

C10S0XX 8位RISC 单片机特性

- 35条精简指令集
- 1K 程序 FLASH 存储空间
- 256x8b 数据 EEPROM
- 数据 EEPROM 可在应用编程
- 64x8b SRAM
- 1 x 带 8 位预分频的定时器 0
- 1 x 带 8 位预分频的定时器 2
- 带 7 位预分频的 WDT，溢出频率约为 16-2048ms
- 上电延迟计数器 PWRT
- 4 个唤醒源，INT、端口变化中断、WDT、数据 EEPROM 写完成
- 内置高速 RC 振荡器，最高频率支持 16M
- 内置低速 RC 振荡器，32K 模式
- 除电源脚 外全部是可用 IO
- 端口变化中断，PA0-PA5
- 支持在线调试
- 程序空间保护
- 工作电压范围：2.0V - 5.5V

1.3. 脚位图

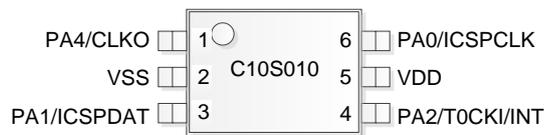


图 1.4 SOT23-6 脚位图

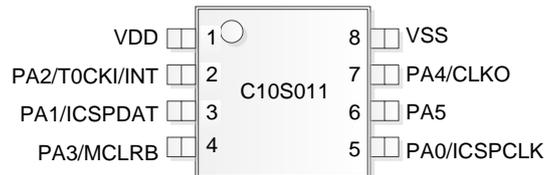


图 1.5 SOP-8 脚位图

目录

特性、脚位.....	1
1. 数字功能框图和程序存储器	6
1.1. 数字功能框图	6
1.2. 程序存储器.....	6
1.3. 脚位图.....	7
2. SFR	8
2.1. 地址映射	8
2.1.1. SFR, BANK0.....	8
2.1.2. SFR, BANK1.....	9
2.1.3. TMR0, 地址 0x01.....	10
2.1.4. STATUS 寄存器, 地址 0x03, 0x83.....	10
2.1.5. PORTA 寄存器, 地址 0x05.....	11
2.1.6. INTCON 寄存器, 地址 0x0B, 0x8B.....	11
2.1.7. PIR1 寄存器, 地址 0x0C	12
2.1.8. TMR2, 地址 0x11	13
2.1.9. T2CON 寄存器, 地址 0x12.....	13
2.1.10. WDTCON 寄存器, 地址 0x18.....	14
2.1.11. MSCKCON 寄存器, 地址 0x1B.....	14
2.1.12. SOSCPR 寄存器, 地址 0x1C, 1D.....	15
2.1.13. OPTION 寄存器, 地址 0x81.....	16
2.1.14. TRISA 寄存器, 地址 0x85	16
2.1.15. PIE1 寄存器, 地址 0x8C	17
2.1.16. OSCCON 寄存器, 地址 0x8F.....	17
2.1.17. PR2 寄存器, 地址 0x92.....	18
2.1.18. WPUA 寄存器, 地址 0x95.....	18
2.1.19. IOCA 寄存器, 地址 0x96.....	18
2.1.20. PCON 寄存器, 地址 0x8E.....	19
2.1.21. EEDAT 寄存器, 地址 0x9A.....	19
2.1.22. EEADR 寄存器, 地址 0x9B.....	19
2.1.23. EECON1 寄存器, 地址 0x9C	19
2.1.24. EECON2 寄存器, 地址 0x9D	20
2.1.25. UCFG0/UCFG1/UCFG2	20
2.1.26. PCL 和 PCLATH.....	22
2.1.27. INDF 和 FSR 寄存器	22
3. 系统时钟源	23
4. 复位时序	23
4.1. POR 上电复位.....	24
4.2. 外部复位 MCLR.....	24

4.3. PWRT (上电计时器)	25
4.4. BOR 低电压复位	25
4.5. 超时动作	26
4.6. 关于 WDT 复位	27
5. BOOT	28
6. 烧录和控制串口	28
7. 看门狗定时器.....	29
7.1. 看门狗.....	29
8. 定时器 0.....	30
8.1. TIMER 0	30
8.2. TIMER 0 定时器模式.....	30
8.3. TIMER 0 计数器模式.....	30
8.3.1. 软件可配置预分频电路	31
8.3.2. 定时器 0 中断.....	31
8.3.3. 用外部时钟驱动定时器 0	32
9. 定时器 2.....	32
9.1. TIMER2.....	32
10. 数据 EEPROM.....	33
10.1. 编程数据 EEPROM 步骤	33
10.2. 关于编程周期.....	33
10.3. 读 EEPROM 步骤	34
11. 慢时钟测量模式	34
12. 中断模式	35
12.1. 中断过程中的现场保存	36
13. 睡眠省电模式.....	37
13.1. 唤醒模式	37
13.2. 看门狗唤醒.....	37
14. I/O 端口.....	38
14.1. PORTA 端口和 TRISA 寄存器	38
14.2. 端口的其他功能	38
14.2.1. 弱上拉.....	38
14.2.2. 状态变化中断.....	38
14.2.3. 端口描述.....	39
15. 芯片的电气特性	45
15.1. 绝对极限参数	45
15.2. 内置高频振荡器 (INTERNAL HIGH FREQUENCY OSC)	45
15.3. 内置低频振荡器 (INTERNAL LOW FREQUENCY OSCILLATOR)	45

15.4. 低电压复位电路 (LVR).....	46
15.5. I/O PAD 电路.....	46
15.6. 总体工作电流 (IVDD).....	46
15.7. AC 电气特性.....	47
15.8. 时序图.....	47
15.9. 直流和交流特性图表.....	48
16. MCU 的指令架构.....	52
16.1. 指令集列表.....	52
17. 封装信息.....	54
17.1 封装标识信息.....	54
17.2 详细封装.....	54

1. 数字功能框图和程序存储器、脚位图

1.1. 数字功能框图

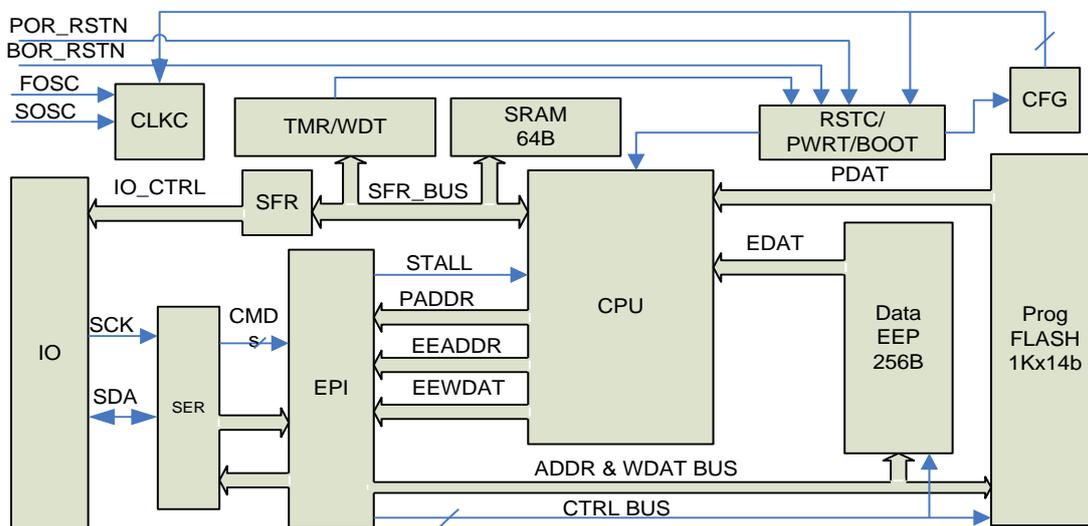


图 1.1 数字功能框图

1.2. 程序存储器

地址寄存器为 13 位 (0x0000 ~ 0x1FFF)，最多支持 8K 地址空间。程序存储器一共 1Kx14b (0x0000 ~ 0x03FF)，加上额外 ID 和配置信息区，共 32x14b，它们由 FLASH 构成。一个 PAGE 是 16 个字，一共有 66 个 PAGE。

其中 0~0x03FF 对主程序区访问，其中未实现部分 0x400~0x1FFF 保留。ID 和配置信息区从 0x2000 开始，到 0x201F 结束。

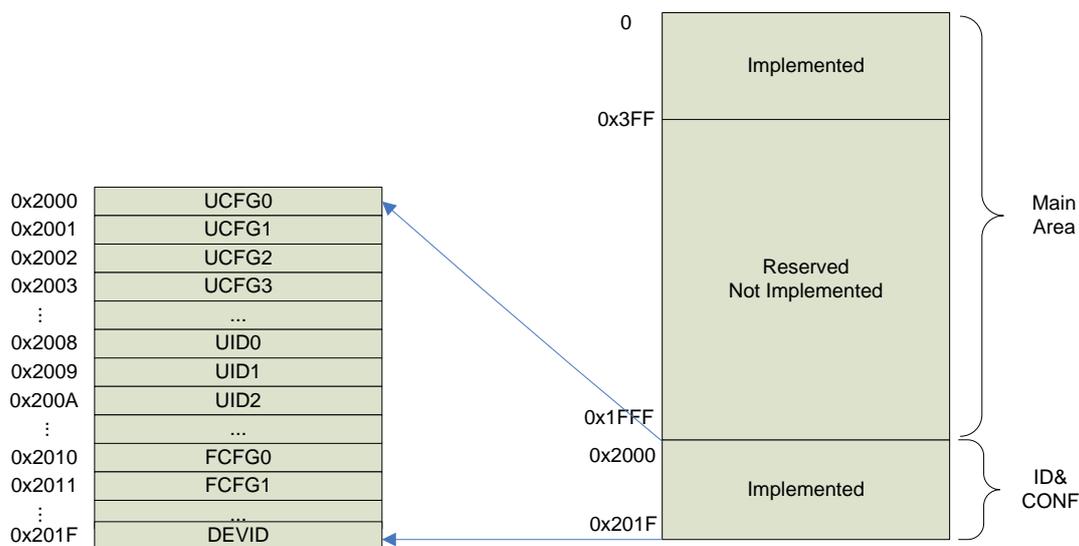


图 1.2 程序空间地址映射

芯片管脚详细描述:

管脚名	功能名	输入信号类型	输出信号类型	具体描述	上下拉
PA0/ICSPCLK	PA0	ST	CMOS	具有可编程上拉和电平变化中断的 I/O	可配置上拉
	ICSPCLK	ST	---	Debug/烧录模式串口 clock 信号 (Fmax=6MHz)	
PA1/ICSPDAT	PA1	ST	CMOS	具有可编程上拉和电平变化中断的 I/O	可配置上拉
	ICSPDAT	ST	CMOS	Debug/烧录模式串口 data 信号 (Fmax=6MHz)	
PA2/T0CKI/INT	PA2	ST	CMOS	具有可编程上拉和电平变化中断的 I/O	可配置上拉
	T0CKI	ST	---	Timer 0 源头时钟输入 (Fmax=4MHz)	
	INT	ST	---	外部中断输入	
PA3/MCLR _B	PA3	ST	---	带电平变化中断的输入	无内部上拉
	MCLR _B	ST	---	外部复位输入	自动开启 上拉
PA4/CLKO	PA4	ST	CMOS	具有可编程上拉和电平变化中断的 I/O	可配置上拉
	CLKO	---	CMOS	测试时钟输出	
PA5	PA5	ST	CMOS	具有可编程上拉和电平变化中断的 I/O	可配置上拉

注意:

1. **ST**: 带 **CMOS** 电平的施密特触发器输入

2. SFR

2.1. 地址映射

2.1.1. SFR, BANK0

Address (HEX)	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Reset, POR
BANK0										
0	INDF	使用 FSR 的内容对数据存储器进行访问（非物理寄存器）								xxxx xxxx
1	TMR0	TIMER0 计数器								xxxx xxxx
2	PCL	程序计数器低 8 位								0000 0000
3	STATUS	-	-	PAGE	/TF	/PF	Z	HC	C	--01 1xxx
4	FSR	间接寻址指针寄存器								xxxx xxxx
5	PORTA	-	-	PA5	PA4	PA3	PA2	PA1	PA0	--00 x000
6	-	-	-	-	-	-	-	-	-	---- ----
7	-	-	-	-	-	-	-	-	-	---- ----
8	-	-	-	-	-	-	-	-	-	---- ----
9	-	-	-	-	-	-	-	-	-	---- ----
A	PCLATH	-	-	-	-	-	程序计数器高 3 位锁存器			---- -000
B	INTCON	GIE	PEIE	TOIE	INTE	PAIE	TOIF	INTF	PAIF	0000 0000
C	PIR1	EEIF	CKMEAIF	-	-	-	-	TMR2IF	-	000- 000-
D	-	-	-	-	-	-	-	-	-	---- ----
E	-	-	-	-	-	-	-	-	-	---- ----
F	-	-	-	-	-	-	-	-	-	---- ----
10	-	-	-	-	-	-	-	-	-	---- ----
11	TMR2	TIMER2[7:0] timer2 module register								0000 0000
12	T2CON	-	TOUTPS[3:0]				TMR2ON	T2CKPS[1:0]		-000 0000
13	-	-	-	-	-	-	-	-	-	---- ----
14	-	-	-	-	-	-	-	-	-	---- ----
15	-	-	-	-	-	-	-	-	-	---- ----
16	-	-	-	-	-	-	-	-	-	---- ----
17	-	-	-	-	-	-	-	-	-	---- ----
18	WDTCON	-	-	-	WDTPS[3:0]			SWDTEN		---0 1000
19	-	-	-	-	-	-	-	-	-	---- ----
1A	-	-	-	-	-	-	-	-	-	---- ----
1B	MSCKCON	-	-	-	SLVREN	-	CKMAVG	CKCNTI	-	---0 -00-
1C	SOSCPRH	SOSCPR [7:0]								1111 1111
1D	SOSCPRH	-	-	-	-	SOSCPR [11:8]				---- 1111

1E	-	-	----
1F	-	-	----
40~7F	SRAM · 64Bytes (user usable)		xxxx xxxx

注意:

1. 未实现, 保留, 读 0

2.1.2. SFR, BANK1

Address (HEX)	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Reset, POR
BANK1										
80	INDF	使用 SFR 的内容对数据存储寄存器进行访问 (非物理寄存器)								xxxx xxxx
81	OPTION	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111
82	PCL	程序计数器低 8 位								0000 0000
83	STATUS	-	-	PAGE	/TF	/PF	Z	HC	C	--01 1xxx
84	FSR	间接寻址指针寄存器								xxxx xxxx
85	TRISA	-	-	TRISA[5:4]		-	TRISA[2:0]		-	--11 -111
86	-	-								----
87	-	-								----
88	-	-								----
89	-	-								----
8A	PCLATH	程序计数器高 3 位锁存器								---- -000
8B	INTCON	GIE	PEIE	T0IE	INTE	PAIE	T0IF	INTF	PAIF	0000 0000
8C	PIE1	EEIE	CKMEAI E	-	-	-	-	TMR2IE	-	000- 000-
8D	-	-								----
8E	PCON	-	-	-	-	-	-	/POR	/BOR	---- --qq
8F	OSCCON	LFMOD	IRCF[2:0]			-	HFIOFR	LFIOFR	-	0101 -00-
90	-	-								----
91	-	-								----
92	PR2	PR2[7:0] timer2 period register								1111 1111
93	-	-								----
94	-	-								----
95	WPUA	-	-	WPUA[5:4]		-	WPUA[2:0]		-	--11 -111
96	IOCA	-	-	IOCA[5:0]			-			--00 0000
97	-	-								----
98	-	-								----
99	-	-								----
9A	EEDAT	EEDAT[7:0]								0000 0000
9B	EEADR	EEADR[7:0]								0000 0000
9C	EECON1	-	-	WREN3	WREN2	WRERR	WREN1	-	RD	--00 x0-0
9D	EECON2	-	-	-	-	-	-	-	WR	---- --0

9E	-	-	-----
9F	-	-	-----
A0~BF	-		-----
C0~FF	Access BANK0 SRAM, 40~7F		xxxx xxxx

注意:

1. **INDF** 不是物理寄存器;
2. 灰色部分表示没有实现;
3. 未实现的寄存器位不要写 1, 因为后续产品可能会使用到。

2.1.3. TMR0, 地址 0x01

Bit	7	6	5	4	3	2	1	0
Name	TMR0[7:0]							
Reset	xxxx xxxx							

Bit	Name	Function
7:0	TMR0[7:0]	Timer 0 计数结果寄存器

2.1.4. STATUS 寄存器, 地址 0x03, 0x83

Bit	7	6	5	4	3	2	1	0
Name	-	-	PAGE	/TF	/PF	Z	HC	C
Reset	-	-	0	1	1	x	x	x

Bit	Name	Function
7:6	NA	没实现, 读 0
5	PAGE	PAGE: 寄存器存储区选择位 (用于直接寻址) 1 = Bank 1 (80h - FFh) 0 = Bank 0 (00h - 7Fh)
4	/TF	/TF: 超时状态位 1 = 上电后, 执行了 CLRWDT 指令或 SLEEP 指令 0 = 发生 WDT 超时溢出
3	/PF	/PF: 掉电标志位 1 = 上电复位后或执行了 CLRWDT 指令 0 = 执行了 SLEEP 指令
2	Z	Z: 零标志位 1 = 算术运算或逻辑运算的结果为零 0 = 算术运算或逻辑运算的结果不为零
1	HC	HC: 半进位/ 借位位 (ADDWF、ADDWI、SUBWI 和 SUBWF 指令)。 对于借位, 极性是相反的。 1 = 结果的第 4 低位向高位发生了进位 0 = 结果的第 4 低位未向高位发生进位

0	C	C: 进位/ 借位位 (ADDWF、ADDWI、SUBWI 和 SUBWF 指令) 1 = 结果的最高位发生了进位 0 = 结果的最高位未发生进位
---	---	---

/TF	/PD	条件
1	1	上电或者低电压复位
0	U	WDT 复位
0	0	WDT 唤醒
U	U	正常运行下发生 MCLR 复位
1	0	睡眠状态下发生 MCLR 复位

2.1.5. PORTA 寄存器，地址 0x05

Bit	7	6	5	4	3	2	1	0
Name	-	-	PA5	PA4	PA3	PA2	PA1	PA0
Reset	-	-	0	0	x	0	0	0

Bit	Name	Function
7	-	-
6	-	-
5	PA[5]	PORTA5 数据
4	PA[4]	PORTA4 数据
3	PA[3]	PA3 只有输入功能，不存在相应的数据寄存器
2	PA[2]	PORTA2 数据
1	PA[1]	PORTA1 数据
0	PA[0]	PORTA0 数据

2.1.6. INTCON 寄存器，地址 0x0B，0x8B

Bit	7	6	5	4	3	2	1	0
Name	GIE	PEIE	TOIE	INTE	PAIE	TOIF	INTF	PAIF
Reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	GIE	GIE: 全局中断使能 1 = 允许所有未屏蔽的中断 0 = 禁止所有中断
6	PEIE	PEIE: 外设中断使能 1 = 允许所有未屏蔽的外设中断 0 = 禁止所有外设中断
5	TOIE	TOIE: 定时器0溢出中断使能 1 = 允许 Timer0 中断

		0 = 禁止 Timer0 中断
4	INTE	INTE: 外部中断使能 1 = 允许 PA2/INT 外部中断 0 = 禁止 PA2/INT 外部中断
3	PAIE	PAIE: PORTA电平中断使能位 1 = 允许 PORTA 电平变化中断 0 = 禁止 PORTA 电平变化中断
2	T0IF	T0IF: 定时器0溢出中断标志位 1 = Timer0 寄存器已经溢出（必须用软件清零） 0 = Timer0 寄存器没有溢出
1	INTF	INTF: PA2/INT外部中断标志位 1 = 发生了 PA2/INT 外部中断（必须用软件清零） 0 = 未发生 PA2/INT 外部中断
0	PAIF	PAIF: PORTA 电平变化中断标志位 1 = 至少一个 PORTA <5:0> 引脚的电平状态发生了改变（必须用软件清零） 0 = 没有一个 PORTA <5:0> 引脚的电平状态发生改变

2.1.7. PIR1 寄存器，地址 0x0C

Bit	7	6	5	4	3	2	1	0
Name	EEIF	CKMEAIF	-	-	-	-	TMR2IF	-
Reset	0	0	-	-	-	-	0	-

Bit	Name	Function
7	EEIF	EEIF: EE写中断标志位 1 = EE 写操作完成（必须软件清零） 0 = EE 写操作未完成
6	CKMEAIF	CKMEAIF: 快时钟测量慢时钟操作完成中断标志位 1 = 快时钟测量慢时钟操作完成（必须软件清零） 0 = 快时钟测量慢时钟未完成
5	-	-
4	-	-
3	-	-
2	-	-
1	TMR2IF	TMR2IF: Timer2 与PR2比较相等中断标志位 1 = timer2的值等于PR2（必须软件清零） 0 = timer2 的值不等于 PR2
0	-	-

2.1.8. TMR2, 地址 0x11

Bit	7	6	5	4	3	2	1	0
Name	TMR2[7:0]							
Reset	0000 0000							

Bit	Name	Function
7:0	TMR2[7:0]	Timer 2 计数结果寄存器

2.1.9. T2CON 寄存器, 地址 0x12

Bit	7	6	5	4	3	2	1	0
Name	-	TOUTPS[3:0]				TMR2ON	T2CKPS[1:0]	
Reset	NA	0000				0	00	

Bit	Name	Function
7	-	-
6: 3	TOUTPS[3:0]	<p>TOUTPS<3:0>: Timer2 Output Postscaler Select bits 定时器2 输出后分频比选择</p> <p>0000 = 1:1 后分频比 0001 = 1:2 后分频比 0010 = 1:3 后分频比 0011 = 1:4 后分频比 0100 = 1:5 后分频比 0101 = 1:6 后分频比 0110 = 1:7 后分频比 0111 = 1:8 后分频比 1000 = 1:9 后分频比 1001 = 1:10 后分频比 1010 = 1:11 后分频比 1011 = 1:12 后分频比 1100 = 1:13 后分频比 1101 = 1:14 后分频比 1110 = 1:15 后分频比 1111 = 1:16 后分频比</p>
2	TMR2ON	<p>TMR2ON: Timer2 On bit 打开定时器2</p> <p>1 = Timer2 is on 打开 0 = Timer2 is off 关闭</p>
1: 0	T2CKPS[1:0]	<p>T2CKPS<1:0>: Timer2 Clock Prescale Select bits 定时器2驱动 时钟预分频比选择</p> <p>00 = Prescaler is 1</p>

		01 = Prescaler is 4 1x = Prescaler is 16
--	--	---

2.1.10. WDTCON 寄存器，地址 0x18

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	WDTPS[3:0]				SWDTEN
Reset	-	-	-	0	1	0	0	0

Bit	Name	Function
7:5	N/A	-
4:1	WDTPS[3:0]	WDTPS<3:0> : 看门狗定时器周期选择: Bit Value = 看门狗定时器驱动时钟之预分频比值 0000 = 1:32 0001 = 1:64 0010 = 1:128 0011 = 1:256 0100 = 1:512 (复位值) 0101 = 1:1024 0110 = 1:2048 0111 = 1:4096 1000 = 1:8192 1001 = 1:16384 1010 = 1:32768 1011 = 1:65536 1100 = 1:65536 1101 = 1:65536 1110 = 1:65536 1111 = 1:65536
0	SWDTEN	软件打开/关闭看门狗定时器: 1 = 打开 0 = 关闭

2.1.11. MSCKCON 寄存器，地址 0x1B

Bit	7	6	5	4	3	2	1	0
MSCKCON	-	-	-	SLVREN	-	CKMAVG	CKCNTI	-
Reset	-	-	-	0	0	0	0	-

Bit	Name	Function
7-5	-	-
4	SLVREN	<p>1: 编译选项里 LVREN 使能时 SLVREN=1 为工作时开启 LVR 睡眠时自动关闭 LVR SLVREN=0 为始终开启 LVR</p> <p>2: 编译选项里 LVREN 关闭时, SLVREN=X 均为关闭 LVR</p>
3	-	保留位, 不要向此位写 1
2	CKMAVG	<p>快时钟测量慢时钟周期的测量平均模式</p> <p>1 = 打开平均模式。(自动测量 4 次, 并取其平均值)</p> <p>0 = 关闭平均模式</p>
1	CKCNTI	<p>Clock Count Init —使能快时钟测量慢时钟周期</p> <p>1 = 使能快时钟测量慢时钟周期</p> <p>0 = 关闭快时钟测量慢时钟周期</p> <p>注: 这一位在测量完毕后会自动归零</p>
0	-	保留位, 不要向此位写 1

2.1.12. SOSCPRL 寄存器, 地址 0x1C, 1D

SOSCPRL, 地址 0x1C

Bit	7	6	5	4	3	2	1	0
Name	SOSCPRL[7:0]							
Reset	8'hff							

SOSCPRH, 地址 0x1D

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	SOSCPRH[11:8]			
Reset	-	-	-	-	4'hf			

Bit	Name	Function
0x1C: 7:0 0x1D: 3:0	SOSCPRH[11:0]	<p>低速振荡器周期 (单位: 快时钟周期数)</p> <p>用于慢时钟测量功能</p>

2.1.13. OPTION 寄存器，地址 0x81

Bit	7	6	5	4	3	2	1	0
Name	/PAPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
Reset	1	1	1	1	1	1	1	1

Bit	Name	Function
7	/PAPU	PAPU: PORTA 上拉使能位 1 = 使能 PORTA 上拉 0 = PORTA 上拉由各端口锁存器值使能
6	INTEDG	INTEDG: 中断沿选择位 1 = 在 INT 引脚的上升沿中断 0 = 在 INT 引脚的下降沿中断
5	T0CS	T0CS: Timer0 时钟源选择位 1 = PA2/T0CKI 引脚的跳变 0 = 内部指令周期(FOSC/4)
4	T0SE	T0SE: Timer0 时钟源边沿选择位 1 = 在 PA2/T0CKI 引脚的下降沿递增 0 = 在 PA2/T0CKI 引脚的上升沿递增
3	PSA	PSA: 预分频器分配位 1 = 预分频器分配给 WDT 0 = 预分频器分配给 Timer0 模块
2:0	PS2	PS<2:0>: 预分频比选择位
		位值 TIMER0 分频比 WDT 分频比
	PS1	000 1 : 2 1 : 1
		001 1 : 4 1 : 2
		010 1 : 8 1 : 4
		011 1 : 16 1 : 8
		100 1 : 32 1 : 16
		101 1 : 64 1 : 32
PS0	110 1 : 128 1 : 64	
	111 1 : 256 1 : 128	

2.1.14. TRISA 寄存器，地址 0x85

Bit	7	6	5	4	3	2	1	0
Name	-	-	TRISA[5]	TRISA[4]	-	TRISA[3]	TRISA[2]	TRISA[1]
Reset	-	-	1	1	-	1	1	1

Bit	Name	Function
7:6	NA	没实现，读 0

5:4	TRISA[5:4]	TRISA<5:4> : PA<5:4> Port I/O 三态控制位 1 = Port 引脚配置为输入管脚 0 = Port 引脚配置为输出管脚
3	-	Note: 仅作为输入 读为 1
2:0	TRISA[2:0]	TRISA<2:0> : PA<2:0> Port I/O 三态控制位 1 = Port 引脚配置为输入管脚 0 = Port 引脚配置为输出管脚

2.1.15. PIE1 寄存器，地址 0x8C

Bit	7	6	5	4	3	2	1	0
Name	EEIE	CKMEAIE	-	-	-	-	TMR2IE	-
Reset	0	0	-	-	-	-	0	-

Bit	Name	Function
7	EEIE	EEIE : EE写中断使能位 1 = 使能EE 写操作完成中断 0 = 关闭 EE 写操作完成中断
6	CKMEAIE	CKMEAIE : 快时钟测量慢时钟操作完成中断使能位 1 = 使能快时钟测量慢时钟操作完成中断 0 = 关闭快时钟测量慢时钟操作完成中断
5:2	-	-
1	TMR2IE	TMR2IE : Timer2 与PR2比较相等中断使能位 1 = 使能 timer2的值等于PR2 中断 0 = 关闭使能 timer2 的值等于 PR2 中断
0	-	-

2.1.16. OSCCON 寄存器，地址 0x8F

Bit	7	6	5	4	3	2	1	0
Name	LFMOD	IRCF[2:0]			-	HFIOFR	LFIOFR	-
Reset	0	3'b101			-	0	0	-

Bit	Name	Function
7	LFMOD	低频内振模式: 1 = 256K 振荡频率模式 0 = 32K 振荡频率模式 注意: 目前此版本 LFMOD 没有作用, 请保持此位为 0。 不管此位为何值, WDT 固定使用 32KHz 频率。
6:4	IRCF[2:0]	内部振荡器频率选择 111 = 16 MHz

		110 =8 MHz 101 =4 MHz(default) 100 =2 MHz 011 =1 MHz 010 =500 kHz 001 =250 kHz 000 =32 kHz (LFINTOSC)
3	-	保留位, 读 1 写无效
2	HFIOFR	高速内部时钟状态 1 = HFINTOSC is ready 0 = HFINTOSC is not ready
1	LFIOFR	低速内部时钟状态 1 = LFINTOSC is ready 0 = LFINTOSC is not ready
0	-	-

2.1.17. PR2 寄存器, 地址 0x92

Bit	7	6	5	4	3	2	1	0
Name	PR2[7:0]							
Reset	1	1	1	1	1	1	1	1

Bit	Name	Function
7:0	PR2[7:0]	Timer 2 周期(比较)寄存器 (详见 timer2 描述章节)

2.1.18. WPUA 寄存器, 地址 0x95

Bit	7	6	5	4	3	2	1	0
Name	-	-	WPUA5	WPUA4	-	WPUA2	WPUA1	WPUA0
Reset	-	-	1	1	-	1	1	1

Bit	Name	Function
7-6, 3	NA	没实现, 读 0
5, 4, 2, 1, 0	WPUAx	Port A 弱上拉使能 1 = 使能 PORT A 端口弱上拉 0 = 断开 PORT A 端口弱上拉

2.1.19. IOCA 寄存器, 地址 0x96

Bit	7	6	5	4	3	2	1	0
-----	---	---	---	---	---	---	---	---

Name	-	IOCA[5:0]
Reset		6'h00

Bit	Name	Function
7-6	N/A	-
5-0	IOCA[5:0]	端口状态触发中断设置 1 = 使能端口状态触发中断 0 = 关闭端口状态触发中断

2.1.20. PCON 寄存器，地址 0x8E

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	/POR	/BOR
Reset	-	-	-	-	-	-	q	q

Bit	Name	Function
7:2	-	保留位，读 0
1	/POR	上电复位标志，低有效 0: 发生了上电复位 1: 没发生上电复位或者由软件置 1
0	/BOR	低电压复位标志，低有效 0: 发生了低电压复位 1: 没发生低电压复位或者由软件置 1

2.1.21 EEDAT 寄存器，地址 0x9A

Bit	7	6	5	4	3	2	1	0
Name	EEDAT[7:0]							
Reset	0	0	0	0	0	0	0	0

2.1.22 EEADR 寄存器，地址 0x9B

Bit	7	6	5	4	3	2	1	0
Name	EEADR[7:0]							
Reset	0	0	0	0	0	0	0	0

2.1.23. EECON1 寄存器，地址 0x9C

Bit	7	6	5	4	3	2	1	0
-----	---	---	---	---	---	---	---	---

Name	-	-	WREN3	WREN2	WRERR	WREN1	-	RD
Reset	-	-	0	0	x	0	-	0

Bit	Name	Function
7:4	-	保留位，读 0
5	WREN3	数据 EEPROM 写使能 3 和 WREN2、WREN1 结合使用
4	WREN2	数据 EEPROM 写使能 2 和 WREN3、WREN1 结合使用
3	WRERR	数据 EEPROM 写错误标志位 1: 在 EEPROM 编程周期发生了看门狗或者外部复位，中止 0: 在 EEPROM 编程周期正常完成
2	WREN1	数据 EEPROM 写使能 1 WREN3-1=111: 允许软件对 EEPROM 编程，编程完成后各位自动清 0 WREN3-1=其他值: 禁止软件对 EEPROM 编程
1	-	保留位，读 0
0	RD	数据 EEPROM 读控制位 此位是只写，读永远返回 0 写 1: 启动一次数据 EEPROM 读周期 写 0: 不启动读

2.1.24. EECON2 寄存器，地址 0x9D

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	WR
Reset	-	-	-	-	-	-	-	0

Bit	Name	Function
7:1	-	保留位，读 0
0	WR	数据 EEPROM 写控制位 读操作： 1: 数据 EEPROM 编程周期进行中 0: 数据 EEPROM 不处于编程周期 写操作： 1: 启动一次数据 EEPROM 编程周期 0: 无意义

2.1.25. UCFG0/UCFG1/UCFG2

软件不能访问 UCFG0、UCFG1、UCFG2，它们只在上电过程由硬件写入。

UCFG0, PROM 地址 0x2000。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
NA	CPB	MCLRE	PWRTEB	WDTE	NA	NA	FOSC0

位	名称	描述
Bit[7]	NA	保留位, 读 0
Bit[6]	CPB	1: PROM 内容不保护 0: 启动 PROM 内容保护, MCU 能读, 串口不能读
Bit[5]	MCLRE	1: PA3/MCLR 脚执行 MCLR 功能, 是复位脚 0: PA3/MCLR 脚执行 PA3 功能, 是数字输入引脚
Bit[4]	PWRTEB	1: PWRT 禁止 0: PWRT 使能 注意: 由于 PWRT 和看门狗复用了同一个计数器, 当 PWRT 计数溢出之后, 该计数器没有自动清 0。所以建议使能了 PWRT 的应用, 在系统初始化时先执行一下 CLRWDT 指令, 以清该计数器。
Bit[3]	WDTE	1: WDT 使能, 程序不能禁止 0: WDT 禁止, 但程序可通过设置 WDTCON 的 SWDTEN 位将 WDT 使能
Bit[2:1]	NA	保留位, 读 0
Bit[0]	FOSC0	1: INTOSC 模式, PA4 输出系统时钟的 4 分频, PA5 为 IO 引脚 0: INTOSCIO 模式, PA4 为 IO 引脚, PA5 为 IO 引脚

UCFG1, PROM 地址 0x2001。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-	-	-	-	-	RD_CTRL	LVREN1	LVREN0

位	名称	描述
Bit[7:3]	--	保留
Bit[2]	RD_CTRL	输出模式时读端口控制 1: 读数据端口返回的 PAD 上的值 0: 读数据端口返回的 Latch 上的值
BIT[1:0]	LVREN[1:0]	低电压复位选择 00: 低电压复位使能 其它: 禁止低电压复位

UCFG2, PROM 地址 0x2002。

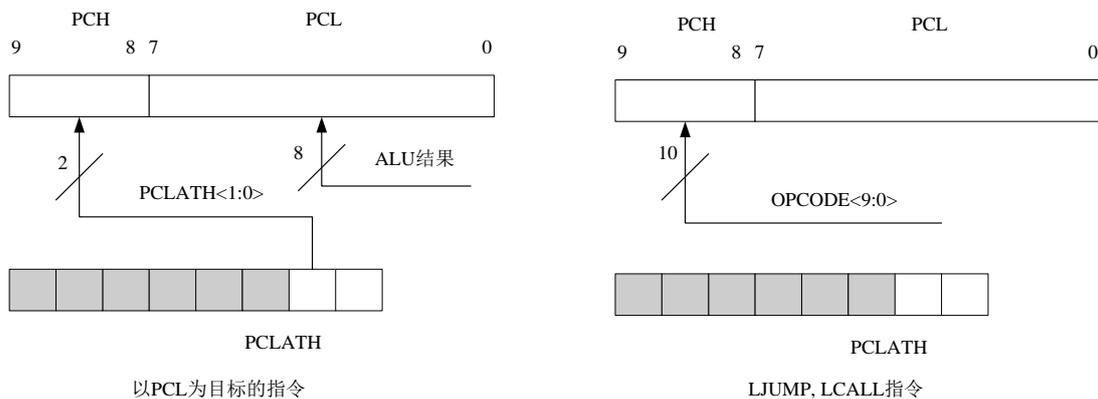
Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	LVRS[3:0]			

位	名称	描述
Bit[7:4]	-	保留位
Bit[3:0]	LVRS[3:0]	低电压复位阈值选择

数值	电压
其他值	保留
0011	2.0V
0100	2.2V
0101	2.5V
0110	2.8V
0111	3.1V
1000	3.6V
1001	4.1V

2.1.26. PCL 和 PCLATH

程序计数器（PC）为 10 位宽。其低 8 位来自可读写的 PCL 寄存器，高 2 位（PC<9:8>）来自 PCLATH，不能直接读写。只要发生复位，PC 就将被清 0。图显示了装载 PC 值的两种情形。注意图右边的 LCALL 和 LJUMP 指令，由于指令中的操作码为 11 位，而芯片的 PC 只有 10 位，所以这时 PCLATH 并不需要用到。



修改 PCL

执行任何以 PCL 寄存器为目标寄存器的指令将同时使程序计数器 PC<9:8>位被 PCLATH 内容取代。这样可通过将所需的高 2 位写入 PCLATH 寄存器来改变程序计数器的所有内容。

计算 LJUMP 指令是通过向程序计数器加入偏移量（ADDWR PCL）来实现的。通过修改 PCL 寄存器跳转到查找表或程序分支表（计算 LJUMP）时应特别谨慎。假定 PCLATH 设置为表的起始地址，如果表长度大于 255 条指令，或如果存储器地址的低 8 位在表的中间从 0xFF 计满返回到 0x00，那么在每次表起始地址与表内的目标地址之间发生计满返回时，PCLATH 必须均必须递增。

注意：PCLATH<3>并没有使用。

2.1.27. INDF 和 FSR 寄存器

INDF 不是物理存在的寄存器，对 INDF 进行寻址将产生间接寻址，可寻址范围为 0~255。

任何使用 INDF 寄存器的指令，实际上是对文件选择寄存器 FSR 所指向的单元进行存取。间接对 INDF 进行操作将返回 0。间接对 INDF 进行写将导致空操作（可能会影响状态标志位）。

3. 系统时钟源

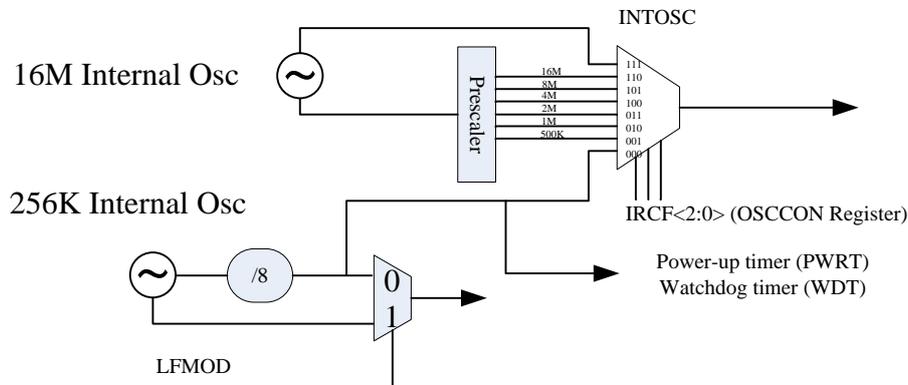


图 3.1 系统时钟源框图

本芯片包含 2 个振荡器作为各种时钟源。包括内部 16M 高速振荡器 (HFINTOSC)，内部 32K/256K (LFINTOSC) 低速振荡器。这些时钟或振荡器结合预分频器可以给系统提供各种频率的时钟源。

系统时钟源的预分频器比例由 OPTION 寄存器里的 IRCF<2:0> 位控制。

注意：

看门狗、系统时钟源 (IRCF=000) 以及 PWRT 统一使用 8 分频之后的输出，即 32KHz，而不管 LFMOD 为何值。

4. 复位时序

有以下几种不同的复位：

- A) 上电复位 POR
- B) WDT(看门狗)复位 – 在常规运行期间
- C) WDT(看门狗)唤醒 – 在睡眠期间
- D) /MCLR 管脚复位 – 在常规运行期间
- E) /MCLR 管脚复位 – 在睡眠期间
- F) 低电压 (BOR) 复位

有些寄存器是不被任何复位影响的；这些寄存器的状态在上电复位时是未知的，也不受复位事件影响。大多数其它寄存器都会在以下复位事件时恢复到其“复位状态”：

- 上电复位 POR
- WDT(看门狗)复位 – 在常规运行期间
- WDT(看门狗)复位 – 在睡眠期间
- /MCLR 管脚复位 – 在常规运行期间
- 低电压 (BOR) 复位

WDT(看门狗)睡眠唤醒不会造成和在常规运行状态下 WDT(看门狗)超时所造成的复位。因为睡眠唤醒本身

就是一种继续的意思，而不是复位/TF 和/PF 位的设置和清零在不同复位条件下的动作是不同的。具体可参考表 4.1 和 4.2。

/MCLR 管脚背后的电路带有防抖功能，能够滤除一些干扰造成的尖细脉冲信号。

下图为复位电路的总体概述框图。

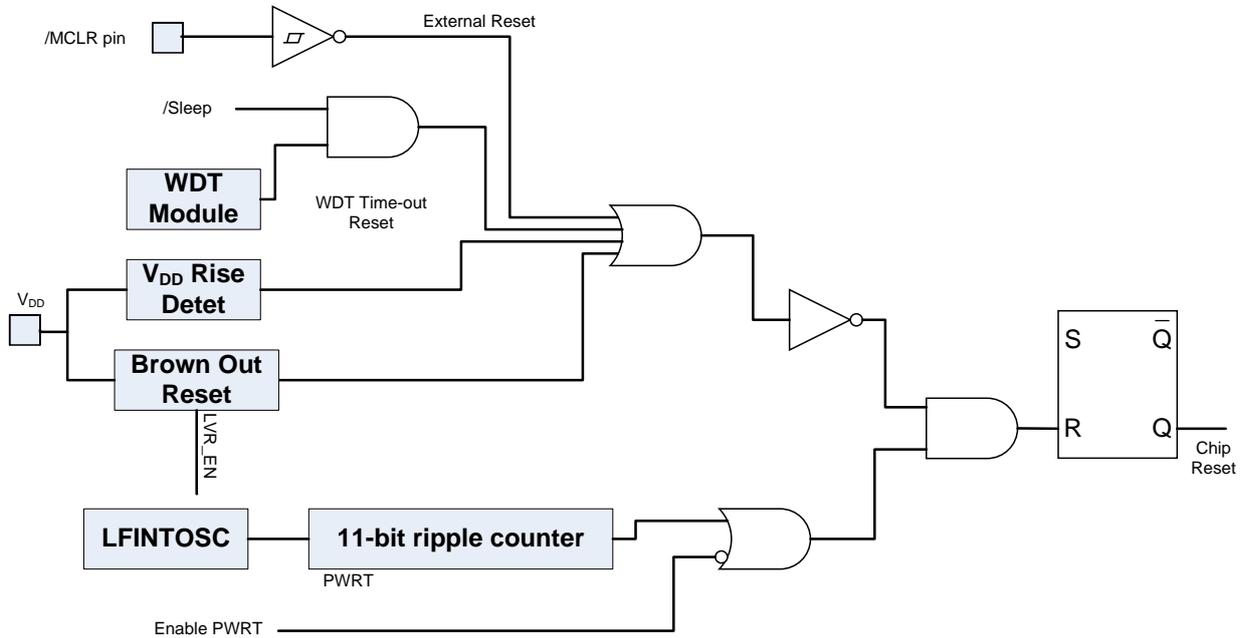


图 4.1 复位功能框图

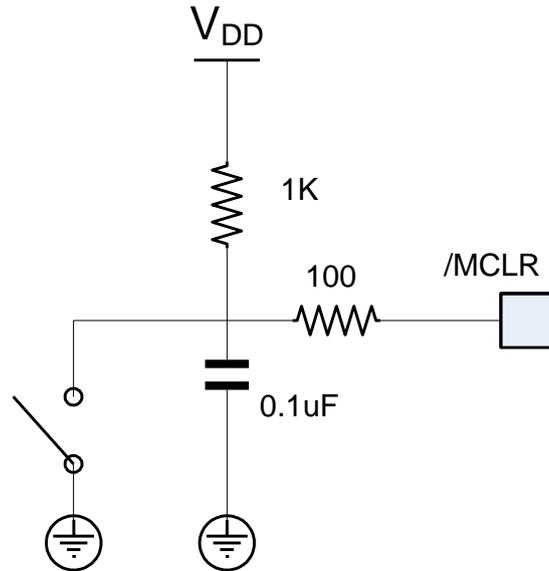
4.1. POR 上电复位

片上的 POR 电路会将芯片保持在复位状态直到 VDD 电源电压达到足够高，为充分利用片内复位电路的功能，用户可以简单地直接在 VDD 和/MCLR 之间接一个电阻。这样外部就无需任何 RC 复位电路。不过这要求 VDD 电压上升时间为最大。

上电完成后，系统复位不会立即释放，还要等一个约 8ms 的延时，期间数字电路保持在复位状态。

4.2. 外部复位 MCLR

需要注意的是，WDT 复位不会把/MCLR 管脚拉低。在/MCLR 管脚上施加超过指标的电压（例如 ESD 事件）会造成/MCLR 复位，而且在管脚上产生超标的大电流，因此我们推荐用户不再直接用一个电阻将/MCLR 和 VDD 连接起来，而是采用以下电路。



在芯片的 CONFIG OPTION 寄存器中有一个 MCLRE 使能位, 将此位清零会使得复位信号由芯片内部产生。当此位为 1 时, 芯片的 PA3/MCLR 脚成为外部复位脚。在这个模式下, /MCLR 脚上有个对 VDD 的弱上拉。

4.3. PWRT（上电计时器）

PWRT 为上电复位, 低电压复位提供一个固定的 64ms（正常情况下）的定时。这个定时器由内部慢时钟驱动。芯片在定时器超时之前都是被保持在复位状态。这段时间能保证 VDD 上升到足够高的电压使得系统能正常工作。PWRT 也可以通过系统 CONFIG 寄存器来使能。在开启低电压复位功能时, 用户应该也打开 PWRT。PWRT 定时是由 VDD 电压超过 VBOR 门限事件启动的。另外需要注意的是, 由于由内部慢时钟驱动, 定时的实际时间长度是随温度, 电压等条件变化而变化的。这个时间不是一个精准参数。

注意:

PWRT 上电复位延时定时器和 WDT 复用同一个计数器, 在 B 版之前, 如果芯片配置了使能 PWRT, 在 PWRT 溢出时该计数器没有自动清 0, 所以应用程序在初始化 WDT 时要执行一条 CLRWDT。

B 版（包括 B 版）之后, PWRT 溢出时会把它清 0。

4.4. BOR 低电压复位

低电压复位由 UCFG1<1:0>位来控制。低电压复位就是指当电源电压低于 VBOR 门限电压时所产生的复位。不过当 VDD 电压低于 VBOR 不超过 TBOR 时间时, 低电压复位可能不会发生。

VBOR 电压在芯片出货之前需要校准, 校准可通过串口写入内部校准寄存器来完成。

如果 BOR（低电压复位）是使能（UCFG1<1:0>=00）的, 那么最大 VDD 电压上升时间的要求就不存在。BOR 电路会将芯片控制在复位状态, 一直到 VDD 电压达到 VBOR 门限电压以上。需要注意的是, 当 VDD 低于系统能正常工作的门限时, POR 电路并不会产生复位信号。在 BOR 使能的情况下, 如果要 BOR 电路产生复位信号, VDD 电压须低于所设阈值(VBOR)并保持 125us 以上。

4.5. 超时动作

在上电过程中，芯片内部的超时动作顺序按以下流程执行：

POR 结束后启动 PWRT 计时

由于计时是由 POR 脉冲结束启动的，如果/MCLR 在低电平状态下保持足够长的时间，超时事件就会发生。那么将/MCLR 拉高会让 CPU 立即开始执行。这在测试或者需要多个 MCU 同步的情况下会很有用。

Power Control Register (PCON)

PCON 寄存器里有 2 位指示哪一种复位发生了。Bit0 是/BOR 指示位，其在上电复位是未知态，软件必须将其置 1，然后检查其是否为 0。Bit1 是/POR 指示位，其在上电复位后为 0，软件必须将其置 1。

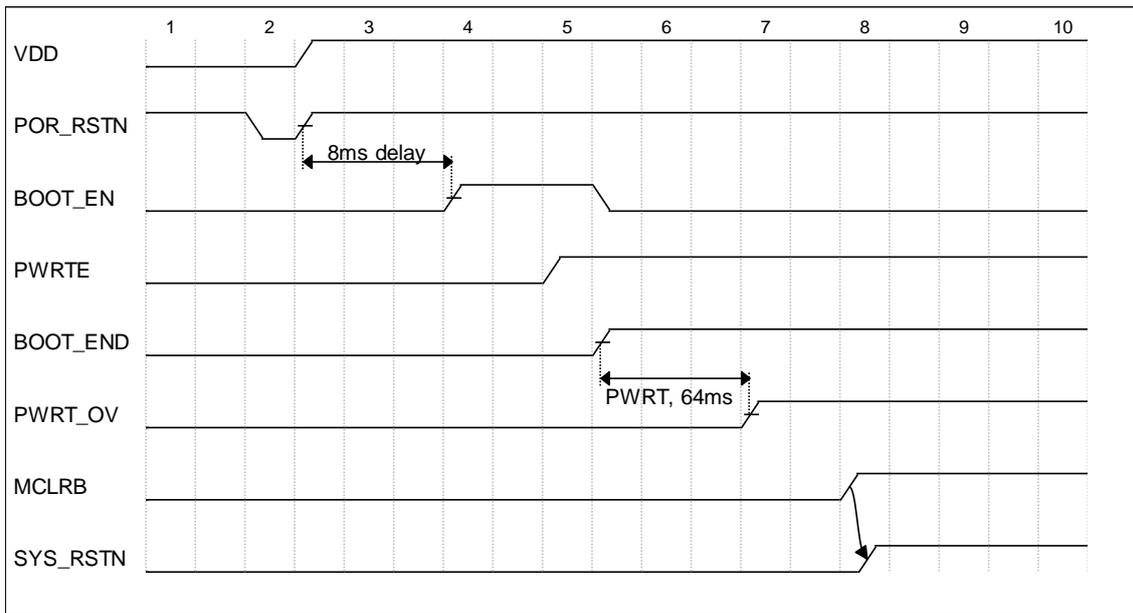


图 4.2 上电复位，使用了 MCLR

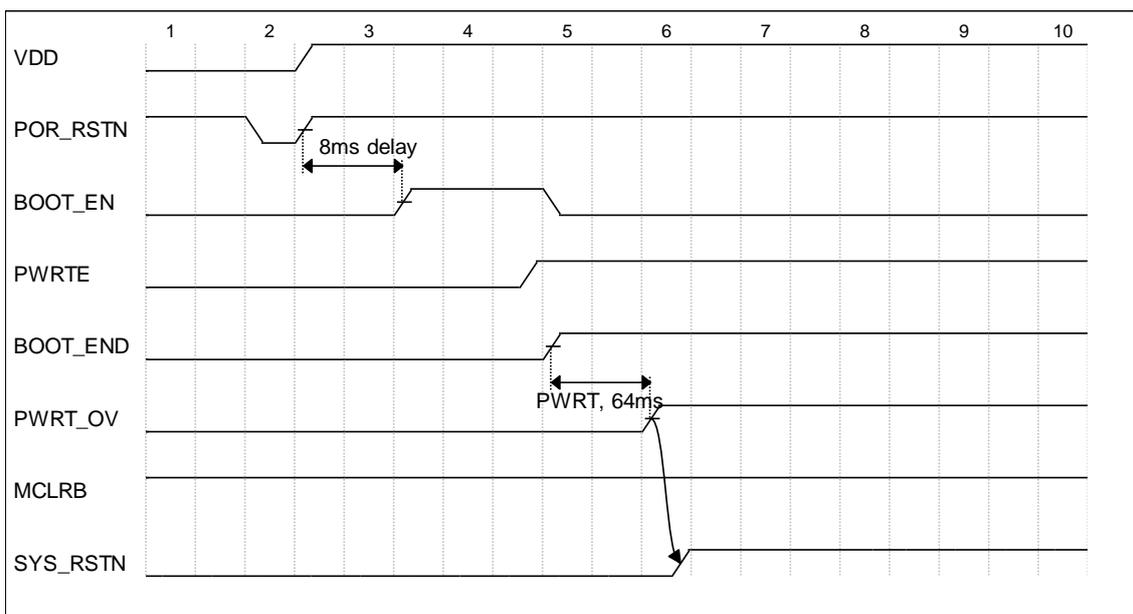


图 4.3 上电复位，没使用 MCLR

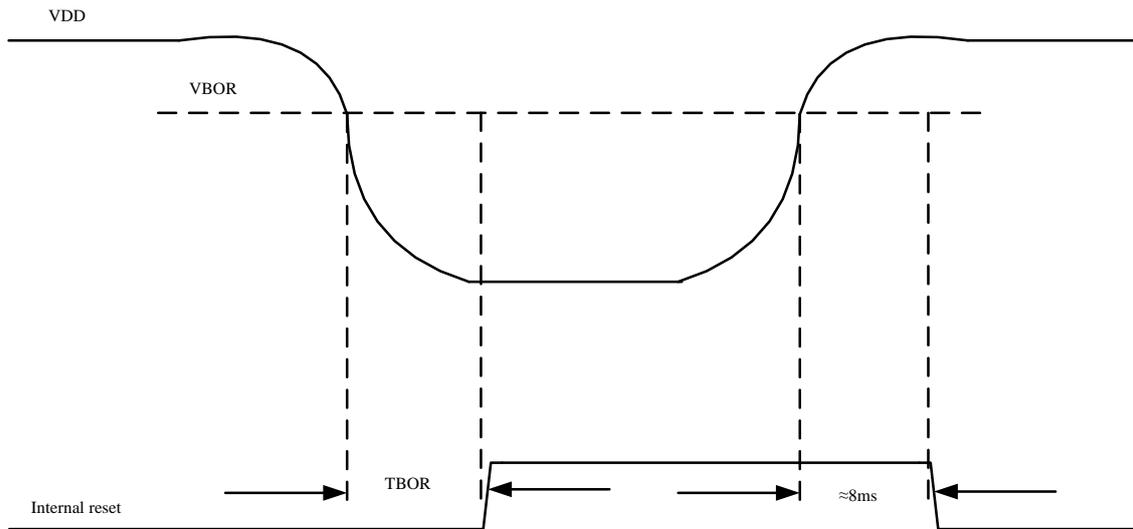


图 4.4 BOR 复位

注意:

1. 上电复位或低电压复位后，并且在 **PWRTEB (UCFG0.4)** 为低时，**PWRT** 有效。它是 **2048** 个内部慢时钟周期，约 **64ms**;
2. **TBOR** 时间约为 **157us**;
3. 电压恢复正常之后，内部复位不会立即释放，而是要等约为 **8ms** 的时间。

振荡器配置	上电复位		低电压复位		睡眠醒来
	/PWRTE=0	/PWRTE=1	/PWRTE=0	/PWRTE=1	
INTOSC	TPWRT	-	TPWRT	-	-

表 4.1 各种情况下的超时

/POR	/BOR	/TF	/PF	条件
0	x	1	1	POR
u	0	1	1	BOR
u	u	0	u	WDT 复位
u	u	0	0	WDT 唤醒
u	u	u	u	常规运行下/MCLR 复位
u	u	1	0	睡眠下/MCLR 复位

表 4.2 STATUS/PCON 位及其意义(u-没变化, x-未知)

4.6. 关于 WDT 复位

在 M 版之前，WDT 复位不会引发 BOOT 过程，复位源释放后，CPU 开始执行指令；

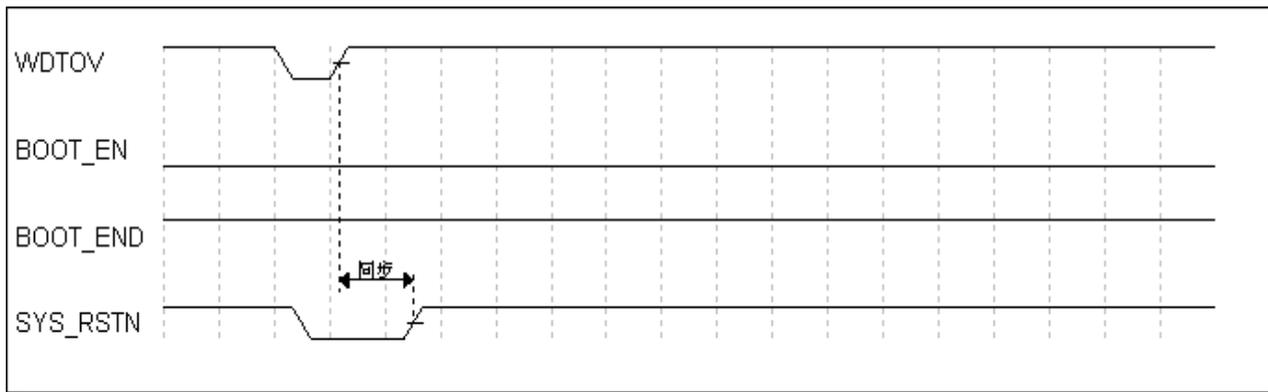


图 4.5 WDT 复位不产生 BOOT

在 M 版（包括 M 版在内），WDT 复位会引发 BOOT 过程，它跟上电复位一样，复位源释放之后，复位控制器还将延时 8ms，

然后对 UCFG0、UCFG1 进行配置，这些步骤完成后，系统复位才真正释放，CPU 开始执行指令。如图 4.6 所示：

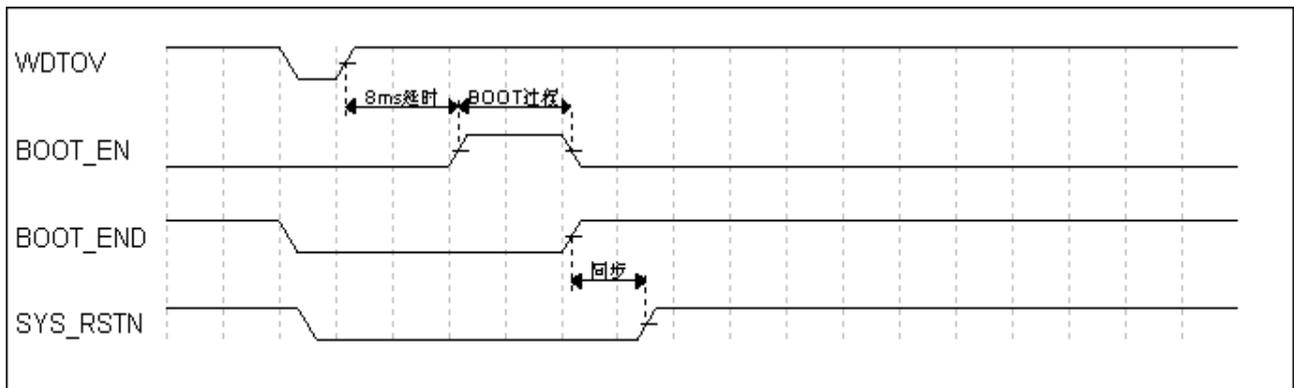


图 4.6 WDT 复位引发 BOOT 过程

5. BOOT

系统复位要等 BOOT 结束后才能释放。

6. 烧录和控制串口

具备一个烧录和控制串口。这个接口为用户和开发人员使用此芯片带来方便。具体请参考相关 调试文档。

7. 看门狗定时器

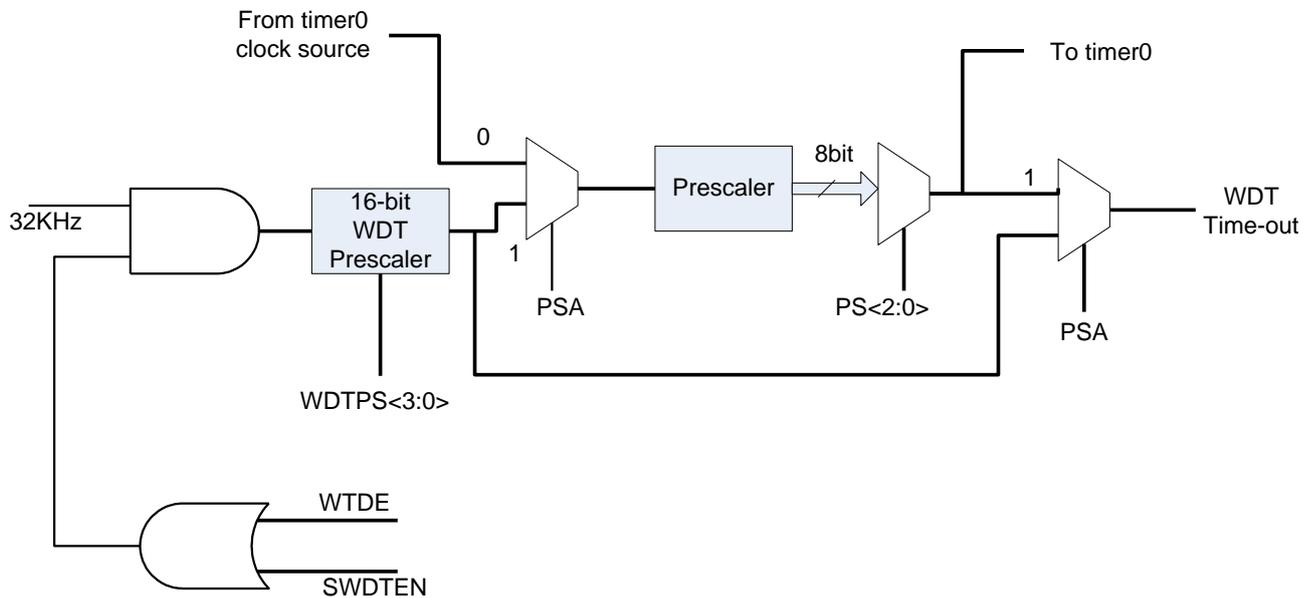


图 7.1 看门狗和定时器 0 框图

7.1. 看门狗

看门狗的时钟源为内部慢时钟（32KHz），它是一个 16 位的计数器，和定时器 0 共用一个 8 位的预分频器，使能位位于配置寄存器 CONFIG OPTION0 的第 3 位，WDTEN，为 1 时表示使能看门狗，为 0 时禁止，由外部串口写入控制。

清看门狗指令 CLRWDT、SLEEP 会清除看门狗计数器。

在使能了看门狗的情况下，MCU 睡眠时看门狗溢出事件可以作为一个唤醒源，而 MCU 正常工作时作为一个复位源。

条件	看门狗状态
WDTE=0	清零
CLRWDT 指令	
进入 SLEEP、退出 SLEEP 时刻	

8. 定时器 0

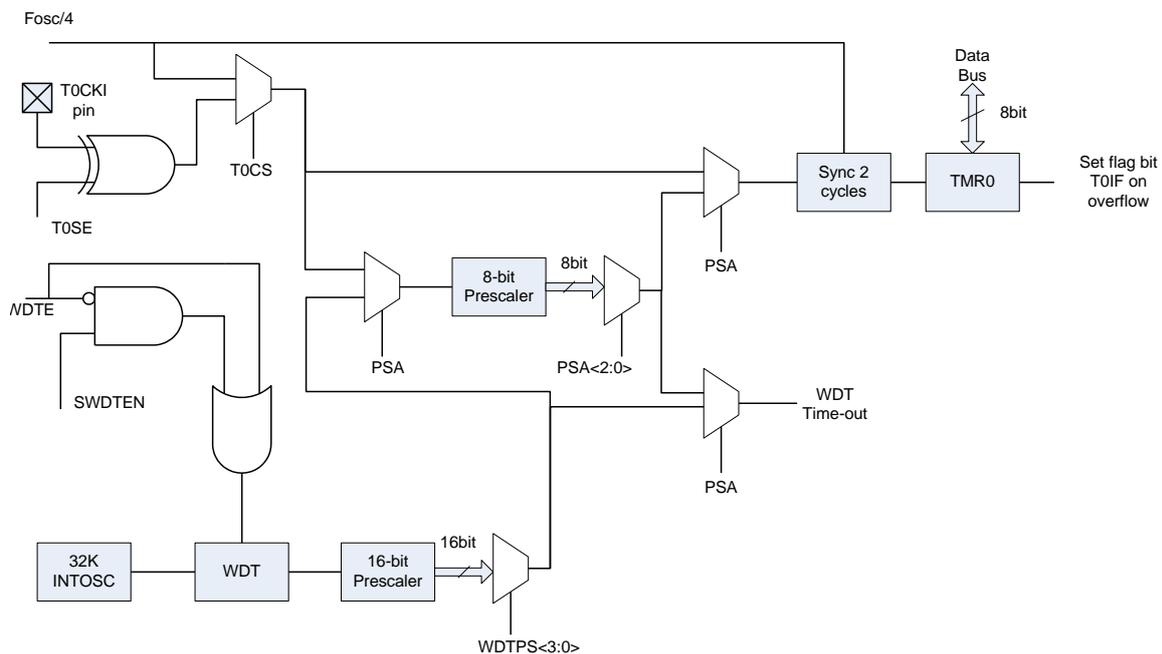


图 8.1 看门狗和定时器 0 框图

8.1. Timer 0

定时器 0 为 8 位，可配置为计数器或定时器使用，当作为外部事件（T0CKI）计数器时，可以配置为上升沿或者下降沿计数。作为定时器时，其计数时钟为系统时钟的 4 分频，即每一指令周期递增一次。

有一个与 WDT 共用的 8 位预分频器，PSA 为 0 时该预分频器分配给定时器 0 使用。

注意：

1. 当改变 PSA 的值时，硬件会自动把预分频器清 0。

8.2. Timer 0 定时器模式

该模式下，定时器 0 在每个指令周期加 1(不带预分频)。软件可以清零 OPTION 寄存器里的 T0CS 位以进入定时器模式。当软件对 TMR0 进行写操作时，定时器在写后面 2 个周期内不会递增。

8.3. Timer 0 计数器模式

该模式下，定时器 0 由每个 T0CKI 管脚的上升沿或下降沿触发加 1(不带预分频)。具体哪一钟沿触发由 OPTION 寄存器里的 T0SE 位决定。软件可以将 OPTION 寄存器里的 T0CS 位置 1 以进入计数器模式。

8.3.1. 软件可配置预分频电路

芯片在定时器 0 和 watchdog 定时器前面有一个分频电路，可以分配给 Timer0 或者 watchdog 定时器用，但二者不能同时使用这个预分频电路。具体分配给 Timer0 还是 watchdog 由 OPTION 寄存器里的 PSA 位决定，PSA 为 0 时，预分频分配给 Timer0 使用。在 Timer0 预分频模式下，总共有 8 个预分频比（1:2 到 1:256）可以通过 OPTION 寄存器里的 PS[2:0]位设置。

预分频电路既不可读也不可写。任何对 TMR0 寄存器的写动作会清零预分频电路。

当预分频电路分配给 watchdog 时，1 条 CLRWDT 指令会清零预分频电路。

8.3.1.1 在定时器和 watchdog 之间切换预分频电路

由于分频电路可以分配给 Timer0 或者 watchdog 定时器用，在二者之间切换预分频器是有可能导致误复位。在将预分频电路从分配给 TMR0 切换到分配给 watchdog 时，请务必执行以下指令顺序：

```
BANKSEL TMR0 ;
CLRWDT ;Clear WDT
CLRR TMR0 ;Clear TMR0 and
; prescaler
BANKSEL OPTION_REG ;
BSR OPTION_REG,PSA ;Select WDT
CLRWDT ;
;
LDWI b'11111000' ;Mask prescaler
ANDWR OPTION_REG,W ; bits
IORWI b'00000101' ;Set WDT prescaler
STR OPTION_REG ; to 1:32
```

在将预分频电路从分配给 watchdog 切换到分配给 TMR0 时，请务必执行以下指令顺序：

```
CLRWDT ;Clear WDT and
;prescaler
BANKSEL OPTION_REG ;
LDWI b'11110000' ;Mask TMR0 select and
ANDWR OPTION_REG,W ; prescaler bits
IORWI b'00000011' ;Set prescale to 1:16
STR OPTION_REG ;
```

8.3.2. 定时器 0 中断

芯片在定时器 0 从 0xFF 溢出到 0x00 时会置起 TOIF 标志，并产生中断（如果使能了的话）。注意，timer0 中断无法唤醒 CPU 因为在睡眠状态下，定时器是被冻结的。

8.3.3. 用外部时钟驱动定时器 0

在计数其模式下，T0CKI 管脚输入和 Timer0 的寄存器之间的同步是由在 Q2, Q4 内部时钟相位采用预分频输出实现的。所以外部时钟源的周期的高电平时间和低电平时间必须满足相关时序要求。

9. 定时器 2

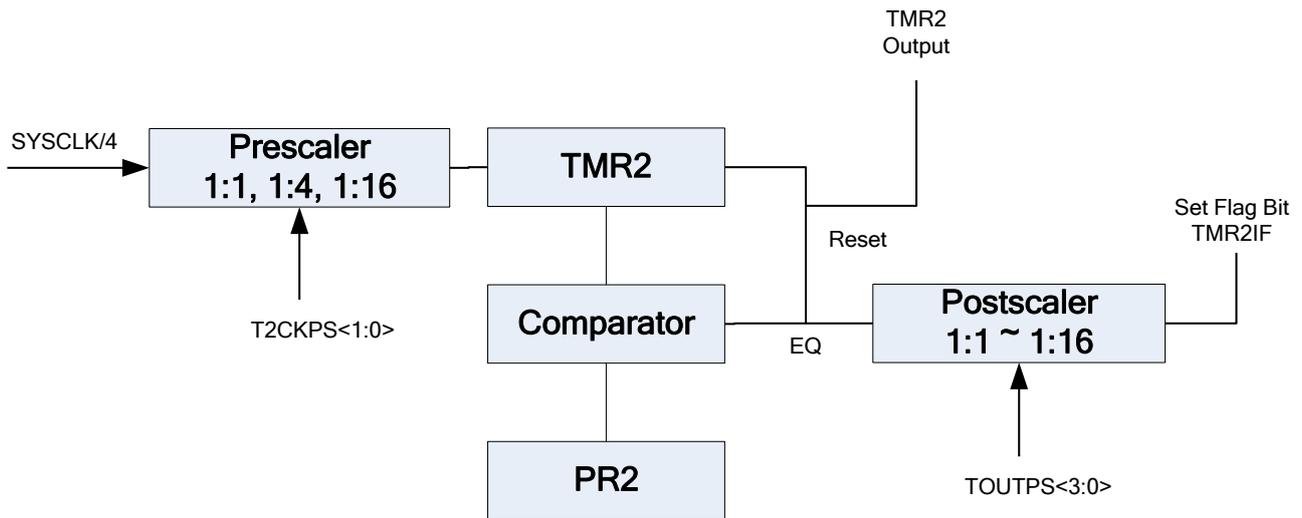


图 9.1 定时器 2 框图

9.1. Timer2

定时器 2 为 8 位定时器包含以下功能:

- 时钟源为指令时钟，即系统时钟的 4 分频
- 8 位计数寄存器
- 8 位周期寄存器
- TMR2 值等同 PR2 时产生中断
- 1:1, 1:4, 1:16 预分频比
- 1:1~1:16 后分频比

图 9.1 为 Timer2 的整体框图。

当 TMR2 等于 PR2 时，TMR2 会在下一个时钟周期归零，后分频开始起作用。后分频结果会用来设置 TMR2IF 标志位。

TMR2 和 PR2 都是可读写寄存器。在复位时，他们的值分别是 0 和 0xFF。

预分频和后分频计数器会在写以下寄存器时清零：

- 写 TMR2
- 写 T2CON
- 任何 reset 动作

注：写 T2CON 并不会清零 TMR2 寄存器。

10. 数据 EEPROM

片内集成有 256 个字节的 EEPROM，通过 EEADR 进行寻址访问。软件可通过 EECON1 和 EECON2 对 EEPROM 进行编程操作，硬件实现了擦除和编程的自定时功能，无需软件查询，节省有限的代码空间，同时利用此特性，启动编程周期之后可以进入睡眠模式，以降低功耗。

数据 EEPROM 在使用（无论是读还是写）之前必须进行以下初始化操作：在未使用到的 EEPROM 某个单元写两次 0xAA，后续程序不要对此单元操作。如：

SYSTEM_INIT:

.....

.....

LDWI 0x55

STR EEPROM_ADDR

LDWI 0xAA

STR EEPROM_DATA

LCALL EEPROM_write

LCALL EEPROM_write

.....

10.1. 编程数据 EEPROM 步骤

- a. 把 INTCON 的 GIE 位清 0；
- b. 判断 GIE 是否为 1，是则重复 a 步骤，否则可以进行下一步；
- c. 往 EEADR 写入目标地址；
- d. 往 EEDAT 写入目标数据；
- e. 把位 WREN3/WREN2/WREN1 全部置 1；
- f. 把位 WR 置 1（EECON2.0，此后 WR 会维持高）；
- g. 写过程不能改变 WREN3/2/1 的值，否则编程终止；
- h. 等大概 2ms 之后编程自动完成，WR 自动清 0，WREN3、WREN2、WREN1 自动清 0；
- i. 如果想再次编程，重复步骤 c~h 即可；

注意：

1. 编程过程中读操作无效。

10.2. 关于编程周期

启动数据 EEPROM 的编程操作后，2ms 的编程计时开始，在这段时间内，CPU 并不会暂停，而是继续执行程序。

10.3. 读 EEPROM 步骤

要读取数据存储单元，用户必须将地址写入 EEADR 寄存器，然后将 EECON1 寄存器的控制位 RD 置 1。在紧接着的下一周期，EEDAT 寄存器就被 EEPROM 数据写入。因此该数据可由下一条指令读取。EEDAT 将保持这个值直到用户下一次从该单元读取或向该单元写入数据时（在写操作过程中）。

下面是读取 EEPROM 的一段示例程序：

```
BANKSEL EEADR
LDWI dest_addr
STR EEADR
BSR EECON1, RD
LDR EEDAT, W
```

11. 慢时钟测量模式

此功能可以比较精准的测量内部慢时钟周期。

在此模式下，TIMER2 的预分频、后分频自动变为 1:1，组成一个 12 位的定时器，其时钟源为系统时钟，启动测量后在 1 个或 4 个慢时钟周期进行计数。计数结束后结果自动存到 SOSCPR 寄存器。

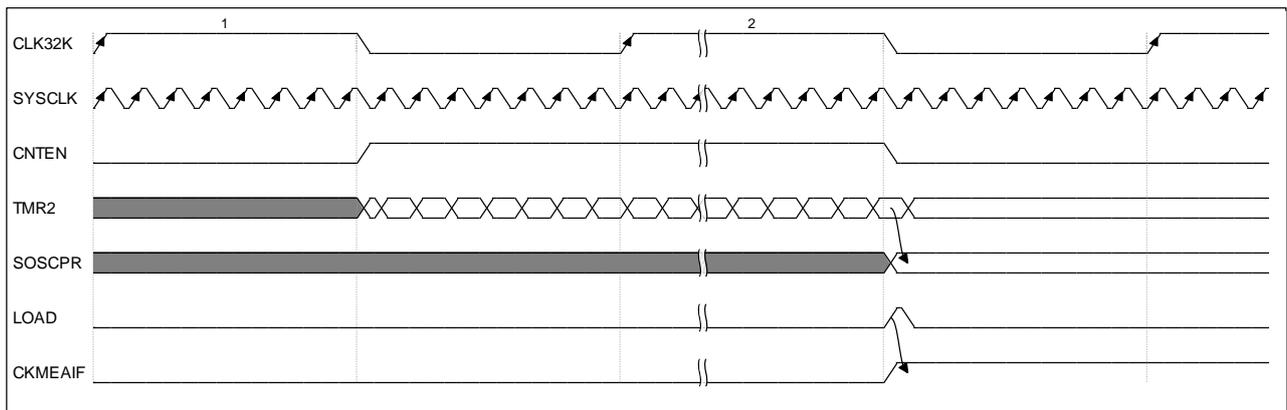


图 11.1 慢时钟测量时序图

操作步骤：

1. 把 T2CON.2 置 1，使能 TIMER2；
2. 如果选择 4 次平均，则把 MSCKCON.2 置 1，否则把它清 0；
3. 置位 MSCKCON.1，开始测量；
4. 测量结束后 MSCKCON.1 自动清 0，中断标志置 1；
5. 可以用查询中断标志位 CKMEAIF 或中断的方式等待结果，此时，SOSCPR 的值即为测量结果。

注意：

1. 做慢时钟测量时，最好把系统时钟选择为 16M，这样分辨率更高，测量结果也更接近实际；
2. 当 MSCKCON.1 为 1 时，表示将测量 4 个慢时钟周期，硬件没有做移位操作。故软件应对其右移 2 位以得到 1 个慢时钟周期的结果。

12. 中断模式

总共有 5 种中断源：

- PA2/INT 管脚进来的外部中断
- Timer0 溢出中断
- PORTA 变化中断
- Timer2 比对相等中断
- EEPROM 数据写中断

中断控制寄存器（INTCON）和外围中断请求寄存器（PIR1）记录了中断标志位。INTCON 同时也包含全局中断使能位 GIE。

当中断被服务后，以下动作自动发生：

- GIE 被清零，从而关闭中断
- 返回地址被推上堆栈
- 程序指针被加载 0004h 地址

中断返回指令，RETFIE，退出中断函数时同时设置 GIE 位，重新使能未屏蔽的中断。

INTCON 寄存器包含以下中断标志位：

- INT 管脚中断
- PORTA 变化中断
- Timer0 溢出中断

PIR1 中包含着外围中断标志位。PIE1 中包含着其对应的中断使能位。

外部中断包括 INT 管脚进来的或者 PORTA 变化中断的延时一般为 3 到 4 个指令周期。具体视中断发生的实际情况而定。

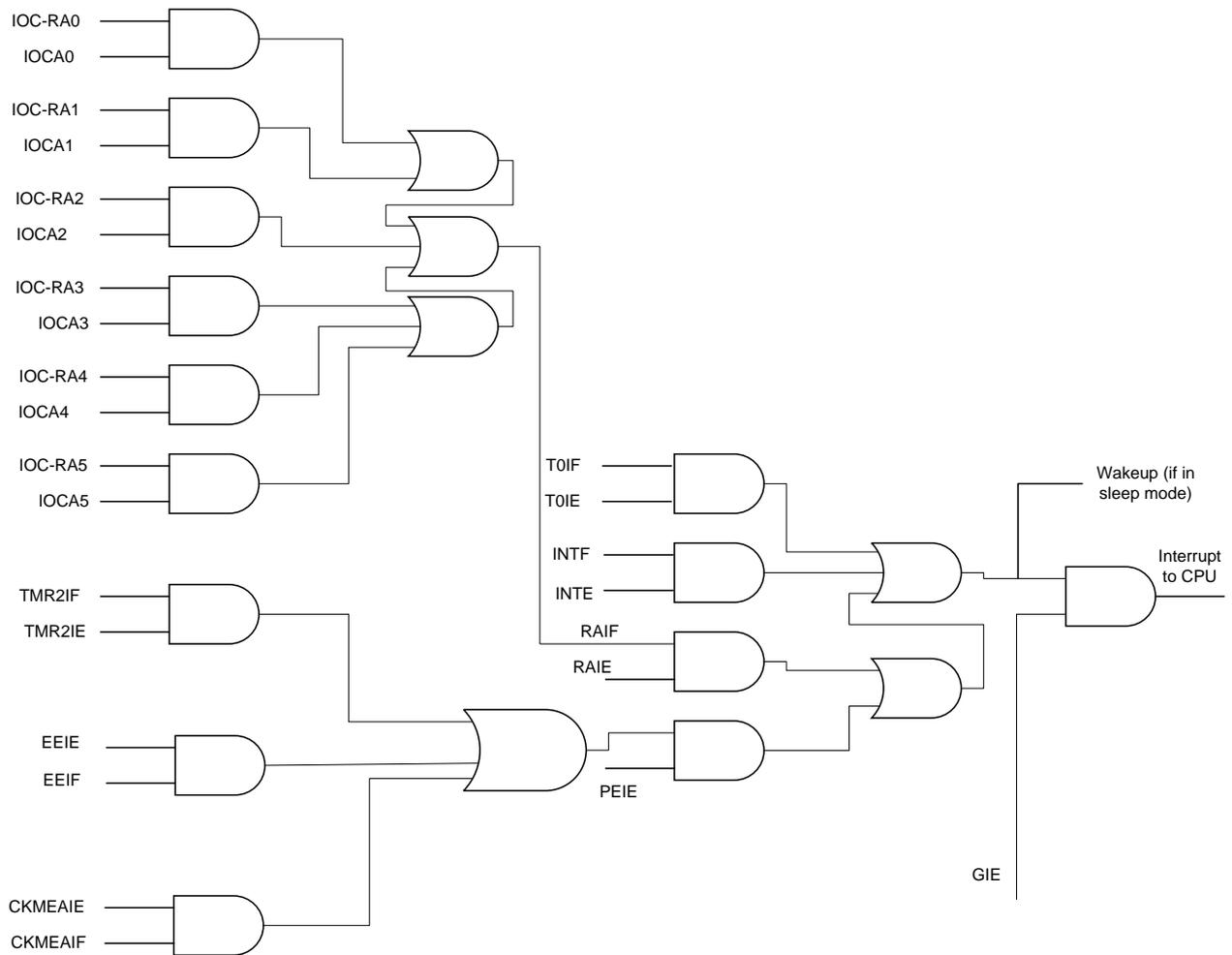


图 12.1 中断产生电路架构框图

12.1. 中断过程中的现场保存

在中断过程中，只有返回 PC 被自动保存在堆栈上。一般来说，用户可能需要保存重要的寄存器值在堆栈上，例如 W，STATUS 寄存器等。这些必须由软件来完成。临时寄存器 W_TEMP 和 STATUS_TEMP 应该被放置在 40~7F 的最后 64byte 里。这 64 个公共 RAM 落在两个页区间，因此可以稍微节省代码。

13. 睡眠省电模式

芯片在执行完 SLEEP 指令后进入睡眠状态。

为了达到最低睡眠功耗，软件应该将所有 IO 置高或低，而且没有外部电路从 IO 耗电。I/O 作为输入的，外部电路应将其拉高或拉低，避免翻转耗电。/MCLR 应该在高电平。

13.1. 唤醒模式

以下事件可以唤醒芯片：

- /MCLR 管脚上有外部复位
- WDT 超时
- PA2/INT 管脚上有中断，PORTA 变化或其他外围中断

清除看门狗指令 CLRWDT、SLEEP 会清除看门狗计数器。

在使能了看门狗的情况下，MCU 睡眠时看门狗溢出事件可以作为一个唤醒源，而 MCU 正常工作时作为一个复位源。

13.2. 看门狗唤醒

看门狗工作在内部慢时钟（32KHz），它是一个 10 位的计数器，和定时器 0 共用一个 8 位的预分频器，使能位位于配置寄存器 UCFG0 的第 3 位，WDTEN，为 1 时表示使能看门狗，为 0 时禁止。

清除看门狗指令 CLRWDT、SLEEP 会清除看门狗计数器。

在使能了看门狗的情况下，MCU 睡眠时看门狗溢出事件可以作为一个唤醒源，而 MCU 正常工作时作为一个复位源。

14. I/O 端口

本芯片共包含 6 个 GPIO。这些 IO 除了作为普通输入/输出端口以外还通常具备一些与内核周边电路通讯的功能，具体见下。

14.1. PORTA 端口和 TRISA 寄存器

PORTA 是一个 6 位双向端口。与其相应的进出方向寄存器就是 TRISA 寄存器。(不过注意到这里第 3 位没有用到，因为 PORTA[3]为单输入方向端口。在 TRISA 寄存器中将某一位设置为“1”会将该对应 PORTA 端口设置为输入端口(此时，输出驱动电路会被关断)。反之，将某一位设置为“0”会将该对应 PORTA 端口设置为输出端口。在置为输出端口时，输出驱动电路会被打开，输出寄存器里的数据会被放置到输出端口。在 PORTA 上进行读动作时，PORTA 内容会是反映输入端口的状态。在 PORTA 上进行写动作时，PORTA 内容会被写入输出寄存器。所有的写操作都是“读-更改-写”这样一个微流程，即数据被读，然后更改，再写入输出寄存器的过程。当 MCLRE 为 1 时，PORTA[3]读的值为 0。

14.2. 端口的其他功能

芯片在 PORTA 的每个端口都有一个状态变化中断选项和弱上拉选项。

14.2.1. 弱上拉

PORTA 的每个端口(除了 PORTA[3])都有一个可以单独设置的内部弱上拉功能。控制 WPUAx 寄存器里的位就可使能或关断这些弱上拉电路。当 GPIO 被设置为输出时，这些弱上拉电路会被自动关断。弱上拉电路在上电复位期间可以被置为关断。这是由 OPTION 寄存器中的/PAPU 位决定的。PORTA[3]内部也有弱上拉功能，它是在将 PORTA[3]设置为/MCLR 功能时自动使能的。当 PORTA[3]被设置为 GPIO 时，该弱上拉电路被自动关断。

14.2.2. 状态变化中断

PORTA 的每个端口都可以被单独设置成一个中断源(端口状态变化触发中断)。控制 IOCAx 寄存器里的位就可使能或关断这些端口的中断功能。端口状态变化触发中断的功能在上电复位时无效的。当端口状态变化触发中断的功能被使能时，当前端口电平值会被与上次读动作所读取数据寄存器的旧值作对比。所有错误匹配结果会被或在一起形成中断标志位 INTCON 寄存器中的 PAIF 标志位。该中断可以将芯片从睡眠状态中唤醒。用户需要在中断服务程序中执行以下程序来清除该标志位：

- A) 对 PORTA 进行一次读或写得动作，这将结束任何错误匹配的状态。
- B) 清零 PAIF 标志位。

错误匹配的条件会一直设置 PAIF 位。对 PORTA 做一次读就可以结束任何错误匹配的状态，使得 PAIF 能

被清零。数据寄存器里保持的上一次读的值不会被/MCLR 或低电压复位所影响。只要错误匹配状态存在，PAIF 位就会被置 1。

14.2.3. 端口描述

PORTA 的每个端口都包含着不同的复用功能。其具体功能和控制在这一节里描述。

14.2.3.1. PORTA[0]

图 14.1 描述了此端口的内部电路结构。PA[0]可以被配置为以下功能端口：

- GPIO
- 调试时钟接口

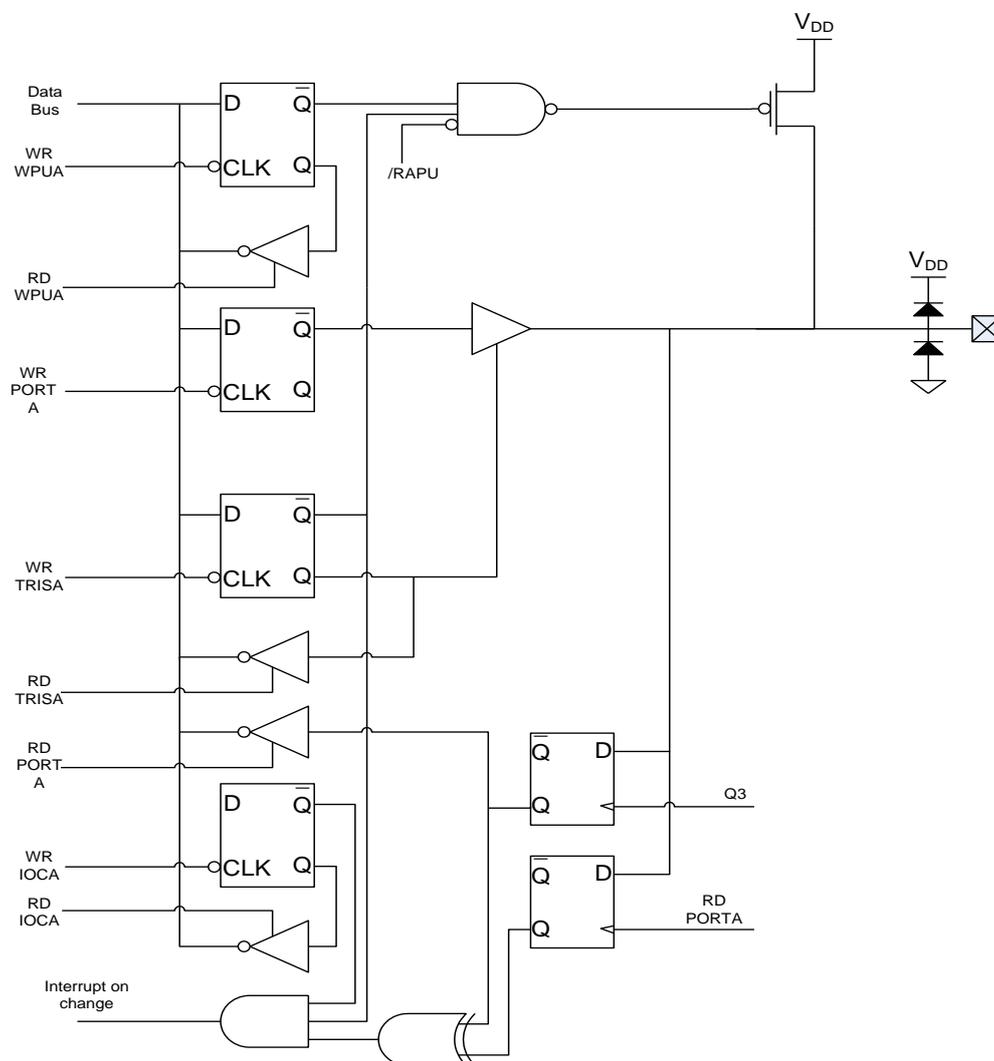


图 14.1 PA0 架构框图

14.2.3.2. PORTA[1]

图 14.2 描述了此端口的内部电路结构。PA[1]可以被配置为以下功能端口：

- GPIO
- 调试数据接口

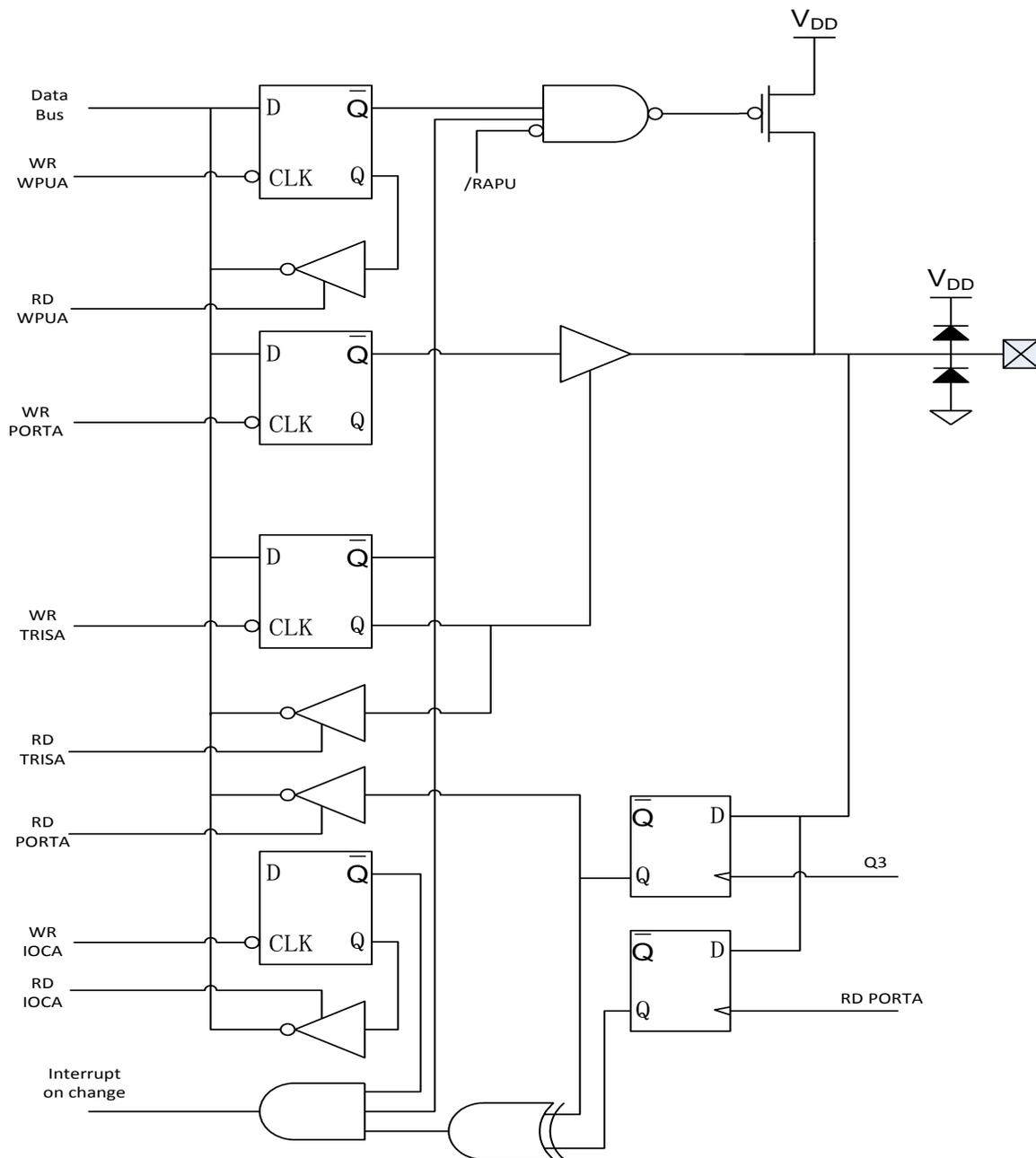


图 14.2 PA1 架构框图

14.2.3.4. PORTA[3]

图 14.4 描述了此端口的内部电路结构。PA[3]可以被配置为以下功能端口：

- 数字输入
- 外部复位

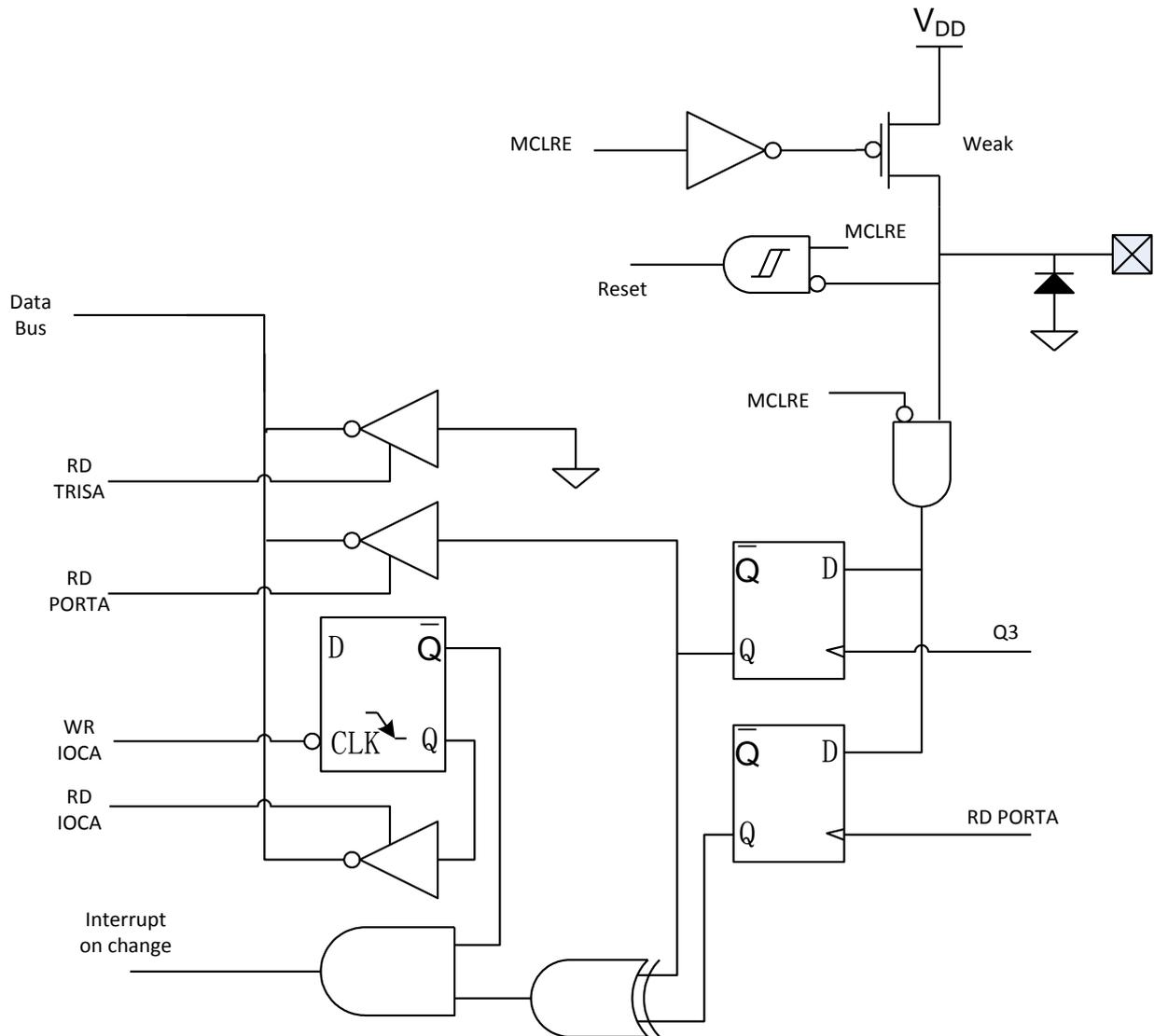


图 14.4 PA3 架构框图

14.2.3.5. PORTA[4]

图 14.5 描述了此端口的内部电路结构。PA[4]可以被配置为以下功能端口：

- GPIO
- 指令时钟输出
- 模拟测试信号输入输出

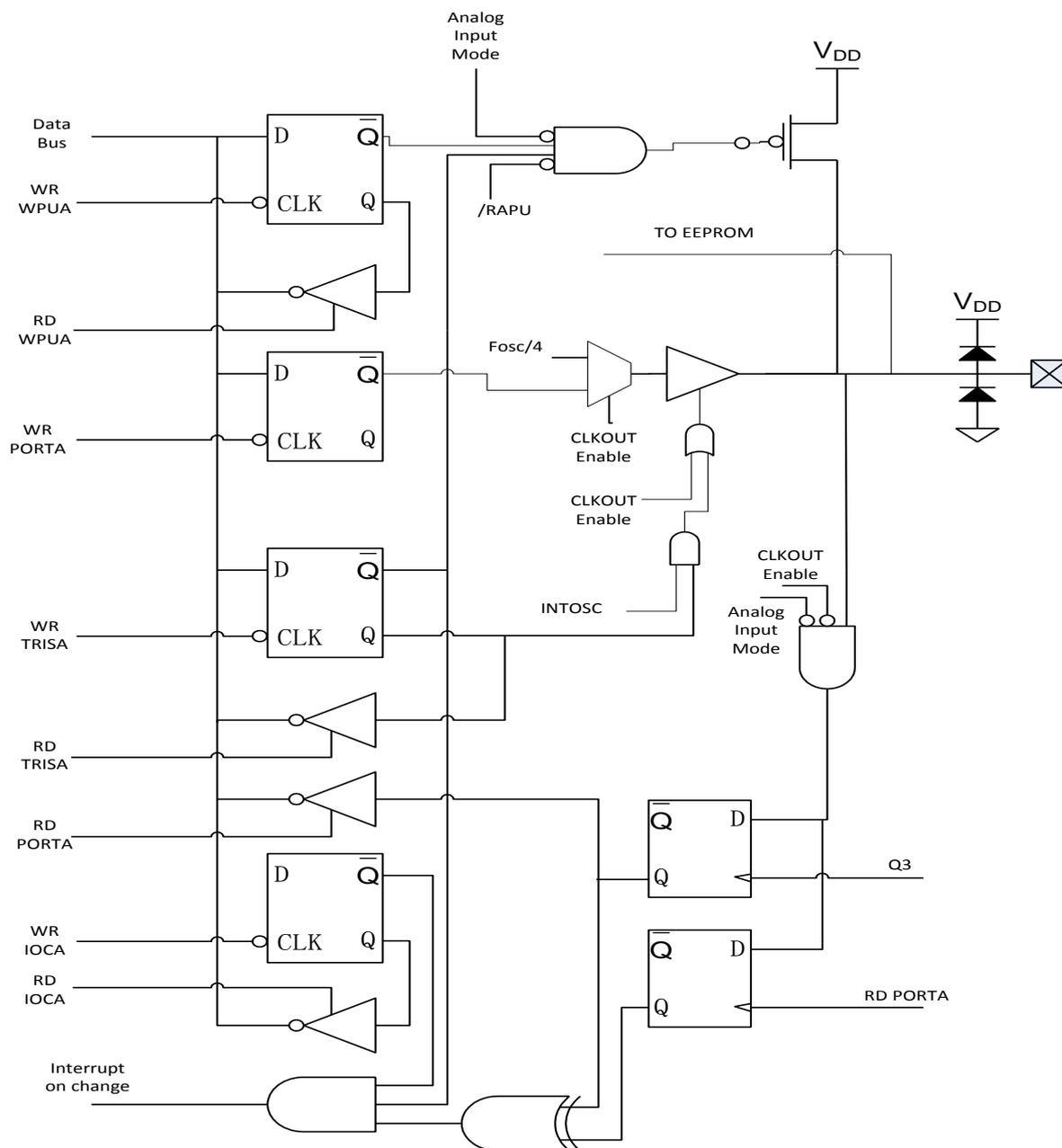


图 14.5 PA4 架构框图

14.2.3.6. PORTA[5]

图 14.6 描述了此端口的内部电路结构。PA[5]可以被配置为以下功能端口:

- GPIO

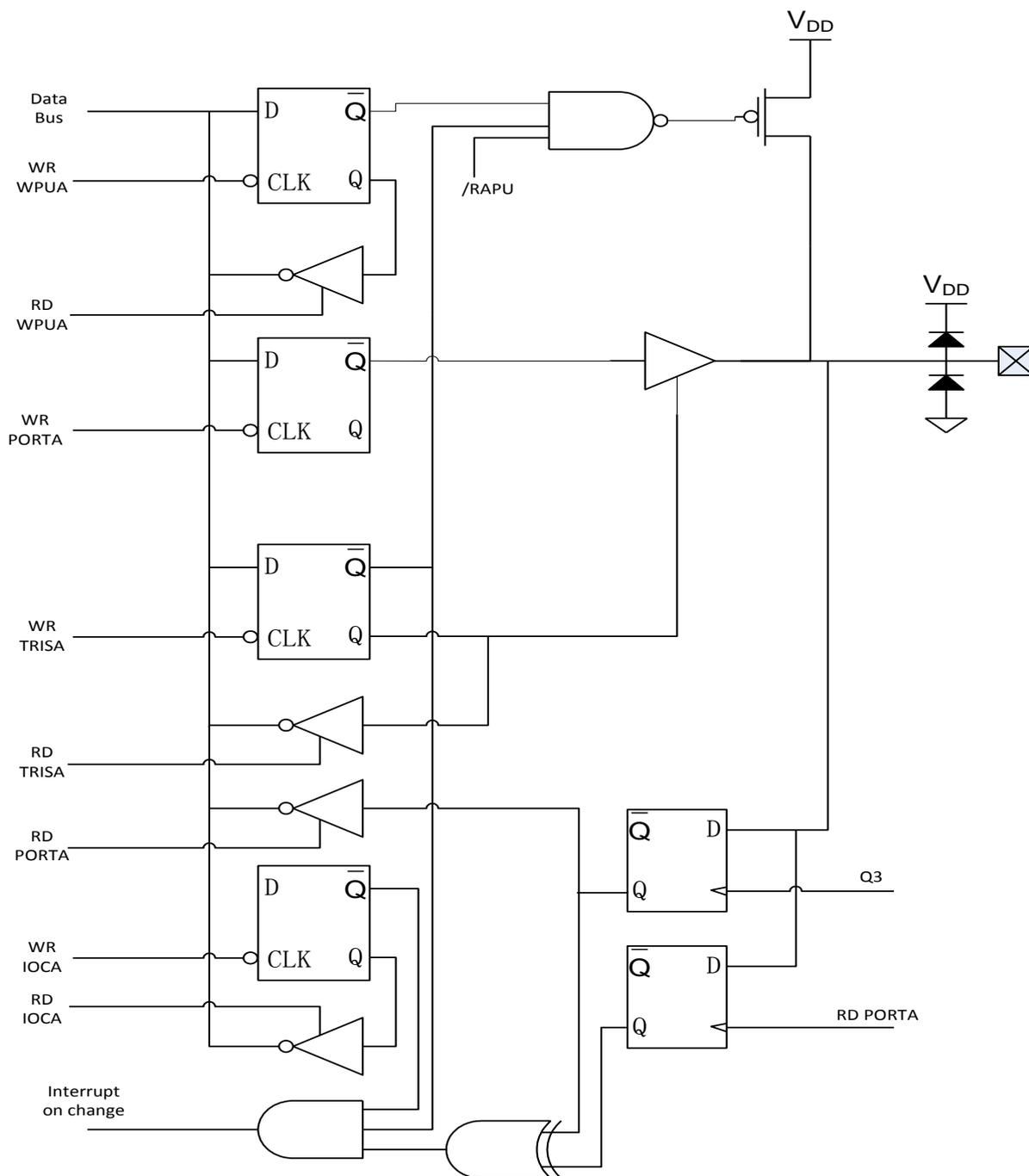


图 14.6 PA5 架构框图

15. 芯片的电气特性

15.1. 绝对极限参数

工作温度	-40~+85℃
储存温度	-40~+125℃
电源电压	$V_{SS}-0.3V \sim V_{SS}+6.0V$
端口输入电压	$V_{SS}-0.3V \sim V_{DD}+0.3V$

注意：如果运行条件超过了上述“绝对极限参数值”，即可能对器件造成永久性损坏。上述值仅为运行条件的极大值，我们不建议器件运行在该规范范围以外。器件长时间工作在绝对极限参数条件下，其稳定性可能受到影响。

15.2. 内置高频振荡器 (Internal High Frequency Osc)

Symbol	特性	最小值 ⁽¹⁾	典型	最大值	单位	条件/备注
Fosc	(校准)频率	15.76	16	16.24	MHz	校准偏差
	频率容差	-1%	—	1%	—	2.0~5.5V, 25℃
		-9.4%	—	8.1%	—	-40~85℃, 3V
启动时间	—	—	2.5	—	us	3V, 25℃
Ivdd 工作电流	—	—	30	—	uA	3V, 25℃

(1) 数据基于特性值，并未生产测试

注 1: 为了确保振荡器的容差，必须尽量靠近器件在 VDD 和 VSS 之间连接去耦电容。建议并联一个 1uF 电容。

15.3. 内置低频振荡器 (Internal Low Frequency Oscillator)

此低频振荡器有双模模式，一种模式下振动频率为 32KHz，另一种模式下振动频率为 256KHz。振荡频率模式由 OSCCON 寄存器中的 LFMOD 位控制，0 为 32KHz 模式，1 为 256KHz 模式。由于 WDT 固定使用 32KHz，故此位并无实际意义。

Symbol	特性	最小值 ⁽¹⁾	典型	最大值 ⁽¹⁾	单位	条件/备注
Sosc	频率 (LFMOD=0)	—	32	—	KHz	3V, 25℃
	频率容差	-4.2%	—	4.2%	—	2.0~5.5V, 25℃
		-14.9%	—	12.9%	—	-40~85℃, 3V
启动时间	—	—	4.6	—	us	3V, 25℃
Ivdd 工作电流	—	—	1.1	—	uA	3V, 25℃

(1) 数据基于特性值，并未生产测试

15.4. 低电压复位电路 (LVR)

电气参数	最小值	典型	最大值 ⁽¹⁾	单位	条件/备注
Ivdd 工作电流	—	16.94	—	uA	3.3V
LVR threshold 可设置范围	1.8	2.0	2.2	V	TT, 25°C
	2.0	2.2	2.4		
	2.6	2.8	3.0		
LVR delay	—	125	157	us	2.0V~5.5V, 25°C

(1) 数据基于特性值，并未生产测试。

15.5. I/O PAD 电路

除特殊说明之外，特性测试条件为 25°C 环境温度。

电气参数	最小值	典型	最大值 ⁽¹⁾	单位	条件/备注
VIL	0	—	0.3*VDD	V	—
VIH	0.7*VDD	—	VDD	V	—
输入漏电流	-1	0	1	uA	V _{DD} =5V
IOL	—	28.85	29.24	mA	V _{DD} =5V, V _{OL} =0.5V
IOH	—	15.31	15.41	mA	V _{DD} =5V, V _{OH} =4.5V
上拉电阻 R _{PH}	—	25	—	kΩ	5V
	—	35	—		3.3V

(1) 数据基于特性值，并未生产测试。

15.6. 总体工作电流 (Ivdd)

典型值的测试条件为 25°C 环境温度，所有 IO 输出固定值。

电气参数	最小值	典型	最大值 ⁽¹⁾	单位	条件/备注
正常模式	—	0.520	0.544	mA	16MHz@3V
	—	0.384	0.400	mA	8MHz@3V
	—	0.317	0.330	mA	4MHz@3V
	—	0.204	0.214	mA	32KHz@3V
休眠模式 (Sleep, WDT OFF, LVR OFF)	—	6.55	8	uA	3.3V
休眠模式 (Sleep, WDT ON, LVR OFF)	—	7.9	9	uA	3.3V
休眠模式 (Sleep, WDT OFF, LVR ON)	—	17.07	37.65	uA	3.3V
休眠模式 (Sleep, WDT ON, LVR ON)	—	18.86	48.65	uA	3.3V

(1) 数据基于特性值，并未生产测试。

15.7. AC 电气特性

电气参数	最小值	典型	最大值	单位	条件/备注
指令周期 (Tins)	250	—	—	ns	系统时钟 HFINTOSC
	125	—	—	us	系统时钟 LFINTOSC
T0CKI 输入周期	(Tins+40)/N 和 20 中较大者	—	—	ns	N = 预分频值 (2, 4, ..., 256)
上电复位保持时间 (Tdrh)	—	8.2	—	ms	T=25°C, PWRT disable
外部复位脉冲宽度 (Trst)	2000	—	—	ns	T=25°C
WDT 周期 (Twdt)	—	1	—	ms	无预分频, WDTPS<3:0>=0000

注：除特殊说明，特性测试条件为：T=-40~85°C，VDD=2.0~5.5V。

15.8. 时序图

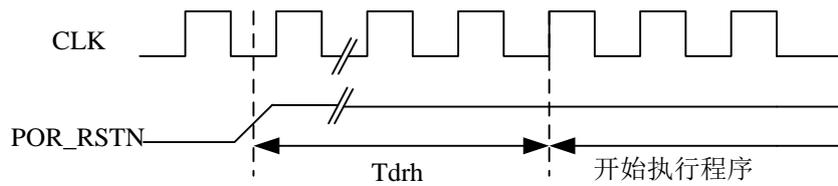


图 15.1 上电复位时序图

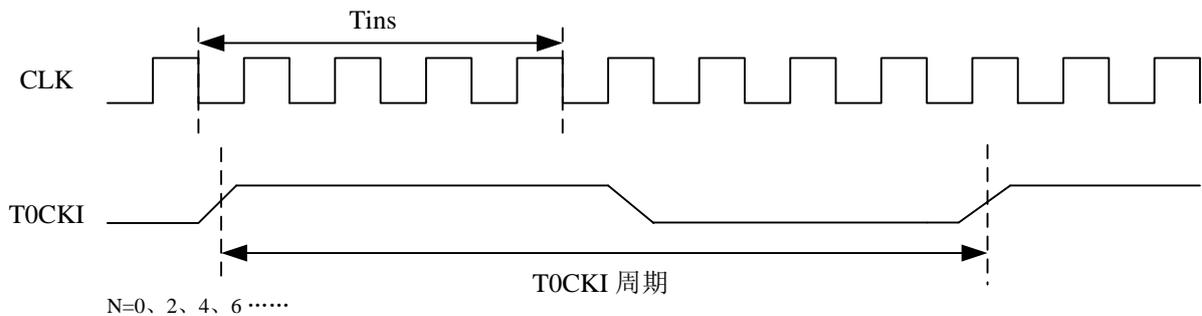


图 15.2 T0CKI 时序图 (4T 指令周期)

15.9. 直流和交流特性图表

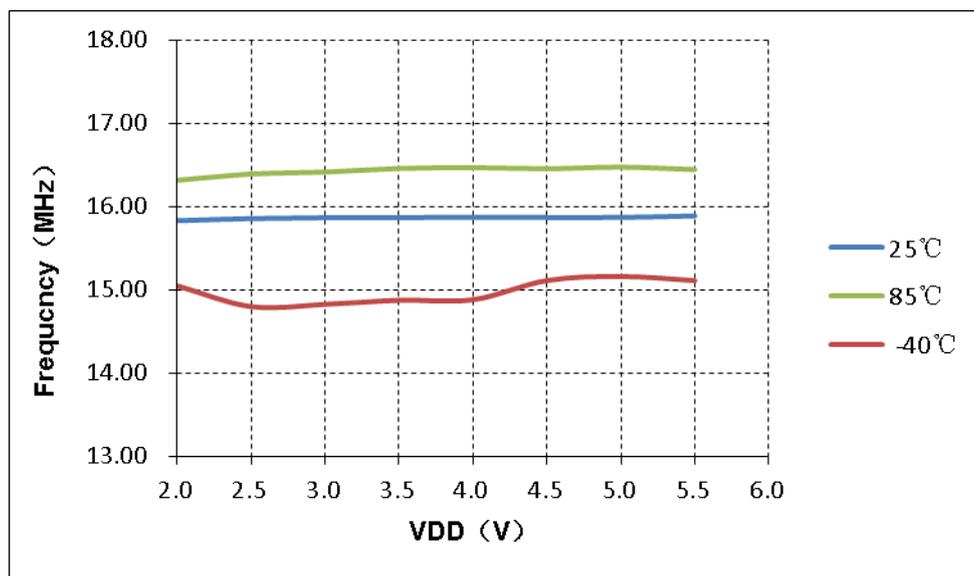


图 15.3 不同温度下 Fosc 与 VDD 曲线

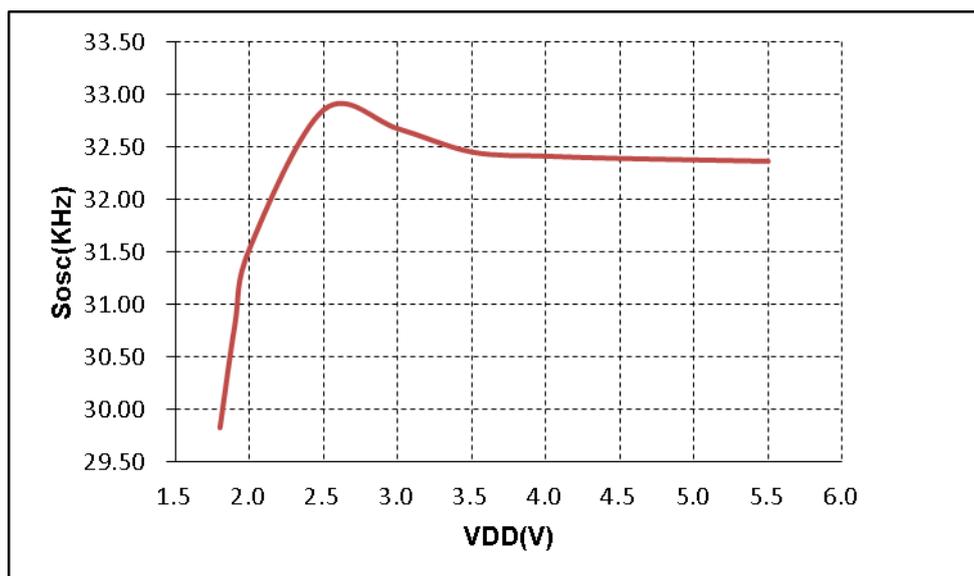


图 15.4 Sosc 与 VDD 关系曲线(T=25°C)

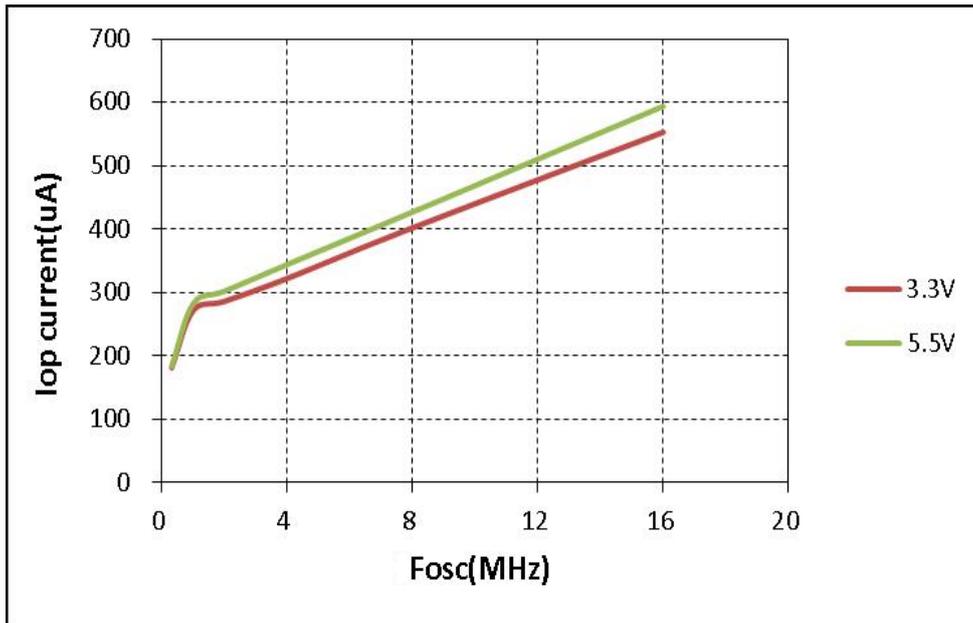


图 15.5 不同 VDD 下芯片工作电流与 Fosc 曲线 (T=25°C)

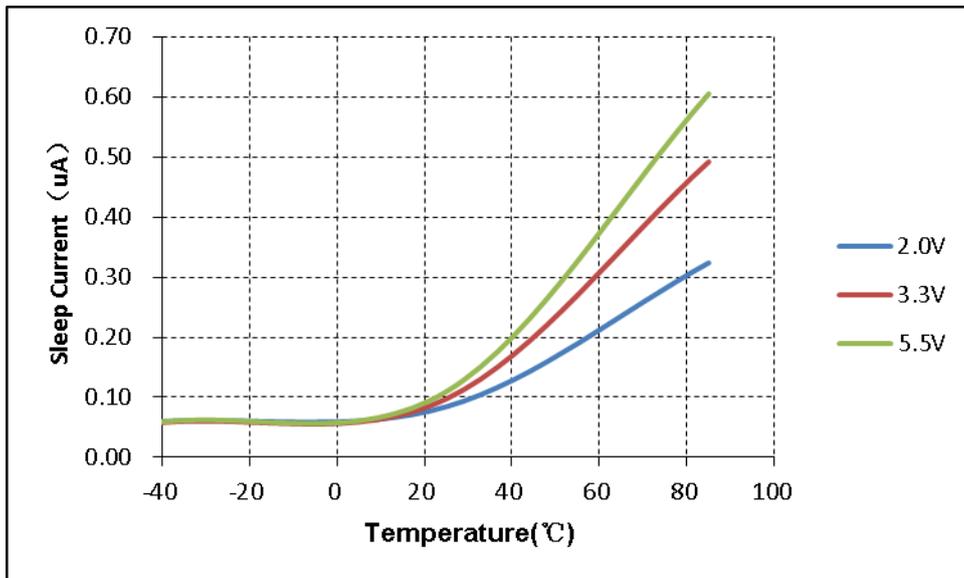


图 15.6 不同 VDD 下睡眠电流与温度曲线 (WDT 和 LVR 禁止)

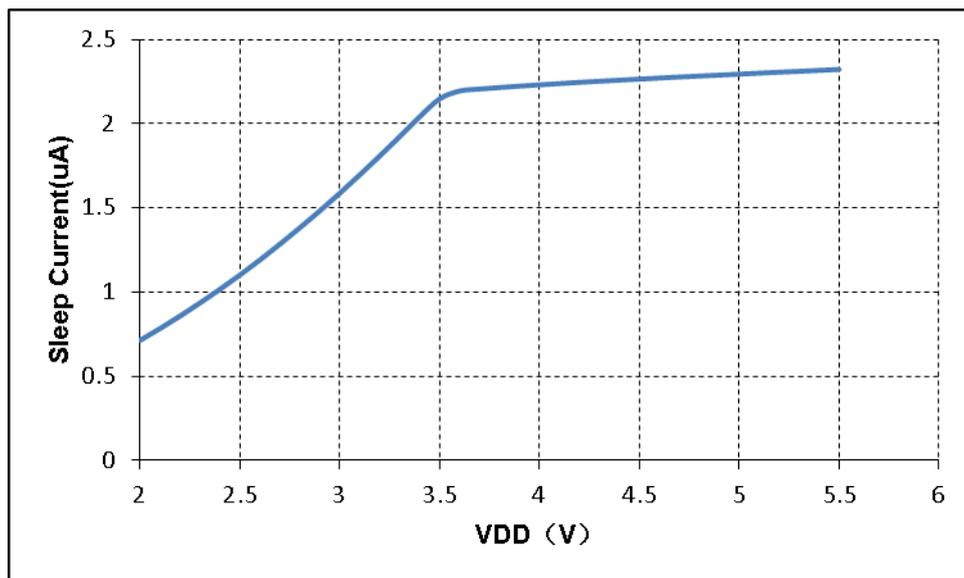


图 15.7 睡眠电流与 VDD 曲线 (WDT 使能, LVR 禁止)

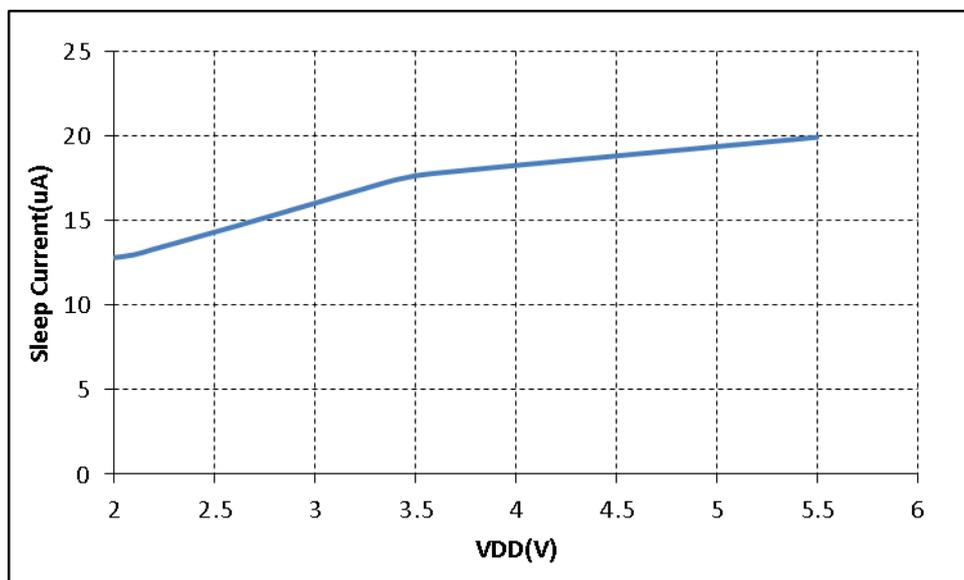


图 15.8 睡眠电流与 VDD 曲线 (WDT 禁止, LVR 使能)

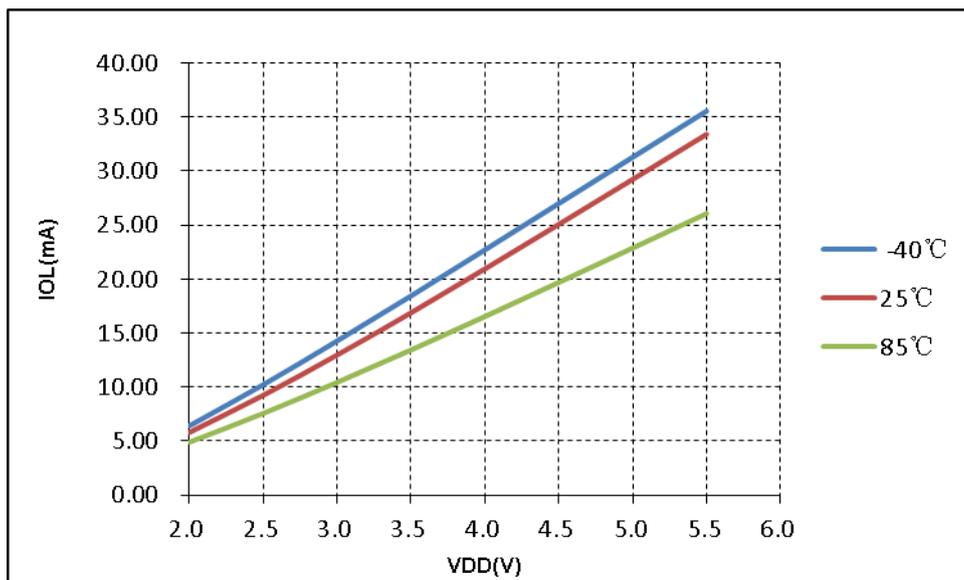


图 15.9 不同温度下 IOL 与 VDD 曲线 (VOL=0.5V)

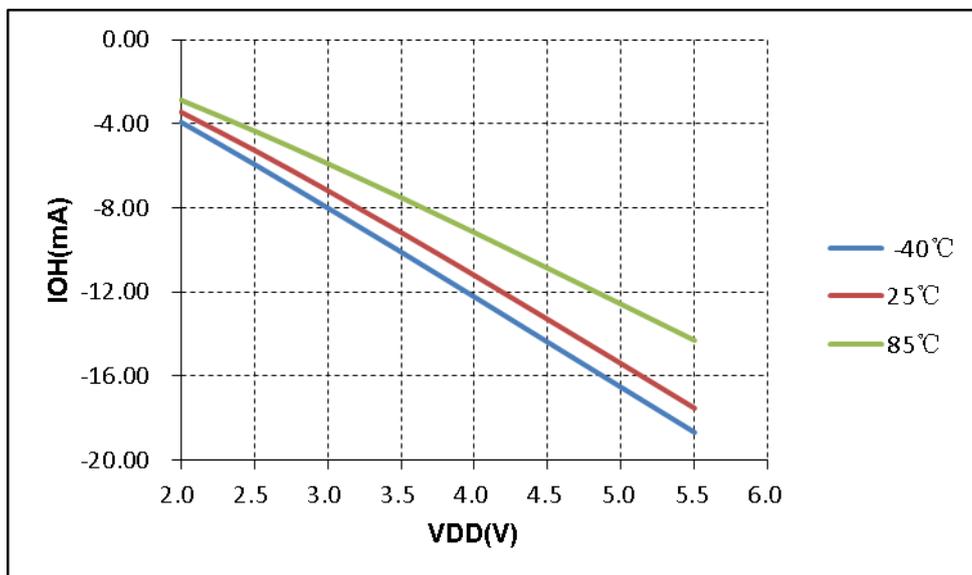


图 15.10 不同温度下 IOH 与 VDD 曲线 (VOH=4.5V)

16. MCU 的指令架构

本芯片采用精简指令集架构。

16.1. 指令集列表

以下是各指令的描述。

汇编语法	功能	运算	状态位
BCR R, b	Bit clear	0-> R(b)	NONE
BSR R, b	Bit set	1-> R(b)	NONE
BTSC R, b	Bit test, skip if 0	Skip if R(b)=0	NONE
BTSS R, b	Bit test, skip if 1	Skip if R(b)=1	NONE
NOP	No operation	None	NONE
CLRWDT	Clear WDT	0-> WDT	/PF, /TF
SLEEP	ENTER SLEEP MODE	0-> WDT, STOP OSC	/PF, /TF
STTMD	Store W TO TMODE	W-> TMODE	NONE
CTLIO R	Control IO direction reg	W-> IODIRr	NONE
STR R(MOVWF)	Store W to reg	W-> R	NONE
LDR R, d(MOVF)	Load reg to d	R-> d	Z
SWAPR R,d	Swap halves reg	[R(0-3)R(4-7)]-> d	NONE
INCR R, d	Increment reg	R+ 1-> d	Z
INCRSZ R, d	Increment reg, skip if 0	R+ 1-> d	NONE
ADDWR R, d	Add W and reg	W+ R-> d	C, HC, Z
SUBWR R, d	Sub W from reg	R- W-> d R+ /W+ 1-> d	C, HC, Z
DECR R, d	Decrement reg	R- 1-> d	Z
DECRSZ R, d	Decrement reg, skip if 0	R- 1-> d	NONE
ANDWR R, d	AND W and reg	R& W-> d	Z
IORWR R, d	Inclu.OR W and reg	W R-> d	Z
XORWR R, d	Exclu.OR W and reg	W^ R-> d	Z
COMR R, d	Complement reg	/R-> d	Z
RRR R, d	Rotate right reg	R(n)-> R(n-1), C-> R(7), R(0)-> C	C
RLR R, d	Rotate left reg	R(n)-> R(n+1), C-> R(0), R(7)-> C	C
CLRW	Clear working reg	0-> W	Z
CLRR R	Clear reg	0-> R	Z
RETI	Return from interrupt	Stack-> PC, 1-> GIE	NONE

RET	Return from subroutine	Stack-> PC	NONE
LCALL N	Long CALL subroutine	N-> PC, PC+1-> Stack	NONE
LJUMP N	Long JUMP address	N-> PC	NONE
LDWI I(MOVLW)	Load immediate to W	I-> W	NONE
ANDWI I	AND W and imm	W& I-> W	Z
IORWI I	Inclu.OR W and imm	W I-> W	Z
XORWI I	Exclu.OR W and imm	W^ I-> W	Z
RETW I	Return, place imm to W	Stack-> PC, I-> W	NONE
ADDWI I	Add imm to W	W+I-> W	C, HC, Z
SUBWI I	Subtract W from imm	I-W-> W	C, HC, Z

Table1, OP Code Field Description

Field	Description
R(F)	SFR/GPR address
W	Working register
b	Bit address within the 8-bit register/ram
I/Imm(k)	Immediate data
X	Don't care, may be 0 or 1
d	Destination select 0: store result in W 1: store result in register/ram
N	Immediate program address
PC	Program counter
TMODE	SFR TMODE
IODIRr	SFR IODIR, r can be A, B, C
C	Carry bit
HC	Half carry
Z	Zero flag
/PF	Power down flag
/TF	Time out flag

