtrlBGP, 0x2862)

0x2862, R/W, BGP 控制寄存器, CtrlBGP				
位		默认值	功能	说明
bit[7:6]	CURRIT<1:0>	0	模拟全局偏置电流调整	00, 1; 01, -33%; 10, -45%; 11, -60%
bit[5:4]	保留	0		必须保证为默认值。
bit[3:1]	REST<2:0>	0	调整 BandGap 温度系数	000, 0ppm; 001, +4ppm; 010, +8ppm; 011, +12ppm; 100, -16ppm; 101, -12ppm; 110, -8ppm; 111, -4ppm
bit0	BGPCHOPN	0	使能斩波去直	0, 使能。

表 4-12 ADC 控制寄存器 5 (CtrlADC5, 0x2863)

0x2863, R/W, ADC 控制寄存器 5, CtrlADC5				
位		默认值	功能	说明
bit7	PTATEN	0	选择温度传感器	1, 温度传感器 1; 0, 温度传感器 0 建议置 1。
bit[6:5]	保留	0		必须保证为默认值。
bit4	RESDIV	0	使能 M 通道输入信号分压选择, 分压系数为 0.226	1, 使能。默认为 0。
bit3	保留	0		必须保证为默认值。
bit[2:0]	MEAS<2:0>	0	选择 M 通道测试功能	000, 经由 AVSS 引脚测地; 001, 测温度; 010, 测经 BAT 引脚输入的系统电池电压信号; 011/100/101, 保留; 110, 测经 M1 引脚 (Pin55) 输入的外部直流信号; 111, 测经 M2 引脚 (Pin56) 输入的外部直流信号。

注意: 当 Pin55~56 用于外部直流信号测量输入时, 寄存器 SegCtrl4 (0x2C23) 的 bit7~bit6 必须置 0, 关闭这两个引脚上的 SEG 输出。

表 4-13 ADC 控制寄存器 6 (CtrlADC6, 0x2864)

0x2864, R/W, ADC 控制寄存器 6, CtrlADC6				
位		默认值	功能	说明
bit[7:4]	保留	0		必须保证为默认值。
bit3	ADCMPDN	0	开关 M 通道 ADC	0, 关闭 M 通道 ADC; 1, 开启 M 通道 ADC
bit2	ADCUPDN	0	开关电压通道 ADC。	0, 关闭电压通道 ADC; 1, 开启电压通道 ADC。
bit1	ADCBPDN	0	开关 IB 通道 ADC。	0, 关闭 IB 通道 ADC; 1, 开启 IB 通道 ADC。
bit0	ADCAPDN	0	开关 IA 通道 ADC。	0, 关闭 IA 通道 ADC; 1, 开启 IA 通道 ADC。

表 4-14 LDO 控制寄存器 (CtrlLDO, 0x2866)

0x2866, R/W, LDO 控制寄存器, CtrlLDO				
位		默认值	功能	说明
bit[7:6]	保留	0		必须保证为默认值。
bit5	LDO33PD	0	在掉电状态下, 使能 LDO33 切换到跟随 5V 电源的模式, 以降低功耗。	1, 关闭 LDO33。必须在 VDCIN 低于 1.0V 时, 在 LDO33PD 位写入 1, 8ms 后, LDO33 停止工作。
bit4	P3V5	0	LDO33 电压调节	0, 3.3V; 1, 3.5V
bit[3:0]	保留	0		必须保证为默认值。

表 4-15 时钟控制寄存器 (CtrlCLK, 0x2867)

0x2867, R/W, 时钟控制寄存器, CtrlCLK				
位		默认值	功能	说明
bit7	PLLPDN	0	开关 PLL 电路	0, 关闭; 1, 正常工作 开启 PLL 之前必须先开启 BGP 电路。
bit6	BGPPDN	0	开关 BandGap 电路和电流通道。	0, 关闭; 1, 正常工作 开启 PLL 之前必须先开启 BGP 电路。
bit[5:4]	ADCLKSEL<1:0>	0	过采样 ADC 时钟频率选择, 以 204.8kHz 为基准。	00, ×1; 01, ×2; 10, ×4; 11, ×8
bit[3:2]	MEACLKSEL<1:0>	0	电能计量时钟频率选择, 以 819.2kHz 为基准。	00, ×1; 01, ×2; 10, ×4; 11, ×8
bit[1:0]	MCUCLKSEL<1:0>	0	MCU 时钟频率选择, 以 819.2kHz 为基准。	00, ×1; 01, ×2; 10, ×4; 11, ×8

表 4-16 PLL 控制寄存器 (CtrlPLL, 0x2868)

0x2868, R/W, PLL 控制寄存器, CtrlPLL				
位		默认值	功能	说明
bit7	MCU26M	0	当 MCU13M=1 时, 使能 MCU 时钟频率加倍 (×2)	1, 使能。
bit6	MCU13M	0	使能 MCU 时钟频率加倍 (×2)	1, 使能。
bit5	PLLSEL	0	50Hz 或 60Hz 应用选择	0, 50Hz; 1, 60Hz

0x2868, R/W, PLL 控制寄存器, CtrIPLL

位		默认值	功能	说明
bit4	PLLLF	0	PLL 降频	1, 降频 1/2
bit3	MEAINV	0	电能计量时钟反相使能	1, 使能。默认为 0。
bit2	MCUINV	0	MCU 时钟反相使能	1, 使能。默认为 0。
bit[1:0]	PLLIB<1:0>	0	PLL 电荷泵偏置电流调节	00, ×1; 01, ×2; 10, ×2; 11, ×3

第5章 时钟

在 V9811 中，时钟系统包括 RC 振荡电路提供的 RC 时钟、晶振电路提供的 OSC 时钟和模拟锁相环（PLL）电路提供的 PLL 时钟。它们可分别作为 MCU 时钟、电能计量时钟和 ADC 时钟的时钟源。

5.1 时钟控制电路

根据芯片的各个功能单元所使用的时钟类型，时钟控制电路的输出可以分为以下几个时钟。

- 时钟 1：可关闭，供 CPU、RAM、FLASH、中断/定时器/UART/IO 使用，由 OSC 或 PLL 时钟作时钟源；
- 时钟 2：可关闭，供电能计量模块使用，由 OSC 或 PLL 时钟作时钟源；
- 时钟 3：可关闭，供 LCD 使用，由 OSC 时钟作时钟源；
- 时钟 4：不可关闭，供 WDT 使用，由 RC 时钟作时钟源；
- 时钟 5：不可关闭，供 RTC 使用，由 OSC 时钟作时钟源。

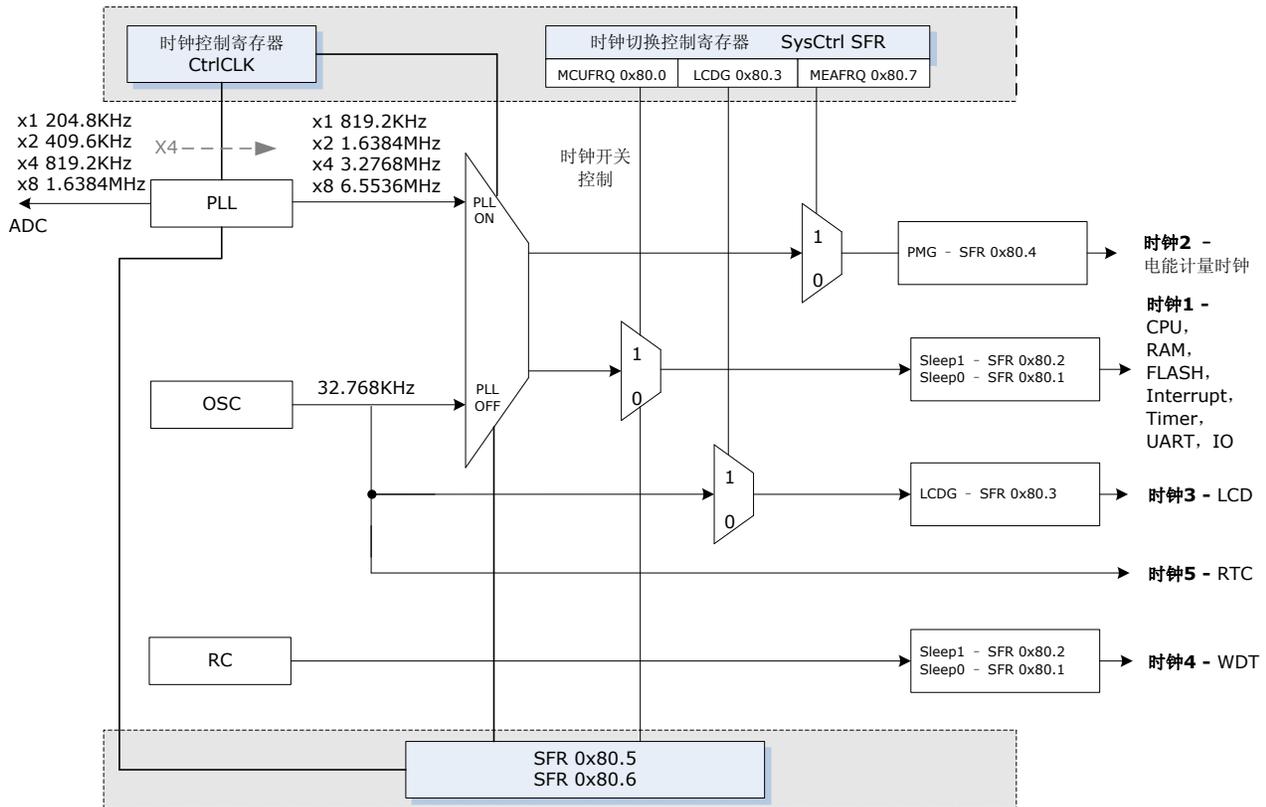
各个时钟的时钟源、开关和时钟频率调整分别由时钟切换控制寄存器（SysCtrl SFR, 0x80）和时钟控制寄存器（CtrlCLK, 0x2867）控制，如图 5-1 所示。

表 5-1 时钟控制

寄存器	位	功能	说明
时钟切换控制寄存器 (SysCtrl SFR, 0x80)	bit7	MEAFRQ	选择电能计量时钟源 0, OSC 时钟； 1, PLL 时钟。 配置这个位切换电能计量时钟源；读取该位的值以显示当前电能计量时钟源。
	bit4	PMG	开关电能计量时钟（时钟 2） 0, 开启； 1, 关闭。 当时钟 1 和时钟 2 的时钟源为 PLL 时钟时，时钟 3 才可被关闭。
	bit3	LCDG	开关 LCD 时序产生电路的时钟（时钟 3） 0, 开启； 1, 关闭。 当时钟 1 和时钟 2 的时钟源为 PLL 时钟时，时钟 3 才可被关闭。
	bit[2:1]	SLEEP1/SLEEPO	开启或者关闭时钟 1。 11/01, 系统进入浅睡眠； 10, 系统进入深睡眠
	bit0	MCUFRQ	选择 MCU 时钟源（时钟 1）。 0, OSC 时钟； 1, PLL 时钟。配置这个位切换 MCU 时钟源；读取该位的值以显示当前 MCU 的时钟源。
时钟控制寄存器 (CtrlCLK, 0x2867)	bit[5:4]	ADCLKSEL<1:0>	过采样 ADC 时钟频率选择，以 204.8kHz 为基准。 00, ×1； 01, ×2； 10, ×4； 11, ×8
	bit[3:2]	MEACLKSEL<1:0>	电能计量时钟频率选择，以 819.2kHz 为基准。 00, ×1； 01, ×2； 10, ×4； 11, ×8

寄存器	位	功能	说明
	bit[1:0]	MCUCLKSEL<1:0>	MCU 时钟频率选择, 以 819.2kHz 为基准。 00, ×1; 01, ×2; 10, ×4; 11, ×8

普通操作



快速操作

图 5-1 时钟控制电路框图

5.2 RC 时钟产生电路

V9811 内部的 RC 振荡电路可以产生一个 32kHz 的 RC 时钟, 不可关闭, 仅供 WDT 电路使用。

5.3 OSC 时钟产生电路

V9811 片内集成了起振电路, 片外仅需一个 32768Hz 的晶体, 即可形成晶振电路, 产生 32.768kHz 的 OSC 时钟。

晶振电路始终工作, 是一个低功耗模块。芯片复位后, 该电路消耗电流为 0.6μA。

在 V9811 的 OSC 时钟产生电路中存在晶体起振检测电路。当 XRESETEN=1 (bit5, CtrlCry2, 0x2861) 时, 如果晶体起振检测电路检测到晶体停振, 则片内的起振电路复位。

V9811 内部起振电路电阻和电容可配, 通过修改配置, 用户可对 OSC 的输出时钟频率进行微调。

表 5-2 晶振电路寄存器配置

寄存器	位	说明	推荐配置
晶振控制寄存器 0 (CtrlCry0, 0x285F)	CAPSEL<7:0>	晶体起振频率调节	0
晶振控制寄存器 1 (CtrlCry1, 0x2860)	XTRSEL<1:0>	起振电路电阻调节	使用 RTC 时, 必须将该寄存器配置为 0x03
	CSEL	晶体起振电容调节	
	XEASYN	MOS 电阻大小调节	
晶振控制寄存器 2 (CtrlCry2, 0x2861)	XTLDOV	晶振电路供电电压选择	0x20
	XRESETEN	起振电路复位使能位	

5.4 PLL 时钟产生电路

在 V9811 中, 晶振电路产生的 OSC 时钟作为基准时钟输入 PLL 电路, 然后, PLL 电路将其倍频产生若干时钟供 MCU、电能计量模块和 ADC 使用。开启 MCU, 然后才能开启 PLL。开启前, PLL 输出频率为 32768Hz。

正常计量时, MCU 时钟频率应为 13107.2kHz, 电能计量时钟频率为 3276.8kHz, ADC 时钟频率为 819.2kHz, 此时整个芯片功耗约 12.8mA。

表 5-3 MCU/ADC/电能计量时钟频率配置

寄存器	位	功能	说明
时钟控制寄存器 (CtrlCLK, 0x2867)	bit[5:4]	ADCLKSEL<1:0>	过采样 ADC 时钟频率选择, 以 204.8kHz 为基准。 00, ×1; 01, ×2; 10, ×4; 11, ×8
	bit[3:2]	MEACLKSEL<1:0>	电能计量时钟频率选择, 以 819.2kHz 为基准。 00, ×1; 01, ×2; 10, ×4; 11, ×8
	bit[1:0]	MCUCLKSEL<1:0>	MCU 时钟频率选择, 以 819.2kHz 为基准。 00, ×1; 01, ×2; 10, ×4; 11, ×8
PLL 控制寄存器 (CtrlPLL, 0x2868)	bit7	MCU26M	当 MCU13M=1 时, 使能 MCU 时钟频率加倍 (×2) 1, 使能。
	bit6	MCU13M	使能 MCU 时钟频率加倍 (×2) 1, 使能。

电能计量时钟频率应为 ADC 时钟频率的 4 倍, MCU 时钟独立于其它两个时钟。

用户可以按以下流程开启 PLL:

1. 时钟控制寄存器 (CtrlCLK, 0x2867) 开关 PLL 时钟并设置 PLL 时钟频率;
2. 等待 PLL 设置频率并等待锁定。用户可通过查询 PLL 状态寄存器 (PLLLCK SFR, 0xA3) 的 PLLLCK 位 (bit0) 判断 PLL 是否已锁定;
3. PLL 锁定后, 连接 PLL 时钟作为 MCU 和电能计量模块的时钟源。连接 PLL 作为时钟源方法: 将时钟切换控制寄存器 (SysCtrl SFR, 0x80) 的 MCUFRQ 位 (Bit0) 和 MEAFRQ 位 (Bit7) 置 1。

PLL 不同频率间切换要严格按照以下流程：

1. MCU 和电能计量时钟源切换为 32.768kHz 的 OSC 时钟输出；
2. 关闭 PLL；
3. 调整时钟频率；
4. 打开 PLL；
5. 等待 PLL 锁定；
6. 连接 PLL 时钟输出作为时钟源。

默认情况下，V9811 支持 50Hz 电网应用。用户可通过 PLL 控制寄存器（CtrlPLL，0x2868）将 V9811 配置为适应 60Hz 电网应用，60Hz 电网应用下的 PLL 的输出频率是 50Hz 电网应用下的 PLL 输出频率的 1.2 倍，此时，用户应重新设置通信的波特率、定时器等与时钟频率相关的设置。

表 5-4 电网应用选择

寄存器	位		功能	说明
PLL 控制寄存器（CtrlPLL，0x2868）	bit5	PLLSEL	50Hz 或 60Hz 应用选择	0, 50Hz; 1, 60Hz

5.5 切换 PLL 时钟与 OSC 时钟

用户可以采用普通操作方式切换或开关电能计量时钟，采用普通或快速操作方式切换或开关 MCU 时钟。普通操作方式需要程序按步骤访问各个控制寄存器。快速操作方式只需要访问一个寄存器就可以由硬件完成所需要的操作步骤，适用于对低功耗要求比较高的应用。用户可以相互替换或结合两种操作方式进行操作。

表 5-5 时钟控制电路中的时钟

	时钟 1	时钟 2	时钟 3	时钟 4	时钟 5
复位后状态	OSC	OSC	OSC	RC	OSC
OSC→PLL	PLL	PLL	OSC	RC	OSC
PLL→OSC	OSC	OSC	OSC	RC	OSC
关闭方法	浅睡眠或深睡眠	PMG = 1 SysCtrl SFR, Bit 4	LCDG = 1 SysCtrl SFR, Bit 3	不可关闭	不可关闭

5.5.1 MCU 和电能计量时钟源普通切换流程

5.5.1.1 MCU 时钟源切换

1. 复位后的状态

复位等级 1 的复位发生后，模拟控制寄存器和时钟切换控制寄存器（SysCtrl SFR，0x80）被复位。此时，PLL 时钟产生电路关闭（PLLPDN=0，bit7，CtrlCLK）；时钟 1 处于开启状态（SLEEP1/SLEEP0=00，bit[2:1]，SysCtrl SFR，0x80）；时钟 1 使用 OSC 时钟作时钟源（MCUFRQ=0，Bit0，SysCtrl SFR）。

2. 从 OSC 时钟切换为 PLL 时钟

访问模拟控制寄存器，开启 PLL 时钟产生电路并设置输出频率，如 3.2768MHz，向 MCUFRQ 位（Bit0，SysCtrl SFR）写入 1，时钟 1 的时钟源从 OSC 时钟切换到 PLL 时钟，此切换过程只需要一个 PLL 时钟周期。

在未开启 PLL 时钟产生电路时，向 MCUFRQ 写入 1 无法将时钟 1 的时钟源切换为 PLL 时钟。

将时钟源切换到 PLL 时钟后，如果 PLL 因为非正常原因被程序关闭，PLL 时钟频率自动变为 32768Hz，但是 MCUFRQ 位依然保持为 1，此时程序可以依靠 PLLLCK 位（Bit0，PLLLCK SFR，0xA3）判断当前的 PLL 时钟状态。

3. 从 PLL 时钟切换为 OSC 时钟

向 MCUFRQ 位（Bit 0， SysCtrl SFR）写入 0，将时钟 1 的时钟源从 PLL 时钟切换到 OSC 时钟。从程序向 MCUFRQ 写入 0 到时钟切换电路真正完成切换，最长需要一个 OSC 时钟周期，在这段时间内，对模拟控制寄存器的写入操作无效。所以，程序在向 MCUFRQ 写入 0 后，应该不断读出这一位的值，直到读出的值从 1 变为 0，表明时钟切换电路已经完成了切换；或者等待一段合适的时间，然后再执行后续的任务，比如关闭 PLL 电路等。

4. 关闭时钟

当系统状态寄存器（Systate SFR，0xA1）的 PWRUP 位（Bit0）为 0，且时钟 1 的时钟源为 OSC 时钟，即 MCUFRQ = 0（Bit 0， SysCtrl SFR）时，向 SLEEP0 位（Bit1， SysCtrl SFR）或者 SLEEP1（Bit2， SysCtrl SFR）写入 1，都可以关闭时钟 1。关闭时钟 1 后，使用时钟 1 的电路单元（CPU、RAM、FLASH、扩展中断、扩展定时器/UART 和 IO）停止工作。

5.5.1.2 电能计量时钟源切换

1. 复位后的状态

复位等级 1 的复位发生后，模拟控制寄存器和时钟切换控制寄存器（SysCtrl SFR，0x80）被复位。此时，PLL 时钟产生电路关闭（PLLPDN=0，bit7，CtrlCLK）；时钟 2 使用 OSC 时钟作时钟源（MEAFRQ=0，Bit7，SysCtrl SFR）。

2. 从 OSC 时钟切换为 PLL 时钟

访问模拟控制寄存器，开启 PLL 并设置输出频率，如 3.2768MHz，向 MEAFRQ 位（Bit7， SysCtrl SFR）写入 1，时钟 2 的时钟源从 OSC 时钟切换到 PLL 时钟，切换过程只需要一个 PLL 时钟周期。在没有开启 PLL 的情况下，即使向 MEAFRQ 写入 1 也无法将时钟源切换到 PLL 时钟。将时钟源切换到 PLL 时钟之后，如果 PLL 因为非正常原因被程序关闭，PLL 时钟输出频率自动变为 32768Hz，但是 MEAFRQ 依然保持为 1，此时程序可以依靠 PLLLCK 位（Bit0，PLLLCK SFR，0xA3）判断当前的时钟状态。

3. 从 PLL 时钟切换为 OSC 时钟

向 MEAFRQ 位（Bit7， SysCtrl SFR）写入 0，将时钟 2 的时钟源从 PLL 时钟切换到 OSC 时钟。从程序向 MEAFRQ 写入 0 到时钟切换电路真正完成切换动作，最长需要一个 OSC 时钟周期，在这段时间内，对模拟控制寄存器的写入操作无效。所以，程序在向 MEAFRQ 写入 0 后，应该不断读出这一位的值，直到读出的值从 1 变为 0，表明时钟切换电路已经完成了切换；或者等待一段合适的时间，然后再执行后续的任务，比如关闭 PLL 电路等。

4. 关闭时钟

当时钟 2 的时钟源为 PLL 时钟时，向 PMG 位（bit4， SysCtrl SFR）写入 1，则时钟 2 被关闭。时钟 2 关闭后，电能计量模块停止工作。

5.6 MCU 时钟源快速切换流程

MCU 时钟源切换的快速操作方式通过访问时钟切换控制寄存器 (SysCtrl SFR, 0x80) 中的 FWC 位 (Bit6) 和 FSC (Bit5) 实现。

表 5-6 MCU 时钟源快速切换控制位

寄存器	Bit6	Bit5	说明
	FWC	FSC	
时钟切换控制寄存器 (SysCtrl SFR, 0x80)	0	0	系统工作状态由程序决定, 并由程序决定是否开启 PLL, 以及是否切换时钟 1 的时钟源
	1	0	开启 PLL, 输出保持为 3.2768MHz, 时钟 1 的时钟源从 OSC 时钟切换为 PLL 时钟
	x	1	时钟 1 的时钟源从 PLL 时钟切换到 OSC 时钟, 关闭 PLL, 并关闭时钟 1 (供 CPU 及其外设使用), 系统进入浅睡眠状态。
备注	<ol style="list-style-type: none"> 当 FSC=0 时, 用户对 FWC 的配置才有效。当 FSC/FWC=01 时, 寄存器 CtrlCLK(0x2867) 和 MCUFRQ 位 (bit0, SysCtrl SFR, 0x80) 无效, 此时, PLL 输出保持 3.2768MHz 不变。先向 FSC/FWC 写入 01, 再向 FSC/FWC 写入 00, 时钟 1 的时钟源从 PLL 时钟切换为 OSC 时钟, 但是 PLL 保持开启。当 MCUFRQ=1 时, 向 FSC/FWC 写入 00 不能切换时钟 1 的时钟源; 无论 FWC 被配置为何值, 当 FSC 写入 1 时, 系统立即进入浅睡眠状态。 		

5.6.1.1 复位后的状态

复位等级 1 或复位等级 2 的复位发生后, FWC/FSC 为 00, 系统工作状态由程序决定, 并由程序决定是否开启 PLL, 以及是否进行时钟源切换。

5.6.1.2 从 OSC 时钟切换为 PLL 时钟

向 FWC 写入 1, 可以触发硬件自动完成开启 PLL 并将时钟 1 的时钟源从 OSC 时钟切换到 PLL 时钟的操作。硬件自动开启 PLL 后, 其输出频率为 3.2768MHz。向 FWC 写入 1 的指令执行后, 切换立即完成。

向 FWC 写入 1, 开启 PLL 并输出 3.2768MHz 时钟后, 请按下述步骤将 PLL 输出时钟频率配置为 13.1072MHz 或 26.2144MHz:

- 向 FWC 写入 0, 将时钟 1 的时钟源从 PLL 输出切换为 OSC 输出, 此时, PLL 保持开启;
- 通过 MCUCLKSEL<1:0>位 (bit[1:0], CtrlCLK, 0x2867) 将 PLL 时钟频率配为 6.5536MHz;
- 将 MCU13M 位 (bit6, CtrlPLL, 0x2868) 置 1, 将 PLL 时钟频率调整为 13.1072MHz; 或, 先将 MCU13M 位置 1, 再将 MCU26M (bit7, CtrlPLL, 0x2868) 置 1, 将 PLL 时钟频率调整为 26.2144MHz;
- 向 MCUFRQ 写入 1, 将 MCU 时钟源切换为 PLL 输出。

5.6.1.3 从 PLL 时钟切换为 OSC 时钟，并关闭时钟

当系统状态寄存器（Systate SFR, 0xA1）的 PWRUP 位（Bit0）为 0 时，向 FSC 写入 1，可以触发硬件自动完成将时钟 1 的时钟源从 PLL 时钟切换为 OSC 时钟，关闭 PLL，并关闭时钟 1，系统进入浅睡眠状态。

5.6.2 MCU 时钟源切换普通操作方式 V.S 快速操作方式

在复位等级 1 或者复位等级 2 的复位发生后，时钟 1 的时钟源为 OSC 时钟，不能进行正常计量，但能进行常数计量。这是一个过渡状态。在对低功耗要求比较高的应用条件下，缩短芯片处于这一状态的时间，可以减少平均功耗。在检测到掉电之后，缩短关闭各个电路单元的时间，也可以减少平均功耗。由于普通操作方式需要程序按步骤访问各个控制寄存器，快速操作方式只需要访问一个寄存器位就可以由硬件完成所需要的操作步骤，所以快速操作方式可以减少上述两种情况下的操作时间，从而降低平均功耗。快速操作方式与普通操作方式以“或”操作的方式发生作用。

当 FSC/FWC=01 时，时钟控制寄存器（CtrlCLK, 0x2867）和 MCUFRQ 位（bit0, SysCtrl SFR, 0x80）无效，此时，PLL 时钟频率保持 3.2768MHz 不变。先向 FSC/FWC 写入 01，再向 FSC/FWC 写入 00，时钟 1 的时钟源从 PLL 时钟切换为 OSC 时钟，但是 PLL 时钟保持开启。当 MCUFRQ=1 时，向 FSC/FWC 写入 00 不能切换时钟 1 的时钟源；无论 FWC 被配置为何值，当 FSC 写入 1 时，系统立即进入浅睡眠状态。

快速操作方式与普通操作方式可以结合进行，二者之间又有相互影响。使用快速操作方式与普通操作方式进行结合操作：先利用快速操作方式开启 PLL 并切换时钟，以节省功耗；当将时钟 1 的时钟源切换到 PLL 时钟后，再利用普通操作方式中的 MCUFRQ 位来保持这个状态，以便不受其他因素的影响。

```
FWC = 1;           // 开启 PLL，切换时钟到 PLL
MCUFRQ = 1;       // 执行此指令时，时钟 1 已经在使用 PLL 时钟
```

表 5-7 分别列出了采用普通操作方式、快速操作方式和结合操作方式进行 MCU 时钟源切换的对比。

表 5-7 普通操作、快速操作和结合操作进行 MCU 时钟源切换的比较

操作	普通操作方式	快速操作方式	结合操作
开启 PLL，切换时钟	访问模拟控制寄存器，开启 PLL MCUFRQ = 1;	FWC = 1	FWC = 1 MCUFRQ=1;
切换时钟，关闭 PLL，关闭时钟 1	MCUFRQ = 0; while(MCUFRQ == 1){}; 访问模拟控制寄存器，关闭 PLL SLEEPO = 1;	FSC = 1	MCUFRQ=0; FSC = 1

图 5-2 中箭头所指的部分，表示使用快速操作方式或者结合操作方式完成从发生 IO 休眠唤醒复位开始，至时钟 1 的时钟源切换为 3.2768MHz 的 PLL 时钟结束，全过程需要约 800 ~ 900μs，整个过程包含复位时间、程序最初跳转指令执行时间以及向 FWC 写入 1 的指令的执行时间。

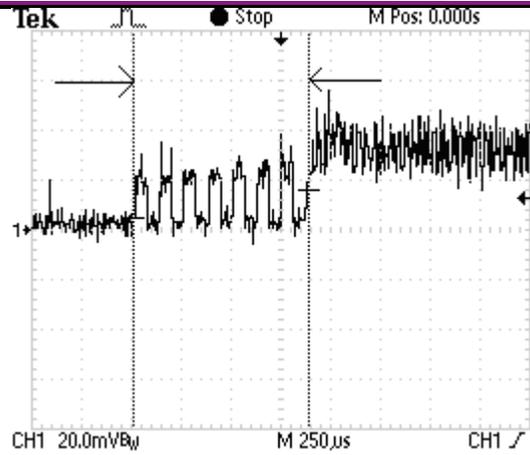


图 5-2 使用快速操作方式开启 PLL 并将时钟 1 的时钟源切换到 PLL 时钟输出

图 5-3 中箭头所指的位置，表示使用快速操作方式或者结合操作方式将时钟源从 PLL 时钟切换到 OSC 时钟，并关闭时钟的全部时间小于 30µs。

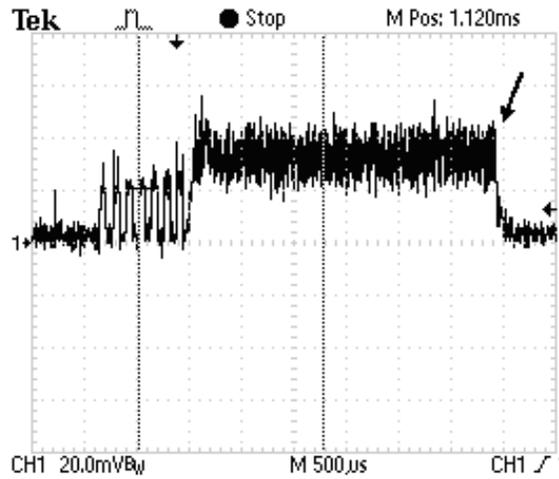


图 5-3 使用快速操作方式将时钟 1 的时钟源切换到 OSC 时钟输出，然后关闭 PLL，关闭时钟 1

第6章 功耗

根据时钟 1 的状态，芯片可以处在三种系统状态：

- OSC 状态：等级 1 复位发生后的系统状态；
- 工作状态：PLL 开启并被用作时钟 1 的时钟源的系统状态；
- 休眠状态：在系统状态寄存器（Systate SFR, 0xA1）中的 PWRUP 位（bit0）为 0 的时候，将时钟 1 的时钟源切换为 OSC 时钟，并关闭时钟 1 的系统状态。休眠状态又分为浅睡眠和深睡眠两种状态。在浅睡眠状态，发生 IO/RTC 休眠唤醒复位或恢复供电复位可以使芯片回到 OSC 状态。在深睡眠状态，发生 IO 休眠唤醒复位可以使芯片回到 OSC 状态。

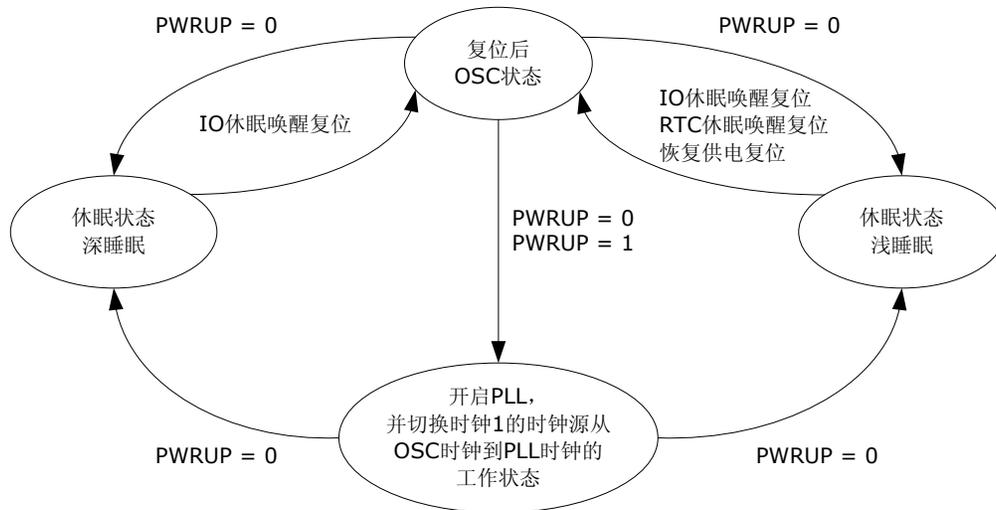


图 6-1 系统状态转换图

系统状态寄存器（Systate SFR, 0xA1）的各个 bit 分别表示当前的供电状态和导致上次系统复位的原因等。软件可以根据这个寄存器的状态决定程序的执行流程。当发生 POR 复位、RSTn 低电平有效或 WDT 溢出复位时，系统状态寄存器（Systate SFR, 0xA1）被复位。

VDCIN 引脚上的输入电压与 PWRUP 标志位和 PWRDN 标志位的关系如图 6-2 所示。

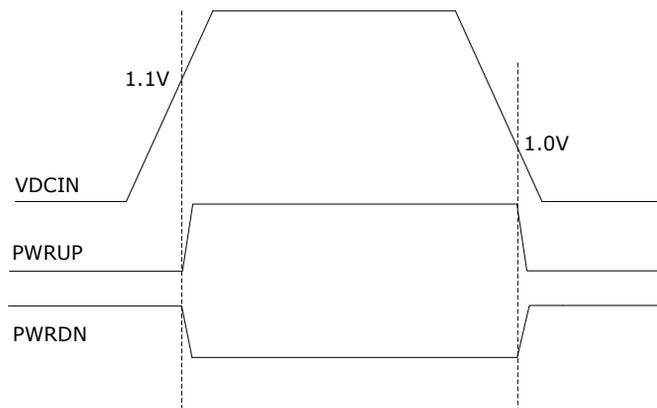


图 6-2 VDCIN 输入电平与 PWRUP 和 PWRDN 标志位的关系

表 6-1 供电状态和系统复位原因指示

寄存器	位		说明
Systate SFR 0xA1	bit5	POR	1, 表示上次的复位是由 POR、片外输入事件 (RSTn) 或 WDT 溢出复位引起。 当其它复位发生时, 此标志位清零
	bit3	IO	1, 表示上次的复位是在浅睡眠或者深睡眠状态下, 由 IO 事件引起。
	bit2	RTC	1, 表示上次的复位是在浅睡眠状态下, 由 RTC 的定时唤醒引起。
	bit1	PWRDN	PWRDN 指示灯默认开启。 当 VDCIN 引脚上的输入电压小于 1.0V 时, PWRDN 位为 1, 表示发生了掉电; 当 VDCIN 引脚上的输入电压大于 1.1V 时, PWRDN 位为 0, 表示没有发生掉电。 如果使能了 PWRDN 中断, 掉电事件会向 MCU 提起中断。
	bit0	PWRUP	当 VDCIN 引脚上的输入电压大于 1.1V 时, PWRUP 位为 1, 表示正常供电, 或者从电池供电状态切换到了正常供电状态; 当 VDCIN 引脚上的输入电压小于 1.0V 时, PWRUP 位为 0, 表示当前是电池供电, 或者从正常供电状态切换到了电池供电状态。

系统整体功耗取决于各模块。各个模块有可关闭和不可关闭部分, 部分模块的功耗与工作电压相关, 部分模块的功耗与工作频率相关。

表 6-2 各模块功耗的影响因素

模块	上电时的状态	是否可关闭	功耗影响因素	
			时钟频率	数字电源电路输出
LDO33	开启	VCDIN 低于 1.0V 时, 在 LDO33PD 位写入 1 (Bit5, CtrlLDO, 0x2866), 使 LDO33 停止工作	否	否
数字电源电路	开启	不可关闭	否	否
OSC	开启	不可关闭	否	否
MCU	运行	可挂起	是	是
低功耗参考电路	开启	不可关闭	否	否
RTC	开启	不可关闭	否	否
PLL	关闭	可关闭	否	否
BGP	关闭	可关闭	否	否
掉电监测电路	开启	不可关闭	否	否
测温电路	关闭	可关闭	否	否
电池电压测量电路	关闭	可关闭	否	否
LCD	驱动关闭, 扫	可关闭	否, 与驱动	否

模块	上电时的状态	是否可关闭	功耗影响因素	
			时钟频率	数字电源电路输出
	描时序开启		模式有关	
ADC	关闭	可关闭	是	否
电能计量电路	通道关闭, 计量时序开启	可关闭	是	是

6.1 OSC 状态

当等级 1 复位发生后, 系统进入 OSC 状态, 此时, LDO33 开启, OSC 开启, 数字电源电路开启, CPU 运行。

表 6-3 OSC 状态

模块	上电时的状态	是否可关闭	当前状态
LDO33	开启	当 VCDIN 输入电压低于 1.0V 时, 在 LDO33PD 位写 1 (Bit5, CtrlLDO, 0x2866), LDO33 停止工作	开启
数字电源电路	开启	不可关闭	开启
OSC	开启	不可关闭	开启
MCU	运行	可挂起	运行
低功耗参考电路	开启	不可关闭	开启
RTC	开启	不可关闭	开启

6.2 工作状态

在 OSC 状态下, 开启 PLL, 将时钟 1 的时钟源从 OSC 时钟切换到 PLL 时钟, 芯片进入工作状态。在工作状态下, 用户可根据需要通过软件配置 PLL 输出频率, 开启不同数量的 ADC 通道, 开启相应计量通道进行计量, 开启 LCD 驱动进行显示, 并使用 CPU 的其他外设等。

在工作状态下, 当 PLL 开启, MCU 时钟频率为 13.1072MHz, 电能计量时钟频率为 3276.8kHz, ADC 时钟频率为 819.2kHz, 即, 寄存器 CtrlCLK (0x2867) 配置为 0xEB, 寄存器 CtrlPLL (0x2868) 的 MCU13M 位 (bit6) 置 1 时, 芯片正常工作。此时, 芯片的功耗由开启的 ADC 通道数量决定。

表 6-4 正常工作状态 (MCU 工作频率为 13.1072MHz)

模块	上电时的状态	是否可关闭	当前状态
LDO33	开启	当 VCDIN 输入电压低于 1.0V 时, 在 LDO33PD 位写 1 (Bit5, CtrlLDO, 0x2866), LDO33 停止工作	开启
数字电源电路	开启	不可关闭	开启
OSC	开启	不可关闭	开启

模块	上电时的状态	是否可关闭	当前状态
低功耗参考电路	开启	不可关闭	开启
RTC	开启	不可关闭	开启
PLL	关闭	是	开启
BGP	关闭	是	开启
掉电监测电路	开启	不可关闭	开启
测温电路	关闭	是	关闭
电池电压测量电路	关闭	是	关闭
ADC	关闭	是	开启 4 路 ADC 通道
电能计量电路	通道关闭, 时钟开启	是	开启 4 路 ADC 通道
LCD	驱动关闭, 扫描时序开启	是	开启
MCU	运行	可挂起	运行
总功耗	12.8mA		

6.3 休眠状态

在系统状态寄存器 (Systate SFR, 0xA1) 中的 PWRUP 位 (bit0) 为 0 的时候, 将时钟 1 的时钟源切换为 OSC 时钟, 并关闭时钟 1 的系统状态, 即为休眠状态。休眠状态又分为浅睡眠和深睡眠两种状态。

在浅睡眠或者深睡眠状态下, RTC 继续工作, 所有的存储器和 CPU 及其外设停止工作并进入低功耗模式, LDO33 关闭, 但是, LCD 和电能计量模块需要手动关闭。如果在进入浅睡眠或者深睡眠之前, 程序已经关闭了 ADC 和 PLL、LCD 驱动/LCD 时序产生电路以及电能计量模块, 并把 IO 设置为禁止输出/屏蔽输入, 则系统处于最省电的状态。如果未关闭 LCD 驱动以及 LCD 时序产生电路, LCD 在浅睡眠或者深睡眠状态下仍可以正常进行显示。

在浅睡眠状态下, RTC/IO 休眠唤醒复位或恢复供电复位都可以使芯片回到 OSC 状态。在深睡眠状态下, 只有发生 IO 休眠唤醒复位可以使芯片回到 OSC 状态。只有在进入浅睡眠或者深睡眠之前将 IOWKUP1 (Pin21) 或 IOWKUP2 (Pin22) 设置为输入, 并在这两个 IO 端口中的任何一个上发生高电平到低电平的跳变 (此跳变的高低电平需各自持续至少 4 个 OSC 时钟周期), 才会产生 IO 休眠唤醒复位。

6.3.1 休眠/唤醒 - 使用普通操作方式

使用普通操作方式进行休眠唤醒操作时, 系统状态的转换完全由程序控制。

程序先查询系统状态寄存器 (Systate SFR, 0xA1) 中的 PWRUP 位 (bit0) 是否为 0。在查询到 PWRUP 为 0 后, 如果此时已经通过普通流程将时钟 1 (MCU 时钟) 的时钟源切换为 OSC 时钟, 即 MCUFREQ = 0 (Bit 0, SysCtrl SFR), 则向 SLEEP0 位 (Bit1, SysCtrl SFR) 写入 1, 系统进入浅睡眠; 向 SLEEP1 位 (Bit2, SysCtrl SFR) 写入 1, 系统进入深睡眠。

系统进入深睡眠/浅睡眠后, 用户可采用 RTC/IO 休眠唤醒复位或恢复供电复位使其回复到 OSC 状态, 然后, 使用普通操作方式开启 PLL 并对时钟 1 的时钟源进行切换, 从而使系统进入工作状态。

表 6-5 系统休眠状态设置

寄存器	SLEEP1 (Bit2)	SLEEP0 (Bit1)	系统状态
时钟切换控制寄存器 (SysCtrl SFR, 0x80)	0	1	浅睡眠
	1	1	浅睡眠
	1	0	深睡眠

6.3.2 休眠/唤醒 - 使用快速操作方式

使用快速操作方式只能进入浅睡眠状态，不能进入深睡眠状态。

程序先查询系统状态寄存器 (Systate SFR, 0xA1) 中的 PWRUP 位 (bit0) 是否为 0。在查询到 PWRUP 为 0 后，用户应先在 LDO33PD 位写入 1 (Bit5, CtrlLDO, 0x2866)，关闭 LDO33，使 LDO33 的输入电压与 VDD5 引脚上的电压相等，再向 FSC 写入 1，触发硬件自动完成将时钟 1 的时钟源从 PLL 时钟切换到 OSC 时钟，关闭 PLL，一直到进入浅睡眠状态的操作。

系统进入浅睡眠后，用户可采用 RTC/IO 休眠唤醒复位或恢复供电复位使其回复到 OSC 状态，然后，使用快速操作方式开启 PLL 并切换时钟 1 的时钟源，从而使系统进入工作状态。

6.3.3 休眠功耗

在休眠状态下，系统的最低功耗为 11 μ A，此时的系统工作状态如表 6-6 所示。

休眠模式下，LCD 模块（无屏）开启时的芯片工作电流为 22 μ A，RTC 模块的工作电流为 2.2 μ A。

表 6-6 休眠状态下系统功耗

模块	上电状态	是否可关闭
数字电源电路	开启	不可关闭
OSC	开启	不可关闭
低功耗参考电路	开启	不可关闭
RTC	开启	不可关闭
掉电监测电路	开启	不可关闭
RC 时钟	开启	不可关闭
功耗	11 μ A	

第7章 电源系统

V9811 电源供电系统有以下特点：

1. 采用 5.0V/3.3V 外电源供电；
2. 芯片内部模拟电路和 GPIO 口等由 3.3VLDO 供电；
3. 芯片内部数字电路、PLL、RTC 和晶体起振电路等由数字电源电路供电；
4. 支持低电压监测，实时监测电池电压。

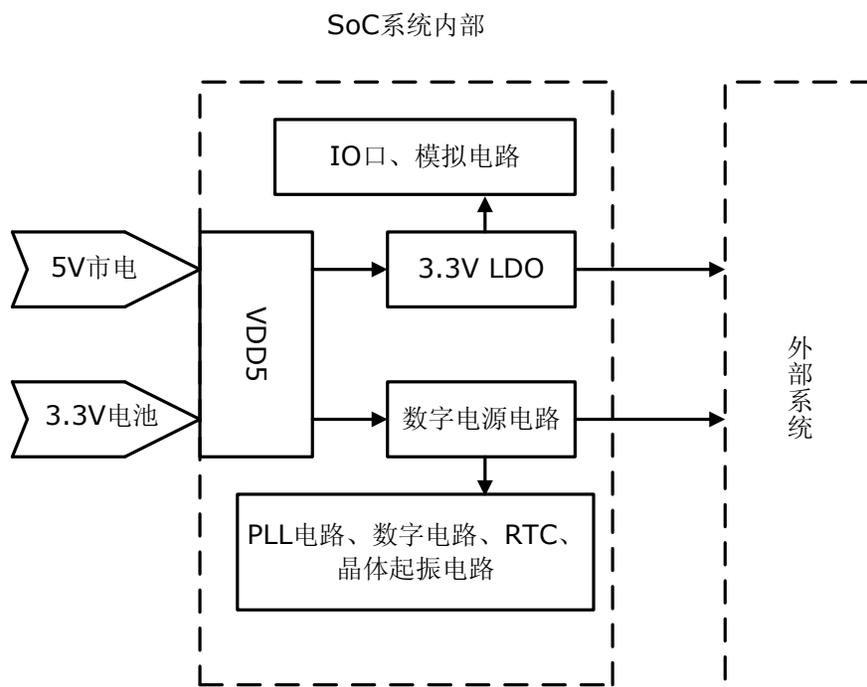


图 7-1 电源供电系统

7.1 3.3V 稳压电路（LDO33）

芯片内部模拟电路和 GPIO 口由 3.3VLDO 供电，该电压由 LDO33 产生。

当引脚 V3V 上输入低电平时，系统采用 5V 单电源供电；当引脚 V3V 上输入高电平时，系统采用 3.3V 单电源供电，此时，LDO33 始终关闭。

采用 5V 单电源供电时，如果外电源大于 3.3V 时，LDO33 保持 3.3V 的输出电压，避免因 5V 电源波动带来的模拟电路性能变化，如图 7-2 所示。当 VDCIN 引脚上的输入电平低于 1.0V，系统状态寄存器（Systate SFR, 0xA1）的 PWRUP 标志位（bit0）为 0，即系统掉电时，用户可将 LDO33PD 位（Bit5, CtrlLDO, 0x2866）置 1，约 8ms 后，LDO33 关闭，LDO33 的输出电压与 5V 主电源的输入电压相等，降低功耗。

表 7-1 LDO33 控制位

寄存器	位	说明
LDO 控制寄存器 (CtrlLDO, 0x2866)	bit5 LDO33PD	1, 关闭 LDO33。当 VDCIN 的输入电压低于 1.0V 时, 在 LDO33PD 位写 1, 8ms 后, LDO33 停止工作。

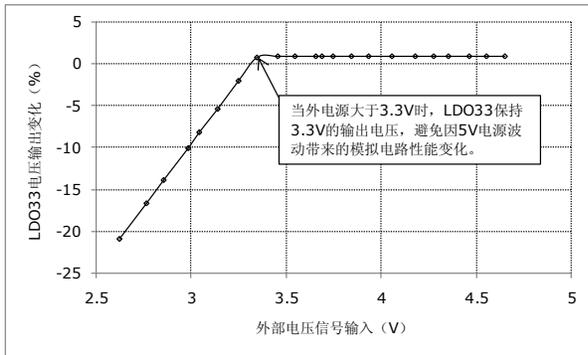


图 7-2 LDO33 电压输出与外部电压信号输入

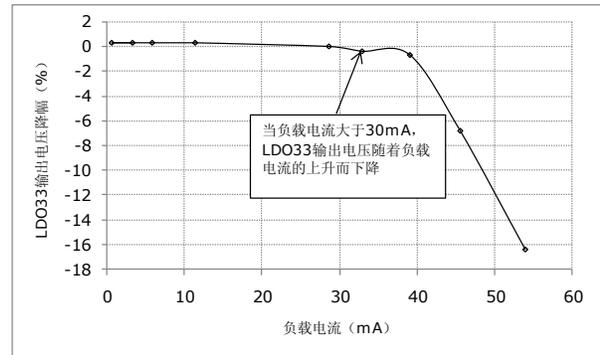


图 7-3 LDO33 电压输出与功耗

LDO33 具有 30mA 的驱动能力, 即, 当模拟电路以及 GPIO 口的负载电流小于 30mA 时, LDO33 能保持输出电压稳定, 当负载电流大于 30mA, LDO33 输出电压随着负载电流的上升而下降 (如图 7-3 所示)。

LDO33 在片外需连接一个由一个 10 μ F 和一个 0.1 μ F 的去耦电容组成的并联电路。

7.2 数字电源电路

芯片内的数字电路、PLL 电路、RTC 和晶体起振电路等均由数字电源电路供电。当 5V 电源比数字电源电路输出电压高 200mV 时, 数字电源电路能保持输出电压稳定, 避免因 5V 电源波动带来的数字电源波动。该电路在掉电状态下仍能正常工作, 从而保证了数字电路供电稳定。数字电源电路具有 35mA 的驱动能力, 即, 当数字电路上的负载电流小于 35mA 时, 数字电源电路仍能保持输出电压稳定; 当负载电流大于 35mA 时, 数字电源电路的输出电压会随着负载电流的增加而明显下降。

数字电源电路在片外需连接耦合电容, 建议采用 10 μ F 和 0.1 μ F 并联的结构。

7.3 掉电监测电路

V9811 内置一个电源监测模块, 可以实时监测 VDCIN 引脚的输入信号。一般, 5V 电源经电阻分压后输入 VDCIN 引脚。当 VDCIN 引脚上的电平低于 1V 时, 掉电监测电路输出高电平, 通知 MCU 系统已经开始掉电, MCU 提起掉电中断, 同时在系统状态寄存器 (Systate SFR, 0xA1) 中 PWRDN 位 (bit1) 置位。

7.4 上电复位和 RSTN 复位

在上电期间, 芯片提供 2 个复位源以保证可靠的复位: 一个是 RSTn 引脚输入的片外复位信号, 另一个就是上电复位电路提供的 POR 复位信号。上电复位电路监测 DVCC 引脚的电平, 当 DVCC 引脚上的输出电压大于 1.8V 时, 复位信号释放, 而在此之前, 芯片会一直处于复位状态。阈值电平 1.8V 是一个典型值, 具体的值会因芯片而异。

该电路始终工作, 不需 MCU 参与。

7.5 电池供电

V9811 系统及其内置的 RTC 均可由电池供电。当 VDCIN 引脚上的输入电压小于 1.0V 时, PWRUP=0(Bit0, Systate SFR), 表示当前 5V 电源发生掉电, 甚至从正常供电状态切换到电池供电状态, 而是否发生这种电源切换则取决于 5V 电源和电池的两个二极管的压降, 如图 7-4 所示。

在长时间不用时, 电池会进入钝化状态。所以, 用户应隔一段时间(时间的长短由电池的特性决定)将电池放电控制寄存器 (CtrlBAT, 0x285C) 的 BATDISC 位置 1, 使能系统供电电池放电, 以防止电池进入钝化状态。电池放电过程中, 电池上消耗的电流为 3mA。放电持续时间不应过长, 以防消耗过多电池电量。放电结束后, 用户应将相应的供电电池放电使能位 (BATDISC) 置 0。

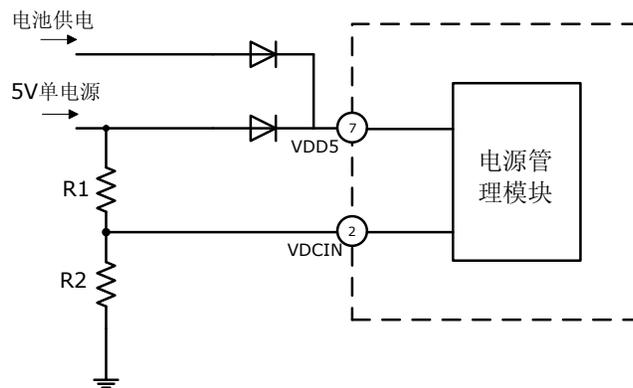


图 7-4 5V 电源与电池切换

第8章 电压基准电路 (BANDGAP)

V9811 内置一个 BandGap 电路，输出一个随温度变化较小的基准电压：1.185V (20ppm/°C)，为 ADC 等提供基准电压和电流。在开启这些电路之前应先开启电压基准电路。

该基准电压温度特性通过以下方面进行调整（无先后顺序要求）：

- 默认情况下，系统使能斩波去直 BGPCHOP (BGPCHOPN=0, bit0, CtrlBGP, 0x2862)。使能斩波去直可以消除 BandGap 电路的直流偏置，有效开启后会导致 BandGap 的输出值发生-50~+50mV 不等的变化，并且可以带来一定程序的温度系数改善。
- 调整 BandGap 温度系数 (REST<2:0>)。经测试，设置 REST<2:0>为 000，温度系数最佳。

表 8-1 BandGap 电路温度特性调整

寄存器	位		说明
BGP 控制寄存器 (CtrlBGP, 0x2862)	bit[3:1]	REST<2:0>	调整 BandGap 温度系数。 000, 0ppm; 001, +4ppm; 010, +8ppm; 011, +12ppm; 100, -16ppm; 101, -12ppm; 110, -8ppm; 111, -4ppm
	bit0	BGPCHOPN	使能斩波去直。0, 使能。

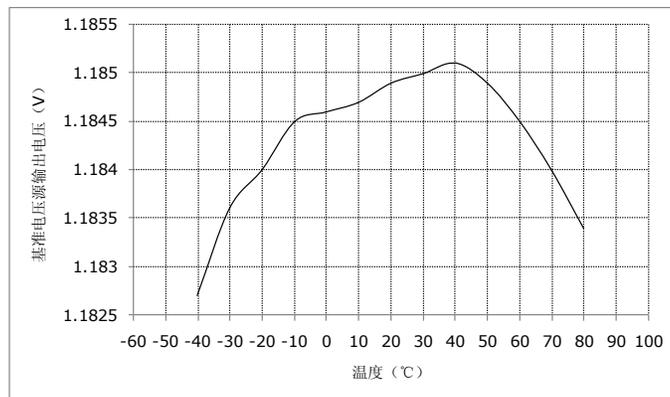


图 8-1 基准电压源温度特性曲线

第9章 电能计量

9.1 寄存器列表和使用说明

9.1.1 访问计量控制/数据寄存器

1. 读写缓存寄存器

当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，读写缓存寄存器被复位。

表 9-1 读写缓冲寄存器与数据的对应关系

数据	ACK	INVD	DATA[31:24]	DATA[23:16]	DATA[15:8]	DATA[7:0]
读写缓冲寄存器	BUFF5	BUFF4	BUFF3	BUFF2	BUFF1	BUFF0
地址	0x2885	0x2884	0x2883	0x2882	0x2881	0x2880

2. 读写操作

用户应按照下述步骤对计量控制/数据寄存器进行读操作：

- 向寄存器 INVD 先写入 0xCC，再写入 0；
- 对目标地址进行有效的读操作；
- 最多 16 个电能计量时钟周期，或者当 ACK 标志位从 1 变为 0 后，硬件电路会把目标地址的数据 DATA 加载到读写缓存寄存器中，数据与缓存寄存器的对应关系如上表所示；
- 读取缓存寄存器得到所需的数据。

用户应按照下述步骤对计量控制/数据寄存器进行写操作：

- 向寄存器 INVD 先写入 0xCC，再写入 0；
- 将所需的数据 DATA 写入读写缓存寄存器，数据与缓存寄存器的对应关系如上表所示；
- 对目标地址进行有效的写操作；
- 最多 16 个电能计量时钟周期，或者当 ACK 标志位从 1 变为 0 后，硬件电路会把缓存寄存器中的数据写入目标地址。

9.1.2 计量控制寄存器

当发生 POR、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位有效时，计量控制寄存器被复位。

表 9-2 PM 控制寄存器 1 (PMCtrl1, 0x2878)

0x2878, R/W, PM 控制寄存器 1, PMCtrl1				
位		默认值	功能	说明
bit7	PVA	0	控制功率/有效值计算电路访问相应的计量数据寄存器 仅供测试使用	0, 功率/有效值计算电路正常访问相应的计量数据寄存器; 1, 禁止功率/有效值计算电路访问相应的计量数据寄存器
bit6	PHCEN	0	使能角差校正	0, 禁止角差校正; 1, 使能角差校正 当角差校正使能时, IA、IB 两个通道根据各自的角差校正值同时进行角差校正。
bit5	SELI	0	选择电流计量通道	0, IA 通道的信号进入计量通道 I1, IB 通道的信号进入计量通道 I2; 1, IA 通道的信号进入计量通道 I2, IB 通道的信号进入计量通道 I1
bit4	PREN	0	控制有功/无功功率、有效值和视在功率计算	0, 停止有功/无功功率、有效值和视在功率计算; 1, 使能有功/无功功率、有效值和视在功率计算
bit3	ONM	0	M 通道数字信号输入使能	0, 屏蔽 M 通道的数字信号输入, 使得 M 通道的数字输入信号为 0; 1, 使能 M 通道数字信号输入
Bit2	ONI2	0	I2 通道数字信号输入使能	0, 屏蔽 I2 通道的数字信号输入, 使得 I2 通道的数字输入信号为 0; 1, 使能 I2 通道数字信号输入
Bit1	ONI1	0	I1 通道数字信号输入使能	0, 屏蔽 I1 通道的数字信号输入, 使得 I1 通道的数字输入信号为 0; 1, 使能 I1 通道数字信号输入
Bit0	ONU	0	电压通道 (U) 数字信号输入使能	0, 屏蔽 U 通道的数字信号输入, 使得 U 通道的数字输入信号为 0; 1, 使能 U 通道数字信号输入

表 9-3PM 控制寄存器 2 (PMCtrl2, 0x2879)

0x2879, R/W, PM 控制寄存器 2, PMCtrl2				
位		默认值	功能	说明
bit7	PGACS	0	I1 通道数字增益符号位	0, 正号; 1, 负号
bit[6:4]	PGAC2~PGAC0	0	I1 通道数字增益控制	取值范围为 0 ~ 5。增益计算方法为 $Gain = 2^{PGACx}$
bit3	PGAUS	0	U 通道数字增益符号位	0, 正号; 1, 负号
bit[2:0]	PGAU2~PGAU0	0	U 通道数字增益控制	取值范围为 0 ~ 5。增益计算方法为 $Gain = 2^{PGAUX}$

表 9-4 PM 控制寄存器 3 (PMCtrl3, 0x287A)

0x287A, R/W, PM 控制寄存器 3, PMCtrl3				
位		默认值	功能	说明
bit7	X0EN	0	过零点中断输出使能位	0, 禁止过零点中断输出; 1, 允许过零点中断输出。
bit6	BPFEN	0	使能电压/电流有效值计算电路上的带通滤波器	0, 禁止; 1, 使能 使能位于电压/电流有效值计算电路上的带通滤波器时, 此带通滤波器可以提高有效值的准确度, 但是会损失谐波成分, 在小信号的时候导致较大的数字截断噪声, 并导致系统响应时间变长
bit5	LPFEN	0	使能次谐波抑制低通滤波器	0, 禁止; 1, 使能 使能位于两路功率计算电路上的低通滤波器, 此低通滤波器可以抑制 12.5Hz 次谐波, 但是会导致系统响应时间变长。
bit4	DBLEN	0	有功/无功功率计算选择	0, 使能 I1 通道同时进行有功和无功功率计算。 1, 使能 I1 和 I2 通道进行有功功率计算 DBLEN = 0, 标记“E1”的寄存器存储 I1 通道的有功数据, 标记“E2”的寄存器存储 I1 通道的无功数据; DBLEN = 1, 标记“E1”的寄存器存储 I1 通道的有功数据, 标记“E2”的寄存器存储 I2 通道的有功数据。
bit3	PGANS	0	I2 通道数字增益符号位	0, 正号; 1, 负号

0x287A, R/W, PM 控制寄存器 3, PMCtrl3

位		默认值	功能	说明
bit[2:0]	PGAN2~PGAN0	0	I2 通道数字增益控制	取值范围为 0 ~ 5。增益计算方法为 $Gain = 2^{PGANx}$

表 9-5 角差校正控制寄存器 1 (PHCCtrl1, 0x287B)

0x287B, R/W, 角差校正控制寄存器 1, PHCCtrl1

位		默认值	功能	说明
bit7	PHCA7	0	IA 通道角差校正符号位	1, 负号。表示电压超前, 电流滞后; 0, 正号。表示电流超前, 电压滞后
bit6	PHCA6	0		未使用
bit[5:0]	PHCA<5:0>	0	IA 通道角差校正值的绝对值	角差校正的分辨率为 0.02197265625°/bit, 总校正量为 ±1.4°。

表 9-6 角差校正控制寄存器 2 (PHCCtrl2, 0x287C)

0x287C, R/W, 角差校正控制寄存器 2, PHCCtrl2

位		默认值	功能	说明
bit7	PHCB7	0	IB 通道角差校正符号位	1, 负号。表示电压超前, 电流滞后; 0, 正号。表示电流超前, 电压滞后
bit6	PHCB6	0		未使用
bit[5:0]	PHCB<5:0>	0	IB 通道角差校正值的绝对值	角差校正的分辨率为 0.02197265625°/bit, 总校正量为 ±1.4°。

表 9-7 PM 控制寄存器 4 (PMCtrl4, 0x287D)

0x287D, R/W, PM 控制寄存器 4, PMCtrl4

位		默认值	功能	说明
bit7	CRPENR	0	E2 计量通道起动/潜动判断使能	0, 禁止; 1, 使能

0x287D, R/W, PM 控制寄存器 4, PMCtrl4				
位		默认值	功能	说明
bit6	CRPEN	0	E1 计量通道起动/潜动判断使能	0, 禁止; 1, 使能
bit5	CFENR	0	E2 能量 CF 脉冲输出使能	0, 禁止; 1, 使能
bit4	CFEN	0	E1 能量 CF 脉冲输出使能	0, 禁止; 1, 使能
bit3	EGYEN	0	能量累加和脉冲计数使能	0, 禁止; 1, 使能
bit2	CFXCG	0	CF 脉冲输出引脚选择	0, E1 能量 CF 脉冲从引脚 CF1 输出, E2 能量 CF 脉冲从引脚 CF2 输出; 1, E2 能量 CF 脉冲从引脚 CF1 输出, E1 能量 CF 脉冲从引脚 CF2 输出
bit[1:0]	PSEL1/PSEL0	0	正相 E1 能量桶功率源选择	00 或 11, E1 功率; 01, I1 通道电流有效值; 10, 常数功率

表 9-8 脉冲输出控制寄存器 (CFCtrl, 0x287E)

0x287E, R/W, 脉冲输出控制寄存器, CFCtrl				
位		默认值	功能	说明
bit[7:6]	CFQR1/CFQR0	0	E2 能量脉冲产生速率控制, 可用于小信号加速校表	00, 1 倍; 01, 4 倍; 10, 8 倍; 11, 16 倍
bit[5:4]	CFQ1/CFQ0	0	E1 能量脉冲产生速率控制, 可用于小信号加速校表	00, 1 倍; 01, 4 倍; 10, 8 倍; 11, 16 倍
bit[3:2]	CFSELR1/CFSELR0	0	E2 能量脉冲源选择	01, 正相 E2 能量; 10, 反相 E2 能量; 00/11, 正相以及反相 E2 能量的绝对值之和
bit[1:0]	CFSEL1/CFSEL0	0	E1 能量脉冲源选择	01, 正相 E1 能量; 10, 反相 E1 能量; 00/11, 正相以及反相 E1 能量的绝对值之和

表 9-9 防潜标志寄存器 (CRPST, 0x287F)

0x287F, R, 防潜标志寄存器, CRPST				
位		默认值	功能	说明
Bit7	CRPST	0	E1 计量通道起动/潜动状态指示	0, E1 计量通道处于正常计量状态; 1, E1 计量通道处于潜动状态
Bit6	CRPSTR	0	E2 计量通道起动/潜动状态指示	0, E2 计量通道处于正常计量状态; 1, E2 计量通道处于潜动状态
Bit[5:0]	保留	0		

9.1.3 计量数据寄存器

9.1.3.1 信号波形数据寄存器

表 9-10 信号波形数据寄存器 (R/W)

地址	寄存器		R/W	数据格式	说明
0x1005	DATAOIU	瞬时电压原始值寄存器	R/W	32-bit 二进制补码	当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，信号波形数据寄存器被复位。 在电能计量时钟频率为 3.2768MHz 时，信号波形数据寄存器的值在每个信号周期内更新 64 次，瞬时信号波形数据的收敛时间为 60 ms。 所有寄存器均可读/可写。
0x1051	DATAIAU	瞬时电压交流值寄存器	R/W	32-bit 二进制补码	
0x103A	DATAIDU	瞬时电压直流值寄存器	R/W	32-bit 二进制补码	
0x100A	DATAOII1	I1 瞬时电流原始值寄存器	R/W	32-bit 二进制补码	
0x1052	DATAIAI1	I1 瞬时电流交流值寄存器	R/W	32-bit 二进制补码	
0x1041	DATAIDI1	I1 瞬时电流直流值寄存器	R/W	32-bit 二进制补码	
0x100F	DATAOII2	I2 瞬时电流原始值寄存器	R/W	32-bit 二进制补码	
0x1053	DATAIAI2	I2 瞬时电流交流值寄存器	R/W	32-bit 二进制补码	
0x1048	DATAIDI2	I2 瞬时电流直流值寄存器	R/W	32-bit 二进制补码	

9.1.3.2 功率/有效值数据寄存器

表 9-11 功率/有效值数据寄存器 (R/W)

地址	寄存器	R/W	数据格式	说明
0x10D3	RMSIU	瞬时电压有效值寄存器	R/W	32-bit 二进制补码
0x10D8	RMSU	秒平均电压有效值寄存器	R/W	32-bit 二进制补码
0x10D1	DATAIP	瞬时 E1 功率寄存器	R/W	32-bit 二进制补码
0x10D4	RMSII1	I1 瞬时电流有效值寄存器	R/W	32-bit 二进制补码
0x10D6	DATAP	秒平均 E1 功率寄存器	R/W	32-bit 二进制补码
0x10D9	RMSI1	I1 秒平均电流有效值寄存器	R/W	32-bit 二进制补码
0x10DB	DATAAP1	I1 秒平均视在功率寄存器	R/W	32-bit 二进制补码
0x10D2	DATAIQ	瞬时 E2 功率寄存器	R/W	32-bit 二进制补码 当 DBLEN=0 时, 积分器导致额外的 1.568 倍增益, 可以通过比差进行校正 当 DBLEN=1 时, 无额外增益
0x10D5	RMSII2	I2 瞬时电流有效值寄存器	R/W	32-bit 二进制补码
0x10D7	DATAQ	秒平均 E2 功率寄存器	R/W	32-bit 二进制补码 当 DBLEN=0 时, 积分器导致额外的 1.568 倍增益, 可以通过比差进行校正 当 DBLEN=1 时, 无额外增益
0x10DA	RMSI2	I2 秒平均电流有效值寄存器	R/W	32-bit 二进制补码
0x10DC	DATAAP2	I2 秒平均视在功率寄存器	R/W	32-bit 二进制补码

发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时, 功率/有效值数据寄存器被复位。

当电能计量时钟频率为 3.2768MHz 时, 瞬时功率以及有效值数据寄存器的值在每个信号周期内更新两次, 瞬时功率数据的收敛时间为 100 ms, 瞬时有功值数据的收敛时间为 130 ms; 秒平均数据寄存器的值每 1.28 秒更新一次。所有寄存器均可读/可写。

无论 DBLEN 为何值, 标记 E1 的寄存器均用于存储 E1 通道的有功数据。

DBLEN=0 时, 标记 E2 的寄存器用于存储 E1 通道的无功数据; DBLEN=1 时, 标记 E2 的寄存器用于存储 E2 通道的有功数据。

无功计量时, 如果电流信号为正值, 则 E1 或 E2 瞬时功率寄存器的值为负, 功率值累加到反相 E1 或 E2 能量桶; 如果电流信号为负值, 则 E1 或 E2 瞬时功率寄存器的值为正, 功率值累加到正相 E1 或 E2 能量桶。

9.1.3.3 功率/有效值比差寄存器

表 9-12 功率/有效值比差寄存器 (R/W)

地址	寄存器		R/W	数据格式	说明
0x10E8	SCP	E1 功率比差寄存器	R/W	32-bit 二进制补码	发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，功率/有效值比差寄存器被复位。 比差校正范围 (-∞%~+49.9%)。所有寄存器均可读/可写。 无论 DBLEN 为何值，标记 E1 的寄存器均用于配置 E1 通道的比差数据。 DBLEN=0 时，标记 E2 的寄存器用于配置 E1 通道的比差数据； DBLEN=1 时，标记 E2 的寄存器用于配置 E2 通道的比差数据。
0x10EB	SCI1	I1 电流有效值比差寄存器	R/W	32-bit 二进制补码	
0x10E9	SCQ	E2 功率比差寄存器	R/W	32-bit 二进制补码	
0x10EC	SCI2	I2 电流有效值比差寄存器	R/W	32-bit 二进制补码	
0x10EA	SCU	电压有效值比差寄存器	R/W	32-bit 二进制补码	

9.1.3.4 功率二次补偿值寄存器

E1 及 E2 功率二次补偿寄存器被用于调整因串扰带来的计量误差。

表 9-13 功率二次补偿寄存器 (R/W)

地址	寄存器		R/W	数据格式	备注
0x10ED	PARAPC	E1 功率二次补偿寄存器	R/W	32-bit 二进制补码	发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，功率二次补偿寄存器被复位。所有寄存器均可读/可写。功率二次补偿的校表误差范围为-50%~+50%。 无论 DBLEN 为何值，E1 标记寄存器用于配置 E1 计量通道的功率二次补偿值。 DBLEN=0 时，E2 标记的寄存器用于配置 E1 计量通道的功率二次补偿值； DBLEN=1 时，E2 标记的寄存器用于配置 E2 计量通道的功率二次补偿值。
0x10EE	PARAQC	E2 功率二次补偿寄存器	R/W	32-bit 二进制补码	

9.1.3.5 带通滤波器系数寄存器

表 9-14 带通滤波器系数寄存器 (0x10EF)

地址	寄存器	R/W	数据格式	默认值
0x10EF	PARABPF 带通滤波器系数寄存器	R/W	32-bit 二进制补码	0x889374BC

发生 POR 复位、RSTn 低电平有效或 WDT 溢出时，带通滤波器系数寄存器被复位。

在复位后，此系数寄存器的默认值为 0x889374BC。

如果使能了电压/电流有效值计算通道上的带通滤波器 (BPFEN, bit6, PMCtrl3, 0x287A)，则，在降频计量时，电压/电流有效值带通滤波器的采样频率变为 819.2Hz，中心频率变为 12.5Hz，对 50Hz 信号有较大衰减，影响了电压/电流有效值的计量精度，也影响了线电压频率测量的准确度。所以，在从正常计量切换为降频计量时，在关闭能量累加、CF 脉冲输出及起动/潜动判断后，用户应将该寄存器配置为 0x911D3C9C。恢复到正常工作模式时，用户应将该寄存器配置为默认值 0x889374BC。

9.1.3.6 能量桶/脉冲计数器

表 9-15 能量桶/脉冲计数器 (R/W)

地址	寄存器	R/W	数据格式	说明
0x10F0	PPCNT 正相 E1 能量桶	R/W	32-bit 无符号正数	<p>发生 POR 复位、RSTn 低电平有效或 WDT 溢出时，所有能量桶和脉冲计数器被复位。</p> <p>能量桶的实际位宽为 42 位，读操作时，读出的为高 32 位的数据；写操作时，访问的是高 32 位，同时低 10 位会被写入 0，然后再参与能量桶运算。</p> <p>脉冲计数器每累加 2 次输出一个 CF 脉冲。</p> <p>当电能计量时钟频率为 3.2768MHz、1.6384MHz 或 819.2KHz 时，能量桶每秒钟累加 12800 次；当电能计量时钟频率为 32768Hz 时，能量桶每秒钟累加 2979 次。</p> <p>无论 DBLEN 为何值，标记 E1 的寄存器均用于存储 E1 计量通道的</p>
0x10F1	NPCNT 反相 E1 能量桶	R/W	32-bit 无符号正数	
0x10F2	PPCFCNT 正相 E1 能量脉冲计数器	R/W	32-bit 无符号正数	
0x10F3	NPCFCNT 反相 E1 能量脉冲计数器	R/W	32-bit 无符号正数	
0x10F6	PQCNT 正相 E2 能量桶	R/W	32-bit 无符号正数	

地址	寄存器		R/W	数据格式	说明
0x10F7	NQCNT	反相 E2 能量桶	R/W	32-bit 无符号正数	有功数据。 DBLEN=0 时, 标记 E2 的寄存器用于存储 E1 计量通道的无功数据; DBLEN=1 时, 标记 E2 的寄存器用于存储 E2 计量通道的有功数据。
0x10F8	PQCFCNT	正相 E2 能量脉冲计数器	R/W	32-bit 无符号正数	无功计量时, 如果电流信号为正值, 则 E1 或 E2 瞬时功率寄存器的值为负, 功率值累加到反相 E1 或 E2 能量桶; 如果电流信号为负值, 则 E1 或 E2 瞬时功率寄存器的值为正, 功率值累加到正相 E1 或 E2 能量桶。
0x10F9	NQCFCNT	反相 E2 能量脉冲计数器	R/W	32-bit 无符号正数	

9.1.3.7 门限值寄存器/常数功率值寄存器

表 9-16 门限值寄存器/常数功率值寄存器 (R/W)

地址	寄存器		R/W	数据格式	说明
0x10F4	GATEP	E1 能量脉冲门限值寄存器	R/W	32-bit 无符号正数	无论 DBLEN 为何值, 标记 E1 的寄存器用于配置 E1 计量通道的门限值。
0x10F5	GATECP	E1 能量起动/潜动判断门限值寄存器	R/W	32-bit 无符号正数	
0x10FA	GATEQ	E2 能量脉冲门限值寄存器	R/W	32-bit 无符号正数	DBLEN=0 时, 标记 E2 的寄存器用于配置 E1 计量通道的门限值; DBLEN=1 时, 标记 E2 的寄存器用于配置 E2 计量通道的门限值。
0x10FB	GATECQ	E2 能量起动/潜动判断门限值寄存器	R/W	32-bit 无符号正数	
0x10FC	DATA CP	常数功率值寄存器	R/W	32-bit 二进制补码	

发生 POR 复位、RSTn 低电平有效或 WDT 溢出时, 所有门限值寄存器和常数功率值寄存器被复位。所有寄存器均可读/可写。

因为所有能量桶的实际位宽为 42 位, 所以能量脉冲门限值寄存器的内容自动在低位被补 10 个 0, 然后再参与能量桶运算。

使能起动/潜动判断后, 防潜动能量桶的输入固定为 1。当电能计量时钟频率为 3.2768MHz、1.6384MHz 或 819.2kHz 时, 防潜动能量桶每秒钟累加 12800 次; 当电能计量时钟频率为 32768Hz 时, 防潜动能量桶每秒钟累加 2979 次。

当防潜动能量桶的累加值先达到起动/潜动判断门限值时, E1/E2 能量桶被清空, 即, E1/E2 计量通道进入潜动状态。当 E1/E2 能量桶的累加值先达到能量脉冲门限值时, 防潜动能量桶被清空, 即, E1/E2 计量通道进行能量计量。

当选择常数功率进行能量累加的时候 (PSEL1/PSEL0 = 10), 常数功率值寄存器的值会被累加进入正相 E1 能量桶。

9.1.3.8 频率值寄存器

表 9-17 频率值寄存器 (DATAFREQ, 0x10FD)

0x10FD, R, 频率值寄存器, DATAFREQ							
	值	bit15	bit14	bit1	bit0
默认值	0x0000	0	0	0	0	0	0

发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，频率值寄存器复位。该寄存器的值为 16-bit 无符号正数，只读。

当电能计量时钟频率为 3.2768MHz 时，频率值寄存器的数据更新时间是 320ms，收敛时间是 410ms。频率测量的精度可达到 0.01Hz，频率的测量范围是 45Hz ~ 70Hz。

9.1.3.9 M 通道数据寄存器

表 9-18 M 通道数据寄存器

地址	寄存器		R/W	数据格式
0x10CE	DATAOM	M 通道原始数据值寄存器	R/W	32-bit 二进制补码
0x10CF	DATADM	M 通道直流数据值寄存器	R/W	32-bit 二进制补码
0x10D0	DATAADM	M 通道直流秒平均数据值寄存器	R/W	32-bit 二进制补码

发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，M 通道数据寄存器复位。

当电能计量时钟频率为 3.2768MHz 时，DATAOM 的数据每秒钟刷新 3200 次，收敛时间是 60ms；DATADM 的数据每秒钟刷新 20 次，收敛时间是 150ms；DATAADM 的数据每 1.28 秒刷新一次，收敛时间是 3s。

降频采样时，收敛时间应乘以相应的降频系数。

9.3 模拟信号输入

V9811 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，接线方式如图 9-2（采用 AGND 当成共模电压，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。

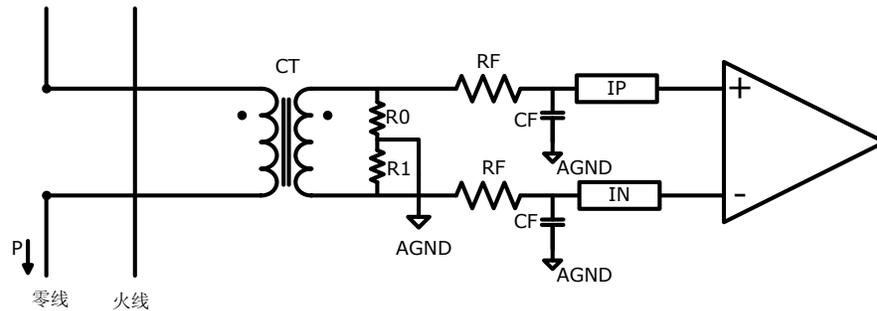


图 9-2 采用 CT 输入电流

也可以采用锰铜电阻分流网络输入电流，采用 AGND 接地。

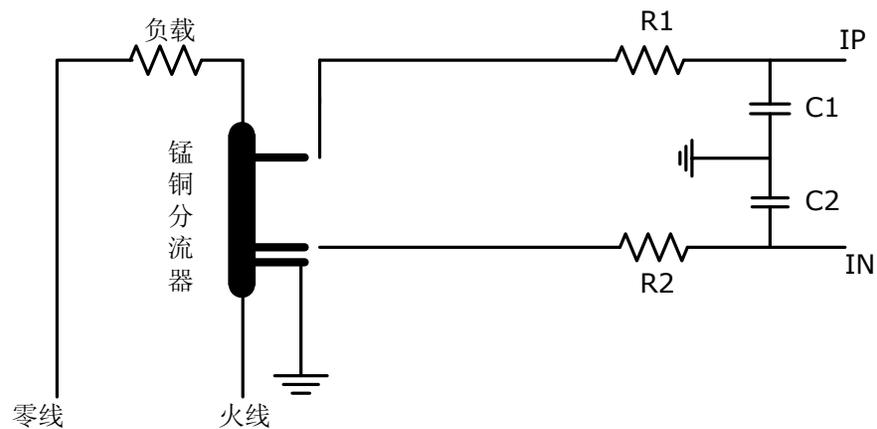


图 9-3 采用锰铜电阻分流网络输入电流

V9811 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。

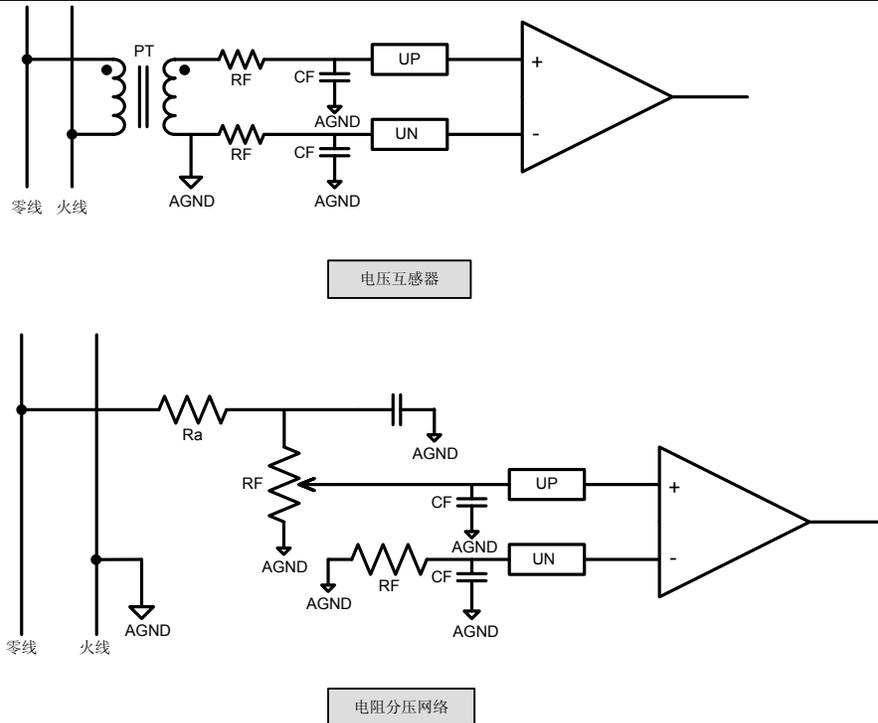


图 9-4 电压输入方式

V9811 三个差分输入通道的电压输入范围是 $\pm 200\text{mV}$ （幅值），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模数转换器（以下简称 ADC）之间的量程相匹配，用户可通过设置 ADC 控制寄存器 0（CtrlADC0，0x2858）对电流和电压通道进行模拟增益配置。

表 9-19 电压/电流通道模拟增益配置

寄存器	位	说明
ADC 控制寄存器 0 (CtrlADC0, 0x2858)	bit6	ADCGU 电压通道模拟增益控制。默认为 0。 0, $\times 1$; 1, $\times 2$
	bit[5:3]	ADCGB<2:0> IB 通道模拟增益控制。默认为 0。 000, $\times 1$; 001, $\times 4$; 010, $\times 8$; 011, $\times 16$; 111, $\times 40$
	bit[2:0]	ADCGA<2:0> IA 通道模拟增益控制。默认为 0。 000, $\times 1$; 001, $\times 4$; 010, $\times 8$; 011, $\times 16$; 111, $\times 40$

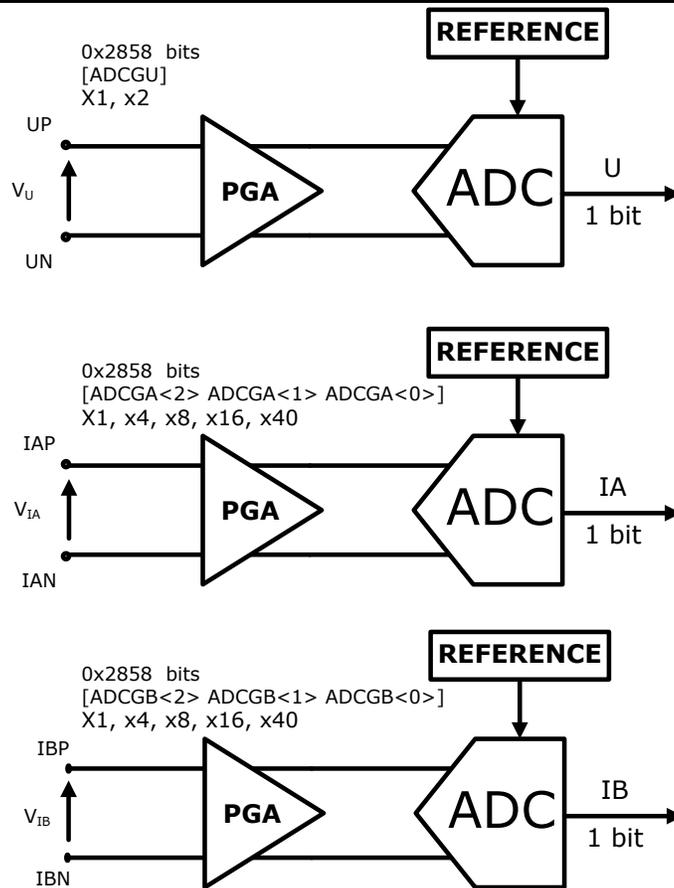


图 9-5 电流/电压通道模拟增益配置

9.4 模拟/数字转换

V9811 中的电压/电流通道 ADC 都采用的是二阶的 Σ/Δ ADC 结构，其满量程输入范围是 $\pm 1100\text{mV}$ 。用户可通过 ADC 控制寄存器 6 (CtrlADC6, 0x2864) 开启或关闭各路通道的 ADC。

表 9-20 开关电压/电流通道 ADC

寄存器	位	说明
ADC 控制寄存器 6 (CtrlADC6 , 0x2864)	Bit0	ADCAPDN 开关 IA 通道 ADC。默认为 0。 0, 关闭; 1, 开启
	Bit1	ADCBPDN 开关 IB 通道 ADC。默认为 0。 0, 关闭; 1, 开启
	Bit2	ADCUPDN 开关电压通道 ADC。默认为 0。 0, 关闭; 1, 开启

ADC 时钟频率由时钟控制寄存器 (CtrlCLK, 0x2867) 中的控制位 ADCLKSEL<1>和 ADCLKSEL<0>决定。ADC 时钟频率必须是电能计量时钟频率的 1/4。ADC 时钟频率默认为 204.8kHz。

表 9-21 ADC 时钟频率配置

寄存器	位	说明
时钟控制寄存器 (CtrlCLK, 0x2867)	Bit[5:4] ADCLKSEL<1:0>	ADC 时钟频率选择, 204.8KHz 为基准。默认为 0。00, ×1; 01, ×2; 10, ×4; 11, ×8

经过模拟/数字转换器, 各通道的模拟输入信号转换为 22_20-bit 的数字信号, 即, 数据位数为 22-bit, 其中, bit21 和 bit20 为符号位。如果 bit[19:0] 的值对应为整数 A, 则此数据对应的数字信号的值为 $A/2^{20}$ (十进制, 表示[-2, +2]范围内的一个数)。

模拟信号经 ADC 转换为数字信号后被输入角差校正模块进行角差校正, 以消除因采样电路和 ADC 的失配引起的电压/电流信号之间的相位差。

9.5 角差校正

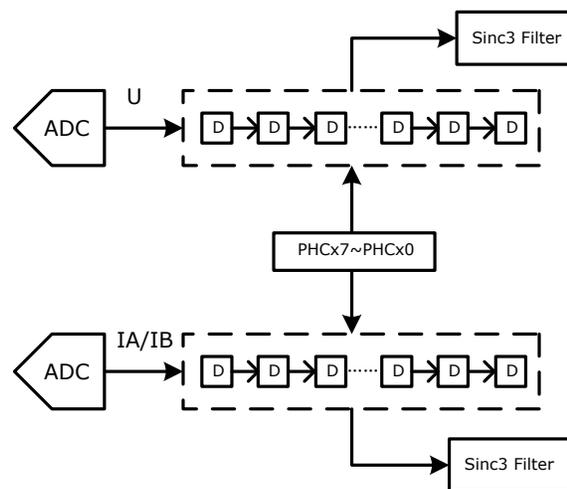


图 9-6 角差校正原理图

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。

在 V9811 中, 当 ADC 时钟频率是 819.2kHz 时, 角差校正的最小分辨率是 0.02197265625 度, 总校正量为 ±1.4 度。

表 9-22 角差校正配置

寄存器	位	说明
PM 控制寄存器 1 (PMCtrl1, 0x2878)	bit6 PHCEN	使能角差校正。0, 禁止; 1, 使能。
角差校正控制寄存器 1 (PHCCtrl1, 0x287B)	Bit[7:0] PHCA7~PHCA0	IA 通道角差校正值。Bit7 为符号位, bit[5:0] 为角差校正值的绝对值。
角差校正控制寄存器 2 (PHCCtrl2, 0x287C)	Bit[7:0] PHCB7~PHCB0	IB 通道角差校正值。Bit7 为符号位, bit[5:0] 为角差校正值的绝对值。

9.6 电流计量通道选择

经过角差校正后，IA 和 IB 两路电流信号可以选择进入不同的计量通道参与后续的运算。

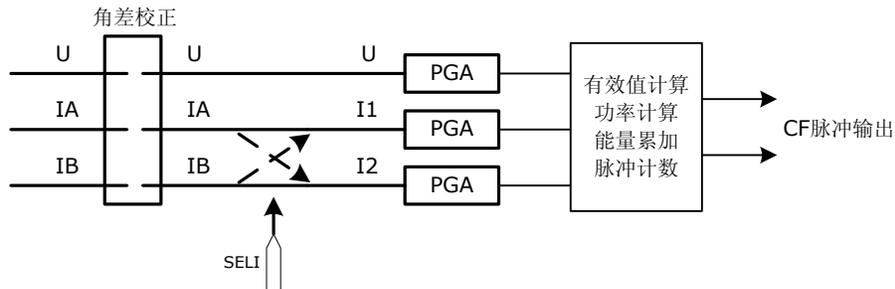


图 9-7 电流计量通道选择

表 9-23 电流计量通道选择控制位

寄存器	位	功能说明
PM 控制寄存器 1 (PMCtrl1, 0x2878)	SELI Bit5	电流计量通道选择。默认值为 0 0, 电流通道 A (IA) 的信号进入计量通道 I1, 电流通道 B (IB) 的信号进入计量通道 I2; 1, 电流通道 A (IA) 的信号进入计量通道 I2, 电流通道 B (IB) 的信号进入计量通道 I1

9.7 数字信号输入

过采样 Σ/Δ ADC 输出 1bit 码流伴有大量的高频噪声，通常用 CIC 滤波器[抽取滤波器(Decimation Filter)]来抑制该噪声，并将系统时钟的采样频率由 FCLK 降低到 FCLK/N(N 是抽取滤波器的抽取率)，而数据长度由 1-bit 扩展为位宽为 $(1+\log_2 N)$ 的数据流。

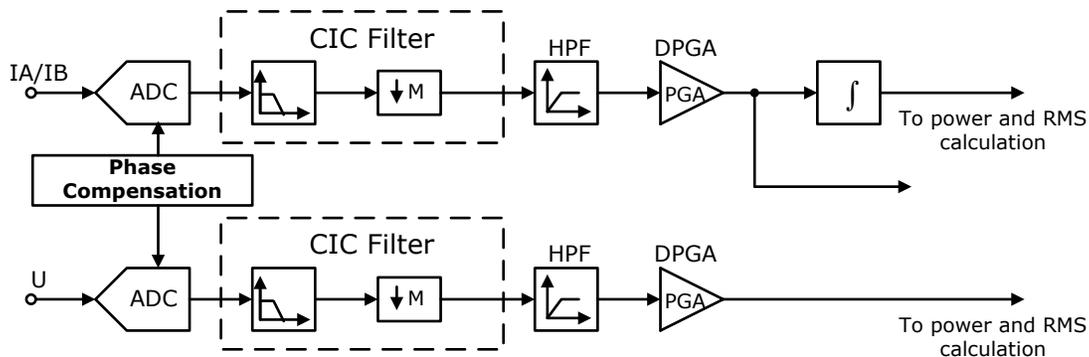


图 9-8 数字信号输入

通过配置 PM 控制寄存器 1 (PMCtrl1, 0x2878) 中的 bit[2:0]，用户可以决定是否将 ADC 输出的数字信号输入抽取滤波器中进行降采样处理。开启该功能时，ADC 的输出信号被累加进入抽取滤波器；关闭该功能时，ADC 输出 0，抽取滤波器输出为常数。

表 9-24 数字信号输入使能

寄存器	bit	功能说明
PM 控制寄存器 1 (PMCtrl1, 0x2878)	ONI2 Bit2	I2 通道数字信号输入使能。默认值为 0 0, 屏蔽 I2 通道的数字信号输入, 使 I2 通道的输入信号为 0; 1, 使 I2 通道得到相应的输入数据。
	ONI1 Bit1	I1 通道数字信号输入使能。默认值为 0 0, 屏蔽 I1 通道的数字信号输入, 使 I1 通道的输入信号为 0; 1, 使 I1 通道得到相应的输入数据。
	ONU Bit0	U 通道数字信号输入使能。默认值为 0 0, 屏蔽 U 通道的数字信号输入, 使 U 通道的输入信号为 0; 1, 使 U 通道得到相应的输入数据。

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 V9811 中, 高通滤波器不可被旁路。当 ADC 时钟频率为 819.2kHz 时, 该高通滤波器的响应时间是 60ms。

用户可通过 PM 控制寄存器 2 (PMCtrl2, 0x2879) 和 PM 控制寄存器 3 (PMCtrl3, 0x287A) 对经高通滤波处理的电压/电流信号进行数字增益配置, 从而增大信号输入下对截断噪声的抑制能力。

表 9-25 计量通道信号数字增益配置

寄存器	bit	功能说明
PM 控制寄存器 3 (PMCtrl3, 0x287A)	PGANS Bit3	计量通道 I2 数字增益符号位 0, 正号; 1, 负号
	PGAN2~PGAN0 Bit[2:0]	计量通道 I2 数字增益控制, 取值范围为 0~5, 增益计算方法为 $Gain=2^{PGANx}$
PM 控制寄存器 2 (PMCtrl2, 0x2879)	PGACS Bit7	计量通道 I1 数字增益符号位 0, 正号; 1, 负号
	PGAC2~PGAC0 Bit[6:4]	计量通道 I1 数字增益控制, 取值范围为 0~5, 增益计算方法为 $Gain=2^{PGACx}$
	PGAUS Bit3	电压通道 (U) 数字增益符号位 0, 正号; 1, 负号
	PGAU2~PGAU0 Bit[2:0]	电压通道 (U) 增益控制, 取值范围为 0~5, 增益计算方法为 $Gain=2^{PGAUx}$

此时获得的电压/电流的数字信号的值分别根据如下公式计算得到:

$$U_a = PGAd_u \times PGA_{u_a} \times A_{u_a} \times \sin \omega t \div 1.185 = DU_a \times \sin \omega t$$

$$I_a = PGAd_i \times PGA_{i_a} \times A_{i_a} \times \sin(\omega t + \varphi) \div 1.185 = DI_a \times \sin(\omega t + \varphi)$$

其中,

PGAd_{ua} 或 PGAd_{ia} 为各通道的数字增益;



PGAua 或 PGAia 为各通道的模拟增益；
Aua 或 Aia 为各通道信号的输入信号的幅度（V）；
1.185 为基准电压（V）。

9.8 有效值计算和校正

用户可通过 PREN 位（bit4, 0x2878, PMCtrl1）开启或关闭电压/电流有效值计算电路，也可以通过 BPFEN 位（bit6, 0x287A, PMCtrl3）开启或关闭有效值计算电路上的带通滤波器，并通过调节带通滤波器系数，来提高有效值的准确度。

表 9-26 有效值计算的寄存器配置

寄存器	bit	说明
PM 控制寄存器 1 (PMCtrl1, 0x2878)	PREN Bit4	使能功率/有效值/视在功率计算。默认为 0（禁止）。
PM 控制寄存器 3 (PMCtrl3, 0x287A)	BPFEN Bit6	使能电压/电流有效值计算电路上的带通滤波器。默认为 0（禁止）。
带通滤波器系数寄存器 (0x10EF)		带通滤波器系数寄存器。 正常计量时，该寄存器的值应采用默认值 0x889374BC。降频计量时，在关闭能量累加、CF 脉冲输出和起动/潜动判断后，该寄存器的值应配为 0x911D3C9C。 恢复到正常计量后，该寄存器应恢复为默认值。

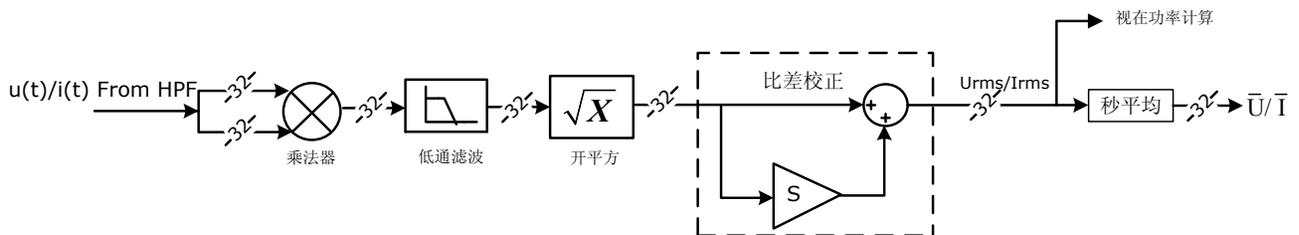


图 9-9 电流/电压有效值计算信号处理

电流/电压有效值计算的信号流如图 9-9 所示。来自高通滤波器输出的电流/电压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，得到电流/电压的平方，然后再做开平方运算，得到的 32 位数据，该数据经过比差校正后，存于瞬时电压/电流有效值寄存器（R/W）。

电压/电流有效值的计算公式：

$$U_{rms} = \frac{\sqrt{2}}{2} \times U_a$$

$$I_{rms} = \frac{\sqrt{2}}{2} \times I_a$$

其中，Urms 和 Irms 为电压和电流的有效值；

Ua 和 Ia 为电压和电流的数字信号值。

设原始的有效值为 RMS'，校正补偿后的有效值为 RMS，比差校正值为 S，则，补偿后的值为：

$$RMS = RMS' \times (1 + S)$$

用户可在功率/有效值比差寄存器 (R/W) 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会进行秒平均，存于秒平均电压/电流有效值寄存器。

上述所有寄存器均为 32 位补码数据，可读可写。当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，这些寄存器被复位。当电能计量时钟频率为 3.2768MHz 时，瞬时电压/电流有效值寄存器更新频率为 100Hz，稳定时间为 130ms；秒平均电压/电流有效值寄存器的值每 1.28s 更新一次。

9.9 视在功率计算

视在功率计算与电压/电流有效值计算同时开启或关闭。

经过比差校正后的电压/电流有效值会被用于视在功率计算：

$$S = I_{rms} \times U_{rms}$$

其中，

S 为视在功率（单位：VA）；

I_{rms} 为经过比差校正后的电流有效值；

U_{rms} 为经过比差校正后的电压有效值。

计算得到的视在功率代数，进一步平均后，得到各通道视在功率的平均值，存于秒平均视在功率寄存器中，其更新时间为 1.28s。上述寄存器为 32 位补码数据，可读可写。当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，这些寄存器被复位。

9.10 有功/无功功率计算和校正

有功/无功功率计算与电压/电流有效值计算同时开启或关闭。

在 V9811 中，当两路电流计量通道的信号用于功率计算时，用户可通过 DBLEN 位 (bit4, 0x287A, PMCtrl3) 选择电流计量通道的信号处理方式。如表 9-27 所示，当 DBLEN 位写入 0 时，仅 I1 通道的电流信号同时进行有功和无功功率计算；当 DBLEN 位写入 1 时，I1 和 I2 两路通道均进行有功功率计算。

表 9-27 功率计算选择

控制位	功率计算选择		数据存储	
	I1 通道	I2 通道	“E1”标记寄存器	“E2”标记寄存器
0	有功/无功	×	有功数据	无功数据
1	有功	有功	I1 有功数据	I2 有功数据

注：表中所述“E1”/“E2”标记寄存器，包括：瞬时功率寄存器，秒平均功率寄存器，功率比差寄存器，功率二次补偿值寄存器，正相/反相能量桶和脉冲计数器，和能量脉冲门限值和起动/潜动判断门限值寄存器等。

9.10.1 有功功率计算

当 DBLEN=0 时，I1 通道电流信号与电压通道信号同时进行有功和无功功率计算；当 DBLEN=1 时，I1 和 I2 通道电流信号与电压通道信号均进行有功功率计算。

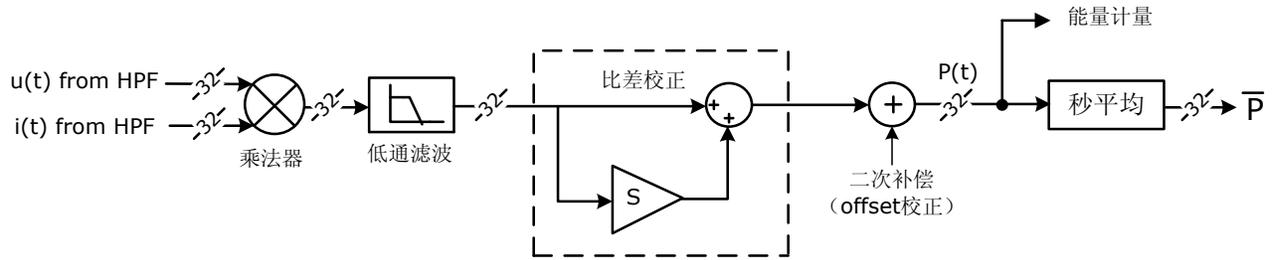


图 9-10 有功功率计算

来自高通滤波器输出的电流和电压信号相乘，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，该数据经过比差校正，再经过二次补偿（offset 校正）后，存于瞬时 E1 或 E2 功率寄存器，参与后续有功能量累加。

有功功率的计算公式如下：

$$P = \frac{1}{2} \times U_a \times I_a$$

其中，P 为有功功率（W）；

U_a 和 I_a 为电压和电流的数字信号的值。

二次补偿后的有功功率数据会进行秒平均，存于秒平均 E1 或 E2 功率寄存器。

上述所有寄存器均为 32 位补码数据，可读可写。当发生 POR 复位、RSTN 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，这些寄存器被复位。

当电能计量时钟频率为 3.2768MHz，ADC 时钟频率是 819.2KHz 时：

1. 有功功率计算电路上的低通滤波器的响应时间是 40ms；
2. 瞬时功率寄存器更新频率为 100Hz，稳定时间为 100ms；
3. 秒平均数据寄存器的值每 1.28 秒更新 1 次。

9.10.2 无功功率计算

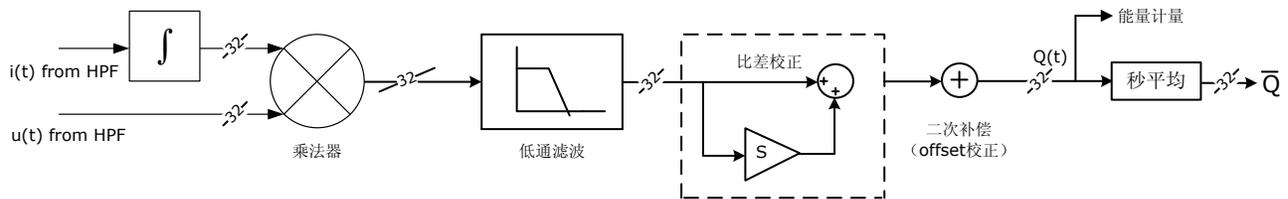


图 9-11 无功功率计算

当 DBLEN=0 时，I1 通道的电流信号同时进行有功功率和无功功率计算。

无功功率计算与有功功率计算类似，但，采用锰铜电阻分流网络或 CT 输入电流时，输入的电流信号需要经过一个数字积分网络，经该网络处理后输出的电流信号用于无功功率计算。该积分器会带来额外的 1.568 倍增益，可经由比差校正消除。

经低通滤波器处理后的 32 位无功功率数据先后经过比差校正和二次补偿（offset 校正）后，存于瞬时 E2 功率寄存器，参与后续无功能量累加。二次补偿后的无功功率数据也会进行秒平均，存于秒平均 E2 功率寄存器。

当电能计量时钟频率为 3.2768MHz，ADC 时钟频率是 819.2KHz 时：

1. 无功功率计算电路上的低通滤波器的响应时间是 40ms;
2. 瞬时功率寄存器的更新频率为 100Hz, 稳定时间为 100ms;
3. 秒平均数据寄存器的值每 1.28 秒更新 1 次。

9.10.3 功率校正

在 V9811 中, 为了修正各个通道间的系统偏差, 消除功率计算过程中因串扰而引起的功率值失调, 同时修正小信号下各个数据的线性度, 根据计算得到的功率值均需要进行比差校正和二次补偿 (offset 校正) 后才会存入对应的数据寄存器中。

设原始的功率值为 P' , 校正补偿后的功率值为 P , 比差校正值为 S , 二次补偿值为 C , 则, 补偿后的值为:

$$P = P' \times (1 + S) + C$$

用户可在“功率/有效值比差寄存器 (R/W)”和“功率二次补偿寄存器 (R/W)”中设置功率的比差校正值和二次补偿值。这些寄存器的数据均为 32-bit 补码, 可读可写。当发生 POR 复位、RSTN 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时, 上述寄存器均被复位。

9.11 能量累加和 CF 脉冲输出

9.11.1 能量累加和脉冲产生

用户可通过 PM 控制寄存器 4 (PMCtrl4, 0x287D) 的 EGYEN 位 (bit3) 开启或关闭能量累加和脉冲计数, 通过 PSEL1~PSEL0 位 (bit[1:0]) 选择累加到正相 E1 能量桶的功率源; 通过脉冲输出控制寄存器 (CFCtrl, 0x287E) 中的 CFSELR1~CFSELR0 (bit[3:2]) 和 CFSEL1~CFSEL0 (bit[1:0]) 选择 E2 和 E1 能量脉冲源。

V9811 支持脉冲常数加倍。通过增大脉冲常数, 使能量桶门限值减小, 从而加快脉冲产生速度, 用户可通过 CFQR1~CFQR0 (bit[7:6]) 和 CFQ1~CFQ0 (bit[5:4]) 配置能量脉冲的产生速度。当输入小信号时, 提高能量脉冲的产生速度, 从而缩短校表所需要的时间。

如图 9-12 所示, 当使能能量累加和脉冲计数时, V9811 可根据功率符号将功率分别累加到正反相 E1/E2 能量桶, 当能量桶的值超过相对应的能量脉冲门限值时, 产生一个脉冲, 同时在对应的脉冲计数器中加 1, 并从能量桶累积数据中减去一个能量脉冲门限值。

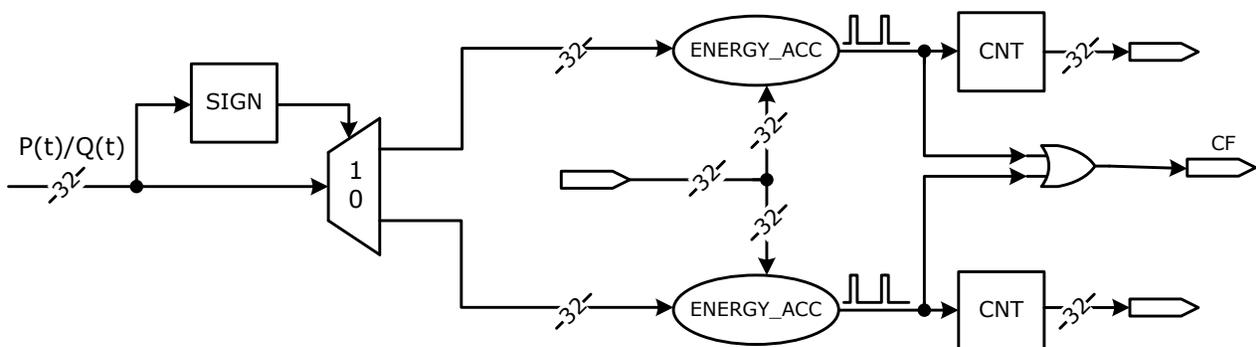


图 9-12 有功/无功能量累加和脉冲计数

表 9-28 能量累加/功率源选择/脉冲源选择/能量脉冲产生速率控制位

寄存器	位		说明
PM 控制寄存器 4 (PMCtrl4, 0x287D)	Bit3	EGYEN	能量累加和脉冲计数使能。 0, 禁止; 1, 使能
	Bit[1:0]	PSEL1 ~PSEL0	正相 E1 能量桶功率源选择。 00 或 11, E1 功率; 01, I1 通道电流有效值; 10, 常数功率
脉冲输出控制寄存器 (CFCtrl, 0x287E)	Bit[7:6]	CFQR1~CFQR0	E2 能量脉冲产生速率控制, 可用于小信号加速校表。 00, 1 倍; 01, 4 倍; 10, 8 倍; 11, 16 倍
	Bit[5:4]	CFQ1~CFQ0	E1 能量脉冲产生速率控制, 可用于小信号加速校表。 00, 1 倍; 01, 4 倍; 10, 8 倍; 11, 16 倍
	bit[3:2]	CFSELR1~CFSELR0	E2 能量脉冲源选择。 01, 正相 E2 能量; 10, 反相 E2 能量; 00/11, 正相以及反相 E2 能量的绝对值之和
	Bit[1:0]	CFSEL1~CFSEL0	E1 能量脉冲源选择。 01, 正相 E1 能量; 10, 反相 E1 能量; 00/11, 正相以及反相 E1 能量的绝对值之和

当电能计量时钟频率为 3.2768MHz、1.6384MHz 或 819.2kHz 时, 能量桶每秒钟累加 12800 次; 当电能计量时钟频率为 32768Hz 时, 能量桶每秒钟累加 2979 次。

所有能量桶、能量脉冲计数器和能量脉冲门限值寄存器有效数据均为 32-bit 无符号正数, 可读可写。能量桶的实际位宽为 42 位, 读操作时, 读出高 32 位; 写操作时, 访问高 32 位, 同时低 10 位会被自动补 0。因此, 能量脉冲门限值寄存器的内容会被自动在低位补 10 个 0 后, 再参与能量桶运算。

当发生 POR 复位、RSTn 低电平有效或 WDT 溢出复位时, 所有能量桶、能量脉冲计数器和能量脉冲门限值寄存器均被复位。

9.11.2 CF 脉冲输出

用户可通过 PM 控制寄存器 4 (PMCtrl4, 0x287D) 的 CFENR 位 (bit5) 和 CFEN 位 (bit4) 开启或关闭 E2 和 E1 计量通道的 CF 脉冲输出, 并通过 CFXCG (bit2) 选择 CF1 和 CF2 脉冲输出引脚。

当 CF 脉冲输出使能时, CF 脉冲输出, 脉冲计数器每累加 2 次输出一个 CF 脉冲。

V9811 提供三个 CF 脉冲输出引脚: Pin26 (P9.5/CF2)、Pin25 (P9.6/CF1) 和 Pin22 (P1.3/CFx)。其中,

- Pin26 和 Pin25 仅在 P9 功能选择寄存器 (P9FS SFR, 0xAD) 的 bit5 和 bit6 置 1 的时候才用于 CF1 和 CF2 脉冲的输出;
- Pin22 用作 CF 脉冲输出则取决于 P1.3 功能选择寄存器 (P13FS, 0x28C7, R/W) 的值:
当 P13FS=1 时, 该引脚用于 CF2 脉冲输出;
当 P13FS=4 时, 该引脚用于 CF1 脉冲输出。

表 9-29 CF 脉冲输出配置

寄存器	位		说明
PM 控制寄存器 4 (PMCtrl4, 0x287D)	Bit5	CFENR	E2 能量 CF 脉冲输出使能 0, 禁止; 1, 使能
	Bit4	CFEN	E1 能量 CF 脉冲输出使能 0, 禁止; 1, 使能
	Bit2	CFXCG	CF 脉冲输出引脚选择 0, E1 能量 CF 脉冲从引脚 CF1 输出, E2 能量 CF 脉冲从引脚 CF2 输出; 1, E2 能量 CF 脉冲从引脚 CF1 输出, E1 能量 CF 脉冲从引脚 CF2 输出

当电能计量时钟频率为 3.2768MHz, ADC 时钟频率为 819.2kHz 时, CF 脉冲的最高输出频率是 6.4kHz, CF 正常输出的脉冲宽度是 80ms, 当不能保证 80ms 脉冲宽度, 且为 1/2 占空比的时候, 输出占空比 50% 的方波。

CF 脉冲输出频率与能量脉冲产生频率成正比。通过脉冲输出控制寄存器 (CFCtrl, 0x287E) 中的 bit[7:4] (CFQR1~CFQR0 和 CFQ1~CFQ0) 配置能量脉冲的产生速度, 即可调整 CF 脉冲输出速度。

9.12 起动/潜动判断

在 V9811 中, 用户可通过 CRPENR 位和 CRPEN 位 (bit7 和 bit6, 0x287D, PMCtrl4) 使能 E2 和 E1 计量通道的起动/潜动判断功能。

V9811 内部有一个防潜动能量桶, 当起动/潜动判断使能后, 该能量桶的输入固定为 1, 而 E1/E2 能量桶输入为 E1/E2 功率或常数功率值。当电能计量时钟频率为 3.2768MHz、1.6384MHz 或 819.2kHz 时, E1/E2 能量桶和潜动能量桶每秒累加 12800 次; 当电能计量时钟频率为 32768Hz 时, E1/E2 能量桶和潜动能量桶每秒累加 2979 次。当防潜动能量桶的累加值先达到起动/潜动判断门限值时, E1/E2 能量桶被清空, E1/E2 计量通道进入潜动状态。当 E1/E2 能量桶的累加值先达到 E1/E2 能量脉冲门限值时, 防潜动能量桶被清空, E1/E2 计量通道进入电能计量状态。

用户可通过 CRPST 位和 CRPSTR 位 (bit15 和 bit14, 0x287F, CRPST) 来判断 E1/E2 计量通道是否处于潜动状态。

表 9-30 起动/潜动判断控制位和标志位

寄存器	bit	说明
PM 控制寄存器 4 (PMCtrl4, 0x287D)	CRPENR Bit7	E2 计量通道起动/潜动判断使能。默认为 0 (禁止)。
	CRPEN Bit6	E1 计量通道起动/潜动判断使能。默认为 0 (禁止)。
防潜标志寄存器 (CRPST, 0x287F)	CRPST Bit7	E1 计量通道起动/潜动状态指示, 只读。 0, 正常计量; 1, 潜动
	CRPSTR Bit6	E2 计量通道起动/潜动状态指示, 只读。 0, 正常计量; 1, 潜动

当发生 POR 复位、RSTn 低电平有效, 或 WDT 溢出复位时, E1/E2 能量起动/潜动判断门限值寄存器被复位。

9.13 线电压频率测量

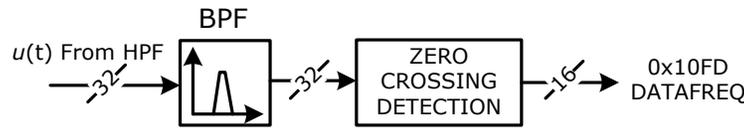


图 9-13 线电压频率测量原理

V9811 支持线电压频率测量。其实现原理是先将高通滤波之后的线电压信号 $u(t)$ 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，存于频率值寄存器（DATAFREQ, 0x10FD）。线电压频率测量的精度可达到 0.01Hz，频率的测量范围是 45Hz~70Hz。

注：在降频计量时，带通滤波器的采样频率降为 819.2kHz，中心频率降为 12.5Hz，对 50Hz 信号有较大衰减，影响了线电压频率测量的精度。所以，从正常计量切换为降频计量时，用户应将带通滤波器系数寄存器（0x10EF）配置为 0x911D3C9C。

线电压频率值直接从频率值寄存器（DATAFREQ, 0x10FD）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：

$$f = f_{\text{ADC}} / \text{Freq}$$

其中， f ：信号频率，Hz；

f_{ADC} ：ADC 时钟频率，Hz；

Freq：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。当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，频率值寄存器被复位。

当电能计量时钟频率为 3.2768MHz，ADC 时钟频率为 819.2KHz 时，频率值寄存器的更新时间为 320ms，稳定时间为 410ms。

V9811 可配置为 50Hz 电网或 60Hz 电网应用。在 50Hz 电网下，当信号频率为 50Hz 时，频率值寄存器输出 0x4000（十进制：16384）；在 60Hz 电网下，当信号频率为 60Hz 时，频率值寄存器输出 0x4000。

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。

表 9-31 信号频率大小与频率值寄存器输出值的线性关系

电网	信号频率	频率值寄存器读数
50Hz	25Hz	0x8000
	50Hz	0x4000
	100Hz	0x2000
60Hz	30Hz	0x8000
	60Hz	0x4000
	120Hz	0x2000

9.14 M 通道测量

9.14.1 M 通道结构

M 通道的外部 and 内部连接图如图 9-14 和图 9-15 所示。M 通道可配置成测量内部地/温度/电池电压/外部电压信号。

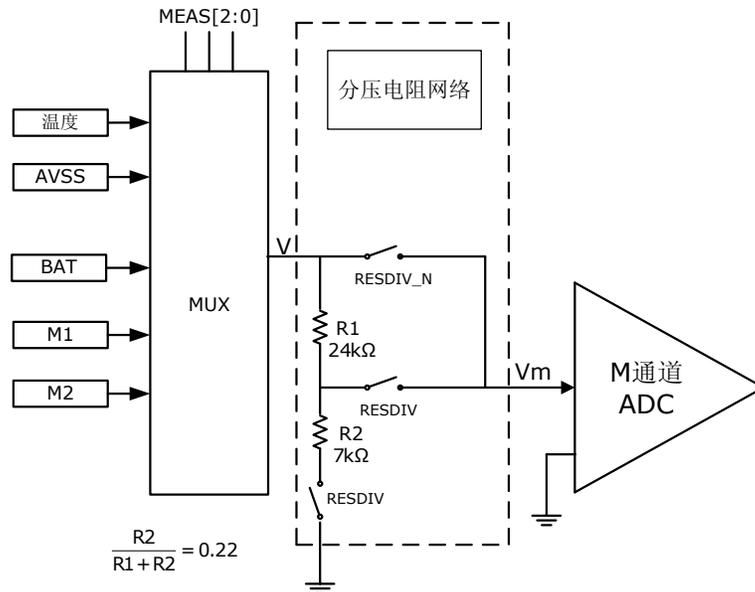


图 9-14 M 通道外部连接图

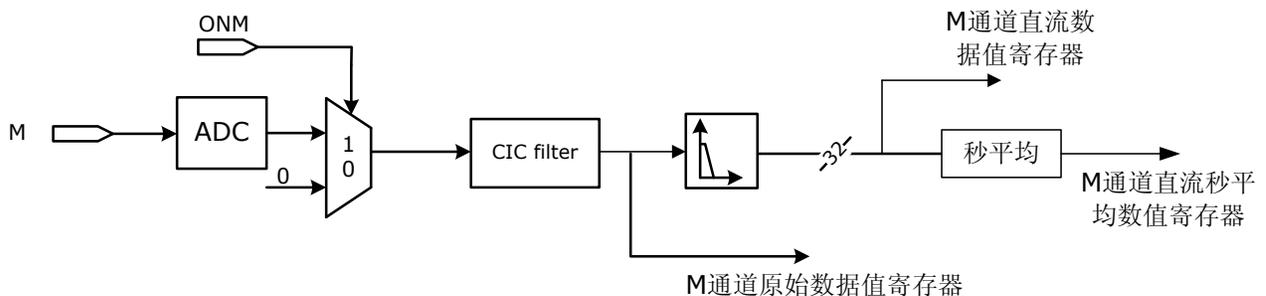


图 9-15 M 通道内部连接图

由于 M 通道只有一个 ADC，所以电池电压、温度和外部电压信号只能交替测量，如图 9-14 所示。用户可通过 ADC 控制寄存器 0 (CtrlADC0, 0x2858) /ADC 控制寄存器 5 (CtrlADC5, 0x2863) /ADC 控制寄存器 6 (CtrlADC6, 0x2864) 和 PM 控制寄存器 1 (PMCtrl1, 0x2878) 配置 M 通道的功能。

表 9-32 M 通道配置

寄存器	位		功能	说明
ADC 控制寄存器 0 (CtrlADC0, 0x2858)	bit7	ADCGM	M 通道的 ADC 增益控制 (电容)	0, ×1; 1, ×2
ADC 控制寄存器 5 (CtrlADC5, 0x2863)	Bit7	PTATEN	选择温度传感器	1, 温度传感器 1; 0, 温度传感器 0
	Bit4	RESDIV	使能 M 通道输入信号分压选择, 分压系数为 0.226	1, 使能。默认为 0。
	Bit[2:0]	MEAS<2:0>	选择 M 通道测试功能	000, 经由 AVSS 引脚测地; 001, 测温度; 010, 测经 BAT 引脚输入的系统电池电压信号; 011/100/101, 保留; 110, 测经 M1 引脚 (Pin55) 输入的外部直流信号; 111, 测经 M2 引脚 (Pin56) 输入的外部直流信号。
ADC 控制寄存器 6 (CtrlADC6, 0x2864)	bit3	ADCMPDN	开关 M 通道 ADC	0, 关闭 M 通道 ADC; 1, 开启 M 通道 ADC
PM 控制寄存器 1 (PMCtrl1, 0x2878)	bit3	ONM	M 通道数字信号输入使能	0, 屏蔽 M 通道的数字信号输入, 使得 M 通道的数字输入信号为 0; 1, 使能 M 通道数字信号输入

当 Pin55~56 用于外部直流信号测量输入时, 则, 寄存器 SegCtrl4 (0x2C23) 的 bit7~bit6 必须置 0, 关闭这两个引脚上的 SEG 输出。

与 M 通道信号测量数据输出相关的寄存器有 3 个, 分别是: M 通道原始数据值寄存器 (DATAOM, 0x10CE)、M 通道直流数据值寄存器 (DATADM, 0x10CF) 和 M 通道直流秒平均数据值寄存器 (DATAADM, 0x10D0)。

当电能计量时钟频率为 3.2768MHz 时 (ADC 时钟频率是 819.2kHz):

1. 寄存器 DATAOM 的值收敛时间是 60ms, 每秒钟刷新 3200 次;
2. 寄存器 DATADM 的值收敛时间是 150ms, 每秒钟刷新 20 次;
3. 寄存器 DATAADM 的值收敛时间是 3s, 每 1.28 秒刷新一次。

上述寄存器的收敛时间与电能计量时钟频率成正比。

9.14.1.1 测量内部地和温度

当 MEAS<2:0>=000 或 001 时, M 通道被用于测量内部地或外部温度信号, 此时, RESDIV 位自动置 0, 内部分压电阻网络关闭。

当 M 通道用于温度测量时，为了获得更好的温度测量结果，ADC 控制寄存器 5 (CtrlADC5, 0x2863) 的 bit7 (PTATEN) 应置 1，使用温度传感器 1。温度测量范围为-40°C~+85°C。

M 通道的测量结果可由寄存器 DATAOM (0x10CE)、DATADM (0x10CF) 和 DATAADM (0x10D0) 读出。

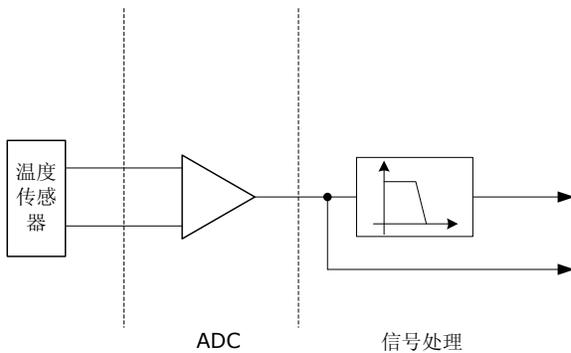


图 9-16 测温电路原理框图

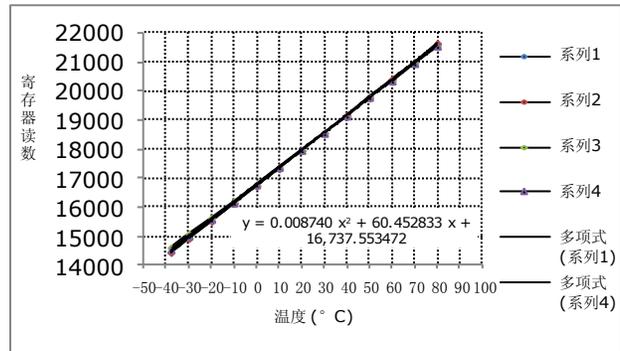


图 9-17 温度测量结果

将电能表放入温控箱，通过温控箱从-40°C 到+80°C 范围内调整，依次读取温度传感器读得的值。通过记录，我们可以看出寄存器读出的值与实际温度值成线性关系。

9.14.2 电池电压和外部信号测量

在 V9811 中，引脚 BAT/M1/M2 是等效的，均可被用于测量系统电池电压和外部电压信号。

如图 9-14 所示，当 M 通道被用于测量系统电池电压和外部电压信号时，电池电压或外部电压信号值为 V，进入 M 通道 ADC 的电压值为 Vm，两者之间的关系与 ADC 控制寄存器 5 (CtrlADC5, 0x2863) 中的 RESDIV 位 (bit4) 的值有关。

当 RESDIV=1 时，内部分压网络开启，则 Vm 与 V 之间的关系如下：

$$V_m = \frac{R_2}{R_1 + R_2} \times V$$

其中，R2 约为 7kΩ，R1 约为 24kΩ，V 值的取值范围为-0.2~4.0V。此时，分压电阻产生的电流 (I) 为：

$$I = \frac{V}{R_1 + R_2}$$

当 RESDIV=0 时，内部分压网络关闭，则 Vm 与 V 之间的关系如下：

$$V_m = V$$

V 值的范围为-200mV~+1.185V (基准电压)。

M 通道的测量结果可由寄存器 DATAOM (0x10CE)、DATADM (0x10CF) 和 DATAADM (0x10D0) 读出。

当 M 通道被用于测交流信号时，交流信号的测量结果包含在寄存器 DATAOM (0x10CE) 的读出值中，而在另两个寄存器的值中没有相应的信息。

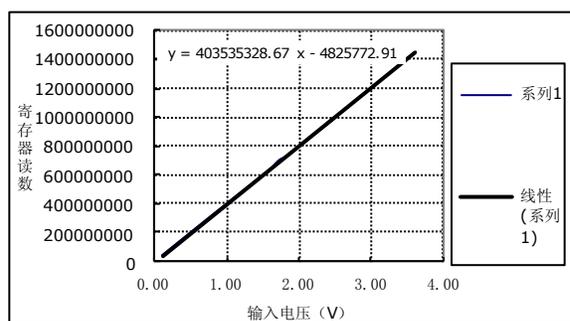


图 9-18 M 通道输入信号经分压后测量结果

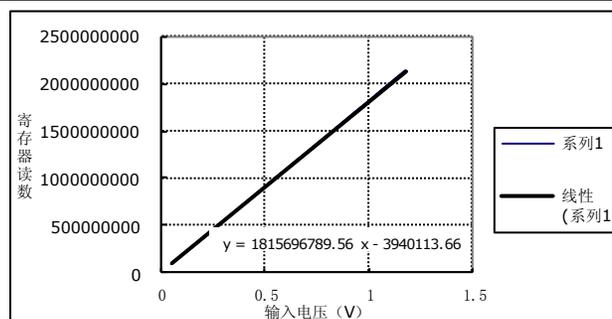


图 9-19 M 通道输入信号未经分压测量结果

图 9-18 和图 9-19 分别给出了开启与关闭内部分压网络的 M 通道信号测量结果。

9.15 计量模块初始化

为了保证计量模块能正常工作，应根据如下流程对计量模块进行初始化。

1. 访问计量控制寄存器，向寄存器 PMCtrl4 (0x287D) 的三个控制位 CRPEN (Bit5)、CFEN (Bit4) 和 EGYEN (Bit3) 写入 0，关闭能量累加、CF 输出以及防潜动电路；向 PMCtrl1 (0x2878) 的 PREN (Bit4) 和 ONx (Bit3~Bit0) 写入 0，关闭计量通道，并停止功率/有效值计算；
2. 访问模拟控制寄存器，配置并开启 ADC；
3. 清空读写缓冲寄存器与数据的对应关系 (0x2880~0x2885)，对计量数据寄存器进行写操作，并向其中写入 0；
4. 在 0x10E4~0x10E6 中写入 0x889374BC；
5. 设置计量数据寄存器（功率/有效值比差寄存器、功率二次补偿值寄存器、E1/E2 脉冲门限值寄存器、E1/E2 能量潜动门限值寄存器），其中，E1/E2 能量起动/潜动判断门限值设为 1；
6. 访问计量控制寄存器，进行如下配置：
 - 选择电流通道：配置 SELI 位 (bit5, PMCtrl1, 0x2878)；
 - 角差校正：配置 PHCEN 位 (PMCtrl1, 0x2878, Bit6) 使能角差校正，然后，配置角差校正控制寄存器 PHCCtrl1 (0x287B) 和 PHCCtrl2 (0x287C) 进行角差校正；
 - 数字增益设置：配置 PGANx 位和 PGANS 位 (Bit0~Bit3, PMCtrl3, 0x287A)；
 - 是否两路电流进行有功计量：配置 DBLEN 位 (Bit4, PMCtrl3, 0x287A)；
 - 是否使用低通滤波器和带通滤波器：配置 LPFEN 位和 BPFEN (Bit5 和 Bit6, PMCtrl3, 0x287A)；
 - 选择正相有功计量类型：配置 PSEL1 位和 PSEL0 位 (Bit1 和 Bit0, PMCtrl4, 0x287D)；
 - CF 输出类型：配置寄存器 CFCtrl (0x287E) 选择 CF 输出类型；
 - 使能起动/潜动判断：向寄存器 PMCtrl4 (0x287D) 的 bit6/bit7 位写入 1。
7. 访问计量控制寄存器，向 PMCtrl1 (0x2878) 的 PREN (Bit4) 和 ONx (Bit3~Bit0) 写入 1，开启计量通道，并使能功率/有效值计算；
8. 等待 100 ms (电能计量时钟频率为 3.2768MHz 或 819.2kHz 情况下为 500 ms)；
9. 访问计量控制寄存器，向寄存器 PMCtrl4 (0x287D) 的 EGYEN (Bit3) 写入 1，使能能量累加和防潜动操作；

10. 等待 2 秒，至系统稳定；
11. 将 E1/E2 能量潜动门限值寄存器设置为正常值；
12. 访问计量控制寄存器，向寄存器 PMCtrl4 (0x287D) 的控制位 CFEN (Bit4) 和 CFENR (bit5) 写入 1，使能 CF 输出。

9.16 校表

9.16.1 相关寄存器的计算公式

与校表相关的计量数据寄存器如表 9-33 所示。

表 9-33 校表相关计量数据寄存器

地址	寄存器	功能说明	长度 (bit)	数据格式
0x10FD	DATAFREQ	频率值寄存器	16	无符号正数
0x10F4	GATEP	E1 能量脉冲门限值寄存器	32	无符号正数
0x10F5	GATECP	E1 能量起动/潜动判断门限值寄存器	32	无符号正数
0x10FA	GATEQ	E2 能量脉冲门限值寄存器	32	无符号正数
0x10FB	GATECQ	E2 能量起动/潜动判断门限值寄存器	32	无符号正数
0x10FC	DATAACP	常数功率值寄存器	32	二进制补码
0x10E8	SCP	E1 功率比差寄存器	32	二进制补码
0x10E9	SCQ	E2 功率比差寄存器	32	二进制补码
0x10EA	SCU	电压有效值比差寄存器	32	二进制补码
0x10EB	SCI1	I1 电流有效值比差寄存器	32	二进制补码
0x10EC	SCI2	I2 电流有效值比差寄存器	32	二进制补码
0x10ED	PARAPC	E1 功率二次补偿寄存器	32	二进制补码
0x10EE	PARAQC	E2 功率二次补偿寄存器	32	二进制补码
0x10D6	DATAP	秒平均 E1 功率寄存器	32	二进制补码
0x10D7	DATAQ	秒平均 E2 功率寄存器	32	二进制补码
0x10D8	RMSU	秒平均电压有效值寄存器	32	二进制补码
0x10D9	RMSI1	I1 秒平均电流有效值寄存器	32	二进制补码
0x10DA	RMSI2	I2 秒平均电流有效值寄存器	32	二进制补码

1. 电压/电流有效值寄存器

包括：I1 和 I2 秒平均电流有效值寄存器 RMSI1 (0x10D9) 和 RMSI2 (0x10DA) 和秒平均电压有效值寄存器 RMSU (0x10D8)。

这几个寄存器的数值与输入信号间的关系如下：

$$\text{公式 9-1} \quad \text{Value} = V \times G \times K$$

其中：



V: 输入信号有效值;

G: 当前增益;

K: 常数, $K=1.8117 \times 10^9$ 。

例 1: 电压通道的采样信号为 18mV, 增益为 1, 则秒平均电压有效值寄存器的值应该为:

$$\text{Value} = 0.018 \times 1 \times 1.8117 \times 10^9 = 0x1F19927$$

例 2: 电流通道的采样信号为 2.35mV, 增益为 16, 则 I1 秒平均电流有效值寄存器的值应该为:

$$\text{Value} = 0.00235 \times 16 \times 1.8117 \times 10^9 = 0x40F6D70$$

2. 功率寄存器

包括: 秒平均 E1 功率寄存器 (DATAP, 地址: 0x10D6) 和秒平均 E2 功率寄存器 (DATAQ, 地址: 0x10D7)

功率寄存器的值可根据以下公式计算:

$$\text{公式 9-2} \quad \text{Value} = V_i \times G_i \times V_v \times G_v \times K \times \cos\theta$$

其中:

V_i 、 V_v : 分别为电流和电压通道输入信号大小;

G_i 、 G_v : 分别为电流和电压通道的增益;

K : 系数。当秒平均 E1/E2 功率寄存器用于存储有功功率的值时, $K=1.5413 \times 10^9$; 当秒平均 E1/E2 功率寄存器用于存储无功功率值时, $K=2.4167 \times 10^9$;

$\cos\theta$: 功率因数

例: 当 V_v 为 18mV, G_v 为 1, V_i 为 2.3mV, G_i 为 16, $\cos\theta$ 为 1, 则用于有功计量时, 秒平均 E1/E2 功率寄存器的值应为:

$$\begin{aligned} \text{Value} &= 0.0023 \times 0.018 \times 1 \times 16 \times 1.5413 \times 10^9 \times 1 \\ &= 0xF941D \end{aligned}$$

3. 频率值寄存器

假设频率值寄存器的值为 Freq, 则实际的信号频率为:

$$\text{公式 9-3} \quad f = \frac{f_{\text{ADC}}}{\text{Freq}}$$

其中, f_{ADC} (Hz) 为 ADC 时钟频率。

4. 能量脉冲门限值寄存器

$$\text{公式 9-4} \quad \text{PGAT} = \frac{P \times T \times 6400}{1024}$$

其中:

P: 秒平均 E1 功率寄存器的值, 由公式 9-2 计算得到;

T: 时间常数, 由公式 $T = \frac{3600 \times 1000}{\text{PulseConstant} \times U_n \times I_n}$ 计算得到。

5. 有效值/功率比差寄存器

$$\text{公式 9-5} \quad S = 2^{31} \left(\frac{1}{1+e} - 1 \right) + S_1 \left(\frac{1}{1+e} \right)$$

其中：

S：比差值，补码形式；

S_1 ：原始比差值；

e：误差。

6. 功率二次补偿寄存器

当电流为 1%I_b、100%I_b 和 1%I_b 的线性度偏差为 e（有符号的百分数，即台体显示的误差值）时，功率二次补偿寄存器的值应计算如下：

设置电流为 1%I_b 时的能量脉冲门限值 PGAT；

然后，根据下述公式计算得到功率二次补偿寄存器的值（C）：

$$\text{公式 9-6} \quad C = 1\% \times e \times P = \text{PGAT} \times 1024 \times \frac{U_n \times I_n \times \text{PulseConstant}}{1000 \times 3600 \times 6400} \times 1\% \times e$$

其中，

e：100%I_b 与 1%I_b 的线性偏差；

PGAT：E1/E2 能量脉冲门限值，用户可设置，由公式 9-4 计算得到。

9.16.2 有功能量校正

1. 粗校

方法一：将 E1 功率比差寄存器 SCP（0x10E8）的值设为 0，选择 I1 通道的信号进行 E1 能量计量，然后根据公式 9-4 计算门限值，并将其写入 E1 能量脉冲门限值寄存器 GATEP（0x10F4）。

注：同一批表采用同一门限值。

方法二：预先将一个门限值 PGAT₁ 写入 E1 能量脉冲门限值寄存器 GATEP（0x10F4），读出误差值 E（带符号），然后根据公式 9-7 计算真正的门限值 PGAT。

$$\text{公式 9-7} \quad \text{PGAT} = \text{PGAT}_1 \times (1 + E)$$

注：门限值为 PGAT₁ 时，误差 E 应是有效数据。

2. 细校

1) 选择通道 I1 进行计量：读取 E1 能量误差，按公式 9-5 计算 E1 功率比差值，并将其写入 E1 功率比差寄存器 SCP（0x10E8）；

2) 采用同样的步骤对通道 I2 进行校表。

注：在完成 E1 能量输出细校的同时也完成对功率值的校正。

9.16.3 电流有效值校正

1. 在通道 I1 的电流有效值比差寄存器（SCI1，地址：0x10EB）中写入 0；

校表台体通 100%I_n 电流；

通过公式 9-1 计算通道 I1 的电流有效值 I₀；

将 I₀ 标定到 I_n，根据公式 I₀ × D = I_n 得到 D（D 为电流有效值比例因数，为固定系数）；

从电表 LCD 读取电流有效值 I₁（I₁ 为电流有效值比差寄存器的实际读数与 D 的乘积）；

2. 根据公式 9-8 和公式 9-5 计算通道 I1 的比差值：



先计算误差：

$$\text{公式 9-8} \quad e = \frac{I_1 - I_n}{I_n}, \text{ 电流单位为 mA, 或者保持一致}$$

再计算比差：根据公式 9-5 计算比差。

3. 通道 I2 的电流有效值比差校正同上。

注：当通入电表的电流小于启动电流时，电表 LCD 上不显示电流有效值 I_1 。

9.16.4 电压有效值校正

1. 在电压有效值比差寄存器（SCU，地址：0x10EA）中写入 0；
2. 校表台体通 100% U_n 电压；
3. 通过公式 9-1 计算电压通道的电压有效值 U_0 ；
4. 将 U_0 标定到 U_n ，根据公式 $U_0 \times D = U_n$ 得到 D （ D 为电压有效值比例因数，为固定系数）；
5. 从电表 LCD 读取电压有效值 U_1 （ U_1 为电压有效值比差寄存器的实际读数与 D 的乘积）；

根据公式 9-9 和公式 9-5 计算电压通道的比差值：

先计算误差：

$$\text{公式 9-9} \quad e = \frac{U_1 - U_n}{U_n}, \text{ 电压单位为 mV, 或者保持一致}$$

再计算比差：根据公式 9-5 计算比差。

9.16.5 角差校正

角差校正就是为了保证，在低功率因数下，计量也能保持一定的精度。角差校正一般在 0.5L 的时候进行。校正公式如下：

$$\text{公式 9-10} \quad N = \frac{3011}{2} \times \text{Error} \times \frac{f_{\text{ADC}}}{819200}$$

其中， f_{ADC} (Hz) 为 ADC 时钟频率。

当角差校正控制寄存器 1（PHCCtrl1, 0x287B）和角差校正控制寄存器 2（PHCCtrl2, 0x287C）的最高位为 1 时，角差校正值为负值。

9.16.6 潜动

当电能计量时钟频率为 3.2768MHz、1.6384MHz 或 819.2KHz 时，能量桶每秒钟累加 12800 次；当电能计量时钟频率为 32768Hz 时，能量桶每秒钟累加 2979 次。

当启动/潜动判断使能后，该能量桶的输入固定为 1，而 E1/E2 能量桶输入为 E1/E2 功率或常数功率值。当电能计量时钟频率为 3.2768MHz、1.6384MHz 或 819.2kHz 时，E1/E2 能量桶和潜动能量桶每秒累加 12800 次；当电能计量时钟频率为 32768Hz 时，E1/E2 能量桶和潜动能量桶每秒累加 2979 次。当防潜动能量桶的累加值先达到启动/潜动判断门限值时，E1/E2 能量桶被清空，E1/E2 计量通道进入潜动状态。当 E1/E2 能量桶的累加值先达到 E1/E2 能量脉冲门限值时，防潜动能量桶被清空，E1/E2 计量通道进入电能计量状态。

一般，我们将潜动条件下的理论出脉冲时间设为启动/潜动判断门限值。

例：一规格为 10(60)、常数 1200 的表，启动电流为 0.4% I_n ，则启动时间为 341 秒。假设当电流小于 0.2% I_n 时系统应进入潜动状态，那么理论上，0.2% I_n 的启动时间为 682 秒。当电能计量时钟频率为 3.2768MHz 时，

起动/潜动判断门限值为 $682 \times \frac{12800}{2} = 0x429A00$ ；当电能计量时钟频率为 32768Hz 时，起动/潜动判断门限值为 $682 \times \frac{2979}{2} = 0xF801F$ 。

9.16.7 校表举例

本节以一款双回路防窃电表为例，对整个校表流程进行说明。

9.16.7.1 表型参数

额定电流：10A

额定电压：220V

脉冲常数：1600

精度等级：1.0

9.16.7.2 设计参数

额定电流时，IA 通道电流采样信号大小：2.5mV

额定电流时，IB 通道电流采样信号大小：5mV

额定电压时，电压通道采样信号大小：18.5mV

9.16.7.3 ADC 增益设置

IA 通道电流增益：16

IB 通道电流增益：8

电压通道增益：1

9.16.7.4 计算

在此例中，假设 DBLEN=1，且，SELI=0，即输入通道 I1 和 I2 的电流信号均进行有功能量计量。下面以通道 I1 为例对校表计算进行说明。

1. E1 能量脉冲门限值计算

1.1 计算功率

根据公式 9-2:

$$\begin{aligned} \text{Value} &= V_i \times G_i \times V_v \times G_v \times 1.5413 \times 10^9 \times \cos \theta \\ &= 0.0025 \times 16 \times 0.0185 \times 1541300000 \\ &= 1140562 \\ &= 0x116752 \end{aligned}$$

1.2 计算能量门限值

根据公式 9-4:

$$\begin{aligned} \text{Value} &= \frac{P \times T \times 6400}{1024} \\ &= \frac{1140562 \times 3600 \times 1000 \times 6400}{1600 \times 10 \times 220 \times 1024} \\ &= 0x6F3E9C \end{aligned}$$

0x6F3E9C 即为能量门限值，并据此将 E1 能量脉冲门限值寄存器（GATEP，0x10F4）设为 0x6F3E9C。

2. 电流有效值门限值计算

2.1 根据公式 9-1 计算电流有效值寄存器 RMSI1（地址：0x10D9）的值：

$$\begin{aligned} \text{Value} &= V \times G \times K \\ &= 0.0025 \times 16 \times 1.8117 \times 10^9 \\ &= 72468000 = 0x451C620 \end{aligned}$$

2.2 计算电流有效值门限值

根据下列公式计算电流有效值门限值：

$$\begin{aligned} \text{IRMSGAT} &= \frac{I \times T \times 6400}{1024} \\ &= \frac{72468000 \times 3600 \times 1000 \times 6400}{1600 \times 10 \times 220 \times 1024} \\ &= 0x1B9C283E \end{aligned}$$

其中，I 为通过公式 9-1 计算得到的电流有效值寄存器的值。

0x1B9C283E 即为门限值，并据此将 E1 能量脉冲门限值寄存器（GATEP，0x10F4）设为 0x1B9C283E。

注：电流有效值门限值和能量脉冲门限值均在同一个寄存器中设置。如果不用电流有效值进行计量防窃电，该值可以不用设置。

3. 有效值比例因数计算

我们可以根据公式 9-1 计算电流、电压有效值寄存器中的数据，但是这些数据只是 ADC 采样数据，如果要将其转化为直观的有效值数据，还需要程序对获取的数据作相关转换。

3.1 I1 电流有效值比例因数

根据公式 9-1 I1 电流有效值寄存器中数据

$$\begin{aligned} \text{Value} &= V \times G \times K \\ &= 0.0025 \times 16 \times 1.8117 \times 10^9 \\ &= 72468000 = 0x451C620 \end{aligned}$$

此时实际电流值应该是 10.00A，因此我们要将该值做如下处理：

$$\begin{aligned} I_{\text{value}} &= \text{Value} \times D \\ 1000 &= 72468000 \times D \\ D &= 1.38 \times 10^{-5} \end{aligned}$$

其中，D 就是通道 I1 电流值比例因数。用户只要把 I1 秒平均电流有效值寄存器的实际读数乘以该比例因数，就能正确显示当前电流有效值。这里默认电流有效值为两位小数。

3.2 通道 I2 电流有效值比例因数

通道 I2 处理方式与通道 I1 相同。

3.3 电压通道有效值比例因数



电压通道处理方式和电流通道处理方式相同。

以上参数在电能表设计时就已经确定，校表不应该改变以上参数。

9.16.7.5 校表流程

1. E1 能量比差校正

1.1 校正条件:

在校表台输入额定电流和额定电压，选择通道 I1 进行计量，功率因数为 1.0。

1.2 计算公式：公式 9-5

1.3 相关寄存器:

E1 功率比差寄存器（地址：0x10E8）

如果此时电能表误差为 4.5%，同时原始比差值为 0x11345682。

$$\begin{aligned} S &= 2^{31} \left(\frac{1}{1+e} - 1 \right) + S_1 \left(\frac{1}{1+e} \right) \\ &= 2147483648 \times \left(\frac{1}{1+0.045} - 1 \right) + 288642690 \times \left(\frac{1}{1+0.045} \right) \\ &= 0xFFFA7CF014 + 0x1076AD6C \\ &= 0xAF39D80 \end{aligned}$$

2. 通道 IA 角差校正

2.1 校正条件:

对校表台输入额定电流和额定电压，选择通道 I1 进行计量，功率因数为 0.5L，角差校正使能。

2.2 计算公式：公式 9-10

2.3 相关寄存器：角差校正控制寄存器 1（PHCCtrl1，0x287B）

该寄存器必须先清零。如果此时误差为 0.36%，计算数据 $S = \frac{3011}{2} \times 0.0036 \times \frac{819200}{819200} = 5$ ，此时，角差为负值，为消除角差，角差校正值应为正值，所以寄存器最高位应写入 0。最终写入寄存器的数据为 0x5。

3. 通道 I1 电流有效值校正

3.1 校正条件:

对校表台输入额定电流和额定电压。

3.2 计算公式：公式 9-5 和公式 9-8

3.3 相关寄存器：I1 电流有效值比差寄存器（SCI1，地址：0x10EB）

如果此时 I_n 为 10.00A，电表 LCD 显示有效值 (I_1) 为 13.43A，原始比差值为 0x21124562，则

$$e = \frac{I_1 - I_n}{I_n} = \frac{1343 - 1000}{1000} = 34.3\%$$

$$\begin{aligned} S &= 2^{31} \left(\frac{1}{1+e} - 1 \right) + S_1 \left(\frac{1}{1+e} \right) \\ &= 2147483648 \times \left(\frac{1}{1+0.343} - 1 \right) + 554845538 \times \left(\frac{1}{1+0.343} \right) \\ &= 0xFFDF4F1B43 + 0x189FFFD0 \\ &= 0xFFF7EF1B13 \end{aligned}$$

4. 电压有效值校正

4.1 校正条件:

对校表台输入额定电流和额定电压。

4.2 计算公式: 公式 9-5 和公式 9-9

4.3 相关寄存器: 电压有效值比差寄存器 (SCU, 地址: 0x10EA)

如果此时 U_n 为 220.1V, 电表 LCD 显示电压有效值 (U_0) 为 230.04V, 原始比差值为 0x10035251。

$$e = \frac{U_0 - U_1}{U_1} = \frac{23004 - 22010}{22010} = 4.5\%$$

$$\begin{aligned} S &= 2^{31} \left(\frac{1}{1+e} - 1 \right) + S_1 \left(\frac{1}{1+e} \right) \\ &= 2147483648 \times \left(\frac{1}{1+0.045} - 1 \right) + 268653137 \times \left(\frac{1}{1+0.045} \right) \\ &= 0xFFFA7CF014 + 0xF52CBB5 \\ &= 0x9CFBBC9 \end{aligned}$$

5. I2 电流与 E2 能量校正

如果有两个通道, 则两个通道的能量比差、角差和电流有效值均需进行校正, E2 与 E1 计量通道的校正方式相同。

第10章 中断控制系统

当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、恢复供电复位、IO/RTC 休眠唤醒复位或调试复位时，中断控制电路寄存器被复位。系统进入浅睡眠或者深睡眠后，中断控制电路寄存器停止工作，进入低功耗状态。

系统提供 4 个外部端口中断输入。使用外部端口中断的时候，需设置 GPIO 寄存器确定 IO 引脚被用于外部中断功能。

10.1 中断控制 SFR

在 V9811 中，与中断控制相关的 SFR 包括：IE SFR（中断允许寄存器）、IP SFR（中断优先级寄存器）、EXIF SFR（扩展中断触发方式选择寄存器）、EICON SFR（扩展特殊寄存器）、EIE SFR（扩展中断允许寄存器）以及 EIP SFR（扩展中断优先级寄存器），它们分别为中断控制电路提供了中断使能、优先级控制和标志位等信息。对上述 SFR 的解释如下。

10.1.1 IE SFR (0xA8)

表 10-1 IE SFR (0xA8)

位	功能
IE.7	EA - 全局中断使能位。EA = 0，禁止所有中断。EA = 1，每一个中断的使能或者屏蔽由这些中断各自的使能位决定。
IE.6	ES1 - UART1 中断使能位。ES1 = 0，禁止 UART1 中断（TI1 和 RI1）。ES1 = 1，使能由 TI1 和 RI1 标志位产生的中断。
IE.5	ET2 - Timer2 中断使能位。ET2 = 0，禁止 Timer2 中断（TF2 或 EXF2）。ET2 = 1，使能由 TF2 或者 EXF2 标志位产生的中断。
IE.4	ES0 - 保留
IE.3	ET1 - Timer1 中断使能位。ET1 = 0，禁止 Timer1 中断（TF1）。ET1 = 1，使能由 TF1 标志位产生的中断。
IE.2	EX1 - 端口中断 1 使能位。EX1 = 0，禁止端口中断 1。EX1 = 1，使能由端口中断 1 产生的中断。
IE.1	ET0 - Timer0 中断使能位。ET0 = 0，禁止 Timer0 中断（TF0）。ET0 = 1，使能由 TF0 标志位产生的中断。
IE.0	EX0 - 端口中断 0 使能位。EX0 = 0，禁止端口中断 0。EX0 = 1，使能由端口中断 0 产生的中断。

10.1.2 IP SFR (0xB8)

标准 8051 提供两个级别的中断优先级：中断级别 1 和中断级别 0。所有中断的优先级可通过 IP SFR 和 EIP SFR 进行设置。中断级别 1 高于中断级别 0。所有的中断都可以被赋予高优先级或者低优先级。除了被赋予的优先级（高或者低）之外，每一个中断都有硬件查询顺序（参看“中断资源列表”，硬件查询顺序号越小，表明该中断的查询优先级越高）。有着相同中断级别（如中断级别 1 或中断级别 0）的中断同时发生的时候，根据它们相应的硬件查询顺序来决定先响应哪个中断。一个中断服务程序只能被另外一个拥有更高优先级的中断所打断。也就是说，中断级别 0 的中断服务程序的执行只能被中断级别 1 的中断打断。一旦某个中断得到了响应，那么它的中断服务程序就只能更高优先级的中断所打断。

表 10-2 IP SFR (0xB8)

位	功能
IP.7	保留。读出值为 1
IP.6	PS1 - UART1 中断 (RI1 或 TI1) 优先级控制位。PS1 = 0, 中断级别 0。PS1 = 1, 中断级别 1。
IP.5	PT2 - Timer2 中断 (TF2 或 EXF2) 优先级控制位。PT2 = 0, 中断级别 0。PT2 = 1, 中断级别 1。
IP.4	PS0 - 保留。
IP.3	PT1 - Timer1 中断 (TF1) 优先级控制位。PT1 = 0, 中断级别 0。PT1 = 1, 中断级别 1。
IP.2	PX1 - 端口中断 1 优先级控制位。PX1 = 0, 中断级别 0。PX1 = 1, 中断级别 1。
IP.1	PT0 - Timer0 中断 (TF0) 优先级控制位。PT0 = 0, 中断级别 0。PT0 = 1, 中断级别 1。
IP.0	PX0 - 端口中断 0 优先级控制位。PX0 = 0, 中断级别 0。PX0 = 1, 中断级别 1。

10.1.3 EIE SFR (0xE8)

表 10-3 EIE SFR (0xE8)

位	功能
EIE.7 ~ 5	保留。读出值为 1。
EIE.4	掉电中断 (中断向量 12) 使能位。0, 禁止; 1, 使能。
EIE.3	中断向量 11 使能位。0, 禁止; 1, 使能。
EIE.2	中断向量 10 使能位。0, 禁止; 1, 使能。
EIE.1	中断向量 9 使能位。0, 禁止; 1, 使能。
EIE.0	中断向量 8 使能位。0, 禁止; 1, 使能。

10.1.4 EIP SFR (0xF8)

表 10-4 EIP SFR (0xF8)

位	功能
EIP.7 ~ 5	保留。读出值为 1
EIP.4	中断向量 12 优先级控制位。0, 中断级别 0; 1, 中断级别 1。
EIP.3	中断向量 11 优先级控制位。0, 中断级别 0; 1, 中断级别 1。
EIP.2	中断向量 10 优先级控制位。0, 中断级别 0; 1, 中断级别 1。
EIP.1	中断向量 9 优先级控制位。0, 中断级别 0; 1, 中断级别 1。
EIP.0	中断向量 8 优先级控制位。0, 中断级别 0; 1, 中断级别 1。

10.1.5 EXIF SFR (0X91)

表 10-5 EXIF SFR (0x91)

位	功能
EXIF.7	IE5 - 中断向量 11 标志位。IE5 = 1, 表示中断向量 11 产生了中断。IE5 必须由软件进行清 0。在中断向量 11 被使能的情况下, 使用软件设置 IE5=1 也会产生中断。
EXIF.6	IE4 - 中断向量 10 标志位。IE4 = 1, 表示中断向量 10 产生了中断。IE4 必须由软件进行清 0。在中断向量 10 被使能的情况下, 使用软件设置 IE4 = 1 也会产生中断。
EXIF.5	IE3 - 中断向量 9 标志位。IE3 = 1, 表示中断向量 9 产生了中断。IE3 必须由软件进行清 0。在中断向量 9 被使能的情况下, 使用软件设置 IE3 = 1 也会产生中断。
EXIF.4	IE2 - 中断向量 8 标志位。IE2 = 1, 表示中断向量 8 发生了中断。IE2 必须由软件进行清 0。在中断向量 8 被使能的情况下, 使用软件设置 IE2 = 1 也会产生中断。
EXIF.3	保留。读出值为 1。
EXIF.2 ~ 0	保留。读出值为 0。

10.1.6 EICON SFR (0XD8)

表 10-6 EICON SFR (0xD8)

位	功能
EICON.7	SMOD1 - UART1 波特率加倍使能位。当 SMOD1 = 1 的时候, UART1 的波特率加倍。
EICON.6	保留。读出值为 1。
EICON.5	保留。
EICON.4	保留。
EICON.3	PFI - 中断向量 12 标志位, 掉电中断。1, 表示发生掉电中断。在退出中断服务程序之前, 必须由软件将此位清 0。否则, 中断会再次发生。在掉电中断被使能的情况, 用软件设置此位也会产生掉电中断。
EICON.2 ~ 0	保留。读出值为 0。

10.2 中断处理

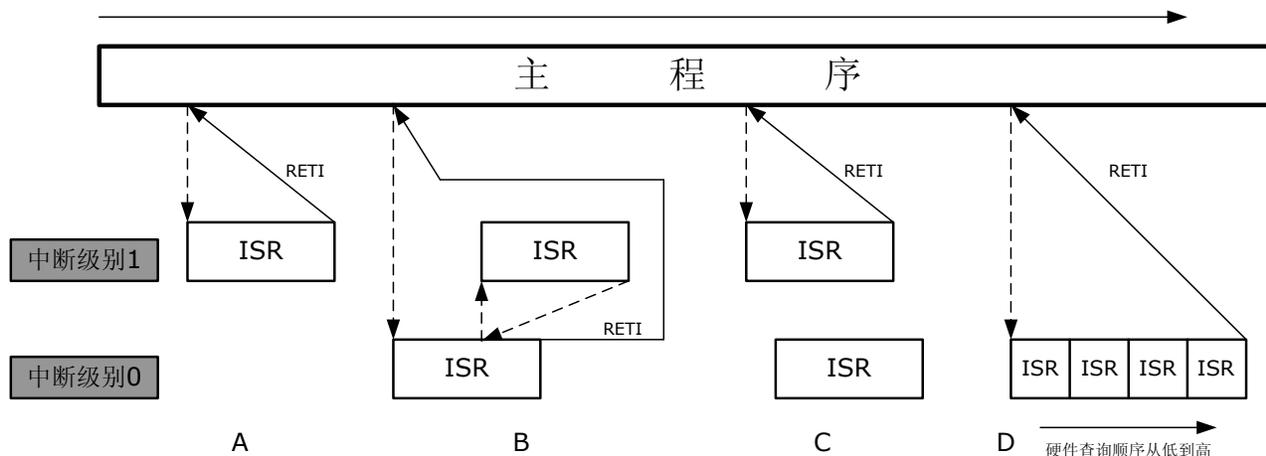


图 10-1 中断处理

图 10-1 描述了 V9811 的中断处理过程。在处理中断时，

- 当一个被使能的中断发生时，CPU 就会跳转到与这个中断相对应的中断向量地址，执行中断服务程序（Interrupt Service Routine, ISR）。只要不发生更高优先级的中断，CPU 会执行中断服务程序直到执行结束为止。每一个中断服务程序都以一个 RETI 指令（从中断返回）结束，如图中 A 所示。
- 在执行了 RETI 指令后，CPU 会返回到中断发生前将要被执行的那一条指令继续执行。在执行中断服务程序以前，CPU 都会执行完当前正在执行的指令。如果正在执行的指令是 RETI，或者是对寄存器 IP SFR、IE SFR、EIP SFR 和 EIE SFR 进行写操作的指令，那么 CPU 在执行完当前正在执行的指令后，会再多执行一条指令，然后才执行中断服务程序。
- 中断级别 1 的优先级高于中断级别 0，所以，中断级别 0 的 ISR 只能被中断级别 1 的 ISR 打断，如图中 B 和 C 所示。当中断级别 0 的 ISR 的执行被中断级别 1 的 ISR 打断后，待执行完中断级别 1 的 ISR 后，CPU 会返回被打断的中断级别 0 的 ISR 继续执行，至 RETI 指令结束，如图中 B 所示。
- 有着相同级别的中断同时发生时，根据它们相应的硬件查询顺序来决定先响应哪个中断（硬件查询顺序号越小，表明该中断的查询优先级越高），如图中 D 所示。
- 中断的响应时间由 CPU 的当前状态决定：
 - 最快的响应时间是五个指令周期：一个指令周期用来检测中断；四个用来执行 LCALL 指令，调用 ISR 中断服务程序；
 - 最大延时（13 个指令周期）发生在 CPU 正在执行 RETI 指令并且后面又紧接着 MUL 或者 DIV 指令的时候，这 13 个指令周期分别是：一个指令周期用来检测中断；三个指令周期用来完成 RETI 指令；五个指令周期用来执行 MUL 或者 DIV 指令；四个指令周期用来执行调用 ISR 的 LCALL 指令。在这种情况下，最大的响应时间是 $13 \times 4 = 52$ 个时钟周期。
- 为了保证能够检测到边沿触发方式的中断，如端口中断 0/1/2/3，相应的中断信号输入端口应该保持 4 个时钟周期的高电平，然后，保持 4 个时钟周期的低电平。如果某一电平触发方式的中断发生时因标志位未置位或一个更高优先级的中断正在处理而无法进入中断服务程序，则，该中断输入信号一直保持有效，直至进入中断服务程序。

10.3 中断扩展

10.3.1 中断向量 8

中断向量 8 被扩展为 7 个中断源，有 5 个寄存器与此相关。

表 10-7 中断向量 8 中断源

中断向量 8			中断源	使能位	中断源标记 (R/W)
向量地址	使能	标记 (R/W)			
43h	IE.7=1 EIE.0=1	IE2 (EXIF.4)	UART2 发送中断	ExInt2IE.0	ExInt2IFG.0
			UART2 接收中断	ExInt2IE.1	ExInt2IFG.1
			UART4 发送中断	ExInt2IE.2	ExInt2IFG.2
			UART4 接收中断	ExInt2IE.3	ExInt2IFG.3
			UART2 普通定时器溢出中断	ExInt2IE.4	ExInt2IFG.4
			UART4 普通定时器溢出中断	ExInt2IE.5	ExInt2IFG.5
			CF1 中断	ExInt2IE.7	ExInt2IFG.7

表 10-8 扩展中断使能寄存器 (ExInt2IE, 0x2843)

0x2843, R/W, 扩展中断使能寄存器, ExInt2IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IE7	-	IE5	IE4	IE3	IE2	IE1	IE0
默认值	0	0	0	0	0	0	0	0

1, 使能; 0, 屏蔽

表 10-9 中断标志 (请求) 寄存器 (ExInt2IFG, 0x2840)

0x2840, R/W, 中断标志 (请求) 寄存器, ExInt2IFG								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IR7	-	IR5	IR4	IR3	IR2	IR1	IR0
默认值	0	0	0	0	0	0	0	0

1, 有中断请求; 0, 无中断请求

扩展中断在被使能后才能在扩展中断标志寄存器中置位。软件在任何时刻都可以访问 ExInt2IFG, 但, 在中断未使能的情况下, 软件写入的扩展中断标志不能提起中断, 只有在中断被使能时, 在这个寄存器中写入 1, 触发软件中断。

写入 0 清除中断标志

表 10-10 扩展中断队列寄存器 (ExInt2OV, 0x2844)

0x2844, R/W, 扩展中断队列寄存器, ExInt2OV								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IPND7	-	IPND5	IPND4	IPND3	IPND2	IPND1	IPND0
默认值	0	0	0	0	0	0	0	0

在 ExInt2IFG 中的标志 (请求) 还没有被软件清除的时候, 如果又发生了同一个中断, 则 ExInt2OV 中相应的位会置 1。这个位只是一个标记, 不会对中断造成任何影响, 需要软件清除。

表 10-11 扩展中断输入类型寄存器 (ExInt2IN, 0x2841)

0x2841, R/W, 扩展中断输入类型寄存器, ExInt2IN								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EDG7I	-	EDG5I	EDG4I	EDG3I	EDG2I	EDG1I	EDG0I
默认值	1	1	1	1	1	1	1	1

1, 边沿触发; 0, 电平触发

用户可任意配置为检测到中断源输入信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时捕捉中断源信号。

表 10-12 扩展中断输出类型寄存器 (ExInt2OUT, 0x2842)

0x2842, R/W, 扩展中断输出类型寄存器, ExInt2OUT								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	-	-	-	EDGO
默认值	-	-	-	-	-	-	-	1

1, 边沿输出; 0, 电平输出

用户可任意配置为检测到中断输出信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时输出中断, 并向 CPU 提起中断, CPU 转入向量 43h 执行中断向量 8 中断服务程序 (Interrupt Service Routine, ISR)。

10.3.2 中断向量 9

中断向量 9 被扩展为 5 个中断源, 有 5 个寄存器与此相关。

表 10-13 中断向量 9 的中断源

中断向量 9			中断源	使能位	中断源标记 (R/W)
向量地址	使能	标记 (R/W)			
4Bh	IE.7=1 EIE.1=1	IE3 (EXIF.5)	UART5 发送中断	ExInt3IE.2	ExInt3IFG.2
			UART5 接收中断	ExInt3IE.3	ExInt3IFG.3
			UART5 普通定时器溢出中断	ExInt3IE.5	ExInt3IFG.5
			RTC 秒中断	ExInt3IE.6	ExInt3IFG.6
			CF2 中断	ExInt2IE.7	ExInt2IFG.7

表 10-14 扩展中断使能寄存器 (ExInt3IE, 0x284B)

0x284B, R/W, 扩展中断使能寄存器, ExInt3IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IE7	IE6	IE5	-	IE3	IE2	-	-
默认值	0	0	0	0	0	0	0	0

1, 使能; 0, 屏蔽

表 10-15 中断标志 (请求) 寄存器 (ExInt3IFG, 0x2848)

0x2848, R/W, 中断标志 (请求) 寄存器, ExInt3IFG								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IR7	IR6	IR5	-	IR3	IR2	-	-
默认值	0	0	0	0	0	0	0	0

1, 有中断请求; 0, 无中断请求

扩展中断在被使能后才能在扩展中断标志寄存器中置位。软件在任何时刻都可以访问 ExInt3IFG, 但在中断未使能的情况下, 软件写入的扩展中断标志不能提起中断, 只有在中断被使能时, 在这个寄存器中写入 1, 触发软件中断。

写入 0 清除中断标志。

表 10-16 扩展中断队列寄存器 (ExInt3OV, 0x284C)

0x284C, R/W, 扩展中断队列寄存器, ExInt3OV								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IPND7	IPND6	IPND5	-	IPND3	IPND2	-	-
默认值	0	0	0	0	0	0	0	0

在 ExInt3IFG 中的标志 (请求) 还没有被软件清除的时候, 如果又发生了同一个中断, 则 ExInt3IFG 中相应的位会置 1。这个位只是一个标记, 不会对中断造成任何影响, 需要软件清除。

表 10-17 扩展中断输入类型寄存器 (ExInt3IN, 0x2849)

0x2849, R/W, 扩展中断输入类型寄存器, ExInt3IN								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EDG7I	EDG6I	EDG5I	-	EDG3I	EDG2I	-	-
默认值	1	1	1	1	1	1	1	1

1, 边沿触发; 0, 电平触发

用户可任意配置为检测到中断源输入信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时捕捉中断源信号。

表 10-18 扩展中断输出类型寄存器 (ExInt3OUT, 0x284A)

0x284A, R/W, 扩展中断输出类型寄存器, ExInt3OUT								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	-	-	-	EDGO
默认值	-	-	-	-	-	-	-	1

1, 边沿输出; 0, 电平输出

用户可任意配置为检测到中断输出信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时输出中断向量 9, 并向 CPU 提起中断, CPU 转入向量 4Bh 执行中断向量 9 中断服务程序 (Interrupt Service Routine, ISR)。

10.3.3 中断向量 10

中断向量 10 被扩展为 4 个中断源, 有 5 个寄存器与此相关。

表 10-19 中断向量 10 中断源

中断向量 10			中断源	使能位	中断源标记 (R/W)
向量地址	使能	标记 (R/W)			
53h	IE.7=1 EIE.2=1	IE4 (EXIF.6)	RTC 非法数据中断	ExInt4IE.0	ExInt4IFG.0
			端口中断 2 (INT2) 下降沿触发	ExInt4IE.2	ExInt4IFG.2
			端口中断 3 (INT3) 下降沿触发	ExInt4IE.3	ExInt4IFG.3
			过零点中断	ExInt4IE.5	ExInt4IFG.5

表 10-20 扩展中断使能寄存器 (ExInt4IE, 0x2853)

0x2853, R/W, 扩展中断使能寄存器, ExInt4IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	IE5	-	IE3	IE2	-	IE0
默认值	0	0	0	0	0	0	0	0

1, 使能; 0, 屏蔽

表 10-21 中断标志 (请求) 寄存器 (ExInt4IFG, 0x2850)

0x2850, R/W, 中断标志 (请求) 寄存器, ExInt4IFG								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-		IR5	-	IR3	IR2		IR0
默认值	0	0	0	0	0	0	0	0

1, 有中断请求; 0, 无中断请求

扩展中断在被使能后才能在扩展中断标志寄存器中置位。软件在任何时刻都可以访问 ExInt4IFG, 但在中断未使能的情况下, 软件写入的扩展中断标志不能提起中断, 只有在中断被使能时, 在这个寄存器中写入 1, 触发软件中断。

写入 0 清除中断标志。

表 10-22 扩展中断队列寄存器 (ExInt4OV, 0x2854)

0x2854, R/W, 扩展中断队列寄存器, ExInt4OV								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-		IPND5	-	IPND3	IPND2		IPND0
默认值	0	0	0	0	0	0	0	0

在 ExInt4IFG 中的标志 (请求) 还没有被软件清除的时候, 如果又发生了同一个中断, 则 ExInt4OV 中相应的位会置 1。这个位只是一个标记, 不会对中断造成任何影响, 需要软件清除。

表 10-23 扩展中断输入类型寄存器 (ExInt4IN, 0x2851)

0x2851, R/W, 扩展中断输入类型寄存器, ExInt4IN								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-		EDG5I	-	EDG3I	EDG2I		EDG0I
默认值	1	1	1	1	1	1	1	1

1, 边沿触发; 0, 电平触发

用户可任意配置为检测到中断源输入信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时捕捉中断源信号。

表 10-24 扩展中断输出类型寄存器 (ExInt4OUT, 0x2852)

0x2852, R/W, 扩展中断输出类型寄存器, ExInt4OUT								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	-	-	-	EDGO
默认值	-	-	-	-	-	-	-	1

1, 边沿输出; 0, 电平输出

用户可任意配置为检测到中断输出信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时输出中断, 并向 CPU 提起中断, CPU 转入向量 53h 执行中断向量 10 中断服务程序 (Interrupt Service Routine, ISR)。

10.3.4 中断向量 11

中断向量 11 被扩展为 4 个中断源, 有 5 个寄存器与此相关。

表 10-25 中断向量 11 中断源

中断向量 11			中断源	使能位	中断源标记 (R/W)
向量地址	使能	标记 (R/W)			
5Bh	IE.7=1 EIE.3=1	IE5 (EXIF.7)	TimerA 溢出中断	ExInt5IE.0	ExInt5IFG.0
			TimerA 捕获中断 0	ExInt5IE.1	ExInt5IFG.1
			TimerA 捕获中断 1	ExInt5IE.2	ExInt5IFG.2
			TimerA 捕获中断 2	ExInt5IE.3	ExInt5IFG.3

表 10-26 扩展中断使能寄存器 (ExInt5IE, 0x28A5)

0x28A5, R/W, 扩展中断使能寄存器, ExInt4IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	IE3	IE2	IE1	IE0
默认值	0	0	0	0	0	0	0	0

1, 使能; 0, 屏蔽

表 10-27 中断标志 (请求) 寄存器 (ExInt5IFG, 0x28A2)

0x28A2, R/W, 中断标志 (请求) 寄存器, ExInt4IFG								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	IR3	IR2	IR1	IR0
默认值	0	0	0	0	0	0	0	0

1, 有中断请求; 0, 无中断请求

扩展中断在被使能后才能在扩展中断标志寄存器中置位。软件在任何时刻都可以访问 ExInt5IFG, 但在中断未使能的情况下, 软件写入的扩展中断标志不能提起中断, 只有在中断被使能时, 在这个寄存器中写入 1, 触发软件中断。写入 0 清除中断标志。

表 10-28 扩展中断队列寄存器 (ExInt5OV, 0x28A6)

0x28A6, R/W, 扩展中断队列寄存器, ExInt5OV								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	IPND3	IPND2	IPND1	IPND0
默认值	0	0	0	0	0	0	0	0

在 ExInt5IFG 中的标志 (请求) 还没有被软件清除的时候, 如果又发生了同一个中断, 则 ExInt5OV 中相应的位会置 1。这个位只是一个标记, 不会对中断造成任何影响, 需要软件清除。

表 10-29 扩展中断输入类型寄存器 (ExInt5IN, 0x28A3)

0x28A3, R/W, 扩展中断输入类型寄存器, ExInt5IN								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	EDG3I	EDG2I	EDG1I	EDG0I
默认值	1	1	1	1	1	1	1	1

1, 边沿触发; 0, 电平触发

用户可任意配置为检测到中断源输入信号的边沿 (上升沿或下降沿) 或电平 (高电平或低电平) 时捕捉中断源信号。

表 10-30 扩展中断输出类型寄存器 (ExInt5OUT, 0x28A4)

0x28A4, R/W, 扩展中断输出类型寄存器, ExInt5OUT								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	-	-	-	EDGO
默认值	-	-	-	-	-	-	-	1

0x28A4, R/W, 扩展中断输出类型寄存器, ExInt5OUT

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
--	------	------	------	------	------	------	------	------

1, 边沿输出; 0, 电平输出

用户可任意配置为检测到中断输出信号的边沿（上升沿或下降沿）或电平（高电平或低电平）时输出中断，并向 CPU 提起中断，CPU 转入向量 5Bh 执行中断向量 11 中断服务程序（Interrupt Service Routine, ISR）。

10.4 中断资源列表

对于扩展中断——中断向量 8、9、10 和 11 而言，中断响应程序应该先清除扩展中断标志位，再清除中断向量 8、9、10 和 11 对应的内核中断标志位。

对于位于同一个中断向量中的扩展中断源，用户可以采用查询方式或者中断方式响应中断事件。在进入中断服务程序后，用户需要对中断标志位和中断使能位进行“与”操作，从而判断中断源。

表 10-31 中断资源列表

硬件查询顺序	向量	中断号	说明	内核 En	扩展 En	内核 Flag	扩展 Flag	
0 (最高)	33h	6	保留					
1	03h	0	端口中断 0 (INT0)，低电平/下降沿触发	IE.0		TCON.1		
2	0Bh	1	Timer 0 中断 (TF0)	IE.1		TCON.5		
3	13h	2	端口中断 1 (INT1)，低电平/下降沿触发	IE.2		TCON.3		
4	1Bh	3	Timer 1 中断 (TF1)	IE.3		TCON.7		
5	23h	4	保留					
6	2Bh	5	Timer 2 中断 (TF2, EXF2)	IE.5		T2CON.7		
7	3Bh	7	UART 1 接收中断 (RI1)	IE.6		SCON1.0		
			UART 1 发送中断 (TI1)			SCON1.1		
8	43h	8	UART2 发送中断	EIE.0		ExInt2IE.0	EXIF.4	ExInt2IFG.0
			UART2 接收中断			ExInt2IE.1		ExInt2IFG.1
			UART4 发送中断			ExInt2IE.2		ExInt2IFG.2
			UART4 接收中断			ExInt2IE.3		ExInt2IFG.3
			UART2 中普通定时器溢出中断			ExInt2IE.4		ExInt2IFG.4
			UART4 中普通定时器溢出中断			ExInt2IE.5		ExInt2IFG.5
			保留					
			CF1 中断			ExInt2IE.7		ExInt2IFG.7

硬件查询顺序	向量	中断号	说明	内核 En	扩展 En	内核 Flag	扩展 Flag
9	4Bh	9	保留	EIE.1		EXIF.5	
			保留				
			UART5 发送中断		ExInt3IE.2		ExInt3IFG.2
			UART5 接收中断		ExInt3IE.3		ExInt3IFG.3
			保留				
			UART5 中定时器溢出中断		ExInt3IE.5		ExInt3IFG.5
			RTC 秒中断		ExInt3IE.6		ExInt3IFG.6
			CF2 中断		ExInt3IE.7		ExInt3IFG.7
10	53h	10	RTC 非法数据中断	EIE.2	ExInt4IE.0	EXIF.6	ExInt4IFG.0
			保留				
			端口中断 2 (INT2), 下降沿触发		ExInt4IE.2		ExInt4IFG.2
			端口中断 3 (INT3), 下降沿触发		ExInt4IE.3		ExInt4IFG.3
			保留				
			过零点中断		ExInt4IE.5		ExInt4IFG.5
			保留				
			保留				
11	5Bh	11	TimerA 溢出中断	EIE.3	ExInt5IE.0	EXIF.7	ExInt5IFG.0
			TimerA 捕获中断 0		ExInt5IE.1		ExInt5IFG.1
			TimerA 捕获中断 1		ExInt5IE.2		ExInt5IFG.2
			TimerA 捕获中断 2		ExInt5IE.3		ExInt5IFG.3
			保留				
			保留				
			保留				
			保留				

硬件查询顺序	向量	中断号	说明	内核 En	扩展 En	内核 Flag	扩展 Flag
			保留				
12 (最低)	63h	12	掉电中断	EIE.4		EICON.3	

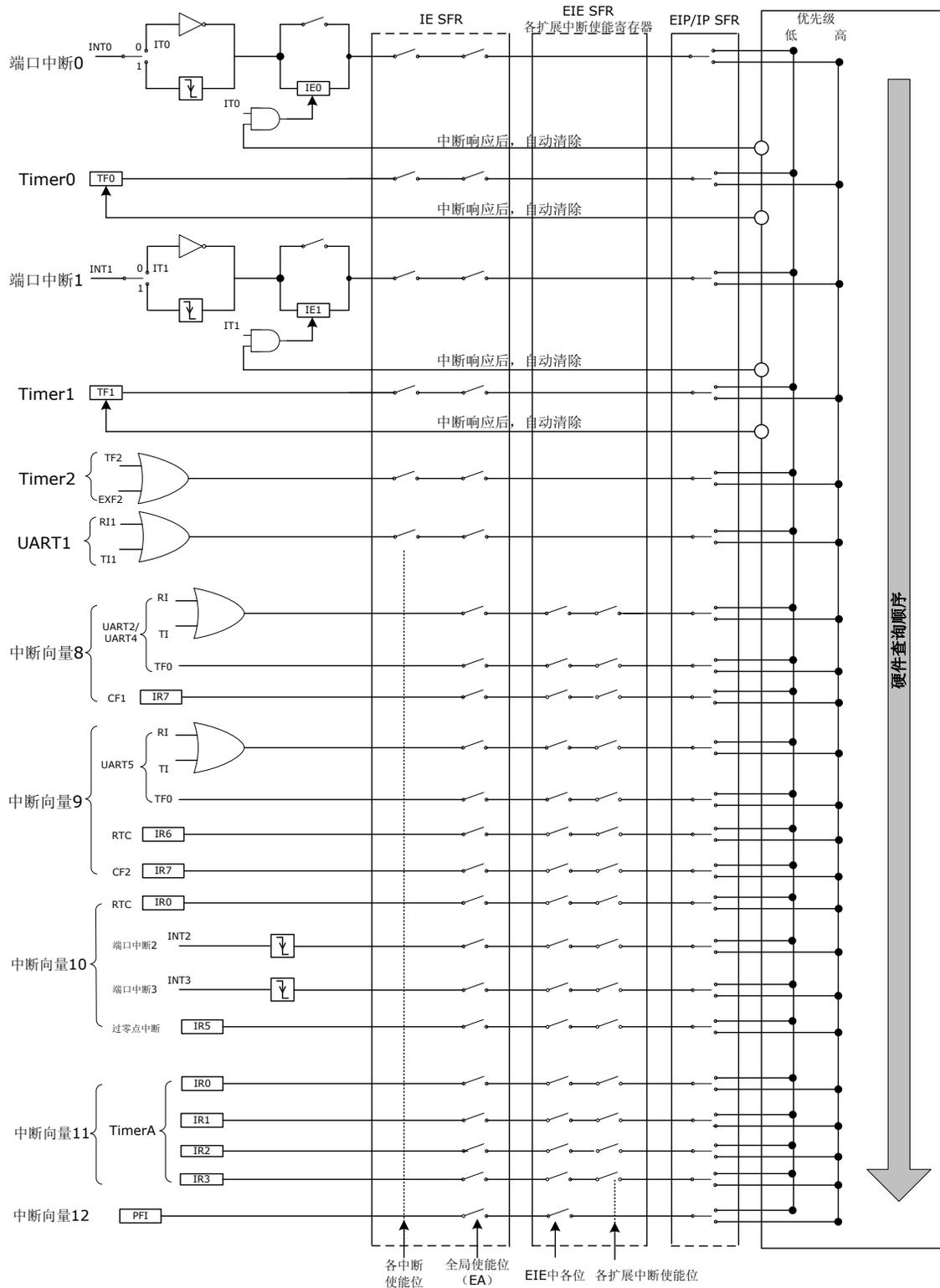


图 10-2 中断控制系统

第11章 TIMERS/UART

当发生 POR 复位、RSTn 低电平有效、WDT 溢出复位、RTC/IO 休眠唤醒复位、恢复供电复位或调试复位时，所有定时器和 UART 串行端口被复位。系统进入浅睡眠或者深睡眠状态后，UART 串行端口和定时器会停止工作，进入低功耗状态。

11.1 定时器/计数器

V9811 可提供最多 10 个硬件定时器：

1. 具有比较/捕获功能的 16 位计数器，TimerA；
2. 8052 的 Timer0、Timer1 和 Timer2。其中，Timer0 和 Timer2 是普通定时器，而，Timer1 除了作为普通定时器外，还可被用作 UART1 的波特率发生器。使用 Timer0 和 Timer1 的计数器功能，以及 Timer2 的计数器/捕获功能时，需要对 GPIO 进行设置；
3. 3 个扩展外设的 UART 串行口（UART2/UART4/UART5）自带的波特率发生器（与 Timer1 兼容，可做普通定时器用，但波特率发生器作普通定时器用时不会提起溢出中断）和普通定时器（与 Timer0 兼容）。

本节仅对 8052 微控制器的 Timer0/Timer1/Timer2 和 TimerA 进行介绍，UART 串行口自带的普通定时器将在“UART”一节中介绍。

用户可通过配置时钟周期控制寄存器（CKCON SFR, 0x8E），选择 Timer0、Timer1 和 Timer2 的时钟源（clk/12 或 clk）；通过配置扩展 UART 的控制/状态寄存器，用户可以选择与其相应的波特率发生器和普通定时器的时钟源（clk/12 或 clk）。

11.1.1 TIMERA

TimerA 是一个 16-bit 计数器，有 3 个比较/捕获模块，具有四种工作模式，而且，有 8 种输出模式的 3 个可配置输出单元。

表 11-1 TimerA 寄存器列表

寄存器		可读 (R) / 可写 (W)
0x2900	TACTL, TimerA 控制寄存器	R/W
0x2902~0x2903	TAR, TimerA 计数寄存器。0x2902 为低位字节，0x2903 为高位字节。	R
0x2904~0x2905	TACCTL0~TACCTH0, TimerA 的捕获/比较控制寄存器 0 0x2904 为低位字节，0x2905 为高位字节	R/W
0x2906~0x2907	TACCTL1~TACCTH1, TimerA 的捕获/比较控制寄存器 1 0x2906 为低位字节，0x2907 为高位字节	R/W
0x2908~0x2909	TACCTL2~TACCTH2, TimerA 的捕获/比较控制寄存器 2 0x2908 为低位字节，0x2909 为高位字节	R/W
0x290A~0x290B	TACCR0, TimerA 的捕获/比较 0 寄存器	R/W
0x290C~0x290D	TACCR1, TimerA 的捕获/比较 1 寄存器	R/W
0x290E~0x290F	TACCR2, TimerA 的捕获/比较 2 寄存器	R/W

下面将对各相关寄存器做出说明。

表 11-2 TimerA 计数寄存器 (TAR, 0x2902~0x2903)

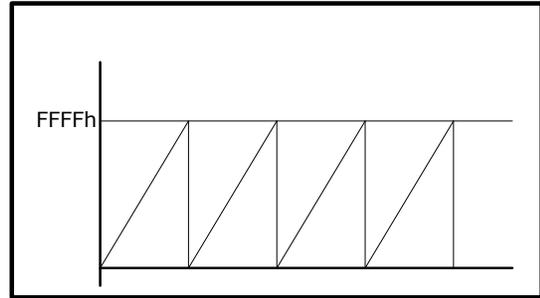
0x2902~0x2903, R, TimerA 计数寄存器, TAR	
TAR	该寄存器的值就是 16 位定时器/计数器的值。在 V9811 中, 该寄存器设定为只读, 不可由软件写入, 但可以由软件复位。另, TimerA 溢出时会提起中断。 0x2902 为 TAR 低位字节, 0x2903 为 TAR 高位字节。
默认值	0

表 11-3 TimerA 控制寄存器 (TACTL, 0x2900)

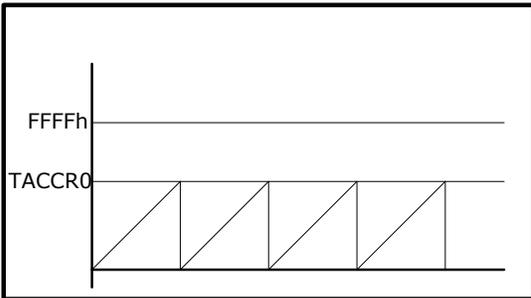
0x2900, R/W, TimerA 控制寄存器, TACTL			
位		默认值	说明
Bit[7:6]	ID1/ID0	0	分频系数: 00, 分频系数为 2; 01, 分频系数为 4; 10, 分频系数为 8; 11, 分频系数为 16
Bit[5:4]	MC1/MC0	0	工作模式选择, 详见图 10-1: 00, Stop Mode, 计数器停止工作; 01, Up Mode, 计数器计数到寄存器 TACCR0 的值后, 再从 0 开始重新计数; 10, Continuous Mode, 计数器计数到 FFFFh 后, 再从 0 开始重新计数; 11, Up/Down Mode, 计数器连续计数到寄存器 TACCR0 的值后, 再递减计数到 0。
Bit3	TSEL	-	时钟源选择。0, $f_{MCU}/128$; 1, f_{MCU}
Bit2	CLR	0	清除计数器的值(TAR)。当 CLR 置位时, TAR 清零, ID1/ID0=00; Up/Down Mode 下, 计数器直接从 0 开始重新计数。
Bit1	TAIE	0	TimerA 的中断使能。当 EIE.3=1, 且, TAIE=1 时, 使能 TimerA 溢出中断; 清零则禁止。
Bit0	TAIFG	0	TimerA 的中断标记。0, 没有中断; 1, 存在中断。 Up Mode: 当定时器从寄存器 TACCR0 的值到 0000h 的时候, TAIFG 置位, 参考图 10-3; Continuous Mode: 当定时器从 FFFFh 到 0000h 的时候, TAIFG 置位; Up/down Mode: 定时器先从 0000h 递增至 TACCR0 的值, 再从 TACCR0 的值递减到 0000h, 递减过程中, 从 0001h 递减到 0000h 的那个时刻, TAIFG 置位。



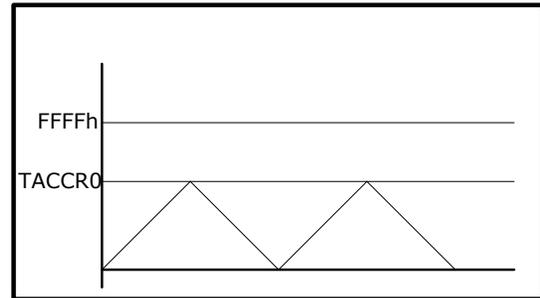
Stop Mode
计时器停止工作。



Continuous Mode
计数器计数到FFFFh，并从0开始重新计数至FFFFh



Up Mode
计数器计数到TACCRO的值，并从0开始重新开始计数至TACCRO的值



Up/Down Mode
计数器连续计数到TACCRO的值，并递减到0000h，再从0000h计数至TACCRO的值

图 11-1 TimerA 的工作模式

当 $MCx > 0$ 时，或有时钟输入时，计时器/计数器工作开启；在 Up mode 或 Up/Down mode 下，如果向寄存器 TACCRO 写入 0，则计时器/计数器停止工作，而向该寄存器写入非零的值，可以重启定时器，从 0000h 开始计数至 TACCRO 的值。

在 Up mode 下，在定时器/计数器运行时改变 TACCRO 值，则：如果新的值大于等于原来的值或定时器目前的计数，则定时器/计数器会计数到新的 TACCRO 的值后再重新从 0 开始计数；如果新的 TACCRO 的值小于定时器目前的计数，则当 TACCRO 值重新设定后，定时器再计一次数后直接从 0 开始重新计数。

在 Up/Down mode 下，在定时器/计数器递减计数时改变 TACCRO 的值，则计时器继续保持递减计数，直到 0000h，然后，从 0000h 递增至新的 TACCRO 的值；在定时器/计数器递增计数时改变 TACCRO 的值，则：如果新的值大于或等于原来的值或定时器目前的计数，则计时器递增计数到新的 TACCRO 的值，再递减到 0000h；如果新的值小于定时器目前的计数，则当 TACCRO 重新设定后，定时器再递增计数一次后开始递减计数到 0000h。

在 Continuous Mode 下，用户可根据自己的需要设定输出频率，如图 11-2 所示。

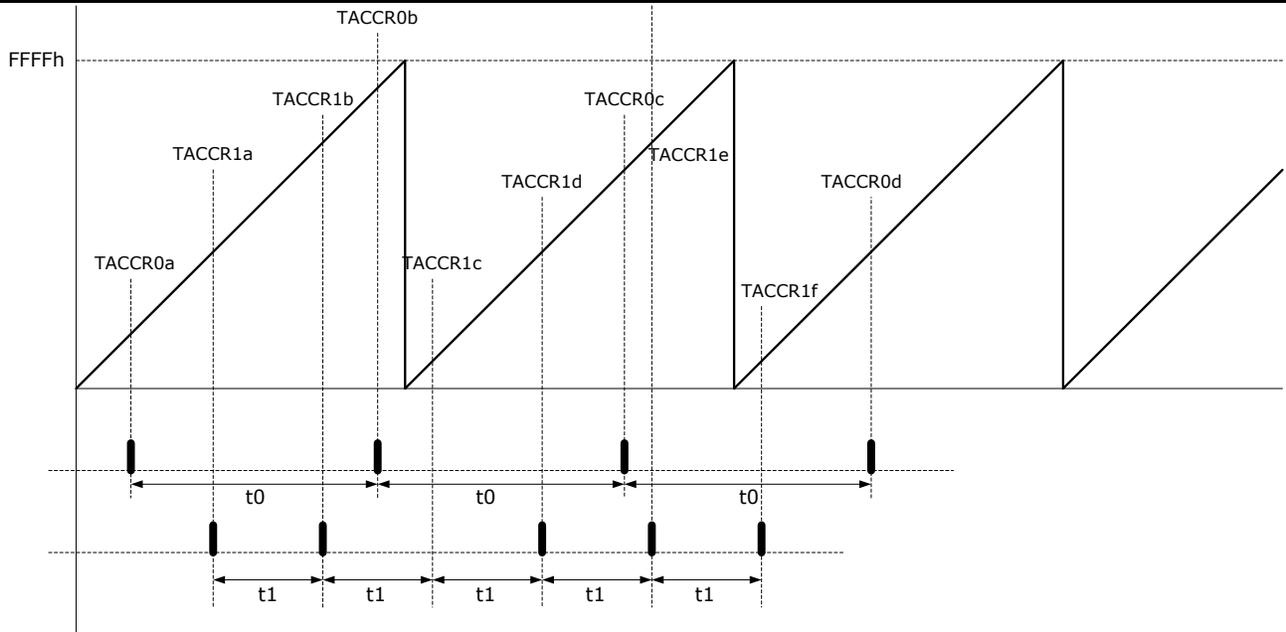


图 11-2 Continuous Mode 的用途

在图 11-2 中，TACCR0a 或 TACCR1a 为 Ta_0 或 Ta_1 时刻的寄存器 TACCR0 或 TACCR1 的值，TACCR0b 或 TACCR1b 为 Tb_0 或 Tb_1 时刻 ($Tb_0 = Ta_0 + t_0$, $Tb_1 = Ta_1 + t_1$) 的 TACCR0 或 TACCR1 的值，依次类推。

上图表明，在 Continuous Mode 下，中断使能时 ($CCIE=1$, Bit4, TACCTLx, 见表 10-4)，用户可设定 TimerA 每隔一段时间（如上图中的 t_0 或 t_1 ）产生一个中断，即，当 $TAR=TACCR0a$ 和 $TACCR1a$ 时，分别产生一个中断，CCIFG (Bit0, TACCTLx, 见表 11-4) 分别置标志位；当 $TAR=TACCR0b$ 和 $TACCR1b$ 时，又分别产生一个中断，依次类推。其中， t_0 和 t_1 相互独立，互不影响。因为在 V9811 中，TimerA 有三个不同的比较/捕获模块，所以，用户最多可设置三个不同的输出频率。在这一应用中，当定时器从 FFFFh 计数到 0000h 的时候，TAIFG 仍会置位。

表 11-4 TimerA 的捕获/比较模块 x*的控制寄存器 (0x2904~0x2909)

0x2904~0x2905/0x2906~0x2907/0x2908~0x2909, R/W, TimerA 的捕获/比较模块 x 的控制寄存器, TACCTLx~TACCTHx				
字节	位		默认值	说明
高位字节 TACCTHx 0x2905 0x2907 0x2909	Bit[15:14]	CM1/CM0	0	捕获模式选择。 00, 禁止捕获模式; 01, 上升沿捕获; 10, 下降沿捕获; 11, 在上升沿和下降沿都捕获
	Bit[13:12]	CCIS1/CCIS0	0	输入信号选择, 这两个位定义捕获信号的来源, 在比较模式下无效。复位后, 该位初始值为 0。 00, 选择输入引脚 TA0/TA1/TA2 上的信号; 01/10, 输入 0; 11, 输入 1。
	Bit11	SCS	0	用于使捕获输入信号与定时器时钟同步。复位后, 该位初始值为 0。 0, 非同步的捕获; 1, 同步的捕获
	Bit10	SCCI	0	选中的输入信号 (引脚 TA0/TA1/TA2 上的输入信号、1 或 0) 随着比较信号 EQUx 锁存进入有一个对用户透明的锁存器, 通过这个位, 可以读出它的值。复位后, 该位初始值为 0。
	Bit9	保留	0	只读, 读出值永远是 0。
低位字节 TACCTLx 0x2904 0x2906	Bit8	CAP	0	捕获/比较模式选择。 0, 比较模式, 用于产生 PWM 输出信号或在特定的时间间隔内产生中断信号; 1, 捕获模式, 用于计算速度或测量时间, 将定时器的值 (TAR) 捕获进入寄存器 TACCR0/1/2。
	Bit7	OUTMOD2	0	选择输出模式, 脉冲输出详见图 11-3。 000, 相应引脚 TAx 上输出 OUT 位的值 001, 当 TAR=TACCRx (x=0~2) 时, 相应引脚 TAx 的输出置位。置位一直保持, 至定时器被复位, 或用户选择了其它输出模式并输出。

0x2904~0x2905/0x2906~0x2907/0x2908~0x2909, R/W, TimerA 的捕获/比较模块 x 的控制寄存器, TACCTLx~TACCTHx

字节	位		默认值	说明		
0x2908	Bit6	OUTMOD1	0	010, 当 TAR=TACCRx (x=1~2) 时, 相应引脚 TAx 的输出翻转; 当 TAR=TACCR0 时, 引脚 TAx 的输出复位。该输出模式不适用于引脚 TA0 上的输出。		
				011, 当 TAR=TACCRx (x=1~2) 时, 相应引脚 TAx 的输出置位; 当 TAR=TACCR0 时, 引脚 TAx 的输出复位。该输出模式不适用于引脚 TA0 上的输出。		
	Bit5	OUTMOD0		100, 当 TAR=TACCRx (x=0~2) 时, 相应引脚 TAx 的输出翻转。		
				101, 当 TAR=TACCRx (x=0~2) 时, 相应引脚 TAx 的输出复位。复位一直保持, 至用户选择了其它的输出模式, 并输出。		
	Bit4	CCIE		0	110, 当 TAR=TACCRx (x=1~2) 时, 相应引脚 TAx 的输出翻转; 当 TAR=TACCR0 时, 引脚 TAx 的输出置位。该输出模式不适用于引脚 TA0 上的输出。	
					111, 当 TAR=TACCRx (x=1~2) 时, 相应引脚 TAx 的输出复位; 当 TAR=TACCR0 时, 引脚 TAx 的输出置位。该输出模式不适用于引脚 TA0 上的输出。	
	Bit4	CCIE			0	中断使能, 使能或者禁止捕获/比较模块 x 提起的中断。 0, 禁止; 1, 使能。
	Bit3	CCI			0	捕获/比较输入信号
选中的输入信号 (引脚 TA0/TA1/TA2 上的输入信号、1 或 0) 可通过这个位读出来。复位后, 该位初始值为 0。						
Bit2	OUT	0	如果 bit7~bit5 配置为 000, 引脚 TA0/TA1/TA2 上输出 OUT 位的值。			
Bit1	COV	0	捕获溢出标志			
			选中捕获模式, COV= 0, 捕获信号的产生被复位, 捕获事件不能将 COV 位置位。			
Bit0	CCIFG		0	选中捕获模式, COV= 1, 如果发生了第二次捕获事件, 而第一次捕获的值还没有被读出, 溢出标志位 COV 置位。必须使用软件对溢出标志位进行复位, 读出捕获的值并不能对它进行复位。		
				捕获模式: 如果 CCIFG 置位, 表示定时器的值 (TAR) 被捕获进入了寄存器 TACCR0/1/2, 参考图 11-3。		
比较模式: 如果 CCIFG 置位, 表示定时器的值 (TAR) 等于寄存器 TACCR0/1/2 中的数据, 参考图 11-3。						

0x2904~0x2905/0x2906~0x2907/0x2908~0x2909, R/W, TimerA 的捕获/比较模块 x 的控制寄存器, TACCTLx~TACCTHx

字节	位	默认值	说明
			<p>在模块 0 中, 当中断请求被接受之后, CCIFG 自动复位。</p> <p>在模块 1 和模块 2 中, 当中断请求被接受之后, 提起中断的标志位自动复位。如果相应的中断使能位被清零, 则不会产生中断, 但是这些标志位还是会置位。这样, 这些标志位必须用软件进行复位。当相应的中断使能位置位的时候, 软件设置 CCIFG 也会产生中断。</p>

x=0~2, 分别表示 TimerA 的捕获/比较模块 0/1/2, 各模块的控制寄存器地址见表 11-1 TimerA 寄存器列表。

在图 11-3 中，用户通过设置 OUTMOD2/OUTMOD1/OUTMOD0 三个位，可以选择不同的输出模式。其中，当这三个位被配为 010/011/110/111 的时候，输出脉冲的频率和占空比改变，从而实现脉冲宽度调制（PWM）。

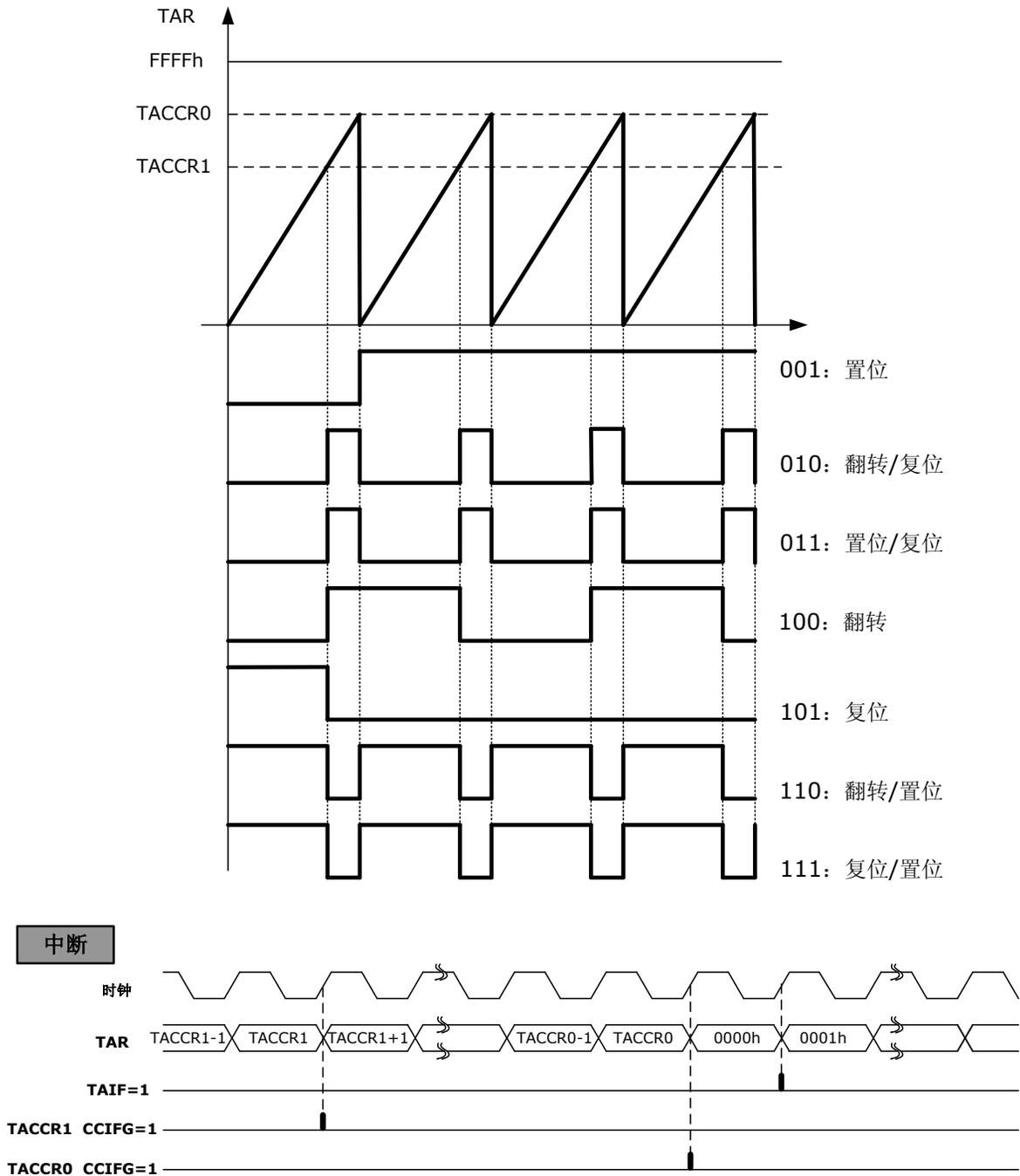


图 11-3 Up Mode, TimerA 比较模式下的 TA1 引脚的输出

11.1.2 TIMER0/TIMER1/TIMER2

11.1.2.1 定时器速率控制

当 CKCON SFR (0x8E) 中的位 CKCON.5、CKCON.4 和 CKCON.3 被设置为 1 时，相应的定时器每个时钟周期 (clk, MCU 时钟周期) 进行一次计数。反之，相应的定时器每 12 个时钟周期进行一次计数。定时器的控制相互独立。复位后这三个位都是 0。

表 11-5 Timer0/Timer1/Timer2 时钟源选择

寄存器	位		说明
时钟周期控制寄存器 (CKCON SFR, 0x8E)	Bit5	T2M	Timer2 时钟源选择。默认为 0。 0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源。
	Bit4	T1M	Timer1 时钟源选择。默认为 0。 0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源。
	Bit3	T0M	Timer0 时钟源选择。默认为 0。 0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源。

11.1.2.2 TIMER0 和 TIMER1

Timer0 和 Timer1 都有四种工作模式，由 TMOD SFR (表 11-6) 和 TCON SFR (表 11-7) 进行控制。四种工作模式分别为：

- 13 位定时器/计数器 (模式 0)
- 16 位定时器/计数器 (模式 1)
- 自动重新加载初值的 8 位定时器/计数器 (模式 2)
- 两个 8 位的定时器/计数器 (模式 3, 仅 Timer0 能工作在这种模式下)

表 11-6 Timer0/1 模式控制寄存器 (TMOD SFR, 0x89)

位	功能	
Bit7 TMOD.7	GATE	Timer1 的门控信号。 当 GATE 置位时, Timer1 只有在引脚 INT1 = 1 (Pin21) 并且 TR1 (TCON.6) = 1 的情况下工作。当 GATE = 0 时, Timer1 在 TR1 = 1 的情况下工作, 而与引脚 INT1 的状态无关。
Bit6 TMOD.6	C/T	Timer1 模式选择位。 C/T = 0, Timer1 作定时器用, 由时钟 (clk 或 clk/12, 由 CKCON.4 位决定) 驱动。C/T = 1, Timer1 作计数器用, 由引脚 T1 (Pin24) 输入信号驱动, 对引脚 T1 上输入信号的 1-0 的跳变进行计数。
Bit[5:4] TMOD.5/TMOD.4	M1/M0	Timer1 模式选择位 00, 模式 0, 13 位定时器/计数器 01, 模式 1, 16 位定时器/计数器

位		功能
		10, 模式 2, 自动重新加载初值的 8 位定时器/计数器 11, 模式 3, 两个 8 位的定时器/计数器
Bit3 TMOD.3	GATE	Timer0 的门控信号。 当 GATE 置位时, Timer0 只有在引脚 INT0 = 1 (Pin22) 并且 TR0 (TCON.4) = 1 的情况下工作。 当 GATE = 0 时, Timer0 在 TR0 = 1 的情况下工作, 与引脚 INT0 的状态无关。
Bit2 TMOD.2	C/T	Timer0 模式选择位。 C/T = 0, Timer0 作定时器用, 由时钟 (clk 或 clk/12, 由 CKCON.3 位决定) 驱动。 C/T = 1, Timer0 作计数器用, 由引脚 T0 (Pin61) 输入信号驱动, 对引脚 T0 上输入信号的 1-0 的跳变进行计数。
Bit[1:0] TMOD.1/TMOD.0	M1/M0	Timer0 模式选择位 00, 模式 0, 13 位定时器/计数器 01, 模式 1, 16 位定时器/计数器 10, 模式 2, 自动重新加载初值的 8 位定时器/计数器 11, 模式 3, 两个 8 位的定时器/计数器

表 11-7 Timer0/1 控制寄存器 (TCON SFR, 0x88)

位		功能
Bit7 TCON.7	TF1	Timer1 溢出标志位。 当 Timer1 计数溢出时, TF1=1; 当 CPU 响应此溢出中断后自动清 0。
Bit6 TCON.6	TR1	Timer1 运行控制位。 TR1=1, 使能 Timer1 进行计数。
Bit5 TCON.5	TF0	Timer0 溢出标志位。 当 Timer0 计数溢出时, TF0=1; 当 CPU 响应此溢出中断后自动清 0。
Bit4 TCON.4	TR0	Timer0 运行控制位。 TR0=1, 使能 Timer0 进行计数。
Bit3 TCON.3	IE1	端口中断 1 边沿检测位。如果将端口中断 1 配置为边沿触发 (IT1 = 1), 当检测到引脚 INT1 (Pin21) 输入信号上的下降沿的时候, IE1 由硬件自动置位; 当 CPU 响应了相应的中断后, IE1 由硬件自动清除。在边沿触发模式中, IE1 也可以由软件进行清除。 如果端口中断 1 配置为电平触发 (IT1 = 0), 当引脚 INT1 上的输入信号为低电平的时候, IE1 由硬件自动置位, 当引脚 INT1 上的输入信号为高电平的时候, IE1 由硬件自动清除。在电平触发模式中, 软件不能对 IE1 位进行写操作。
Bit2 TCON.2	IT1	端口中断 1 类型选择位。 1, CPU 检测引脚 INT1 输入信号上的下降沿 (边沿触发); 0, CPU 检测引脚 INT1 上的输入信号上的低电平 (电平触发)。

位	功能
Bit1 TCON.1	<p>IE0</p> <p>端口中断 0 边沿检测位。</p> <p>如果端口中断 0 配置为边沿触发 (IT0 = 1)，当检测到引脚 INT0 (Pin22) 输入信号上的下降沿的时候，IE0 由硬件自动置位；当 CPU 响应了相应的中断后，IE0 由硬件自动清除。在边沿触发模式中，IE0 也可以由软件进行清除。</p> <p>如果端口中断 0 配置为电平触发 (IT0 = 0)，当引脚 INT0 上的输入信号为低电平的时候，IE0 由硬件自动置位，当引脚 INT0 上的输入信号为高电平的时候，IE0 由硬件自动清除。在电平触发模式中，软件不能对 IE0 位进行写操作。</p>
Bit0 TCON.0	<p>IT0</p> <p>端口中断 0 类型选择位。</p> <p>1, CPU 检测引脚 INT0 上的输入信号上的下降沿 (边沿触发)；</p> <p>0, CPU 检测引脚 INT0 上的输入信号上的低电平 (电平触发)。</p>

11.1.2.2.1 Timer0/1 模式 0

模式 0 对于 Timer0 和 Timer1 而言是相同的。

在模式 0 中，定时器/计数器的低位字节 TL0 SFR (0x8A) 或 TL1 SFR (0x8B) 会从 0 计数到 31，当从 31 开始递增计数时，上述寄存器被清零，且高位字节 TH0 SFR (0x8C) 或 TH1 SFR (0x8D) 加 1。在这一模式下，定时器/计数器仅使用了 13 位，即 TLx (x=0~1) 的 Bit0~Bit4 和 THx (x=0~1) 的全部 8 位。在模式 0 中，TL0 SFR (0x8A) 或 TL1 SFR (0x8B) 的高三位是不确定的，当使用软件确定寄存器的值时，这三位必须被屏蔽。

在寄存器 TCON SFR 中，通道配置控制位 TR0/TR1 (TCON.4/TCON.6)，用户可以开启定时器/计数器。在寄存器 TMOD SFR 中，通过配置 C/T 位 (TMOD.6 或 TMOD.2)，用户可以选择 Timer0/1 是作定时器用还是作计数器用。

在 GATE 位 (TMOD.7 或 TMOD.3) 是 0 或者 GATE 位是 1 并且引脚 INT0 (Pin22) 或者 INT1 (Pin21) 上相应的中断输入信号有效的时候，如果 TRx=1 (x=0~1, TCON.4 或 TCON.6)，则，Timer0/1 运行。

当 13 位定时器/计数器递增计数到 1FFFh，然后再递增计数到全 0 的时候，TCON SFR 中的 TF0 (TCON.5) 或者 TF1 (TCON.7) 置位，并且向 CPU 提起中断。

11.1.2.2.2 Timer0/1 模式 1

模式 1 对于 Timer0 和 Timer1 而言是相同的。在模式 1 中，Timer0/1 被配置成 16 位定时器/计数器。因为定时器/计数器的低位字节 (TL0 SFR 和 TL1 SFR) 的全部 8 位都有效，所以 TLx (x=0~1) 从 0 开始递增计数到 255。当 TLx (x=0~1) 从 255 继续递增计数时，定时器/计数器的低位字节会被清零，而高位字节 (TH0 SFR 和 TH1 SFR) 会加 1。定时器/计数器计数到 FFFFh 后，会递增计数到全 0。其他情况下，模式 1 的操作与模式 0 相同。

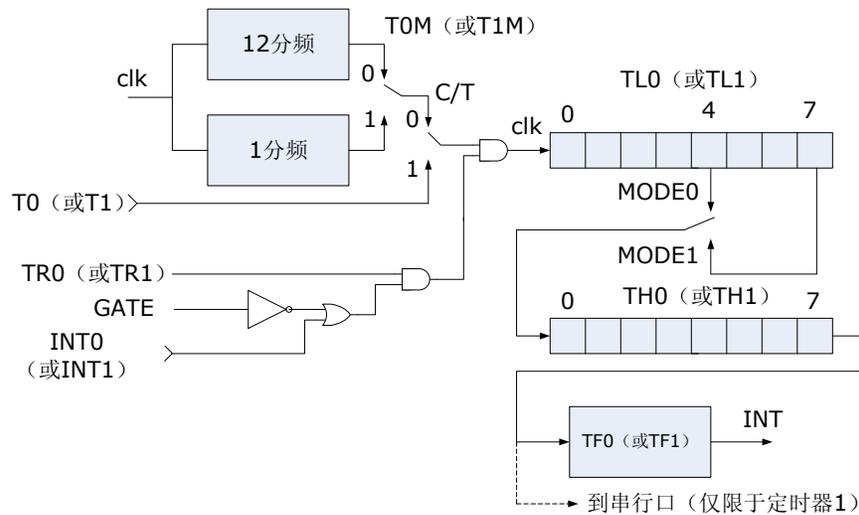


图 11-4 Timer0/1 模式 0/1

11.1.2.2.3 定时器 0/1 模式 2

模式 2 对于 Timer0 和 Timer1 而言是相同的。在模式 2 中，Timer0/1 被配置成可以重新加载初值的 8 位定时器/计数器。其中，定时器/计数器的低位字节 TL0 SFR (0x8A) 或 TL1 SFR (0x8B) 是定时器/计数器，高位字节 TH0 SFR (0x8C) 或 TH1 SFR (0x8D) 保存重新加载初值（如，200）。低位字节可递增至 255，而当 TLx (x=0~1) 从 255 继续递增计数时，TLx 的值会重置为 THx (x=0~1) 中保存的初值（如，200），然后，TLx 从该初值（如 200）开始重新计数，而 THx 保存不变。模式 2 的控制部分与模式 0 和模式 1 相同。

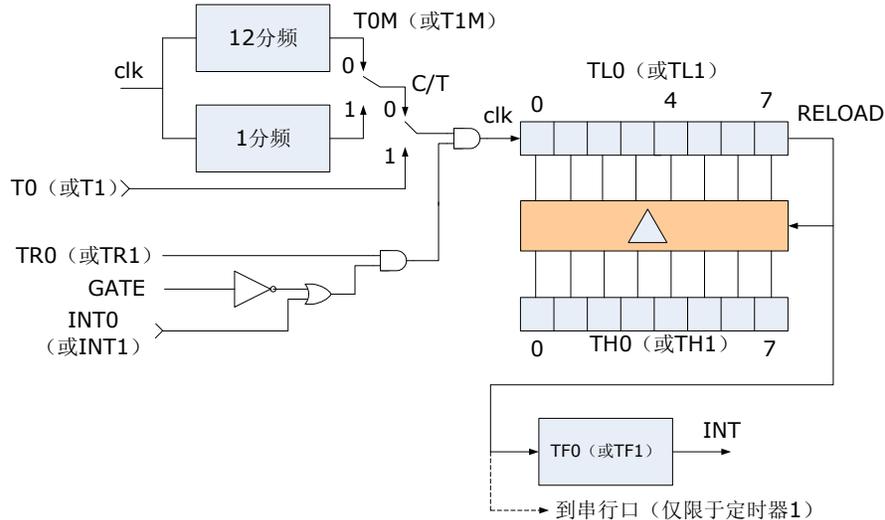


图 11-5 Timer0/1 模式 2

11.1.2.2.4 Timer0/1 模式 3

在模式 3 中，Timer0 的高位字节和低位字节分别作为两个独立的 8 位定时器/计数器使用；Timer1 停止计数，并保持它的值不变。

如图 11-6 所示，TL0 SFR (0x8A) 被配置成 8 位的定时器/计数器，由 Timer0 的控制位控制。在寄存器 TMOD SFR 中，通过配置 C/T 位 (TMOD.2)，用户可以选择 TL0 SFR 是作定时器用还是作计数器用。当 TR0

(TCON.4) = 1, 且引脚 INT0 (Pin22) 为高电平时, 如果 TMOD.3 = 1, 则 Timer0 可用于对引脚 INT0 上的输入信号上低电平脉冲宽度的计数。

TH0 SFR (0x8C) 作为一个独立的 8 位定时器工作, 只能由时钟 (clk 或者 clk/12) 驱动。Timer1 的控制位 TR1 (TCON.6) 和标志位 TF1 (TCON.7) 被用来作为 TH0 SFR 的控制位和标志位。

由于 Timer0 使用了 Timer1 的控制位 (TR1) 和中断标志位 (TF1), 所以当 Timer0 工作在模式 3 中的时候, Timer1 不能完全发挥作用。不过 Timer1 依然可以用作波特率发生器, 位于 TL1 SFR (0x8B) 和 TH1 SFR (0x8D) 中的依然是 Timer1 的计数值。

当 Timer0 工作在模式 3 下时, 通过配置 Timer1 的模式位 (TMOD.5 和 TMOD.4), 用户可控制 Timer1。在模式 0、模式 1 或者模式 2 中, 用户可开启 Timer1; 在模式 3 中, 用户可关闭 Timer1, 此时 Timer1 依然可以使用其模式选择位 C/T1 (TMOD.6) 和时钟源选择位 T1M 位 (CKCON.4)。因此, Timer1 可以对时钟周期 (1 分频或者 12 分频) 进行计数, 也可以对引脚 T1 (Pin24) 上的输入信号上的高到低电平变化进行计数。当 Timer0 工作在模式 3 的时候, Timer1 的 TMOD.7 仍有效。

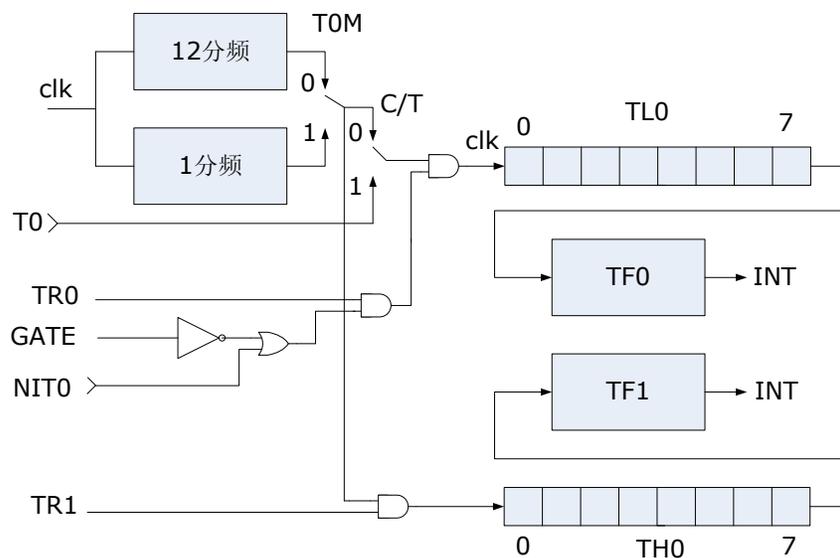


图 11-6 Timer0/1 模式 3

11.1.2.3 TIMER2

Timer2 只能工作在 16 位模式下, 并提供了 Timer0 和 Timer1 所不具有的一些功能。Timer2 的工作模式如下:

- 16 位定时器/计数器;
- 带有捕获功能的 16 位计数器;
- 可以自动加载初值的 16 位计数器;

与 Timer2 有关的 SFR 包括:

- T2CON SFR (0xC8);
- RCAP2L SFR (0xCA) - 当 Timer2 被配置成捕获模式的时候, 用于捕获 TL2 SFR 的值; 或者当 Timer2 被配置成自动重装初值模式的时候, 该寄存器的值被重载入 TL2 SFR;
- RCAP2H SFR (0xCB) - 当 Timer2 被配置成捕获模式的时候, 用于捕获 TH2 SFR 的值; 或者当 Timer2 被配置成自动重装初值模式的时候, 该寄存器的值被重载入 TH2 SFR;

- TL2 SFR (0xCC) – 16 位定时器/计数器的低位字节;
- TH2 SFR (0xCD) – 16 位定时器/计数器的高位字节。

表 11-8 Timer2 控制寄存器 (T2CON SFR, 0xC8)

位		功能
Bit7 T2CON.7	TF2	Timer2 溢出标志位。 当 Timer2 从 FFFFh 溢出的时候, TF2 位由硬件自动置位。TF2 位必须由软件清 0。只有在 bit5 和 bit4 都被清 0 的情况下, TF2 才会被置位。向 TF2 写入 1, 在 Timer2 中断使能的情况下, 会强制触发 Timer2 中断。
Bit6 T2CON.6	EXF2	Timer2 外部中断标志。 EXEN2 置位时, 当引脚 T2EX (Pin23) 上的输入信号的从高到低电平变化引起重新加载初值或者捕获事件时, EXF2 由硬件自动置位。EXF2 必须由软件进行清 0。向 EXF2 写入 1, 在 Timer2 中断使能的情况下, 会强制触发 Timer2 外部中断。
Bit5 T2CON.5	保留	默认为 0
Bit4 T2CON.4	保留	默认为 0
Bit3 T2CON.3	EXEN2	Timer2 外部中断使能。 EXEN2 = 1, 引脚 T2EX 上的输入信号由高到低电平的变化可以引发捕获或者重新加载初值事件发生。EXEN2 = 0 的时候, Timer2 将忽略引脚 T2EX 输入信号上的所有外部事件。
Bit2 T2CON.2	TR2	Timer2 运行控制位。 TR2 = 1, 启动 Timer2。TR2 = 0, 关闭 Timer2。
Bit1 T2CON.1	C/T2	Timer2 模式选择位。 C/T2 = 0, Timer2 作定时器用, 根据 T2M 位的值 (CKCON.5), Timer2 每个时钟周期或者每 12 个时钟周期进行一次计数。 C/T2 = 1, Timer2 作计数器用, 对引脚 T2 (Pin62) 上的输入信号的高到低电平跳变进行计数。
Bit0 T2CON.0	CP/RL2	捕获/重新加载标志位。 当 CP/RL2 = 1, 并且 EXEN2 = 1 时, 引脚 T2EX 上的输入信号发生高到低电平跳变会将 Timer2 当前的计数值捕获进入 RCAP2L SFR (0xCA) 或 RCAP2H SFR (0xCB)。 当 CP/RL2 = 0, 并且 EXEN2 = 1 时, 计数溢出或者引脚 T2EX 上的输入信号发生高到低电平跳变会使 Timer2 自动重新加载初值。如果 RCLK 或者 TCLK 被设置为 1, CP/RL2 将不起作用, 并且 Timer2 将在每一次溢出后都工作在自动重新加载初值模式。

11.1.2.4 TIMER2 的 16 位定时器/计数器模式

Timer2 在 16 位定时器/计数器模式下的工作原理如图 11-7 所示。在寄存器 T2CON SFR 中, 通过配置 C/T2 位 (T2CON.1), 用户可以选择 Timer2 作定时器用还是作计数器用。TR2 位 (T2CON.2) 使能 Timer2 运行。当计数从 FFFFh 递增的时候, TF2 标志位 (T2CON.7) 置位, 并且向 CPU 提起中断。

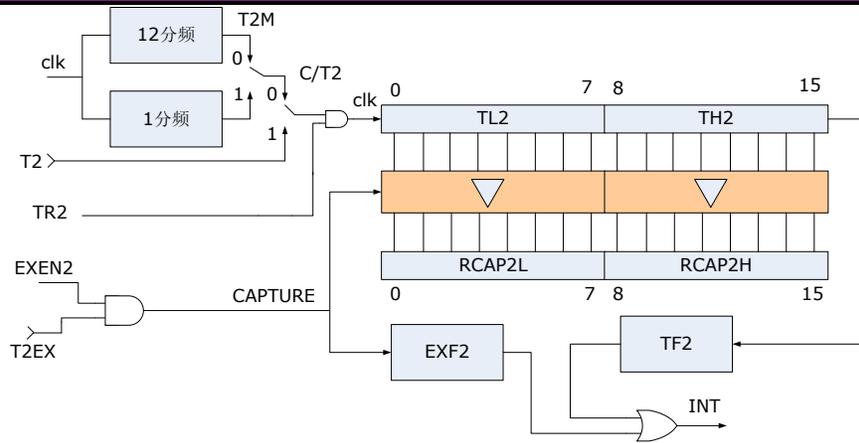


图 11-7 Timer2 的 16 位定时器/计数器模式，带捕获功能的 16 位计数器模式

11.1.2.5 TIMER2 的带有捕获功能 16 位计数器模式

Timer2 的捕获模式与 16 位定时器/计数器模式相同，只是增加了捕获寄存器以及控制信号。T2CON SFR 中的 CP/RL2 位 (T2CON.0) 控制捕获功能的使用。当 CP/RL = 1 并且 EXEN2 = 1 (T2CON.3) 的时候，引脚 T2EX (Pin23) 上的输入信号的高到低电平跳变会导致 Timer2 的值被加载到捕获寄存器 RCAP2L SFR (0xCA) 和 RCAP2H SFR (0xCB) 中。

11.1.2.6 TIMER2 的自动加载初值 16 位计数器模式

当 CP/RL2 = 0 (T2CON.0) 的时候，Timer2 被配置为自动加载初值模式，计数器的控制输入与其它 16 位计数器模式相同。软件事先将所要加载的初始值写入寄存器 RCAP2L SFR (0xCA) 和 RCAP2H SFR (0xCB) 中。当计数器从 FFFFh 递增的时候，Timer2 设置 TF2 位 (T2CON.7) 有效，并将 RCAP2L SFR 中的值重载入 TL2 SFR (0xCC)，将 RCAP2H SFR 中的值重载入 TH2 SFR (0xCD)。

当 Timer2 工作在自动加载初值模式的时候，如果使能位 EXEN2 = 1 (T2CON.3)，那么引脚 T2EX (Pin23) 上输入信号的高到低电平跳变可以强制引发计数初始值的加载。

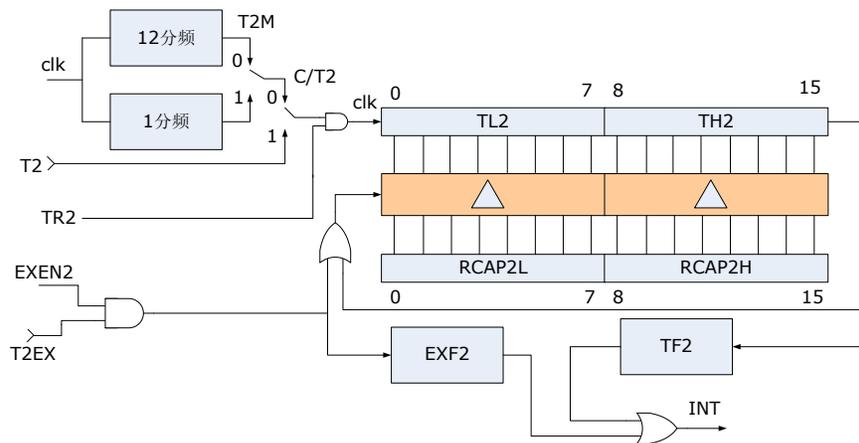


图 11-8 Timer2 自动加载初值的 16 位计数器模式

11.2 UART

在 V9811 中，片上共实现了 4 个 UART 串行口，除了 8052 微控制器的 UART1 外，还包括扩展外设的 UART2/UART4/UART5。

扩展外设的 UART 串行口有四种工作模式，其中，在模式 0 下，RXD 端口只能接收数据，TXD 端口则提供移位时钟；但在其他模式下，上述 3 个扩展外设的 UART 串行口与标准 8051 微控制器的 UART 串行口功能兼容。推荐使用扩展 UART2/UART4/UART5 作串行口用。

11.2.1 UART1

UART1 除了只能由 Timer1 产生波特率，以及不能产生 38kHz 载波之外，其结构与工作模式同 UART2。寄存器 EICON SFR (0xD8) 中的 SMOD1 (EICON.7) 控制 UART1 波特率是否加倍。

与 UART1 相关的 SFR 包括：

- SCON1 SFR (0xC0) - UART1 控制寄存器
- SBUF1 SFR (0xC1) - UART1 缓冲寄存器

表 11-9 UART1 控制寄存器 (SCON1 SFR, 0xC0)

位		功能
Bit[7:6] SCON1.7 SCON1.6	SM0_1 SM1_1	UART1 模式位 00, 模式 0, 8 位移位寄存器, 波特率=clk 或 clk/12 01, 模式 1, 8 位 UART, 波特率由 Timer1 决定 10, 模式 2, 9 位 UART, 波特率=clk/32 或 clk/64 11, 模式 3, 9 位 UART, 波特率由 Timer1 决定
Bit5 SCON1.5	SM2_1	多处理器通信使能位。在模式 2 和模式 3 中, SM2_1 位使能多处理器通信功能。在模式 2 和 3 中, 当 SM2_1 = 1 时, 如果接收到的第九位数据是 0, RI_1 将不置位。在模式 1 中, 当 SM2_1 = 1 时, 只有在接收到有效的停止位后, RI_1 才会置位。在模式 0 中, SM2_1 决定波特率: 当 SM2_1 = 0 的时候, 波特率是 clk/12; 当 SM2_1 = 1 的时候, 波特率是 clk。
Bit4 SCON1.4	REN_1	接收使能位。 当 REN_1 = 1, 使能通过 UART1 串行口接收数据。
Bit3 SCON1.3	TB8_1	存储模式 2 和模式 3 中要发送的第九位数据的值。
Bit2 SCON1.2	RB8_1	在模式 2 和模式 3 中, RB8_1 保存接收到的第九位数据。在模式 1 中, RB8_1 保存停止位。在模式 0 中没有使用 RB8_1。
Bit1 SCON1.1	TI_1	发送中断标志位。表示数据已经发送完成。在模式 0 中, TI_1 在第八位数据发送完成后置位。在其他的模式中, TI_1 在停止位出现在引脚 TXD1 输出信号上的时候置位。TI_1 必须由软件进行清 0。
Bit0 SCON1.0	RI_1	接收中断标志位。表示一个串行数据接收完成。在模式 0 中, RI_1 在接收完第八位数据后置位。在模式 1 中, 根据 SM2_1 位的状态不同, RI_1 在对接收到的停止位进行最后一次

位	功能
	采样后置位，在模式 2 和模式 3 中，RI_1 在对 RB8_1（第九位数据）进行最后一次采样后置位。RI_1 必须由软件进行清 0。

11.2.2 扩展外设 UART

UART2/UART4/UART5 的结构与原理基本相同，但是，仅 UART2 可用于发送带有 38kHz 载波的信号。

这几个 UART 串行口均带有一个波特率发生器（与 Timer1 兼容）以及一个普通定时器（与 Timer0 兼容）。其中普通定时器的定时溢出事件既可以通过中断来响应，也可以通过查询标志位的方式来响应；波特率发生器作为普通定时器使用的时候，并不会提起中断，而是把相应的溢出标志位置位。作为 CPU 的扩展外设，这些串行端口由专用的控制/状态寄存器控制其波特率是否加倍、普通定时器和波特率发生器的时钟源选择、启动和溢出状态等。

11.2.2.1 寄存器

与 UART 串行口相关的寄存器如表 11-10 所示。

表 11-10 UARTx 相关的寄存器

0x2820, R/W	TCON2, UART2 控制/状态寄存器
0x2821, R/W	TMOD2, UART2 相关定时器模式控制寄存器
0x2822, R/W	TH20, UART2 普通定时器高位字节
0x2823, R/W	TH21, UART2 波特率发生器高位字节
0x2824, R/W	TL20, UART2 普通定时器低位字节
0x2825, R/W	TL21, UART2 波特率发生器低位字节
0x2826, R/W	SCON2, UART2 控制寄存器
0x2827, R/W	SBUF2, UART2 缓冲寄存器
0x2830, R/W	TCON4, UART4 控制/状态寄存器
0x2831, R/W	TMOD4, UART4 相关定时器模式控制寄存器
0x2832, R/W	TH40, UART4 普通定时器高位字节
0x2833, R/W	TH41, UART4 波特率发生器高位字节
0x2834, R/W	TL40, UART4 普通定时器低位字节
0x2835, R/W	TL41, UART4 波特率发生器低位字节
0x2836, R/W	SCON4, UART4 控制寄存器
0x2837, R/W	SBUF4, UART4 缓冲寄存器
0x2838, R/W	TCON5, UART5 控制/状态寄存器
0x2839, R/W	TMOD5, UART5 相关定时器模式控制寄存器
0x283A, R/W	TH50, UART5 普通定时器高位字节
0x283B, R/W	TH51, UART5 波特率发生器高位字节
0x283C, R/W	TL50, UART5 普通定时器低位字节
0x283D, R/W	TL51, UART5 波特率发生器低位字节

0x283E, R/W	SCON5, UART5 控制寄存器
0x283F, R/W	SBUF5, UART5 缓冲寄存器

表 11-11 UARTx 控制/状态寄存器 (TCONx)

位		默认值	说明
Bit7	SMOD	0	UARTx 波特率加倍使能位。1, UARTx 的波特率加倍;
Bit6	保留		
Bit5	T1M	0	波特率发生器时钟源选择。0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源;
Bit4	T0M	0	普通定时器时钟源选择。0, 使用 clk/12 作为时钟源; 1, 使用 clk 作为时钟源;
Bit3	TF1	0	波特率发生器溢出标志, 不影响溢出中断。1, 溢出; 0, 无溢出
Bit2	TF0	0	普通定时器溢出标志。1, 溢出; 0, 无溢出
Bit1	TR1	0	波特率发生器的启动位。1, 启动; 0, 关闭
Bit0	TR0	0	普通定时器的启动位。1, 启动; 0, 关闭

表 11-12 UARTx 相关定时器模式控制寄存器 (TMODx)

位		功能
Bit7 TMOD2.7	GATE1	必须等于 0, 此时, 波特率发生器在 TR1 = 1 的情况下工作。
Bit6 TMOD2.6	C/T1	必须等于 0, 此时, 寄存器 TCONx 中的 T1M 位决定波特率发生器是由 clk 驱动还是由 clk/12 驱动。
Bit[5:4] TMOD2.5	T1M1 T1M0	波特率发生器模式选择位 00, 模式 0, 13 位定时器/计数器 01, 模式 1, 16 位定时器/计数器 10, 模式 2, 自动重新加载初值的 8 位定时器/计数器 11, 模式 3, 两个 8 位的定时器/计数器
Bit3 TMOD2.3	GATE0	必须等于 0, 此时, 普通定时器在 TR0 = 1 的情况下工作。
Bit2 TMOD2.2	C/T0	必须等于 0, 此时, 寄存器 TCONx 中的 T0M 位决定普通定时器是由 clk 驱动还是由 clk/12 驱动。
Bit1 TMOD2.1	T0M1 T0M0	普通定时器模式选择位 00, 模式 0, 13 位定时器/计数器 01, 模式 1, 16 位定时器/计数器 10, 模式 2, 自动重新加载初值的 8 位定时器/计数器 11, 模式 3, 两个 8 位的定时器/计数器

表 11-13 UARTx 控制寄存器 (SCONx)

位		功能
Bit[7:6] SCON2.7/SCON2.6	SM0 SM1	UARTx 模式位 00, 模式 0, 8 位移位寄存器, 波特率 = clk 或 clk/12 01, 模式 1, 8 位 UART, 波特率由波特率发生器决定 10, 模式 2, 9 位 UART, 波特率 = clk/32 或 clk/64 11, 模式 3, 9 位 UART, 波特率由波特率发生器决定
Bit5 SCON2.5	SM2	多处理器通信使能位。 在模式 2 和模式 3 中, SM2 位使能多处理器通信功能。在模式 2 和 3 中, 当 SM2 = 1 时, 如果接收到的第九位数据是 0, RI 将不置位。在模式 1 中, 当 SM2 = 1 时, 只有在接收到有效的停止位后, RI 才会置位。在模式 0 中, SM2 决定波特率: SM2 = 0, 波特率是 clk/12; SM2 = 1, 波特率是 clk。
Bit4 SCON2.4	REN	接收使能位。 REN = 1, 使能通过 UARTx 串行口接收数据。
Bit3 SCON2.3	TB8	存储模式 2 和模式 3 中要发送的第九位数据的值。
Bit2 SCON2.2	RB8	在模式 2 和模式 3 中, RB8 保存接收到的第九位数据。在模式 1 中, RB8 保存停止位。在模式 0 中没有使用 RB8。
Bit1 SCON2.1	TI	发送中断标志位。表示数据已经发送完成。在模式 0 中, TI 在第八位数据发送完成后置位。在其他的模式中, TI 在停止位出现在 TXD 输出信号上的时候置位。TI 必须由软件进行清 0。
Bit0 SCON2.0	RI	接收中断标志位。表示一个串行数据接收完成。在模式 0 中, RI 在接收完第八位数据后置位。在模式 1 中, 根据 SM2 位的状态不同, RI 在对接收到的停止位进行最后一次采样后置位, 在模式 2 和模式 3 中, RI 在对 RB8 进行最后一次采样后置位。RI 必须由软件进行清 0。

表 11-14 UARTx 缓冲寄存器 (SBUFx)

寄存器	R/W	位	默认值	说明
SBUFx	R/W	Bit[7:0]	0	UARTx 缓冲寄存器 串行端口缓冲寄存器仅有一个地址, 但实际是两个缓冲器, 写 SBUF 的操作完成待发送数据的加载, 读 SBUF 的操作可获得已接收到的数据, 两个操作分别对应两个不同的寄存器, 一个是只写发送寄存器, 另一个是只读接收寄存器。当工作于标准的异步全双工通信模式下时, UART 串行端口可实现同时读写数据。

11.2.2.2 UART2 发送载波信号

UART2 可以选择发送带有 38kHz 载波的信号, 也可以选择发送不带有载波的普通信号。用户可以通过寄存器 Txd2FS 设置 UART2 发送信号的类型, 通过寄存器 CARRH 和 CARRL 设置 38kHz 载波信号的占空比。

表 11-15 TXD2 类型选择寄存器 (Txd2FS, 0x28CF)

0x28CF, R/W, TXD2 类型选择寄存器, Txd2FS					
位		默认值	说明		
Bit[7:1]	保留	X			
Bit0	TXD2CARRY	0	0, 带有 38kHz 载波; 1, 不带有载波。 当 TXD2CARRY 被清零时, 引脚 TXD2 (Pin59) 发射数位“0”, 发送 38kHz 的载波调制信号。		

表 11-16 载波产生寄存器

寄存器		说明	R/W	位	默认值
0x2898	CARRHH	载波产生寄存器 1 高位字节, 占空比控制, 高电平持续时间	R/W	Bit[7:0]	0
0x2899	CARRHL	载波产生寄存器 1 低位字节, 占空比控制, 高电平持续时间	R/W	Bit[7:0]	0
0x289A	CARRLH	载波产生寄存器 2 高位字节, 占空比控制, 低电平持续时间	R/W	Bit[7:0]	0
0x289B	CARRLL	载波产生寄存器 2 低位字节: 占空比控制, 低电平持续时间	R/W	Bit[7:0]	0

11.2.3 UART 的模式选择

通过配置 UART 控制寄存器中的模式选择位, 如 SCON2 (0x2826) 中的 SM1 和 SM2 位, UART 串行口可工作于四种模式下。

表 11-17 UART 模式选择

模式		同步/异步	波特率时钟	数据位	开始位/停止位	第 9 位的功能
0	8 位移位寄存器	同步	clk 或者 clk/12	8	无	无
1	8 位 UART	异步	由波特率发生器决定	8	1 个开始位, 1 个停止位	无
2	9 位 UART	异步	clk/32 或者 clk/64	9	1 个开始位, 1 个停止位	0, 1, 奇偶校验
3	9 位 UART	异步	由波特率发生器决定	9	1 个开始位, 1 个停止位	0, 1, 奇偶校验

因为 UART1 除了只能由 Timer1 产生波特率, 以及不能产生 38kHz 载波之外, 其结构与工作模式同 UART2。而 UART4~UART5 (使用本身自带的波特率发生器) 除了不能产生 38kHz 载波外, 其结构与工作模式同 UART2。所以, 本节以 UART2 为例, 对 UART 的四种工作模式进行说明。

11.2.3.1 模式 0

在模式 0 下, RXD 只能接收数据, TXD 提供移位时钟。在模式 0 中, 当相应的寄存器 SCON2 (0x2826) 中的 REN (SCON2.4) 置位并且 RI (SCON2.0) 被清 0 的时候, 就开始接收数据。在移位时钟的每个上升沿上, UART 移位接收数据, 直到 8 个数据位全部接收完为止。在移位接收完第八个数据位之后, 再过一个机器周期, RI 置位并且接收过程结束, 直到软件清除 RI 位开始一次新的接收过程。

11.2.3.2 模式 1

模式 1 提供了标准的异步、全双工通信功能, 一帧数据包括 10 位: 一个开始位, 8 个数据位, 一个停止位。对于接收操作, 停止位保存在 RB8 位 (SCON2.2) 中。不管接收还是发送, 都是最低位在前。

模式 1 的波特率是波特率发生器溢出频率的函数。UART2 使用其自身的波特率发生器来产生波特率。每当波特率发生器从它们的最大计数值再次递增计数的时候，就给波特率电路产生了一个时钟。这个时钟被 16 分频后用于产生波特率。波特率计算公式如下：

$$\text{公式 11-1 } \text{BaudRate} = \frac{2^{\text{SMOD}x}}{32} \times \text{Overflow}$$

其中，Overflow 为波特率发生器的溢出频率。UART1 使用 Timer1 作波特率发生器，UART2~UART5 使用本身自带的波特率发生器（与 Timer1 兼容）。而，SMODx 为 SMOD0/1 的值，决定波特率发生器的溢出频率是否被 2 分频，从而实现波特率是否加倍。

一般情况下，波特率发生器被配置为自动加载初值的 8 位定时器模式。波特率发生器初值存储在 TH21（0x2823）中，所以，完整的波特率公式（使用 clk/12 产生波特率的情况下）为：

$$\text{公式 11-2 } \text{BaudRate} = \frac{2^{\text{SMOD}x}}{32} \times \frac{\text{clk}}{12 \times (256 - \text{TH21})}$$

其中，clk 为 MCU 时钟周期，TH21 为寄存器 TH21 的值。

波特率发生器的时钟源选择由寄存器 TCON2（0x2820）中的 T1M（TCON2.5）位决定。当 T1M=1 时，波特率发生器选择 clk 为时钟源，则公式 11-2 变为：

$$\text{公式 11-3 } \text{BaudRate} = \frac{2^{\text{SMOD}x}}{32} \times \frac{\text{clk}}{(256 - \text{TH21})}$$

此时，如果波特率已知，用户可通过下式求得 TH21 的值：

$$\text{公式 11-4 } \text{TH21} = 256 - \frac{2^{\text{SMOD}x} \times \text{clk}}{32 \times \text{BaudRate}}$$

当 T1M=0，且波特率已知时，用户可通过下式求得 TH21 的值：

$$\text{公式 11-5 } \text{TH21} = 256 - \frac{2^{\text{SMOD}x} \times \text{clk}}{384 \times \text{BaudRate}}$$

在模式 1 中，在软件把数据写入寄存器 SBUF2（0x2827）之后，UART2 就开始发送数据。UART2 按照下述顺序向 TXD2 端口（Pin59）上输出数据：开始位，8 位数据位（最低位在前），停止位。在停止位发送完成后 TI 位被置位。

模式 1 中，在 REN 位（SCON2.4）使能的情况下，RXD2 端口（Pin60）接收到开始位的下降沿后就开始了数据接收过程。为了达到这个目的，不管在何种波特率下，都对 RXD2 上的每一位采样 16 次。当检测到开始位的下降沿后，用于产生接收时钟的计数器被复位，以便与接收到的数据位同步。为了排除干扰的影响，UART2 对在每一位的中间位置进行的三次连续采样所获得的值进行判断，只有当在三次采样中至少两次都相同的值，接收到的数据位才会被认为是有效的。对于开始位的判断尤其如此。如果 RXD2 上的下降沿没有被上述判断方法判断为有效（低电平），那么串行口就停止接收数据，并等待 RXD2 上的下一个下降沿。

如果 RI 为 0（SCON2.0），且 SM2 为 1（SCON2.5），停止位的状态是 1（如果 SM2 为 0，那么不用关心停止位的状态），UART2 就把接收到的字节写入寄存器 SBUF2（0x2827）中，把停止位加载到 RB8 位（SCON2.2），并且设置 RI 位。如果不满足上述条件，所接收的数据就会丢失，数据不会加载进入寄存器 SBUF2 和 RB8 位，RI 位也不会被置位。

11.2.3.3 模式 2

模式 2 提供了异步、全双工的通信功能。一帧数据包括 11 个位：一个开始位，8 个数据位，一个可编程的第九位，一个停止位。

不管发送还是接收数据都是最低位在前。对于发送操作，第九位由 TB8 (SCON2.3) 决定。如果要把第九位当作奇偶校验位，需要把奇偶位的值 (Bit0, PSW SFR) 放入 TB8 中。

寄存器 TCON2 中的 SMOD 位 (TCON2.7) 决定是否将 UART2 波特率加倍 (clk/32 或 clk/64)。波特率由下式决定：

$$\text{公式 11-6 BaudRate} = \frac{2^{\text{SMOD}} \times \text{clk}}{64}$$

在模式 2 中，在软件把数据写入寄存器 SBUF2 (0x2827) 之后，UART2 就开始发送数据。UART2 按照下述顺序向 TXD2 端口 (Pin59) 上输出数据：开始位，8 位数据位 (最低位在前)，第九位和停止位。在停止位发送完成后 TI 被置位。

模式 2 中，在 REN 位 (SCON2.4) 使能的情况下，RXD2 端口 (Pin60) 接收到开始位的下降沿后就开始了数据接收过程。为了达到这个目的，不管在何种波特率下，都对 RXD2 端口上的每一位采样 16 次。当检测到开始位的下降沿后，用于产生接收时钟的计数器被复位，以便与接收到的数据位同步。为了排除干扰的影响，UART2 对在每一位的中间位置进行的三次连续采样所获得的值进行判断，只有当在三次采样中至少两次都相同的值，接收到的数据位才会被认为是有效的。对于开始位的判断尤其如此。如果 RXD2 上的下降沿没有被上述判断方法判断为有效 (低电平)，那么 UART2 就停止接收数据，并等待 RXD2 上的下一个下降沿。

如果 RI 为 0 (SCON2.0)，且 SM2 为 1 (SCON2.5)，停止位的状态是 1 (如果 SM2 为 0，那么不用关心停止位的状态)，UART2 就把接收到的字节写入寄存器 SBUF2 (0x2827)，把第九位加载到 RB8 位 (SCON2.2)，并且设置 RI 位。如果不满足上述条件，所接收的数据就会丢失，数据不会加载进入寄存器 SBUF2 和 RB8 位，RI 位也不会被置位。

11.2.3.4 模式 3

模式 3 提供了异步、全双工的通信功能。一帧数据包括 11 个位：一个开始位，8 个数据位，一个可编程的第九位和一个停止位。不管发送还是接收数据都是最低位在前。

模式 3 中数据的发送和接收操作与模式 2 中相同。模式 3 的波特率发生与模式 1 中相同。也就是说，模式 3 是由模式 2 的协议与模式 1 的波特率综合而成。

11.2.3.5 多处理器通信

当 SM2=1 (SCON2.5) 时，使能模式 2 和模式 3 中的多处理器通信功能。在多处理器通信模式中，接收到的第九位保存在 RB8 (SCON2.2) 中，并且在接收到停止位后，如果 RB8 = 1，那么将触发 UART2 接收中断。

多处理器通信的典型应用是一台主机向几台从机中的一台发送一个数据块。主机首先发送代表目标从机的地址字节。在发送地址字节的时候，主机把第九位设置为 1；在发送数据字节的时候，设置第九位为 0。

当 SM2 = 1 时，任何从机接收到数据字节都不会提起中断。不过，地址字节却可以向所有的从机提起中断，每一个从机都可以检查收到的地址字节，判断自己是不是被寻址的从机。地址译码工作必须由软件在中断服务程序中完成。被寻址的从机清除自己的 SM2 位，并准备开始接收数据字节。没有被寻址的从机保持自己的 SM2 位不变，并忽略主机发送来的数据字节。

第12章 LCD

12.1 时序和驱动电路

LCD 时序和驱动电路使用独立的 32.768kHz 的 OSC 时钟。

发生 POR 复位、RSTn 低电平有效或 WDT 溢出复位时，LCD 时序和驱动电路被复位。复位后，LCD 模拟驱动电路被关闭，扫描频率 64Hz，驱动分压电阻 300kΩ，并清空显示缓冲寄存器。显示缓冲寄存器的地址是 0x2C00 ~ 0x2C1D；显示控制寄存器的地址为 0x2C1E；SEG 开关控制寄存器的地址分别为 0x2C1F~0x2C23。控制 LCD 驱动电压的控制位位于 LCD 驱动电压控制寄存器（CtrlLCDV，0x285E）。

用户可通过配置显示控制寄存器（LCDCtrl，0x2C1E）和时钟切换控制寄存器（SysCtrl SFR，0x80）关闭 LCD 扫描时序电路以及驱动电路，使其进入低功耗状态。

表 12-1 显示控制寄存器（LCDCtrl，0x2C1E）

位		默认值	说明
Bit7	ON/OFF	0	开启/关闭 LCD 模拟驱动电路。 1, 开启，模拟驱动电路向 COM 口和 SEG 口输出相应的信号，否则输出高阻。
Bit[6:5]	保留	0	
Bit4	LCDDTYPE	0	LCD 模式选择。 0, 4COM 模式；1, 6COM 模式；
Bit[3:2]	DRV1/DRV0	0	设置每个分压电阻的阻值。 00, 300kΩ；01, 600kΩ；10, 150kΩ；11, 200kΩ
Bit[1:0]	FRQ1/FRQ0	0	设置扫描频率。 11, 512Hz；10, 256Hz；01, 128Hz；00, 64Hz；

表 12-2 SEG 开关控制寄存器 (R/W)

寄存器		bit7	bit6	bit5	Bit4	bit3	bit2	bit1	bit0
0x2C1F	SegCtrl0	-	-	-	-	-	-	SEGON1	SEGON0
0x2C20	SegCtrl1	SEGON15	SEGON14	SEGON13	SEGON12	SEGON11	SEGON10	SEGON9	SEGON8
0x2C21	SegCtrl2	SEGON23	SEGON22	SEGON21	SEGON20	SEGON19	SEGON18	-	-
0x2C22	SegCtrl3	SEGON31	SEGON30	-	-	-	-	-	-
0x2C23	SegCtrl4	SEGON39	SEGON38	-	-	SEGON35	SEGON34	SEGON33	SEGON32
默认值		0	0	0	0	0	0	0	0

0, 关闭 SEG 输出; 1, 开启 SEG 输出。关闭此 SEG 输出后, 此 SEG 所占用的引脚可以作为 GPIO 使用 (“GPIO” 中 “P4 ~ P9”)。

仅当 LCDTYPE=0 (bit4, LCDCtrl, 0x2C1E) 时, SegCtrl0 中的 bit1 和 bit0 有效。

更新显示缓冲寄存器即可完成显示数据更新，且 LCD 显示更新在更新显示缓冲寄存器后实时完成。

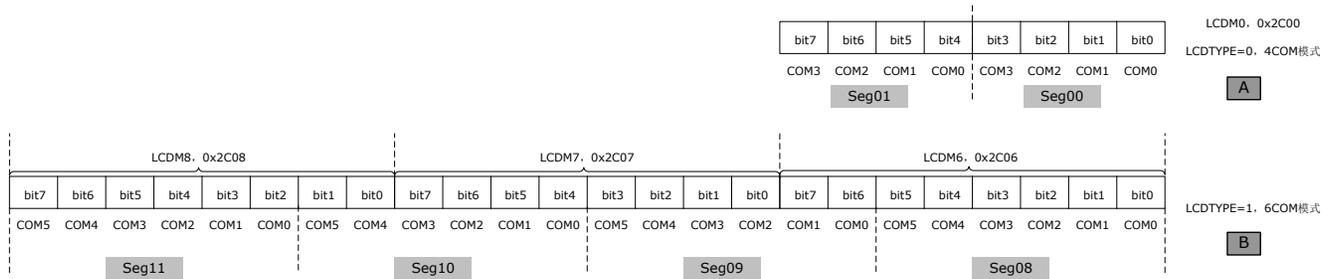


图 12-1 不同模式下，LCD 显示缓冲寄存器的工作原理

当 LCDTYPE=0 时，两个 Seg 对应显示缓冲寄存器中的一个字节。显示缓冲寄存器字节的低 4 位对应 Seg (n)，显示缓冲寄存器字节的高 4 位对应 Seg (n+1)，如图 12-1 中 A 所示。

表 12-3 当 LCDTYPE=0 时，SEG 与显示缓冲寄存器的对应关系

地址	显示缓冲寄存器	SEG	D7	D6	D5	D4	D3	D2	D1	D0
			COM3	COM2	COM1	COM0	COM3	COM2	COM1	COM0
0x2C00	LCDM0	S01 S00	SEG01				SEG00			
0x2C04	LCDM4	S09 S08	SEG09				SEG08			
0x2C05	LCDM5	S11 S10	SEG11				SEG10			
0x2C06	LCDM6	S13 S12	SEG13				SEG12			
0x2C07	LCDM7	S15 S14	SEG15				SEG14			
0x2C09	LCDM9	S19 S18	SEG19				SEG18			
0x2C0A	LCDM10	S21 S20	SEG21				SEG20			
0x2C0B	LCDM11	S23 S22	SEG23				SEG22			
0x2C0F	LCDM15	S31 S30	SEG31				SEG30			
0x2C10	LCDM16	S33 S32	SEG33				SEG32			
0x2C11	LCDM17	S35 S34	SEG35				SEG34			
0x2C13	LCDM19	S39 S38	SEG39				SEG38			

当 LCDTYPE=1 时，因 Pin32 和 Pin33 复用为普通 GPIO 口和 COM 口，所以，LCDM0(0x2C00)和 LCDM1 (0x2C01) 的 bit3~bit0 无效。如图 12-1 中 B 所示，显示缓冲寄存器中 6 个 bit 对应一个 Seg，如，Seg08 对应的是寄存器 LCDM6 (0x2C06) 的 bit5~bit0，Seg09 对应的是寄存器 LCDM6 (0x2C06) 的 bit7~bit6 和 LCDM7 (0x2C07) 的 bit3~bit0，依次类推，如表 12-4 所示。因为系统仅能访问字节，所以访问 Seg09 时，系统需要同时访问 LCDM6 和 LCDM7，依次类推。

表 12-4 当 LCDTYPE=1 时，SEG 与显示缓冲寄存器的对应关系

地址	显示缓冲寄存器	SEG	D7	D6	D5	D4	D3	D2	D1	D0	
0x2C06	LCDM6	S09 S08	SEG09		SEG08						
0x2C07	LCDM7	S10 S09	SEG10						SEG09		
0x2C08	LCDM8	S11 S10	SEG11							SEG10	

地址	显示缓冲寄存器	SEG	D7	D6	D5	D4	D3	D2	D1	D0
0x2C09	LCDM9	S13 S12	SEG13		SEG12					
0x2C0A	LCDM10	S14 S13	SEG14				SEG13			
0x2C0B	LCDM11	S15 S14	SEG15						SEG14	
0x2C0D	LCDM13	S18 --	SEG18				--			
0x2C0E	LCDM14	S19 S18	SEG19						SEG18	
0x2C0F	LCDM15	S21 S20	SEG21		SEG20					
0x2C10	LCDM16	S22 S21	SEG22				SEG21			
0x2C11	LCDM17	S23 S22	SEG23						SEG22	
0x2C16	LCDM22	S30 --	SEG30				--			
0x2C17	LCDM23	S31 S30	SEG31						SEG30	
0x2C18	LCDM24	S33 S32	SEG33		SEG32					
0x2C19	LCDM25	S34 S33	SEG34				SEG33			
0x2C1A	LCDM26	S35 S34	SEG35						SEG34	
0x2C1C	LCDM28	S38 --	SEG38				--			
0x2C1D	LCDM29	S39 S38	SEG39						SEG38	

12.2 LCD 驱动电压

V9811 支持 2.7V~3.3V 共 7 档(0.1V/lsb)驱动电压。用户可通过配置 LCD 驱动电压控制寄存器(CtrlLCDV, 0x285E) 的 bit2~bit0 调整 LCD 的驱动电压。

当 V9811 由 5V 主电源供电时(引脚 V3V 上的电平为低), LDO33 输出电压为 3.3V, 此时 LCD 的驱动电压为配置值。当 V9811 由 3.3V 主电源(引脚 V3V 上的电平为高)或电池供电时, 实际的 LCD 驱动电压为上述配置值与系数 K 的乘积, K 为 LDO33 输出电压与 3.3V 的比值。

发生 POR 复位、RSTn 低电平有效或 WDT 溢出复位时, 驱动电压默认为 3.3V。

表 12-5 LCD 驱动电压配置

寄存器	位	默认值	说明
CtrlLCDV 0x285E	bit[2:0] LCDV<2:0>	0	LCD 驱动电压调整 000, 3.3V; 001, 3.2V; 010, 3.1V; 011, 3.0V; 100, 2.9V; 101, 2.8V; 110, 2.7V; 111, 2.7V

第13章 GPIO

V9811 共实现了 P0~P9 共 10 个 GPIO 口，其中，P0~P8 为普通 GPIO 口，P9 为可快速访问的 GPIO 口。

发生 POR 复位、RSTn 低电平有效或 WDT 溢出复位时，IO 控制寄存器被复位。复位后，如果程序不改变 IO 控制寄存器的配置，所有 IO 口处于禁止输出/屏蔽输入的状态（调试模式下，P0 作 JTAG 端口，复位后，P0.0 为输出，P0.1 ~ P0.3 为输入）。系统进入浅睡眠或者深睡眠状态后，IO 口保持此前的状态不变。

除了由 MODE1 引脚确定的功能外（在 MODE1 引脚为低电平时，P0 端口复用为 JTAG 端口，P0.0 的作用是 TDO，固定为输出；P0.1 的作用是 TDI，固定为输入；P0.2 的作用是 TMS，固定为输入；P0.3 的作用是 TCK，固定为输入），每个 IO 口的输入/输出使能分别由各自的输入/输出使能寄存器独立控制，IO 口的功能则由相对应的功能选择寄存器控制。其中，P0 和调试端口复用；P1、P2 和 P9 与功能端口复用；P3 和 COM 口复用；P4 ~ P9 和 SEG 口复用。

13.1 P0

P0 是 GPIO 与调试端口复用，共 4 个 IO。当 MODE1 引脚接低电平时，P0 作调试端口用。此时，P0.0 的作用是 TDO，P0.1 的作用是 TDI，P0.2 的作用是 TMS，P0.3 的作用是 TCK，其中，TDO 固定为输出，TDI/TMS/TCK 固定为输入。当 MODE1 引脚接高电平时，芯片处于正常工作模式下，P0 输入使能寄存器和输出使能寄存器决定 P0 的每一个 IO 口的输入与输出。

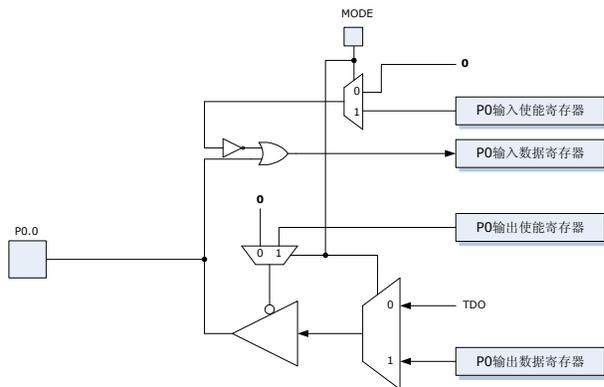


图 13-1 P0.0 端口的结构框图

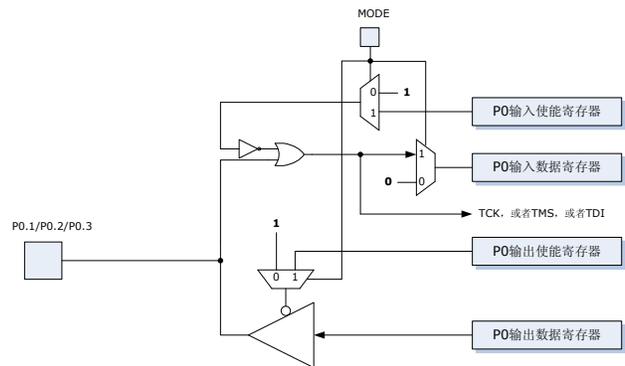


图 13-2 P0.1/P0.2/P0.3 端口的结构框图

表 13-1 P0 输出使能寄存器（P0OE，0x28A8）

0x28A8, R/W, P0 输出使能寄存器, P0OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	P03OEN	P02OEN	P01OEN	P00OEN
默认值	0	0	0	0	1	1	1	1

1, 禁止输出；0, 允许输出

表 13-2 P0 输入使能寄存器 (P0IE, 0x28A9)

0x28A9, R/W, P0 输入使能寄存器, P0IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	P03INEN	P02INEN	P01INEN	P00INEN
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-3 P0 输出数据寄存器 (P0OD, 0x28AA)

0x28AA, R/W, P0 输出数据寄存器, P0OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-				
默认值	0	0	0	0	1	1	1	1

表 13-4 P0 输入数据寄存器 (P0ID, 0x28AB)

0x28AB, R/W, P0 输入数据寄存器, P0ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-				
默认值	0	0	0	0	0	0	0	0

13.2 P1

P1 是 GPIO 与特殊功能端口复用, 共 4 个 IO。在正常工作模式下, P1 输入和输出使能寄存器决定 P1 的每一个 IO 的输入与输出。当 P1 的各个 IO 作特殊功能用时, 其功能由相对应的特殊功能选择寄存器决定。

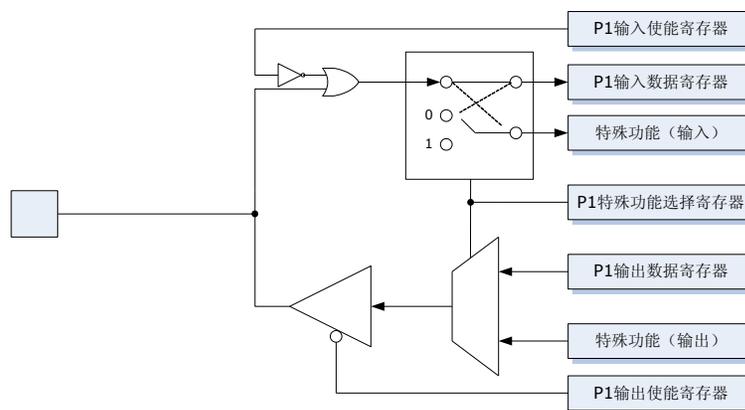


图 13-3 P1 端口的结构框图

表 13-5 P1 输出使能寄存器 (P1OE, 0x28AC)

0x28AC, R/W, P1 输出使能寄存器, P1OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	P14OEN	P13OEN	P12OEN	P11OEN	-
默认值	0	0	0	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-6 P1 输入使能寄存器 (P1IE, 0x28AD)

0x28AD, R/W, P1 输入使能寄存器, P1IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	P14INEN	P13INEN	P12INEN	P11INEN	-
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-7 P1 输出数据寄存器 (P1OD, 0x28AE)

0x28AE, R/W, P1 输出数据寄存器, P1OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-					
默认值	0	0	0	1	1	1	1	1

表 13-8 P1 输入数据寄存器 (P1ID, 0x28AF)

0x28AF, R/W, P1 输入数据寄存器, P1ID								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-					
默认值	0	0	0	0	0	0	0	0

表 13-9 P1.1 功能选择寄存器 (P11FS, 0x28C5, R/W)

位		默认值	说明
Bit[7:3]	保留	0	
Bit[2:0]	P11FNC2/P11FNC1/P11FNC0	0	000, GPIO 010, RXD1, UART1 接收数据输入 011, T1, Timer1 外部输入 100, 端口中断 2

表 13-10 P1.2 功能选择寄存器 (P12FS, 0x28C6, R/W)

位		默认值	说明
Bit[7:3]	保留	0	
Bit[2:0]	P12FNC2/P12FNC1/P12FNC0	0	000, GPIO 001, MEACLK, 电能计量时钟输出 010, TXD1, UART1 发送数据输出 011, T2EX, Timer2 捕获/重载触发信号输入 100, 端口中断 3

表 13-11 P1.3 功能选择寄存器 (P13FS, 0x28C7, R/W)

位		默认值	说明
Bit[7:3]	保留	0	
Bit[2:0]	P13FNC2/P13FNC1/P13FNC0	0	000, GPIO 001, CF2 脉冲输出 010, RXD5, UART5 接收数据输入 011, 端口中断 0 100, CF1 脉冲输出 101, SP, RTC 秒脉冲输出, 每 30s 输出一个脉冲 110, PLLDIV, 高频分频后的系统时钟输出, 可用于高频秒脉冲输出

表 13-12 P1.4 功能选择寄存器 (P14FS, 0x28C8)

位		默认值	说明
Bit[7:2]	保留	0	
Bit[1:0]	P14FNC1/P14FNC0	0	000, GPIO 001, PLLDIV, 高频分频后的系统时钟输出, 可用于高频秒脉冲输出 010, TXD5, UART5 发送数据输出 011, 端口中断 1

13.3 P2

P2 也是 GPIO 与特殊功能端口复用, 共 4 个 IO。在正常工作模式下, P2 输入和输出使能寄存器决定 P2 的每一个 IO 的输入与输出。当 P2 的各个 IO 作特殊功能用时, 其功能由相对应的特殊功能选择寄存器决定。

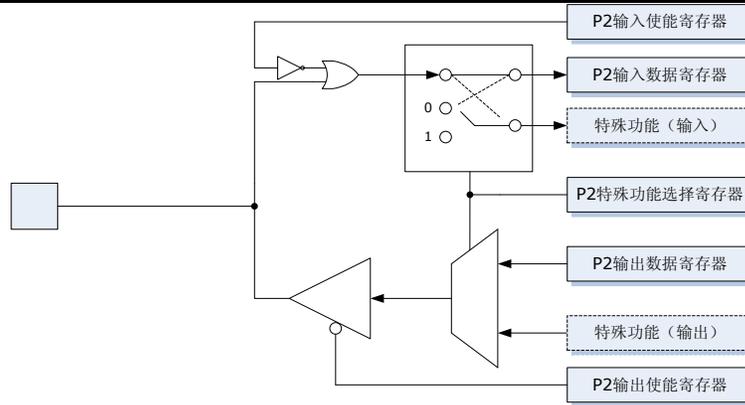


图 13-4 P2 端口的结构框图

表 13-13 P2 输出使能寄存器 (P2OE, 0x28B0)

0x28B0, R/W, P2 输出使能寄存器, P2OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	P25OEN	P24OEN	-	-	P21OEN	P20OEN
默认值	0	0	1	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-14 P2 输入使能寄存器 (P2IE, 0x28B1)

0x28B1, R/W, P2 输入使能寄存器, P2IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	P25INEN	P24INEN			P21INEN	P20INEN
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-15 P2 输出数据寄存器 (P2OD, 0x28B2)

0x28B2, R/W, P2 输出数据寄存器, P2OD								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-						
默认值	0	0	1	1	0	0	1	1

表 13-16 P2 输入数据寄存器 (P2ID, 0x28B3)

0x28B3, R/W, P2 输入数据寄存器, P2ID								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-						
默认值	0	0	0	0	0	0	0	0

表 13-16 P2.0 功能选择寄存器 (P20FS, 0x28C9, R/W)

位		默认值	说明
Bit[7:2]	保留	0	
Bit[1:0]	P20FNC1/P20FNC0	0	000, GPIO 001, OSC, OSC 时钟输出 010, RXD4, UART4 接收数据输入 011, T2, Timer2 外部输入

表 13-17 P2.1 功能选择寄存器 (P21FS, 0x28CA, R/W)

位		默认值	说明
Bit[7:2]	保留	0	
Bit[1:0]	P21FNC1/P21FNC0	0	000, GPIO 001, PLL, PLL 时钟输出 010, TXD4, UART4 发送数据输出 011, T0, Timer0 外部输入

表 13-18 P2.4 功能选择寄存器 (P24FS, 0x28CD, R/W)

位		默认值	说明
Bit[7:2]	保留	0	
Bit[1:0]	P24FNC1/P24FNC0	0	000, GPIO 010, RXD2, UART2 接收数据输入

表 13-19 P2.5 功能选择寄存器 (P25FS, 0x28CE, R/W)

位		默认值	说明
Bit[7:2]	保留	0	
Bit[1:0]	P25FNC1/P25FNC0	0	000, GPIO 010, TXD2, UART2 发送数据输出, 可以选择发送带有 38kHz 载波的信号

13.4 P3

P3 为 GPIO 与 COM 口复用, 共 4 个 IO。当 P3 作 COM 口用时, P3 输出使能寄存器必须禁止输出, 同时屏蔽输入。当 P3 作 GPIO 用时, 必须关闭全部 LCD 模拟驱动电路 (Bit7, LCDCtrl, 0x2C1E)。

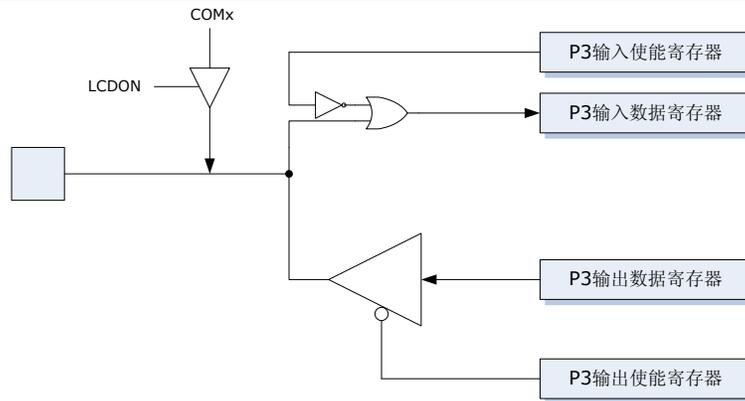


图 13-5 P3 端口的结构框图

表 13-20 P3 输出使能寄存器 (P3OE, 0x28B4)

0x28B4, R/W, P3 输出使能寄存器, P3OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	P33OEN	P32OEN	P31OEN	P30OEN
默认值	0	0	0	0	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-21 P3 输入使能寄存器 (P3IE, 0x28B5)

0x28B5, R/W, P3 输入使能寄存器, P3IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-	P33INEN	P32INEN	P31INEN	P30INEN
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-22 P3 输出数据寄存器 (P3OD, 0x28B6)

0x28B6, R/W, P3 输出数据寄存器, P3OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-				
默认值	0	0	0	0	1	1	1	1

表 13-23 P3 输入数据寄存器 (P3ID, 0x28B7)

0x28B7, R/W, P3 输入数据寄存器, P3ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
	-	-	-	-				
默认值	0	0	0	0	0	0	0	0

13.5 P4

P4 共 2 个 IO 口。当 LCDTYPE=0 时，P4.0 与 P4.1 为 GPIO 与 SEG 口复用；当 LCDTYPE=1 时，P4.0 和 P4.1 为 GPIO 与 COM 口复用。当 P4 作 SEG 口或 COM 口用时，P4 输出使能寄存器必须禁止输出，同时屏蔽输入。当 P4 作 GPIO 用时，必须关闭相应 SEG 引脚的 SEG 输出功能。

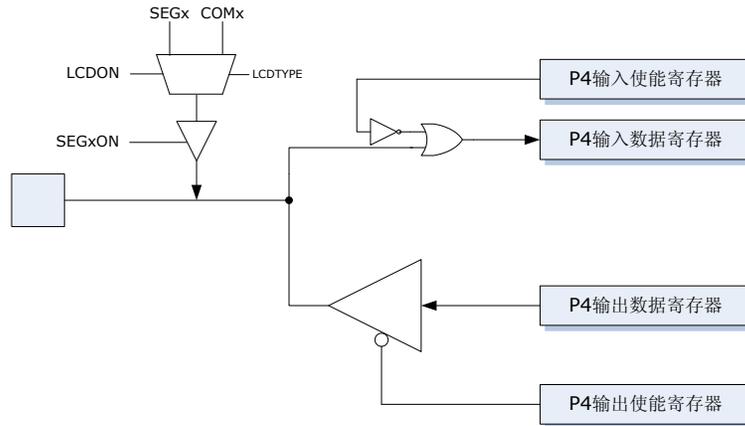


图 13-6 P4 端口的结构框图

表 13-24 P4 输出使能寄存器（P4OE，0x28B8）

0x28B8, R/W, P4 输出使能寄存器, P4OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
							P41OEN	P40OEN
默认值	1	1	1	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-25 P4 输入使能寄存器（P4IE，0x28B9）

0x28B9, R/W, P4 输入使能寄存器, P4IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
							P41INEN	P40INEN
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-26 P4 输出数据寄存器（P4OD，0x28BA）

0x28BA, R/W, P4 输出数据寄存器, P4OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	1	1	1	1	1	1	1	1

表 13-27 P4 输入数据寄存器 (P4ID, 0x28BB)

0x28BB, R/W, P4 输入数据寄存器, P4ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	0	0	0	0	0	0	0	0

13.6 P5

P5 为 GPIO 与 SEG 口复用，共 8 个 IO。当 P5 作 SEG 口用时，P5 输出使能寄存器必须禁止输出，同时屏蔽输入。当 P5 作 GPIO 用时，必须关闭相应 SEG 引脚的 SEG 输出功能。

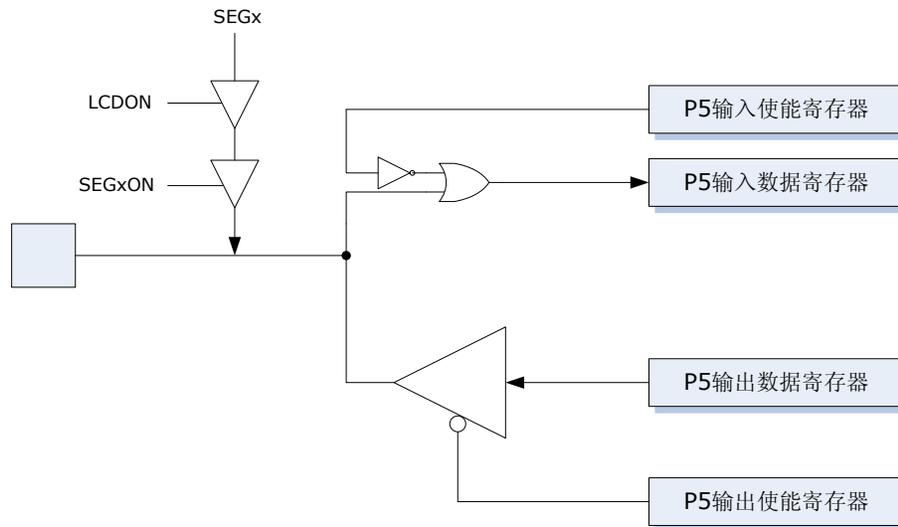


图 13-7 P5 端口的结构框图

表 13-28 P5 输出使能寄存器 (P5OE, 0x28BC)

0x28BC, R/W, P5 输出使能寄存器, P5OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P57OEN	P56OEN	P55OEN	P54OEN	P53OEN	P52OEN	P51OEN	P50OEN
默认值	1	1	1	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-29 P5 输入使能寄存器 (P5IE, 0x28BD)

0x28BD, R/W, P5 输入使能寄存器, P5IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P57INEN	P56INEN	P55INEN	P54INEN	P53INEN	P52INEN	P51INEN	P50INEN
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-30 P5 输出数据寄存器 (P5OD, 0x28BE)

0x28BE, R/W, P5 输出数据寄存器, P5OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	1	1	1	1	1	1	1	1

表 13-31 P5 输入数据寄存器 (P5ID, 0x28BF)

0x28BF, R/W, P5 输入数据寄存器, P5ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	0	0	0	0	0	0	0	0

13.7 P6

P6 为 GPIO 与 SEG 口复用，共 6 个 IO。当 P6 作 SEG 口用时，P6 输出使能寄存器必须禁止输出，同时屏蔽输入。当 P6 作 GPIO 用时，必须关闭相应 SEG 引脚的 SEG 输出功能。

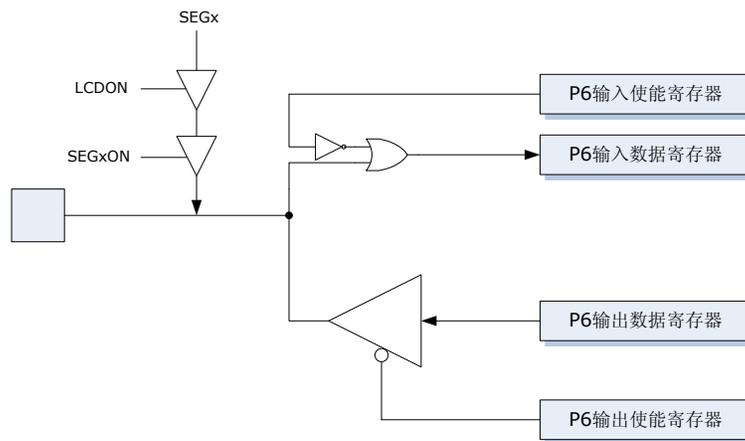


图 13-8 P6 端口的结构框图

表 13-32 P6 输出使能寄存器 (P6OE, 0x28C0)

0x28C0, R/W, P6 输出使能寄存器, P6OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P67OEN	P66OEN	P65OEN	P64OEN	P63OEN	P62OEN		
默认值	1	1	1	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-33 P6 输入使能寄存器 (P6IE, 0x28C1)

0x28C1, R/W, P6 输入使能寄存器, P6IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P67INEN	P66INEN	P65INEN	P64INEN	P63INEN	P62INEN		
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-34 P6 输出数据寄存器 (P6OD, 0x28C2)

0x28C2, R/W, P6 输出数据寄存器, P6OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	1	1	1	1	1	1	1	1

表 13-35 P6 输入数据寄存器 (P6ID, 0x28C3)

0x28C3, R/W, P6 输入数据寄存器, P6ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	0	0	0	0	0	0	0	0

13.8 P7

P7 为 GPIO 与 SEG 口复用, 共 2 个 IO。当 P7 作 SEG 口用时, P7 输出使能寄存器必须禁止输出, 同时屏蔽输入。当 P7 作 GPIO 用时, 必须关闭相应 SEG 引脚的 SEG 输出功能。

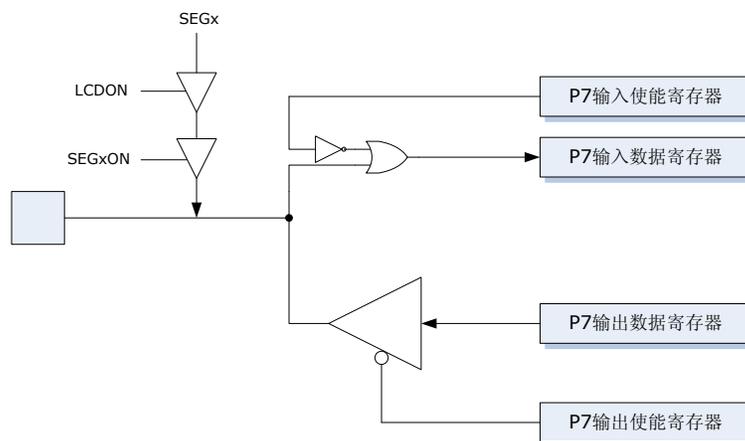


图 13-9 P7 端口的结构框图

表 13-36 P7 输出使能寄存器 (P7OE, 0x28D5)

0x28D5, R/W, P7 输出使能寄存器, P7OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P77OEN	P76OEN						
默认值	1	1	1	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-37 P7 输入使能寄存器 (P7IE, 0x28D6)

0x28D6, R/W, P7 输入使能寄存器, P7IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P77INEN	P76INEN						
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-38 P7 输出数据寄存器 (P7OD, 0x28D7)

0x28D7, R/W, P7 输出数据寄存器, P7OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	1	1	1	1	1	1	1	1

表 13-39 P7 输入数据寄存器 (P7ID, 0x28D8)

0x28D8, R/W, P7 输入数据寄存器, P7ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	0	0	0	0	0	0	0	0

13.9 P8

P8 为 GPIO 与 SEG 口复用, 共 3 个 IO。当 P8 作 SEG 口用时, P8 输出使能寄存器必须禁止输出, 同时屏蔽输入。当 P8 作 GPIO 用时, 必须关闭相应 SEG 引脚的 SEG 输出功能。

表 13 4113 41 P8 输出使能寄存器 (P8OE, 0x28D9)

0x28D9, R/W, P8 输出使能寄存器, P8OE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
						P82OEN	P81OEN	P80OEN
默认值						1	1	1

1, 禁止输出; 0, 允许输出

表 13-40 P8 输入使能寄存器 (P8IE, 0x28DA)

0x28DA, R/W, P8 输入使能寄存器, P8IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
						P82INEN	P81INEN	P80INEN
默认值						0	0	0

1, 允许输入; 0, 禁止输入

表 13-41 P8 输出数据寄存器 (P8OD, 0x28DB)

0x28DB, R/W, P8 输出数据寄存器, P8OD								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	-	-	-	-	-	1	1	1

表 13-42 P8 输入数据寄存器 (P8ID, 0x28DC)

0x28DC, R/W, P8 输入数据寄存器, P8ID								
	bit7	Bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	-	-	-	-	-	0	0	0

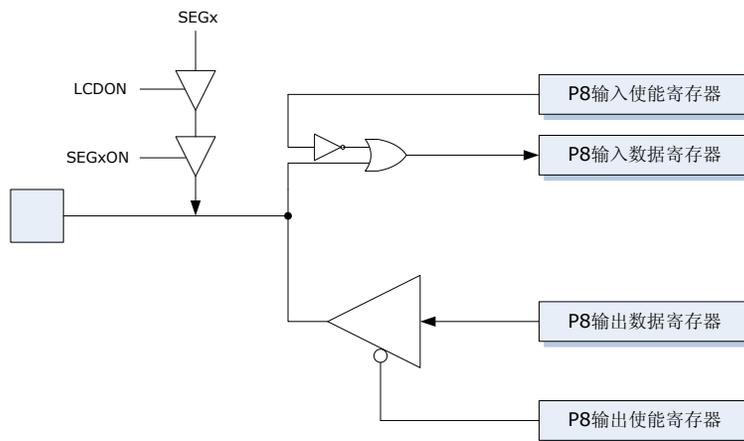


图 13-10 P8 端口的结构框图

13.10 P9

P9 为可快速访问的 GPIO，与特殊功能端口复用，共 6 个 IO 口，其中，P9.0 能用于 SEG 输出。

在正常工作模式下，P9 输入和输出使能寄存器决定 P9 的每一个 IO 的输入与输出。

当 P9 的各个 IO 口作快速 GPIO 口时，当 $f_{MCU}=13.1072MHz$ 时，通信速率约为 200kbps。

当 P9 的各个 IO 作特殊功能用时，其功能由相对应的特殊功能选择位决定。当 P9.0~P9.2 作 TimerA 比较/捕获模块 0/1/2 输入时，这三个端口输入使能。另外，P9.1 可用于标准秒信号输入，使用此功能时，P9.1 输入使能。

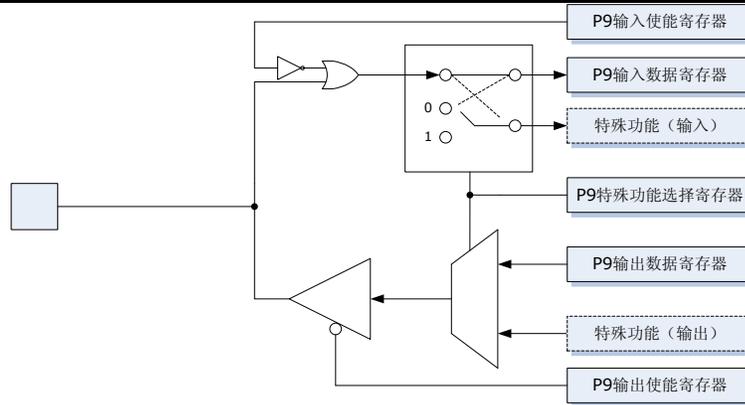


图 13-11 P9 端口的结构框图

表 13-43 P9 输出使能寄存器（P9OE SFR，0xA4）

0xA4, R/W, P9 输出使能寄存器, P9OE SFR								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
		P96OEN	P95OEN		P93OEN	P92OEN	P91OEN	P90OEN
默认值	1	1	1	1	1	1	1	1

1, 禁止输出; 0, 允许输出

表 13-44 P9 输入使能寄存器（P9IE SFR，0xA5）

0xA5, R/W, P9 输入使能寄存器, P9IE								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
		P96INEN	P95INEN		P93INEN	P92INEN	P91INEN	P90INEN
默认值	0	0	0	0	0	0	0	0

1, 允许输入; 0, 禁止输入

表 13-45 P9 输出数据寄存器（P9OD SFR，0xA6）

0xA6, R/W, P9 输出数据寄存器, P9OD SFR								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	1	1	1	1	1	1	1	1

表 13-46 P9 输入数据寄存器（P9ID SFR，0xA7）

0xA7, R/W, P9 输入数据寄存器, P9ID SFR								
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
默认值	0	0	0	0	0	0	0	0

表 13-47 P9 功能选择寄存器 (P9FS SFR, 0xAD)

0xAD, R/W, P9 功能选择寄存器, P9FS SFR		
位		说明
Bit7	保留	
Bit6	P96FNC	P9.6 功能配置。1, CF1 脉冲输出; 0, 快速 IO 口
Bit5	P95FNC	P9.5 功能配置。1, CF2 脉冲输出; 0, 快速 IO 口
Bit4	保留	
Bit3	P93FNC	P9.3 功能配置。1, PLLDIV, 高频分频后的系统时钟输出, 可用于高频秒脉冲输出; 0, 快速 IO 口
Bit2	P92FNC	P9.2 功能配置。1, TA2, TimerA 比较/捕获模块 2 输入/输出, 输入使能; 0, 快速 IO 口
Bit1	P91FNC	P9.1 功能配置。1, TA1, TimerA 比较/捕获模块 1 输入/输出, 输入使能; 0, 快速 IO 口
Bit0	P90FNC	P9.0 功能配置。1, TA0, TimerA 比较/捕获模块 0 输入/输出, 输入使能; 0, 快速 IO 口

第14章 WDT

WDT 使用独立的 32kHz 的 RC 时钟。

发生 POR 复位、RSTn 低电平有效时，WDT 计数被清除。复位结束后，WDT 自动开始运行，1.5 秒之后，如果软件没有清除 WDT 计数，则产生一次复位脉冲，此后，如果软件依然没有清空 WDT 计数，则每 2 秒产生一次复位脉冲。复位持续时间保持 240 μ s。系统进入浅睡眠或者深睡眠状态后，WDT 自动停止工作，IO/RTC 休眠唤醒或恢复供电后，WDT 自动重新开始运行。

WDT 溢出复位会在系统状态寄存器（Systate SFR，0xA1）的 Bit5 置标志位，供程序查询。发生恢复供电复位时，此标志位会被清零。

在调试模式下（MODE1 引脚接低电平），WDT 溢出复位被屏蔽，且不会在 Systate SFR 中置标志位。

程序连续向 WDTEN SFR（0xCE）和 WDTCLR SFR（0xCF）写入 0xA5 和 0x5A，可以清除 WDT 的计数值，使其不产生溢出复位。

0xCE, W, WDTEN SFR	0xA5
0xCF, W, WDTCLR SFR	0x5A

第15章 RTC

V9811 集成的 RTC 具有如下特点：

- 使用独立的 32.768kHz 的 OSC 时钟；
- 提供实时时钟和日历，而且，能自动对闰年闰月及各大小月进行判断，并对其进行自动调整；
- 内部具有校正寄存器，可以根据配置的校正值自动进行计时校正；
- 提供实时校准的秒脉冲输出，同时向 MCU 提起秒中断。

当系统进入浅睡眠状态的时候，RTC 依然可以正常工作，并可以提供日/时/分/秒/500ms/250ms/125ms/62.5ms 间隔的唤醒复位。

15.1 RTC 相关寄存器

RTC 相关寄存器中，仅 RTC 计时寄存器（RTC 时间设置寄存器）和 RTC 校正寄存器不受任何复位影响，其它寄存器均可在发生 POR 复位、RSTn 低电平有效或 WDT 溢出复位时被复位。

表 15-1 RTC 密码使能寄存器（RTCPEN SFR, 0x90）

0x90, RTC 密码使能寄存器, RTCPEN SFR			
位	默认值	R/W	说明
bit[7:0]	0	W	用户必须先向寄存器 RTCPEN SFR 写入 0x96, 才能对寄存器 RTCPWD SFR 进行有效的写操作, 对这两个寄存器的访问必须为连续写操作。对寄存器 RTCPEN SFR 而言, 仅 0x96 为有效数据。

表 15-2 RTC 密码寄存器（RTCPWD SFR, 0x97）

0x97, RTC 密码寄存器, RTCPWD SFR			
位	默认值	R/W	说明
bit[7:1]	0	W	当且仅当 bit[7:1] 被写入 0b0101011 时, bit0 才能被有效写入。
bit0	WE 0	R/W	1, 使能对 INTRTC SFR、RTC 校正寄存器和 RTC 时间设置寄存器的写入操作

表 15-3 系统休眠唤醒间隔寄存器（INTRTC SFR, 0x96）

0x96, 系统休眠唤醒间隔寄存器, INTRTC SFR			
位	默认值	R/W	说明
bit[7:3]	0	R/W	
bit[2:0]	RTC<2:0>	R/W	000, 1 秒; 001, 1 分; 010, 1 小时; 011, 1 天; 100, 500 毫秒; 101, 250 毫秒; 110, 125 毫秒; 111, 62.5 毫秒

表 15-4 RTC 校正寄存器 (RTCCH SFR/RTCCL SFR, 0x94/0x95)

寄存器	位		默认值	R/W	说明
0x94, RTCCH SFR	bit[7:6]	保留	0	R/W	RTC 校正值, 补码。
	bit[5:0]	C<13:8>	0	R/W	
0x95, RTCCL SFR	bit[7:0]	C<7:0>	0	R/W	RTC 内部计时器以 OSC 时钟频率进行累加, 在 1~29 秒, 每累加 32768 次产生一个秒脉冲。在第 30 秒, 累加(32767+C<13:0>)次产生一个秒脉冲。因此, RTC 内部计时器每 30 秒进行一次秒脉冲调整, 平均到 30 秒内, 秒脉冲校正分辨率为 1.02ppm, 调整范围为±8332.3ppm (±12 分钟/天)。

表 15-5 RTC 数据读取寄存器 (RDRTC SFR, 0xDA)

0xDA, RTC 数据读取寄存器, RDRTC SFR			
位	默认值	R/W	说明
bit[7:0]	0	R	当 MCU 要读取 RTC 时间寄存器的值时, 必须先对该寄存器进行读操作, 详见“读 RTC 时间数据”。该寄存器的读出值为 00, 无意义。

表 15-6 高频分频门限值寄存器 (DIVTHH SFR/DIVTHM SFR/DIVTHL SFR, 0xDB/0xDC/0xDD)

寄存器	位		默认值	R/W	说明
0xDB, DIVTHH SFR	bit[7:0]	DIV<23:16>	0	R/W	高频分频门限值寄存器高位字节
0xDC, DIVTHM SFR	bit[7:0]	DIV<15:8>	0	R/W	高频分频门限值寄存器中位字节
0xDD, DIVTHL SFR	bit[7:0]	DIV<7:0>	0	R/W	高频分频门限值寄存器低位字节

表 15-7 高频计数器状态寄存器 (PLLCNT SFR, 0xDE)

0xDE, 高频计数器状态寄存器, PLLCNT SFR				
位	默认值	R/W	说明	
bit[7:2]		0	R/W	
bit[1:0]	STT<1:0>	0	R/W	<p>0x00, 分频器。从 0 计数到高频分频门限值寄存器的值后, 高频计数器清零并重新开始计数。每次清零都会输出脉冲 (可作高频分频时钟输出)。</p> <p>0x01, 清零计数器。当检测到输入的标准秒脉冲信号的上升沿后, 该寄存器的值自动切换为 0x02, 即, 该高频计数器作计数器用。</p> <p>0x02, 当该寄存器的值自动切换为 0x02 后, 高频计数器从 0 开始计数。</p> <p>0x03, 当再次检测到输入的标准秒脉冲信号的上升沿后, 该寄存器的值自动切换为 0x03, 高频计数器停止计数, 高频计数器当下的计数值被送入高频分频门限值寄存器。</p>

计时寄存器（即，RTC 时间设置寄存器）的值采用 BCD 编码，其中 Bit7~Bit4 表示日期的十位数，Bit3~Bit0 表示日期的个位数，如，在 RTCSC SFR 中设置 43 秒时，寄存器的读数应为 0b01000011。RTC 会自动对闰年闰月及各大小月进行判断。所有计时寄存器均可读可写。

表 15-8 计时寄存器（RTC 时间设置寄存器）

寄存器	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0x9A 秒设置寄存器，RTCSC SFR 0~59 秒	-	S40	S20	S10	S8	S4	S2	S1
0x9B 分设置寄存器，RTCMiC SFR 0~59 分	-	M40	M20	M10	M8	M4	M2	M1
0x9C 时设置寄存器，RTCHC SFR 0~23 时	-	-	H20	H10	H8	H4	H2	H1
0x9D 天设置寄存器，RTCDC SFR	-	-	D20	D10	D8	D4	D2	D1
0x9E 周设置寄存器，RTCWC SFR	-	-	-	-	W8	W4	W2	W1
0x9F 月设置寄存器，RTCMoC SFR 1~12 月	-	-	-	Mo10	Mo8	Mo4	Mo2	Mo1
0x93 年设置寄存器，RTCYC SFR 2000~2099 年	Y80	Y40	Y20	Y10	Y8	Y4	Y2	Y1

“周”只能由用户设置，RTC 不会自动判断，一旦设定，RTC 会自动累加。如，用户应手动将 2010 年 1 月 1 日设置为周五，一旦设定，RTC 会自动显示 2010 年 1 月 2 日为周六。W8/W4/W2/W1: 0000, 周日; 0001, 周一; 0010, 周二; 0011, 周三; 0100, 周四; 0101, 周五; 0110, 周六。

系统默认年份从 2000 开始，故，寄存器 RTCYC SFR 仅需设置年份的个位与十位数字，比如，设置 2010 年时，RTCYC 应设置为 0b00010000。

15.2 读写 RTC

15.2.1 写 RTC

在 V9811 中，对 INTRTC SFR、RTC 校正寄存器和计时寄存器的写操作被保护。

用户应严格按照以下步骤对上述寄存器进行写操作：

1. 在 RTCPEN SFR 中写入 0x96，使能寄存器 RTCPWD SFR 的写入操作；
2. 在 RTCPWD SFR 的 bit[7:1] 写入 0b0101011，再向 bit0 写入 1，使能对 INTRTC SFR、RTC 校正寄存器和计时寄存器的写入操作；
3. 对寄存器 INTRTC SFR/RTC 校正寄存器/计时寄存器进行数据配置；
4. 5 个 OSC 时钟后，在 RTCPWD SFR 的 bit[7:1] 写入 0b0101011，再向 bit0 写入 0，恢复对 INTRTC SFR、RTC 校正寄存器和计时寄存器的写入保护，5 个 OSC 时钟后，MCU 自动将上述寄存器的数据写入 RTC，从而完成 RTC 配置。

15.2.2 读 RTC 数据

15.2.2.1 读 RTC 时间数据

读取 RTC 时间数据时，MCU 应先读寄存器 RDRTC SFR (0xDA)，等待至少 5 个 OSC 时钟后，RTC 时间数据被锁存，MCU 再对所有计时寄存器（时间设置寄存器）进行读操作，从而得到当前的时间数据。

15.2.2.2 读 RTC 校正数据

MCU 可直接读取 RTC 校正寄存器的值。

15.3 RTC 计时

初始上电后，RTC 开始计时，而且，只要芯片供电保持有效，计时不受任何复位的影响。

如果用户未对 RTC 进行时间设置，则 RTC 从随机值进行计时；如果用户对 RTC 进行了时间设置，则 RTC 在设置值的基础上进行计时。

15.4 RTC 中断

在 V9811 中，当中断使能时，RTC 可以向 MCU 提起非法数据中断和 RTC 秒中断。

15.4.1 RTC 非法数据中断

当 RTC 非法数据中断被使能时，即，EA=1，EIE.1=1，且 ExInt4IE.0=1，在以下三种情况下 RTC 会向 MCU 提起非法数据中断：

1. 在没有取消写入保护的情况下（没有在 RTCPWD SFR 中写入 0x57），对 INTRTC SFR、RTC 校正寄存器和计时寄存器进行写操作；
2. 取消了写入保护，但是设置了错误的时间；

在上述两种情况下，计时寄存器中的值会保持不变，即本次时间设置无效（时间值用 BCD 编码表示，对于每一个表示时间的 BCD 位，数据 0xF 不被认为是错误数据）。

3. RTC 在运行过程中因为系统原因导致时间数据错误。在这种情况下，用户在取消对 RTC 时间设置寄存器的写保护后，应一次性完成对所有 RTC 时间设置寄存器的配置，再恢复对这些寄存器的写保护，从而有效地完成对时间数据的更正。分次写入时间数据无效。

15.4.2 RTC 秒中断

当 RTC 秒中断被使能时，即，EA=1，EIE.2=1，且 ExInt3IE.6=1，RTC 会向 MCU 提起秒中断。

15.5 高频秒脉冲校正

V9811 集成了一个 24 位的高频计数器（PLL CNT），其工作模式由高频计数器状态寄存器（PLL CNT SFR）控制，该计数器可作为分频器或计数器使用。

当 PLL CNT SFR 为 00 时，这个 24 位高频计数器作高速分频器用。在这个模式下，PLL CNT 从 0 开始计数，每个时钟周期累加 1。用户可通过高速分频门限值寄存器（DIVTHH/DIVTHM/DIVTHL SFR）预设其分频门限值（TH），当 PLL CNT 计数到预设分频门限值时，PLL CNT 清零，重新开始计数，并从引脚 P1.3(Pin22)、P1.4(Pin21)

或 P9.3 (Pin27) 输出系统分频时钟。该分频时钟可作高频秒脉冲，或高频时钟输出。系统分频时钟的频率由公式 15-1 确定：

$$\text{公式 15-1 } f_{\text{DIV}} = \frac{f_{\text{MCU}}}{2 \times (\text{TH} + 1)}$$

其中， f_{DIV} 为系统分频时钟频率，Hz；

f_{MCU} 为系统时钟频率，Hz；

TH 为预设的分频门限值（即寄存器 DIVTHH/DIVTHM/DIVTHL 的值）。当 TH 为 0 时（即默认情况下），系统时钟被 2 分频，系统时钟最高可被 2^{25} 分频。

输出高频秒脉冲时， $f_{\text{DIV}}=1\text{Hz}$ ，公式 15-1 转变为：

$$\text{公式 15-2 } \text{TH} = \frac{f_{\text{MCU}}}{2} - 1$$

上述计算得到的 TH 值写入寄存器 DIVTHH/DIVTHM/DIVTHL，系统即输出秒脉冲。用户可通过程序计算高频分频门限值寄存器的值，实时分频输出高频秒脉冲。

当引脚 P9.1 (Pin57) 被用作标准秒脉冲输入时，PLLNT 可对此标准输入信号进行计数或捕获，判断自身时钟频率的偏差，从而得出校正数据，对 RTC 进行自动校正，其具体过程如下：

1. 向寄存器 PLLCNT SFR 写入 0x01，将 PLLCNT 清零，PLLCNT 作计数器用；
2. 当 PLLCNT 检测到输入的标准秒脉冲信号的上升沿后，PLLCNT 从 0 开始计数（此时，PLLCNT SFR=0x2）；
3. 直至 PLLCNT 检测到输入的标准秒脉冲信号的上升沿后，PLLCNT 停止计数（此时，PLLCNT SFR=0x03），且，计数器当下的计数值被自动写入高频分频门限值寄存器（TH）。

寄存器 PLLCNT SFR 的读出值从 0x01 到 0x03 时，系统完成了一次脉宽测量，此功能可用于 OSC 时钟校正。

15.6 RTC 校正

使用 V9811 时，用户可通过校正高频秒脉冲或温度补偿两种方法求得 RTC 校正值（C），进行 RTC 校正：

- 当 $C > 0$ 时，调慢时钟，将 $(3\text{FFFH}-C)$ 的值写入 RTC 校正寄存器；
- 当 $C < 0$ 时，调快时钟，将 C 值写入 RTC 校正寄存器。

15.6.1 利用高频秒脉冲校正 RTC

RTC 校正在 32768Hz 的 OSC 时钟频率下进行，每 30 秒校正一次（分别在 29 秒/59 秒进行），RTC 校正寄存器的值每 30 秒更新一次。

OSC 时钟频率与 MCU 时钟频率存在如下关系：

$$\text{公式 15-3 } f_{\text{MCU}} = K \times f_{\text{OSC}}$$

其中，K 为系数，100/200/400，当 MCU 时钟频率为 13.1072MHz 时， $K=400$ 。

假设校正寄存器的值为 C，高频分频门限值校正值为 ΔTH ，实测得到的高频秒脉冲周期为 t，则，当 f_{MCU} 为 13.1072MHz 时，三者之间的关系如下：

公式 15-4
$$\Delta TH = \frac{(\frac{1}{t} - 1) \times 32768 \times 400}{2}$$

公式 15-5
$$C = (\frac{1}{t} - 1) \times 32768 \times 30 + 1$$

公式 15-6
$$\frac{\Delta TH}{C - 1} = \frac{200}{30}$$

所以，根据公式 15-6，用户可在校正高频秒脉冲的同时实时 RTC。

15.6.2 温度补偿校正 RTC

由于温度和离散性的影响，不同芯片正常运行时外接的 32768Hz 晶振的频率在不同的时间会有不同的实际振荡频率。RTC 的校正寄存器可以对 RTC 计时进行校正，减小这种差异对最终实际计时带来的影响。

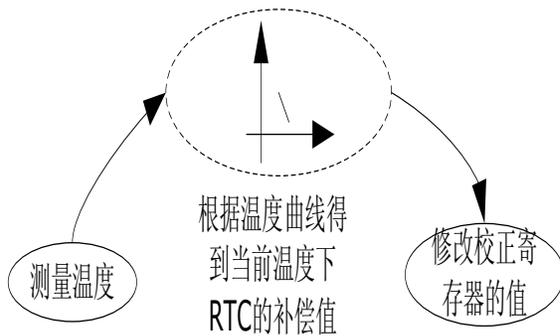


图 15-1 RTC 温度补偿原理

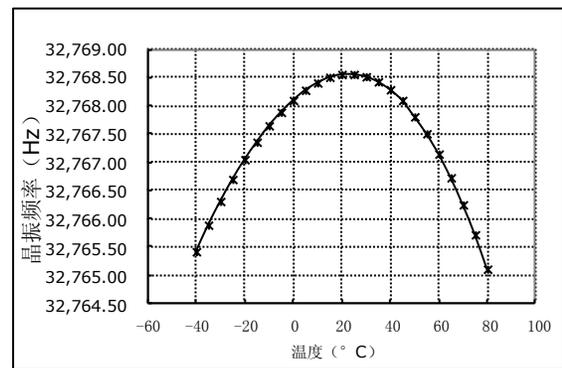


图 15-2 晶振频率与温度的关系

将其中 4 只电能表放入温控箱，通过温控箱从 -40~+80°C 范围内调整，依次用频率计测量引脚 P2.0(Pin62) 输出的晶振频率。通过记录可以看出成抛物线关系，不同表之间系数偏差很小，仅上下偏移，即常数项有偏差（见图 15-2）。

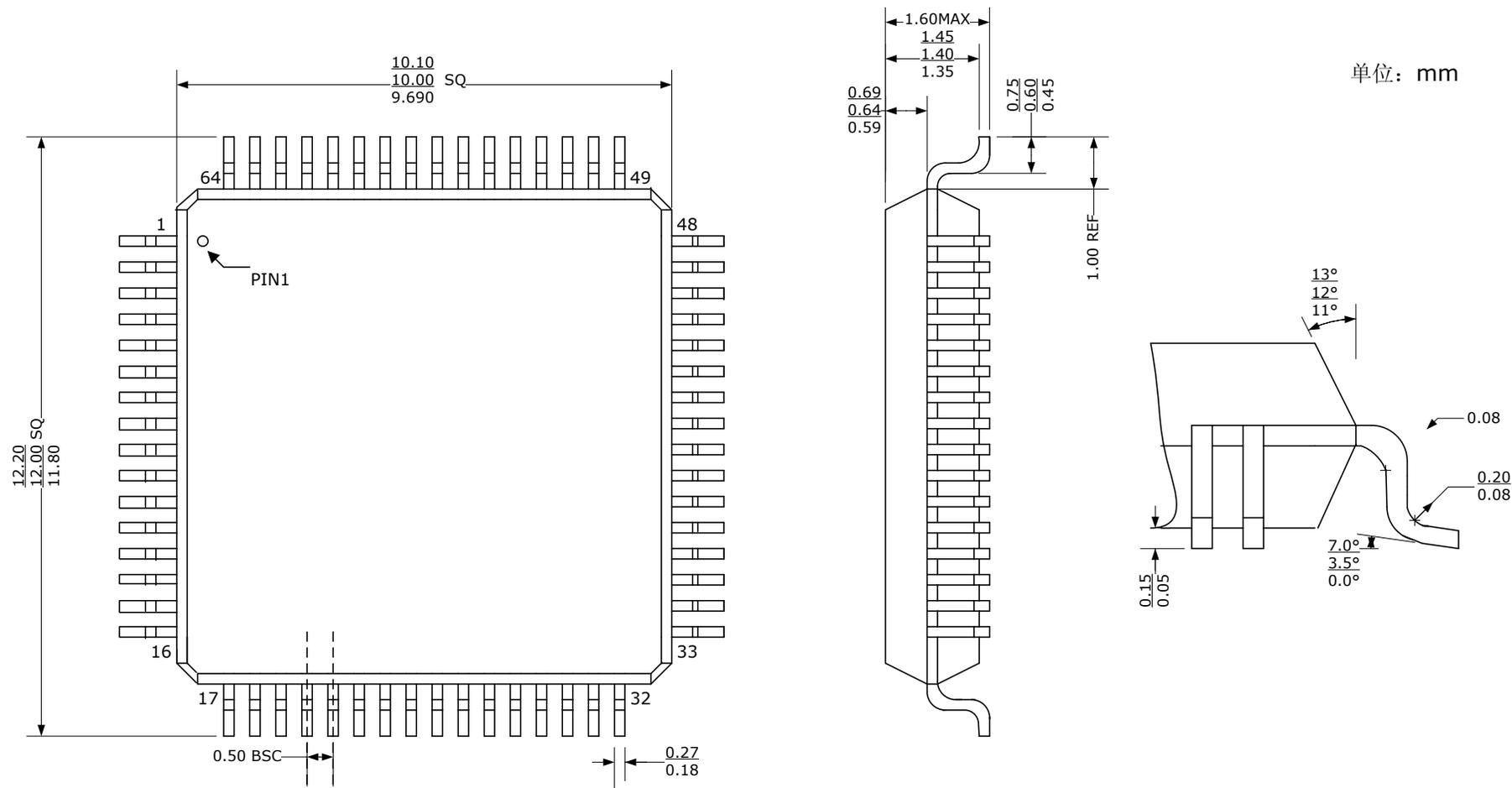
由于这个函数关系式比较复杂，因此用户可选择通过下述方法计算每个温度下对应的 RTC 校正值（C），制表，再通过查表法得到不同温度下的 RTC 校正值。

1. 将不同温度下秒脉冲输出正好为一秒时的晶振值作为基准，如 32768；
2. 设本次测得的晶振频率为 x，与基准值比较，通过下式计算出校正值 C。

RTC 每 30s 的校正精度 $\frac{1}{32768} \times 1000000 = 30.5\text{ppm}$ ，需校正 $\frac{x - 32768}{32768} \times 1000000 \text{ ppm}$ ，则，校正值为

$$C = \frac{x - 32768}{32768} \times 1000000 / (\frac{1}{32768} \times 1000000) = x - 32768。$$

封装尺寸图



图索引

图 2-1 复位电路框图	24
图 3-1 IRAM 和 SFR	27
图 3-2 数据存储空间分配图	29
图 3-3 程序存储空间分配图	30
图 5-1 时钟控制电路框图	45
图 5-2 使用快速操作方式开启 PLL 并将时钟 1 的时钟源切换到 PLL 时钟输出	51
图 5-3 使用快速操作方式将时钟 1 的时钟源切换到 OSC 时钟输出, 然后关闭 PLL, 关闭时钟 1	51
图 6-1 系统状态转换图	52
图 6-2 VDCIN 输入电平与 PWRUP 和 PWRDN 标志位的关系	52
图 7-1 电源供电系统	57
图 7-2 LDO33 电压输出与外部电压信号输入	58
图 7-3 LDO33 电压输出与功耗	58
图 7-4 5V 电源与电池切换	59
图 8-1 基准电压源温度特性曲线	60
图 9-1 电能计量模块信号处理	72
图 9-2 采用 CT 输入电流	73
图 9-3 采用锰铜电阻分流网络输入电流	73
图 9-4 电压输入方式	74
图 9-5 电流/电压通道模拟增益配置	75
图 9-6 角差校正原理图	76
图 9-7 电流计量通道选择	77
图 9-8 数字信号输入	77
图 9-9 电流/电压有效值计算信号处理	79
图 9-10 有功功率计算	81
图 9-11 无功功率计算	81
图 9-12 有功/无功能量累加和脉冲计数	82
图 9-13 线电压频率测量原理	85
图 9-14 M 通道外部连接图	86
图 9-15 M 通道内部连接图	86
图 9-16 测温电路原理框图	88
图 9-17 温度测量结果	88
图 9-18 M 通道输入信号经分压后测量结果	89

图 9-19 M 通道输入信号未经分压测量结果 89

图 10-1 中断处理 101

图 10-2 中断控制系统..... 112

图 11-1 TimerA 的工作模式..... 115

图 11-2 Continuous Mode 的用途 116

图 11-3 Up Mode, TimerA 比较模式下的 TA1 引脚的输出 120

图 11-4 Timer0/1 模式 0/1..... 124

图 11-5 Timer0/1 模式 2..... 124

图 11-6 Timer0/1 模式 3..... 125

图 11-7 Timer2 的 16 位定时器/计数器模式, 带捕获功能的 16 位计数器模式..... 127

图 11-8 Timer2 自动加载初值的 16 位计数器模式..... 127

图 12-1 不同模式下, LCD 显示缓冲寄存器的工作原理 137

图 13-1 P0.0 端口的结构框图..... 139

图 13-2 P0.1/P0.2/P0.3 端口的结构框图..... 139

图 13-3 P1 端口的结构框图..... 140

图 13-4 P2 端口的结构框图..... 143

图 13-5 P3 端口的结构框图..... 145

图 13-6 P4 端口的结构框图..... 146

图 13-7 P5 端口的结构框图..... 147

图 13-8 P6 端口的结构框图..... 148

图 13-9 P7 端口的结构框图..... 149

图 13-10 P8 端口的结构框图 151

图 13-11 P9 端口的结构框图..... 152

图 15-1 RTC 温度补偿原理 160

图 15-2 晶振频率与温度的关系..... 160

表索引

表 1-1 特殊功能寄存器 (SFR) 列表	17
表 1-2 时钟切换控制寄存器 (SysCtrl SFR, 0x80)	20
表 1-3 时钟周期控制寄存器 (CKCON SFR, 0x8E)	21
表 1-4 PLL 状态寄存器 (PLLLCK SFR, 0xA3)	21
表 1-5 系统状态寄存器 (Systate SFR, 0xA1)	21
表 2-1 复位范围	23
表 3-1 寄存器组选择	28
表 3-2 XRAM 高 128 字节写入保护寄存器 (XRamPwd, 0x28A0)	28
表 3-3 ISP 操作对 FLASH 不同空间的访问限制	31
表 3-4 指令集	31
表 3-5 可编程的 MOVX 访问周期	36
表 4-1 ADC 控制寄存器 0 (CtrlADC0, 0x2858)	37
表 4-2 ADC 控制寄存器 1 (CtrlADC1, 0x2859)	37
表 4-3 ADC 控制寄存器 2 (CtrlADC2, 0x285A)	38
表 4-4 ADC 控制寄存器 3 (CtrlADC3, 0x285B)	38
表 4-5 电池放电控制寄存器 (CtrlBAT, 0x285C)	39
表 4-6 ADC 控制寄存器 4 (CtrlADC4, 0x285D)	39
表 4-7 LCD 驱动电压控制寄存器 (CtrlLCDV, 0x285E)	39
表 4-8 晶振控制寄存器 0 (CtrlCry0, 0x285F)	39
表 4-9 晶振控制寄存器 1 (CtrlCry1, 0x2860)	40
表 4-10 晶振控制寄存器 2 (CtrlCry2, 0x2861)	40
表 4-11 BGP 控制寄存器 (CtrlBGP, 0x2862)	40
表 4-12 ADC 控制寄存器 5 (CtrlADC5, 0x2863)	41
表 4-13 ADC 控制寄存器 6 (CtrlADC6, 0x2864)	41
表 4-14 LDO 控制寄存器 (CtrlLDO, 0x2866)	42
表 4-15 时钟控制寄存器 (CtrlCLK, 0x2867)	42
表 4-16 PLL 控制寄存器 (CtrlPLL, 0x2868)	42
表 5-1 时钟控制	44
表 5-2 晶振电路寄存器配置	46
表 5-3 MCU/ADC/电能计量时钟频率配置	46
表 5-4 电网应用选择	47
表 5-5 时钟控制电路中的时钟	47

表 5-6 MCU 时钟源快速切换控制位	49
表 5-7 普通操作、快速操作和结合操作进行 MCU 时钟源切换的比较	50
表 6-1 供电状态和系统复位原因指示	53
表 6-2 各模块功耗的影响因素	53
表 6-3 OSC 状态	54
表 6-4 正常工作状态 (MCU 工作频率为 13.1072MHz)	54
表 6-5 系统休眠状态设置	56
表 6-6 休眠状态下系统功耗	56
表 7-1 LDO33 控制位	58
表 8-1 BandGap 电路温度特性调整	60
表 9-1 读写缓冲寄存器与数据的对应关系	61
表 9-2 PM 控制寄存器 1 (PMCtrl1, 0x2878)	62
表 9-3 PM 控制寄存器 2 (PMCtrl2, 0x2879)	63
表 9-4 PM 控制寄存器 3 (PMCtrl3, 0x287A)	63
表 9-5 角差校正控制寄存器 1 (PHCCtrl1, 0x287B)	64
表 9-6 角差校正控制寄存器 2 (PHCCtrl2, 0x287C)	64
表 9-7 PM 控制寄存器 4 (PMCtrl4, 0x287D)	64
表 9-8 脉冲输出控制寄存器 (CFCtrl, 0x287E)	65
表 9-9 防潜标志寄存器 (CRPST, 0x287F)	66
表 9-10 信号波形数据寄存器 (R/W)	66
表 9-11 功率/有效值数据寄存器 (R/W)	67
表 9-12 功率/有效值比差寄存器 (R/W)	68
表 9-13 功率二次补偿寄存器 (R/W)	68
表 9-14 带通滤波器系数寄存器 (0x10EF)	69
表 9-15 能量桶/脉冲计数器 (R/W)	69
表 9-16 门限值寄存器/常数功率值寄存器 (R/W)	70
表 9-17 频率值寄存器 (DATAFREQ, 0x10FD)	71
表 9-18 M 通道数据寄存器	71
表 9-19 电压/电流通道模拟增益配置	74
表 9-20 开关电压/电流通道 ADC	75
表 9-21 ADC 时钟频率配置	76
表 9-22 角差校正配置	76
表 9-23 电流计量通道选择控制位	77
表 9-24 数字信号输入使能	78
表 9-25 计量通道信号数字增益配置	78

表 9-26 有效值计算的寄存器配置	79
表 9-27 功率计算选择	80
表 9-28 能量累加/功率源选择/脉冲源选择/能量脉冲产生速率控制位	83
表 9-29 CF 脉冲输出配置	84
表 9-30 起动/潜动判断控制位和标志位	84
表 9-31 信号频率大小与频率值寄存器输出值的线性关系	85
表 9-32 M 通道配置	87
表 9-33 校表相关计量数据寄存器	90
表 10-1 IE SFR (0xA8)	98
表 10-2 IP SFR (0xB8)	99
表 10-3 EIE SFR (0xE8)	99
表 10-4 EIP SFR (0xF8)	99
表 10-5 EXIF SFR (0x91)	100
表 10-6 EICON SFR (0xD8)	100
表 10-7 中断向量 8 中断源	102
表 10-8 扩展中断使能寄存器 (ExInt2IE, 0x2843)	102
表 10-9 中断标志 (请求) 寄存器 (ExInt2IFG, 0x2840)	102
表 10-10 扩展中断队列寄存器 (ExInt2OV, 0x2844)	103
表 10-11 扩展中断输入类型寄存器 (ExInt2IN, 0x2841)	103
表 10-12 扩展中断输出类型寄存器 (ExInt2OUT, 0x2842)	103
表 10-13 中断向量 9 的中断源	103
表 10-14 扩展中断使能寄存器 (ExInt3IE, 0x284B)	104
表 10-15 中断标志 (请求) 寄存器 (ExInt3IFG, 0x2848)	104
表 10-16 扩展中断队列寄存器 (ExInt3OV, 0x284C)	104
表 10-17 扩展中断输入类型寄存器 (ExInt3IN, 0x2849)	104
表 10-18 扩展中断输出类型寄存器 (ExInt3OUT, 0x284A)	105
表 10-19 中断向量 10 中断源	105
表 10-20 扩展中断使能寄存器 (ExInt4IE, 0x2853)	105
表 10-21 中断标志 (请求) 寄存器 (ExInt4IFG, 0x2850)	105
表 10-22 扩展中断队列寄存器 (ExInt4OV, 0x2854)	106
表 10-23 扩展中断输入类型寄存器 (ExInt4IN, 0x2851)	106
表 10-24 扩展中断输出类型寄存器 (ExInt4OUT, 0x2852)	106
表 10-25 中断向量 11 中断源	106
表 10-26 扩展中断使能寄存器 (ExInt5IE, 0x28A5)	107
表 10-27 中断标志 (请求) 寄存器 (ExInt5IFG, 0x28A2)	107



表 10-28 扩展中断队列寄存器 (ExInt5OV, 0x28A6)	107
表 10-29 扩展中断输入类型寄存器 (ExInt5IN, 0x28A3)	107
表 10-30 扩展中断输出类型寄存器 (ExInt5OUT, 0x28A4)	107
表 10-31 中断资源列表	109
表 11-1 TimerA 寄存器列表	113
表 11-2 TimerA 计数寄存器 (TAR, 0x2902~0x2903)	114
表 11-3 TimerA 控制寄存器 (TACTL, 0x2900)	114
表 11-4 TimerA 的捕获/比较模块 x*的控制寄存器 (0x2904~0x2909)	117
表 11-5 Timer0/Timer1/Timer2 时钟源选择	121
表 11-6 Timer0/1 模式控制寄存器 (TMOD SFR, 0x89)	121
表 11-7 Timer0/1 控制寄存器 (TCON SFR, 0x88)	122
表 11-8 Timer2 控制寄存器 (T2CON SFR, 0xC8)	126
表 11-9 UART1 控制寄存器 (SCON1 SFR, 0xC0)	128
表 11-10 UARTx 相关的寄存器	129
表 11-11 UARTx 控制/状态寄存器 (TCONx)	130
表 11-12 UARTx 相关定时器模式控制寄存器 (TMODx)	130
表 11-13 UARTx 控制寄存器 (SCONx)	131
表 11-14 UARTx 缓冲寄存器 (SBUFx)	131
表 11-15 TXD2 类型选择寄存器 (Txd2FS, 0x28CF)	132
表 11-16 载波产生寄存器	132
表 11-17 UART 模式选择	132
表 12-1 显示控制寄存器 (LCDCtrl, 0x2C1E)	135
表 12-2 SEG 开关控制寄存器 (R/W)	136
表 12-3 当 LCDTYPE=0 时, SEG 与显示缓冲寄存器的对应关系	137
表 12-4 当 LCDTYPE=1 时, SEG 与显示缓冲寄存器的对应关系	137
表 12-5 LCD 驱动电压配置	138
表 13-1 P0 输出使能寄存器 (P0OE, 0x28A8)	139
表 13-2 P0 输入使能寄存器 (P0IE, 0x28A9)	140
表 13-3 P0 输出数据寄存器 (P0OD, 0x28AA)	140
表 13-4 P0 输入数据寄存器 (P0ID, 0x28AB)	140
表 13-5 P1 输出使能寄存器 (P1OE, 0x28AC)	141
表 13-6 P1 输入使能寄存器 (P1IE, 0x28AD)	141
表 13-7 P1 输出数据寄存器 (P1OD, 0x28AE)	141
表 13-8 P1 输入数据寄存器 (P1ID, 0x28AF)	141
表 13-9 P1.1 功能选择寄存器 (P11FS, 0x28C5, R/W)	141

表 13-10 P1.2 功能选择寄存器 (P12FS, 0x28C6, R/W)	142
表 13-11 P1.3 功能选择寄存器 (P13FS, 0x28C7, R/W)	142
表 13-12 P1.4 功能选择寄存器 (P14FS, 0x28C8)	142
表 13-13 P2 输出使能寄存器 (P2OE, 0x28B0)	143
表 13-14 P2 输入使能寄存器 (P2IE, 0x28B1)	143
表 13-15 P2 输入数据寄存器 (P2ID, 0x28B3)	143
表 13-16 P2.0 功能选择寄存器 (P20FS, 0x28C9, R/W)	144
表 13-17 P2.1 功能选择寄存器 (P21FS, 0x28CA, R/W)	144
表 13-18 P2.4 功能选择寄存器 (P24FS, 0x28CD, R/W)	144
表 13-19 P2.5 功能选择寄存器 (P25FS, 0x28CE, R/W)	144
表 13-20 P3 输出使能寄存器 (P3OE, 0x28B4)	145
表 13-21 P3 输入使能寄存器 (P3IE, 0x28B5)	145
表 13-22 P3 输出数据寄存器 (P3OD, 0x28B6)	145
表 13-23 P3 输入数据寄存器 (P3ID, 0x28B7)	145
表 13-24 P4 输出使能寄存器 (P4OE, 0x28B8)	146
表 13-25 P4 输入使能寄存器 (P4IE, 0x28B9)	146
表 13-26 P4 输出数据寄存器 (P4OD, 0x28BA)	146
表 13-27 P4 输入数据寄存器 (P4ID, 0x28BB)	147
表 13-28 P5 输出使能寄存器 (P5OE, 0x28BC)	147
表 13-29 P5 输入使能寄存器 (P5IE, 0x28BD)	147
表 13-30 P5 输出数据寄存器 (P5OD, 0x28BE)	148
表 13-31 P5 输入数据寄存器 (P5ID, 0x28BF)	148
表 13-32 P6 输出使能寄存器 (P6OE, 0x28C0)	148
表 13-33 P6 输入使能寄存器 (P6IE, 0x28C1)	149
表 13-34 P6 输出数据寄存器 (P6OD, 0x28C2)	149
表 13-35 P6 输入数据寄存器 (P6ID, 0x28C3)	149
表 13-36 P7 输出使能寄存器 (P7OE, 0x28D5)	150
表 13-37 P7 输入使能寄存器 (P7IE, 0x28D6)	150
表 13-38 P7 输出数据寄存器 (P7OD, 0x28D7)	150
表 13-39 P7 输入数据寄存器 (P7ID, 0x28D8)	150
表 13-40 P8 输入使能寄存器 (P8IE, 0x28DA)	151
表 13-41 P8 输出数据寄存器 (P8OD, 0x28DB)	151
表 13-42 P8 输入数据寄存器 (P8ID, 0x28DC)	151
表 13-43 P9 输出使能寄存器 (P9OE SFR, 0xA4)	152
表 13-44 P9 输入使能寄存器 (P9IE SFR, 0xA5)	152

表 13-45 P9 输出数据寄存器 (P9OD SFR, 0xA6)	152
表 13-46 P9 输入数据寄存器 (P9ID SFR, 0xA7)	152
表 13-47 P9 功能选择寄存器 (P9FS SFR, 0xAD)	153
表 15-1 RTC 密码使能寄存器 (RTCPEN SFR, 0x90)	155
表 15-2 RTC 密码寄存器 (RTCPWD SFR, 0x97)	155
表 15-3 系统休眠唤醒间隔寄存器 (INTRTC SFR, 0x96)	155
表 15-4 RTC 校正寄存器 (RTCCH SFR/RTCCL SFR, 0x94/0x95)	156
表 15-5 RTC 数据读取寄存器 (RDRTC SFR, 0xDA)	156
表 15-6 高频分频门限值寄存器 (DIVTHH SFR/DIVTHM SFR/DIVTHL SFR, 0xDB/0xDC/0xDD)	156
表 15-7 高频计数器状态寄存器 (PLLCNT SFR, 0xDE)	156
表 15-8 计时寄存器 (RTC 时间设置寄存器)	157

版本更新说明

时间	版本	修改内容
2012-06-15	V3.0	<ol style="list-style-type: none"> 1. 更新“引脚分布图”(Pin27)、表 1-1/13-43/13-44/13-47 2. 更新“极限参数”(功耗) 3. 更新表 11-1 和图 9-17/9-18/9-19 4. 添加“封装尺寸图”
2012-04-28	V2.1	<ol style="list-style-type: none"> 1. 更新表 15-4 2. 更正公式 15-4/15-5/15-6
2012-04-25	V2.0	<ol style="list-style-type: none"> 1. 更新文档结构/格式 2. 更正引脚 Pin5/Pin27 的功能描述 3. 更正“特殊功能寄存器” 4. 更新“程序存储空间” 5. 更正寄存器 CtrlBGP/CtrlLDO/PARABPF 的描述 6. 更正“数字电源电路” 7. 更正“RTC 校正”
2011-04-23	V1.0	正式发布