
WL2610M用户手册

2. 4G无线收发SOC芯片

WL2610MSOP16

Version 1.0



上海晶曦微电子科技有限公司

本公司保留对产品在可靠性、功能和设计方面的改进作进一步说明的权利。

版本修订记录

版本	更新日期	修改记录
1.0	2021-1-11	New Release

目录

1	芯片简介	6
2	芯片主要特点	6
3	电气特性	7
4	管脚描述	9
5	典型应用	11
6	内部MCU到RF的SPI接口	12
6.1	SPI 默认格式	13
7	射频模块接收/传输流程	14
7.1	RF初始化	14
7.2	发送流程“短包” (<=63BYTES)	14
7.3	接收流程“短包” (<=63BYTES)	14
7.4	发送流程“长包” (<=255BYTES且>63BYTES)	15
7.5	接收流程“长包” (<=255BYTES且>63BYTES)	15
7.6	进入IDLE模式流程	15
7.7	进入SLEEP模式流程	15
7.8	SLEEP唤醒	16
7.9	扫描RSSI流程	16
7.10	传输数据率250K时的流程	16
8	射频相关寄存器信息	16
9	射频数据包	21
10	射频电路电源要求	21
11	射频模块应用注意事项	21
11.1	天线及匹配	21
11.2	I/O电压	22
11.3	POR	22
11.4	状态转换	22
11.5	同步字使用注意事项	22
12	MCU模块系统框图	22
12.1	系统框图	22
12.2	ROM 程序存储器	23
12.3	数据存储器	24
13	功能描述	27
13.1	操作寄存器	27
13.1.1	H/L (间接寻址寄存器)	27
13.1.2	Y/Z (间接寻址寄存器)	27
13.1.3	R 寄存器	28
13.1.4	X (X 寄存器)	28
13.1.5	W (W 寄存器)	28
13.1.6	P_CTRL 寄存器	29
13.1.7	PC (程序计数器) 与堆栈	29
13.1.7.1	PCH 寄存器	30
13.1.7.2	PCL 寄存器	30

13.1.8	PFLAG (状态寄存器)	30
13.1.9	RBANK (RAM 存储区选择寄存器)	31
13.1.10	OPTION	31
13.1.11	WDTR 寄存器	32
13.1.12	OSCM 寄存器	33
13.1.13	REM_REG	33
13.1.14	REVO_TOL	33
13.1.15	REVO_TOH	34
13.1.16	REV1_TOL	34
13.1.17	REV1_TOH	34
13.2	TC0 (16 定时/计数器寄存器)	34
13.2.1	概述	34
13.2.2	TCOCL	35
13.2.3	TCOM	35
13.2.4	TCOR	35
13.2.5	TCOD	36
13.2.6	BUZ 波形	36
13.2.7	PWM 波形	37
13.3	T1 (定时/计数器寄存器)	37
13.3.1	概述	37
13.3.2	T1M 寄存器	38
13.3.3	T1C 寄存器	38
13.3.4	T1M1 寄存器	38
13.4	带死区互补的 12 位 PWM	39
13.4.1	PWM 特性	39
13.4.2	PWM 输出模式	39
13.4.2.1	PWM 互补输出模式	39
13.4.2.2	PWM 独立输出模式	39
13.4.3	PWM 相关寄存器	40
13.4.3.1	PWM0 使能寄存器 PWMOEN	40
13.4.3.2	PWM0 控制寄存器 PWMOC	40
13.4.3.3	PWM0 周期寄存器 PWMOPL、PWMOPH	41
13.4.3.4	PWM0 占空比寄存器 PWMODL、PWMODH	41
13.4.3.5	PWM0 死区时间寄存器 PWMODTL、PWMODTH	42
13.4.3.6	PWM1 占空比寄存器 PWM1DL、PWM1DH	42
13.4.3.7	PWM0 计数器/定时器时间寄存器 PWMOTL、PWMOTH	42
13.4.3.8	PWM0 输出端口选择寄存器	42
13.5	中断	43
13.5.1	概述	43
13.5.2	中断控制寄存器	43
13.5.3	INTRQ	44
13.5.4	PEDGE	44
13.5.5	STKP	45
13.6	MTP	47
13.6.1	MTPEN (MTP 控制寄存器)	47
13.6.2	MTPQ (MTP 标志位)	47
13.6.3	MTPC (MTP 擦除烧写频率控制)	48
13.6.4	MTPAH/L (MTP 地址低高 2 位/低 8 位)	48
13.6.5	MTPD (MTP8 位数据)	48
14	I0 端口	48
14.1	端口模式控制寄存器	48
14.2	端口数据寄存器	49

14.3	端口唤醒控制寄存器.....	49
15	系统时钟.....	49
15.1	概述.....	49
15.2	指令周期 FCPU.....	50
16	系统工作模式.....	50
16.1	概述.....	50
16.2	普通模式.....	52
16.3	低速模式.....	52
16.4	睡眠模式.....	53
16.5	绿色模式.....	53
16.6	系统唤醒.....	53
16.6.1	概述.....	53
16.6.2	唤醒时间.....	54
17	复位.....	54
17.1	概述.....	54
17.2	上电复位.....	55
17.3	看门狗复位.....	55
17.4	寄存器复位状态表.....	55
17.4.1	上电复位状态表.....	55
17.4.2	WDT 复位状态表.....	56
18	指令集.....	57
19	封装形式.....	59

1 芯片简介

WL2610M是一款低成本，高集成度的2.4GHz SOC的无线收发芯片，片上集成发射机、接收机、频率综合器、GFSK调制解调器和低功耗的MCU。WL2610M具有高灵敏度、低功耗以及抗干扰能力强的优点，片上MCU内建了4K*16-bit的OTP-ROM、1K*8-bit的MTP-ROM，内置红外学习电路。

WL2610M在系统应用上外围线路简单，大大降低了系统方案的外围成本。

2 芯片主要特点

- 低功耗的2.4GHz ISM频段、GFSK射频收发器
- 1Mbps/250Kbps的数据率
- DSSS调制解调器
- 强大的C/I提供良好的WIFI共存性能
- 简单的射频与微处理器接口-SPI
- 强大的C/I提供良好的WIFI共存性能
- 可编程数据白化
- 支持FEC，增加通信可靠性
- 支持8/16位CRC
- 支持AGC
- AutoACK和自动重传
- 高效的功耗管理模块（最低约2 μ A待机电流）
- 外接晶振容许50PPM的变化
- 符合FCC/ETSI等国际标准
- 近距离模式，适用于有特定安全需求的环境
- 宽工作电压范围：2.2V~3.6V（8M 2分频），MTP 烧写和擦除是电压 2.5v 以上。
- 工作温度范围：-20℃~80℃
- 工作频率范围：
- 内置 RC 8MHz $\pm$ 2%
- 系统时钟源可配置选择：Fosc/2、Fosc/4、Fosc/8、Fosc/16、Fosc/32
- 4K*16-bit 片内 ROM
- 1K*8-bit 片内 MTP
- 可直接、间接寻址
- 1 个 16 位定时器 TC0，做 PWM、Buzzer 时 8 位有效。
- 1 个 16 位基本定时器 T1
- 1 个 12 位定时器 TM1，可组成 3 路 12 位 PWM 或者 1 组带死区互补 12 位 PWM。
- 内置上电复位时间 PWRT
- 内置 WDT 看门狗
- 内置可校准 IRC 振荡器(8MHZ)
- 一个安全位（代码寄存器中）保护程序不被读出
- 512*8bits 片内寄存器组（SRAM，一般寄存器）
- 3 组双向 I/O 端口
- 8 级用于子程序嵌套的堆栈
- 节能模式（SLEEP 模式）
- 7 种可用中断：TC0、T0、TM1、外中断、P0_P1 端口变化中断、P54 端口变化中断、P50 端口变化中断

- 内置红外发射放大驱动电路
- 内部上电复位电路(POR)
- 内置低电压检测 (LVD)，用于欠压复位 (BOR)
- SOP16封装

3 电气特性

表3.1: 射频电气特性（以下电气特性都是在工作温度25℃，工作电压3.3V条件下得到的）

参数	描述	规格			单位	说明
		最小	典型	最大		
T _{OP}	工作温度	-40		110	℃	
V _{DD}	工作电压	1.9		3.6	V	
F _{xtal}	晶体频率		12		MHz	容许50PPM的变化
I _{DD_TX}	TX工作电流		15		mA	P _{OUT} = 0 dBm
I _{DD_RX}	RX 工作电流		18		mA	
I _{DD_IDLE}	IDLE工作电流		1		mA	
I _{DD_SLP}	Sleep 工作电流		1		μ A	
F _{CLK}	输出时钟		12		MHz	
T _{r_spi}	SPI时钟沿上升下降的时间			25	ns	读写不出错的最低要求
F _{SPI}	SPI时钟速度			12	MHz	
F _{OP}	工作频率	2402		2483	MHz	
VSWR	天线端口差异 (Z ₀ =50Ω)		<2:1		VSWR	
RXS	接收灵敏度	-95 @250K bps	-90 @1Mbps		dBm	BER ≤ 0.1%
RXM	最大输入功率	-20			dBm	BER ≤ 0.1%
R _{data}	数据率	250K		1M	Bit/s	
CI _{co-channel}	同频干扰		9		dB	
CI _{1MHz}	1MHZ同频信号干扰		-1		dB	-60 dBm信号.
CI _{2MHz}	2MHZ同频信号干扰		-27		dB	-60 dBm信号
CI _{3MHz}	3MHZ同频信号干扰		-35		dB	-60 dBm信号
CI _{Image}	Carrier/Interference at image frequency		-20		dB	-67 dBm信号
CI _{3MHzUp}	Carrier/Interference at >3MHz offset		-33		dB	-60 dBm信号

参数	描述	规格			单位	说明
		最小	典型	最大		
OBB_1	带外干扰		-10		dBm	30 MHz to 2000 MHz
OBB_2		-27			dBm	2000 MHz to 2400 MHz
OBB_3		-10			dBm	3000 MHz to 12.75 GHz
P _{out_max}	最大发射功率			8	dBm	
DF1 _{avg}	最大频偏 00001111 pattern		250		kHz	调制特性TX眼图
DF2 _{max}	最大频偏 01010101 pattern		220		kHz	调制特性TX眼图
DF2 _{max} / DF1 _{avg}	TX EYE opening	80			%	1010序列相对于00001111序列
IBS_1	带内辐射 (+/850kHz)			-20	dBm	随机数
IBS_2	带内辐射2MHz偏移			-40	dBm	随机数
IBS_3	带内辐射3MHz 偏移			-55	dBm	随机数
OBS_1	带外辐射			-60	dBm	30 MHz ~ 1 GHz
OBS_2				-45	dBm	1 GHz ~ 12.75 GHz
OBS_3				-60	dBm	1.8 GHz ~ 1.9 GHz
OBS_4				-65	dBm	5.15 Hz ~ 5.3 GHz

表3.2: 极限参数

参数	符号	参数范围	单位
电源电压	V _{cc}	-0.3~5.5	V
输入电压	V _i	-0.3~V _{cc} +0.3	V
输出电压	V _o	-0.3~V _{cc} +0.3	V
最大功耗 (V _{cc} = 5V,空载)	P _a	5	mW
工作温度	T _{opr}	-20~+70	°C
贮存温度	T _{stg}	-40~+125	°C

表3.2: DC 特性 (TEMP=25°C、VDD=5V)

参数	符号	测试条件	最小值	典型值	最大值	单位
电源电压	V _{cc}	8MHz , Fosc/2	2.0		5.5	V
静态电流	I _{DD}	振荡器停振,休眠			1	μA
工作电流	I _{CC}	8MHz/4 分频, 空载		0.7		mA
	I _{cc1}	32KHz 低速,VDD=5V		5.8		uA
	I _{cc2}	32KHz 低速,VDD=3V		1.2		uA

内置高频 RC	F_{HRC}	温度：-20~80°C 电压：1.8~3.6V	15.5	16	16.5	MHz
内置低频 RC	F_{LRC}	VDD=5v		32		KHz
		VDD=3V		16		
I/O 管脚输入高电压	V_{IH}		3.1			V
I/O 管脚输入低电压	V_{IL}				2.0	V
RESET	V_{IH}		3.2			V
	V_{IL}				1.45	V
I/O 管脚	I_{OL}	$V_{OL}=0.2VDD$		24		mA
	I_{OH}	$V_{OH}=0.8VDD$		19		mA
端口上拉电阻	R			50		K Ω
BOR 2.0V	V	温度：-20~80°C	1.7	1.85	2.1	V

表3.2: AC 特性 (TEMP=25°C)

参数	符号	测试条件	最小值	典型值	最大值	单位
工作频率	F	2~5.5V		8M		Hz
看门狗复位时间	Twdt	不分频时		64		ms
端口由低到高 上升时间	T_r	VDD>2.3V		5		us
端口由高到低 下降时间	T_f	VDD>2.3V		40		ns

4 管脚描述

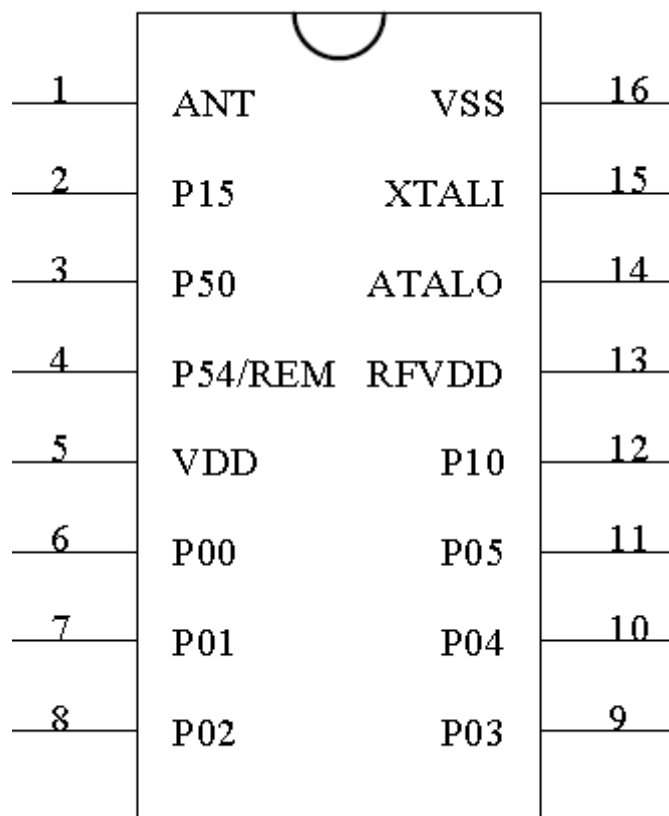


图4.1 WL2610MSOP16封装示意图

表4.1 WL2610MSOP16管脚描述

管脚名	管脚类型	描述
ANT	射频端口	天线端口
P15	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； TC0 计数功能输入脚； T-KEY 扫描脚； SDA 低位脚； 外部中断管脚
P50	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻； 红外学习信号输出脚； T-KEY 扫描脚； SDA 高位脚；
P54/REM	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻； 编码输出端（可通过寄存器设置）； 红外小信号输入端（可通过寄存器设置）； T-KEY 扫描脚；
VDD	电源	电源
P00	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚； 高压烧写脚；

P01	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚； MODEL 脚；
P02	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚；
P03	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚； SCK 脚；
P04	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚；
P05	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚；
P10	I/O	双向输入输出脚，输入时施密特触发，内置上拉电阻，可设置唤醒功能，可设置中断功能； T-KEY 扫描脚；
RFVDD	电源	电源
XTALO	模拟输出	晶振端口
XTALO	模拟输入	晶振端口
VSS	地	地

表4.2 射频与微处理器接口

射频与微处理器接口		
	RF	MCU
1	CE	P14
2	CSN	P13
3	CLK	P11
4	MOSI	P12

5 典型应用

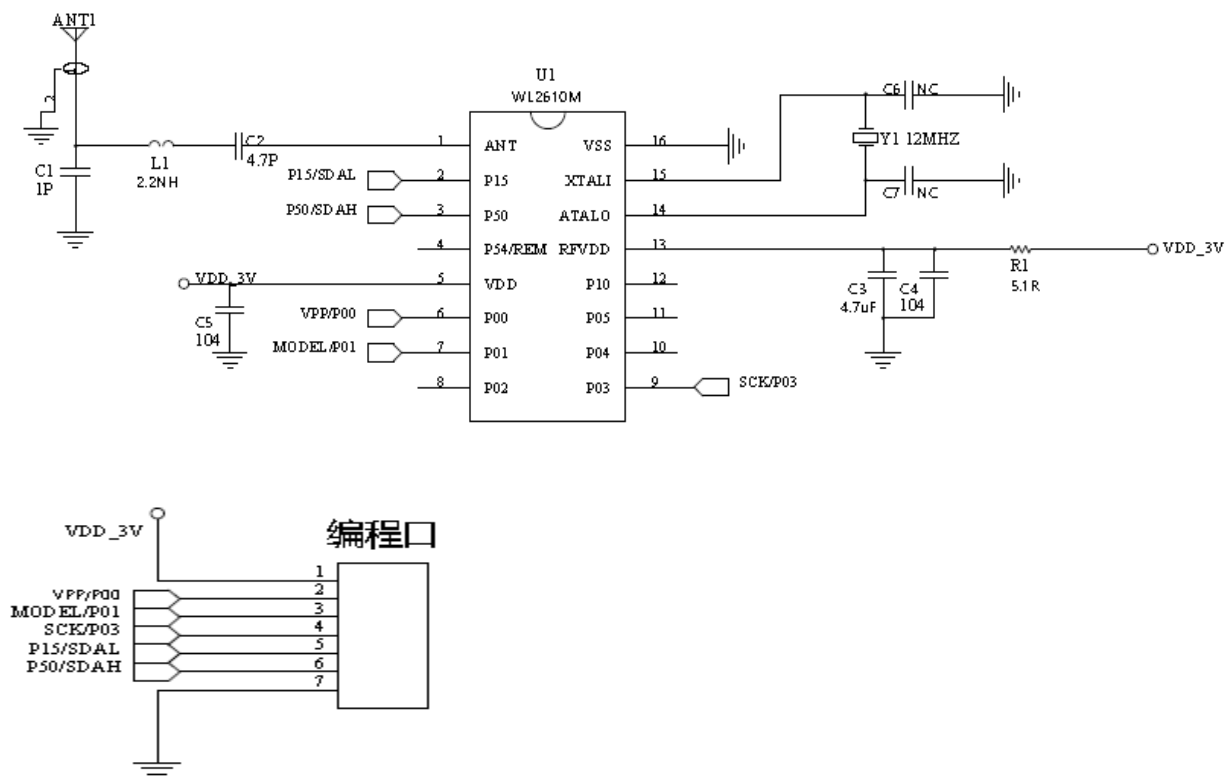


图5.1 WL2610MSOP16典型应用图

表5.1 WL2610MSOP16典型应用元件清单

数量	原理图标识	值	封装	描述
1	L1	2.2nH	0603	murata, $\pm 0.2\text{nH}$
2	C1	1pF	0603	murata, $\pm 5\%\text{PF}$, 50V
3	C2	4.7pF	0603	murata, $\pm 5\%\text{PF}$, 50V
4	C3	4.7uF	0603	murata, $\pm 5\%\text{PF}$, 50V
5	C4	104	0603	murata, $\pm 5\%\text{PF}$, 50V
6	C5	104	0603	murata, $\pm 5\%\text{PF}$, 50V
7	C6, C7	NC		
8	R1	5.1R	0603	murata
9	Y1	12M	3225	$< \pm 50\text{ppm}$ ESR $<60\Omega$
10	U1	WL2610M	SOP16	

6 内部MCU到RF的SPI接口

RF 与 MCU 内部通讯接口

RF	MCU
CE	P14
CSN	P13
CLK	P11
MOSI	P12

6.1 SPI 默认格式

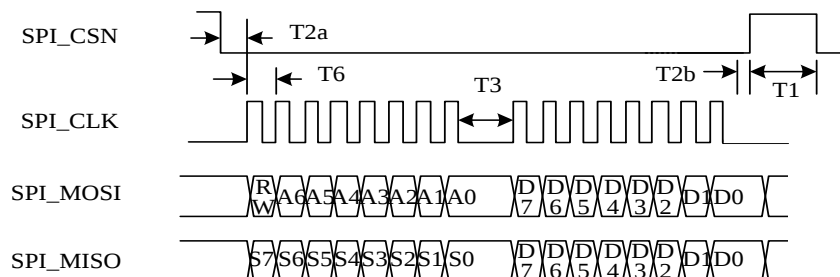


图6.1 4线SPI读写寄存器方式 (CKPHA=1)

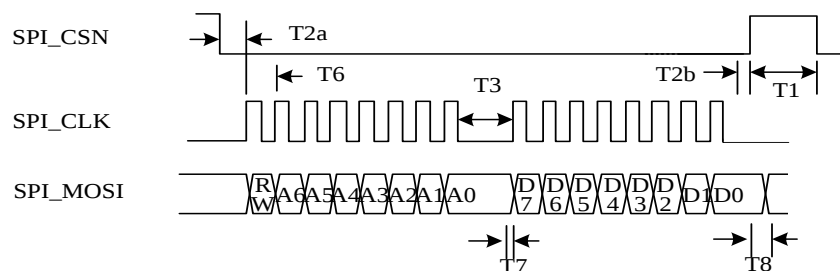


图6.2 3线SPI读寄存器方式 (CKPHA=1)

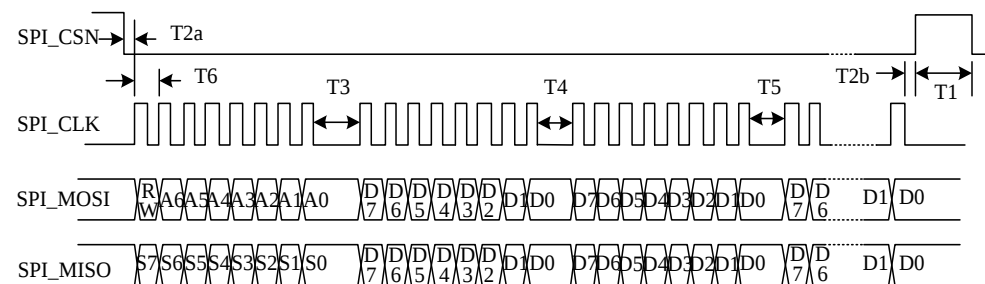


图6.3 4线SPI读写FIFO方式 (CKPHA=1)

Notice:

- 1) SPI读写位：写=0，读=1；
- 2) 访问FIFO寄存器时，可以采用只写一次地址的方式来连续读写数据；
- 3) 访问除FIFO外的其他寄存器时，每个CS任务期间只能访问一个字节；

表6.1 SPI时序要求

参数	描述	规格	单位	说明
----	----	----	----	----

		最小值	典型值	最大值		
T1	最小SPI闲置时间	250			ns	
T2a	CSN建立时间	20			ns	
T2b	CSN保持时间	200			ns	
T3, T4, T5		450			ns	访问FIFO时
T6		83			ns	
T7	SPI时钟上升沿至RF数据输出时间	0		10	ns	
T8	SPI_CSN至MOSI的上升沿到RF数据输入时间			220	ns	

7 射频模块接收/传输流程

7.1 RF初始化

- 1、引脚CE置高电平，使能芯片（如果芯片有引脚CE）；
- 2、选择3线或4线SPI模式（reg94）；
- 3、然后写入以下数据到相关寄存器，以初始化芯片：

寄存器	04	17	18	34	35	45	46	52	53
值	0xC2	0x3A	0x0C	0x08	0x08	0x00	0x09	0x1A	0x40
寄存器	80								
值	0xF8								

- 4、reg65选择编码方式和FEC（发送设备需要与接收设备设置相同的值）；
- 5、reg64、reg81设置数据帧格式，reg72、73、74、75、76、77、78、79设置syncword（发送设备需要与接收设备设置相同的值）。

7.2 发送流程“短包”（≤63bytes）

进行RF初始化后，发送流程需要加入以下步骤：

- 1、reg18设置发射功率增益；
- 2、reg15设置发射频率信道；
- 3、reg104清TX FIFO；
- 4、reg100往FIFO写入数据长度；
- 5、reg100写数据到FIFO，且数据长度不能长于FIFO长度范围；
- 6、清reg15[7]，设置reg14[0]，FIFO的数据即可发出，然后就可以得到一个发送完成的标志，引脚PKT为高或者PKT_FLAG标志位为1(reg97[6])，此时WL1601处于idle模式；
- 7、重复步骤3、4、5、6发送其他数据包，或者进入接收流程。

7.3 接收流程“短包”（≤63bytes）

RF初始化后，接收流程中需要加入以下步骤：

- 1、reg70[5]选择短距离模式或者正常工作模式；
- 2、reg15设置接收频率的信道；
- 3、reg105清RX FIFO；
- 4、清reg14[0]，设置reg15[7]，等待PKT引脚为高或者reg97的PKT_FLAG标志位为1（意为已经接收到1个包）；
- 5、读取reg96检查CRC错误标志位；如果CRC错误标志位为0，则从FIFO（reg100）得到数据，得到的第一个字节就是需要读取的长度；如果CRC错误标志位为1，则重复步骤3、4，等待下一个包；
- 6、重复步骤3、4、5接收新数据，或者进入发送流程；
- 7、MCU应当设置time out，如果超时，表示没有收到数据，应清reg15[7]，使WL1601退出接收状态。

7.4 发送流程“长包”（ $\leq 255\text{bytes}$ 且 $> 63\text{bytes}$ ）

RF初始化后，发送流程（长包）需要加入以下步骤：

- 1、reg18设置发射功率增益；
- 2、reg15设置发送频率信道；
- 3、reg104清TX FIFO；
- 4、FIFO写入数据长度（reg100）；
- 5、FIFO写入数据（reg100）（FIFO是64bytes，可以最多写入63bytes），清reg15[7]，设置reg14[0]；
- 6、reg97检查FIFO标志位和PKT标志位，如果FIFO_FLAG为1而PKT_FLAG为0，则FIFO可以写入新数据（数据长度不能超过64位FIFO满阈值（reg80、reg81））；
- 7、重复步骤6直到MCU将全部数据写入FIFO且FIFO_FLAG为1，清reg14[0]，长包完成发送；
- 8、重复步骤2-7，发送其他数据包，或者进入接收流程。

7.5 接收流程“长包”（ $\leq 255\text{bytes}$ 且 $> 63\text{bytes}$ ）

- 1、选择短距离模式或者正常工作模式reg70[5]；
- 2、reg15设置发送频率信道；
- 3、reg105清RX_FIFO；
- 4、清reg14[0]，设置reg15[7]；
- 5、reg97检查FIFO_FLAG和PKT_FLAG，如果FIFO标志位为1，则从FIFO取数据（reg100），如果在极端环境，MCU需要一个时钟来避免等待FIFO标志位；
- 6、必要时重复步骤5，直到MCU读取了全部数据，清reg15[7]，长包接收完成；
- 7、重复步骤3-6接收其他数据包，或者进入发送流程。

7.6 进入IDLE模式流程

按照以下步骤进入空闲模式：

- 1、清reg14[0]，清reg15[7]。

7.7 进入SLEEP模式流程

RF初始化后，按照以下步骤进入sleep模式：

- 1、reg70设置sleep使能位。

7.8 SLEEP唤醒

从Sleep模式唤醒，进入idle模式的方式有如下2种：

方法1：使SPI_CSN保持低电平大于2ms；

方法2：CE脚复位

7.9 扫描RSSI流程

RF初始化后，以下步骤是RSSI扫描流程：

- 1、设置reg22打开RSSI功能，低2位需为0；
- 2、设置需要扫描的频道数量reg84；
- 3、设置第一个扫描的频道号reg86（也叫RSSI起始信道）；
- 4、设置reg86[7]开始扫描RSSI；
- 5、等待PKT_FLAG为1（reg97），从FIFO（reg100）得到信道的RSSI值。

7.10 传输数据率250K时的流程

RF初始化后，在进入前面章节讲述的发送/接收流程之前，需要加入以下步骤：

- 1、在发送流程之前写入以下数据到相关寄存器

寄存器	64	106	107	108	109	110	111
值	0x68	0x68	0xC3	0x16	0x7C	0x6E	0xA1

- 2、在接收流程之前写入以下数据到相关寄存器

寄存器	64	106	107	108	109	110	111
值	0x68	0x68	0xC3	0x7A	0x89	0xC1	0x97

注：

- 1、reg106的值需要根据使用环境的噪声进行设置，以得到最小的丢包率，取值范围0x60~0xA0H；
- 2、如果要从250Kbps改回1Mbps，上表中的寄存器应当被MCU写回默认值；
- 3、每次收发转换时，reg108~111都需要按照上表对应收发，写入相应值。

8 射频相关寄存器信息

通过SPI访问以下寄存器；其他寄存器仅供内部调试，请保持初始化值。

寄存器14 默认值：0x00 可读写

Bit	Bit Name	Description
7-1	保留	
0	Start_TXFIFO	写1开始发送数据到FIFO，发送完成后，该位自动被硬件清0，如需进入发送模式，则需重写该位为1。

寄存器15 默认值：0x30 可读写

Bit	Bit Name	Description
7	Start_RX	写1进入接收模式，接收到有效数据包后，该位自动被硬件清0，如需进入接收模式，则需重写该位为1。

6-0	Frequency_Channel	The frequency of channel=2402+ Frequency Channel (MHz).
-----	-------------------	---

寄存器18 默认值：0x08 可读写

Bit	Bit Name	Description
7-4	TX power	发射功率增益
3	保留	
2	TX power	发射功率增益
1-0	保留	

寄存器22 默认值：0x03 可读写

Bit	Bit Name	Description
7-2	保留	
1-0	RSSI_DISABLE	=11: 禁止RSSI功能 =00: 使能RSSI功能

寄存器30 默认值：0x80 可读写

Bit	Bit Name	Description
7	SPI_CKPHA	=1: 第二个时钟沿捕捉数据; =0: 第一个时钟沿捕捉数据;
6-0	保留	

寄存器37 默认值：0x00 可读写

Bit	Bit Name	Description
7-2	保留	
1	CLEAR_PKT_FLAG	写1清PKT_FLAG标志位
0	CLEAR_FIFO_FLAG	写1清FIFO_FLAG标志位

寄存器39 默认值：0x00 可读写

Bit	Bit Name	Description
7-1	保留	

0	APL_clk_sel	1: 24MHz frequency of external crystal 0: 12MHz frequency of external crystal
---	-------------	--

寄存器64 默认值: 0x78 可读写

Bit	Bit Name	Description
7-5	PREAMBLE_LEN	Preamble长度=1+preamble_len(byte)
4-3	SYNCWORD_LEN	=11: syncword为8字节 {包含reg79、reg78、reg77、reg76、reg75、reg74、reg73、reg72} =10: syncword为6字节 {包含reg79、reg78、reg77、reg76、reg73、reg72} =01: syncword为4字节 {包含reg79、reg78、reg73、reg72} =00: syncword为2字节 {包含reg73、reg72}
2-0	TRAILER_LEN	Trailer长度=4+ (2*trailer_len) (bits)

寄存器65 默认值: 0x00 可读写

Bit	Bit Name	Description
7-6	保留	
5-4	FEC_TYPE	=00: No FEC =01: FEC13 =10: FEC23
3-0	保留	

寄存器70 默认值: 0x03 可读写

Bit	Bit Name	Description
7	保留	
6	SLEEP_ENABLE	写1进入sleep模式
5	LNA_Off_Mode	=1: 近距离模式 =0: 正常工作模式

4-0	保留	
-----	----	--

寄存器80 默认值：0x20 可读写

Bit	Bit Name	Description
7-3	FIFO_Empty_Threshold	FIFO空阈值
2-0	FIFO_Full_Threshold_H	FIFO满阈值高位

寄存器81 默认值：0x47 可读写

Bit	Bit Name	Description
7-6	FIFO_Full_Threshold_L	FIFO满阈值低位
5-0	SYNCWORD_Threshold	SYNCWORD_Threshold值推荐设为 syncword length-1

寄存器84 默认值：0xFD 可读写

Bit	Bit Name	Description
7-2	Scan_RSSI_CH_NUM	扫描RSSI的信道数
1-0	保留	

寄存器86 默认值：0x00 可读写

Bit	Bit Name	Description
7	Start_Scan_RSSI	写1开始扫描RSSI，扫描完成后该位被硬件自动清零.
6-0	Channel_Offset_of_RSSI_Scan	从 (2402+ Channel_Offset_of_RSSI_Scan) MHz信道开始扫描RSSI

寄存器94 默认值：0x00 可读写

Bit	Bit Name	Description
7	THREE_WIRE_SPI_EN	=1: 切换到3线SPI接口 =0: 切换到4线SPI接口
6-0	保留	

寄存器96 只读

Bit	Bit Name	Description
7	CRC_VERIFY_FLAG	=1: CRC认证错误 =0: CRC认证正确

6-0	保留	
-----	----	--

寄存器97 只读

Bit	Bit Name	Description
7	保留	
6	PKT_FLAG	接收/发送包完成标志位，1表示完成
5	FIFO_FLAG	FIFO空/满标志
4-0	保留	

寄存器100 默认值:0x00 可读写

Bit	Bit Name	Description
7-0	FIFO_DATA	将要发送的数据，或已经收到的数据

寄存器104 默认值: 0x00 可读写

Bit	Bit Name	Description
7	CLR_ W_PTR	写1，清空FIFO写指针； 读取时总为0
6	保留	
5-0	FIFO_W_PTR	FIFO写指针，只读

寄存器105 默认值: 0x00 可读写

Bit	Bit Name	Description
7	CLR_ R_PTR	写1，清空FIFO读指针； 读取时总为0
6	保留	
5-0	FIFO_R_PTR	FIFO读指针，只读

寄存器106 默认值: 0x32 可读写

Bit	Bit Name	Description
7-0	DSSS Threshold	匹配滤波器阈值

寄存器107 默认值: 0x11 可读写

Bit	Bit Name	Description
7	Switch_VER	0: 1Mbps调制解调器模式 1: 250Kbps调制解调器模式

6-4	SF_num[2:0]	调制解调器分频因子
3-0	SF_pattern[3:0]	调制解调器扩频模式

寄存器108 默认值: 0x3E 可读写

Bit	Bit Name	Description
7-0	Match_filter_coeff0 [15:8]	匹配滤波器系数

寄存器109 默认值: 0x68 可读写

Bit	Bit Name	Description
7-0	Match_filter_coeff0 [7:0]	匹配滤波器系数

寄存器110 默认值: 0x85 可读写

Bit	Bit Name	Description
7-0	Match_filter_coeff1 [15:8]	匹配滤波器系数

寄存器111 默认值: 0x76 可读写

Bit	Bit Name	Description
7-0	Match_filter_coeff1 [7:0]	匹配滤波器系数

9 射频数据包

Preamble	SYNC	Trailer	Payload	CRC
----------	------	---------	---------	-----

Preamble: 1-8字节, 可编程

SYNC: 16/24/32/40/48/56/64位, 编程同SYNCWORD设备

Trailer: 0-18位, 可编程

Payload: TX/RX数据, 0-64位

CRC: 可选8/16位CRC

10 射频电路电源要求

避免使用变频DC-DC变换, 特别是变换频率在50kHz以下直流变换器会干扰RF通讯, 电机系统内不要省滤波电容; 2 μ A的sleep电流指的是常温下的情况, 高温下sleep电流会上升到>20 μ A, 这是由MOS的漏电特性决定, 非电路设计问题。

11 射频模块应用注意事项

11.1 天线及匹配

匹配的第一要求是50ohm阻抗线, 第二准则是长线串电容谐振消除, 宽线并电感谐振消除。如

果不按照手册中推荐的封装，而是定制封装，由此造成不同的天线管脚位置，那么匹配需要根据实际进行相应的调整。

11.2 I0电压

I0管脚的电压不应高于VDD的电压。

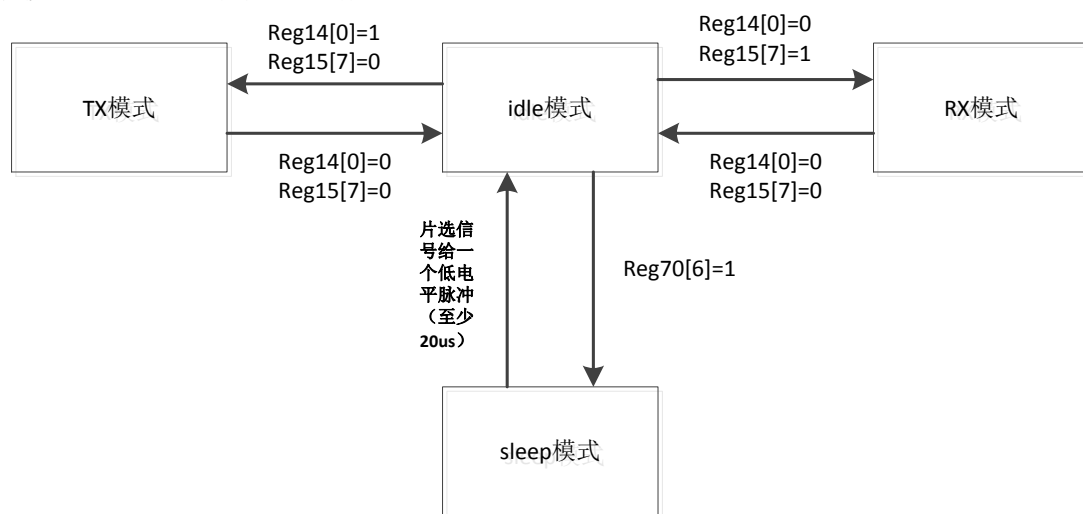
注意：XTALI、XTALO、ANT的电压不得超过2.0V。

11.3 POR

POR 电路要求，下电时电源需要降低到可复位的最高限0.5V，上电时电源爬升速度要求3V/ms。

11.4 状态转换

关于各个模式之间的切换如下图所示：



说明：当启动TX mode之后如果检测到发送结束标志PKT_FLAG之后，会自动进入idle模式。当RX mode之后检测到PKT_FLAG之后也同样会自动进入idle模式，这些是由硬件自动完成的。

注意：

1、Reg14[0]和Reg15[7]不能同时为1。

2、不能在RX或TX状态下发送sleep指令（即写reg70[6]=1），例如，如果在RX状态下没有收到数据而超时，应先清reg15[7]，然后再发送sleep指令，才能使芯片正确进入sleep状态。

11.5 同步字使用注意事项

同步字syncword的字长、阈值、内容的设置会影响射频性能，较长的字长、非规律的内容可以增加通讯距离和收发包成功率，建议使用手册中的同步设置。

同步字的阈值应根据字长作相应调整，字长越短，阈值应设越低值。

12 MCU模块系统框图

12.1 系统框图

12.3 数据存储

地址	RAM
000H	bank0开启
...	通用存储区
07FH	
080H	
...	系统寄存区
...	
0FFH	Bank0的结束区
100H	bank1开始
...	通用存储区
...	
1FFH	Bank1的结束区
200H	bank2开始
...	通用存储区
...	
27FH	Bank2的结束区

特殊寄存器列表:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	L	H	R	Z	Y	X	PFLAG	RBANK	W0	W1	W2	W3	W4	W5	W6	W7
9	@HL	@YZ	P_CTRL	PCL	PCH	OSCM	WDTR	INTRQ		INTEN	REM_REG	P0W	P1W			PEDGE
A	P0M0	P1M0			P5M0	P0	P1			P5	P0M1	P1M1			P5M1	
B					TIM	T1C_L	T1C_H	TCOM	TCOC	TCOR	TCOD/TCOH					
C	PWM0DL	PWM0DH	PWM0PL	PWM0PH	PWM0EN	PWM0C	PWM0TL	PWM0TH	PWM0DIL	PWM0DTH	PWMPSCON		OPTION			
D	PWM1DL	PWM1DH					MTPQ		MTPEN		MTPAH	MTPC	MTPAL		MTPD	
E										T1M1	REV0_T0L	REV0_T0H	REV1_T0L	REV1_T0H		STKP
F	STK7L	STK7H	STK6L	STK6H	STK5L	STK5H	STK4L	STK4H	STK3L	STK3H	STK2L	STK2H	STK1L	STK1H	STK0L	STK0H

特殊功能寄存器地址:

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
80H(r/w)	L	间接寻址寄存器							
81H(r/w)	H	间接寻址寄存器							
82H(r/w)	R	ACC 高8位缓存							
83H(r/w)	Z	间接寻址寄存器							
84H(r/w)	Y	间接寻址寄存器							
85H(r/w)	X	工作寄存器							
86H(r/w)	PFLAG	/T0	/PD	-	-	-	C	DC	Z
87H(r/w)	RBANK	-	-	-	-	-	-	RBANKS1	RBANKS0
88H(r/w)	W0	通用工作寄存器使用							
89H(r/w)	W1	通用工作寄存器使用							
8AH(r/w)	W2	通用工作寄存器使用							
8BH(r/w)	W3	通用工作寄存器使用							
8CH(r/w)	W4	通用工作寄存器使用							

8DH(r/w)	W5	通用工作寄存器使用							
8EH(r/w)	W6	通用工作寄存器使用							
8FH(r/w)	W7	通用工作寄存器使用							
90H(r/w)	@HL	间接寻址寄存器							
91H(r/w)	@YZ	间接寻址寄存器							
92H(r/w)	P_CTRL	-	LVDS1	LVDS0	R_SEL1	R_SEL0	P5_CTRL	P1_CTRL	P0_CTRL
93H(r/w)	PCL	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
94H(r/w)	PCH	-	-	-	-	PC11	PC10	PC9	PC8
95H(r/w)	OSCM	-	-	-	CPUM1	CPUM0	CLKMD	STPHX	-
96H(r/w)	WDTR	看门狗清零寄存器							
97H(r/w)	INTRQ	-	TM1IF	TC0IF	T1IF	P0_P1IF	P54IF	P50IF	INT0IF
99H(r/w)	INTEN	-	TM1IE	TC0IE	T1IE	P0_P1IE	P54IE	P50IE	INT0IE
9AH(r/w)	REM_REG	-	-	-	-	ELE_EN	DRV_S2	DRV_S1	DRV_S0
9BH(r/w)	P0W	-	-	P05W	P04W	P03W	P02W	P01W	P00W
9CH(r/w)	P1W	-	-	P15W	P14W	P13W	P12W	P11W	P10W
9FH(r/w)	PEDGE	-	IRXEN	IRXO	-	INTS	-	PG1	PG0
0A0H(r/w)	P0M0	-	-	P05M0	P04M0	P03M0	P02M0	P01M0	P00M0
0A1H(r/w)	P1M0	-	-	P15M0	P14M0	P13M0	P12M0	P11M0	P10M0
0A4H(r/w)	P5M0	-	-	-	P54M0	-	-	-	P50M0
0A5H(r/w)	P0	-	-	P05	P04	P03	P02	P01	P00
0A6H(r/w)	P1	-	-	P15	P14	P13	P12	P11	P10
0A9H(r/w)	P5	-	-	-	P54	-	-	-	P50
0AAH(r/w)	P0M1	-	-	P05M1	P04M1	P03M1	P02M1	P01M1	P00M1
0ABH(r/w)	P1M1	-	-	P15M1	P14M1	P13M1	P12M1	P11M1	P10M1
0AEH(r/w)	P5M1	-	-	-	P54M1	-	-	-	P50M1
0B5H(r/w)	T1M	TR1	TP1S2	TP1S1	TP1S0	TP1SEL	T1CK	T1GATE	TC0GN
0B6H(r/w)	T1C_L	定时/计数器 T1 低 8 位数据							
0B7H(r/w)	T1C_H	定时/计数器 T1 高 8 位数据							
0B8H(r/w)	TC0M	TC0TR	TC0S2	TC0S1	TC0S0	TC0I1	TC0I0	BUZOUT	PWMOUT
0B9H(r/w)	TC0C	定时/计数器 TC0 的低 8 位数据							
0BAH(r/w)	TC0R	定时/计数器 TC0 自动装载位							
0BBH(r/w)	TC0D/TC0H	定时/计数器 TC0 的 PWM 占空比寄存器/高 8 位数据							
0C0H(r/w)	PWM0DL	PWM0DL<7:0>							
0C1H(r/w)	PWM0DH	-	-	-	-	PWM0DH<3:0>			
0C2H(r/w)	PWM0PL	PWM0PL<7:0>							
0C3H(r/w)	PWM0PH	-	-	-	-	PWM0PH<3:0>			
0C4H(r/w)	PWM0EN	TM1EN	TM1_GN	TM1SEL8	PWM1_O EN	PWM0M	PWM01_ OEN	PWM0_O EN	PWM0_E N
0C5H(r/w)	PWM0C	CK0[3]	CK0[2]	-	-	PWM0S[1:0]		CK0[1]	CK0[0]
0C6H(r/w)	PWM0DL	PWM0DL<7:0>							

0C7H(r/w)	PWM0DH	-	-	-	-	PWM0DH<3:0>			
0C8H(r/w)	PWM0DTL	PWM0DTL<7:0>							
0C9H(r/w)	PWM0DTH	-	-	-	-	PWM0DTH<3:0>			
0CAH(r/w)	PWMPSCON	-	-	PWMPSCON<5:0>					
0D0H(r/w)	PWM1DL	PWM1DL<7:0>							
0D1H(r/w)	PWM1DH	-	-	-	-	PWM1DH<3:0>			
0CCH(r/w)	OPTION	WDT_EN	WDT_pre2	WDT_pre1	WDT_pre0	IE_SEL	P5_scan_w	P1_scan_w	P0_scan_w
0D6H(r/w)	MTPQ	-	-	-	-	MTP_CERS _FLAG	MTP_ERA_ FLAG	MTP_PRO_ FLAG	-
0D8H(r/w)	MTPEN	-	-	MTP_CPEN	-	MTP_CERS	MTP_ERA	MTP_PRO	MTP_RD
0DAH(r/w)	MTPAH	-	-	-	-	-	-	MTP 地址低高 2 位	
0DBH(r/w)	MTPC	-	Pro_clks1	Pro_clks0	Pro_sel4	Pro_sel3	Pro_sel2	Pro_sel1	Pro_sel0
0DCH(r/w)	MTPAL	MTP 地址低 8 位							
0DEH(r/w)	MTPD	MTP8 位数据							
0E9H(r/w)	T1M1	-	-	-	PWM_W2	PWM_W1	INT0_W	T1_OVS	T1_OVW
0EAH(r)	REV0_T0L	学习定时缓存低位，学习信号下降沿锁存							
0EBH(r)	REV0_T0H	学习定时缓存高位，学习信号下降沿锁存							
0ECH(r)	REV1_T0L	学习定时缓存低位，学习信号上升沿锁存							
0EDH(r)	REV1_T0H	学习定时缓存高位，学习信号上升沿锁存							
0EFH(r/w)	STKP	GIE	*	LVD2X	LVD1D8	-	STKPB2	STKPB1	STKPB0
0F0H(r/w)	STK7L	S7PC7	S7PC6	S7PC5	S7PC4	S7PC3	S7PC2	S7PC1	S7PC0
0F1H(r/w)	STK7H	-	-	-	-	S7PC11	S7PC10	S7PC9	S7PC8
0F2H(r/w)	STK6L	S6PC7	S6PC6	S6PC5	S6PC4	S6PC3	S6PC2	S6PC1	S6PC0
0F3H(r/w)	STK6H	-	-	-	-	S6PC11	S6PC10	S6PC9	S6PC8
0F4H(r/w)	STK5L	S5PC7	S5PC6	S5PC5	S5PC4	S5PC3	S5PC2	S5PC1	S5PC0
0F5H(r/w)	STK5H	-	-	-	-	S5PC11	S5PC10	S5PC9	S5PC8
0F6H(r/w)	STK4L	S4PC7	S4PC6	S4PC5	S4PC4	S4PC3	S4PC2	S4PC1	S4PC0
0F7H(r/w)	STK4H	-	-	-	-	S4PC11	S4PC10	S4PC9	S4PC8
0F8H(r/w)	STK3L	S3PC7	S3PC6	S3PC5	S3PC4	S3PC3	S3PC2	S3PC1	S3PC0
0F9H(r/w)	STK3H	-	-	-	-	S3PC11	S3PC10	S3PC9	S3PC8
0FAH(r/w)	STK2L	S2PC7	S2PC6	S2PC5	S2PC4	S2PC3	S2PC2	S2PC1	S2PC0
0FBH(r/w)	STK2H	-	-	-	-	S2PC11	S2PC10	S2PC9	S2PC8
0FCH(r/w)	STK1L	S1PC7	S1PC6	S1PC5	S1PC4	S1PC3	S1PC2	S1PC1	S1PC0
0FDH(r/w)	STK1H	-	-	-	-	S1PC11	S1PC10	S1PC9	S1PC8
0FEH(r/w)	STK0L	S0PC7	S0PC6	S0PC5	S0PC4	S0PC3	S0PC2	S0PC1	S0PC0
0FFH(r/w)	STK0H	-	-	-	-	S0PC11	S0PC10	S0PC9	S0PC8

13 功能描述

13.1 操作寄存器

13.1.1 H/L（间接寻址寄存器）

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
80H(R/W)	L	1、通用工作寄存器； 2、RAM 数据寻址指针@HL (90H)。							
81H(R/W)	H								

例：用 H、L 作为数据指针，访问bank0 中020H 处的内容。

B0MOV H, #00H

B0MOV L, #20H

B0MOV A, @HL

例：对bank0中的数据进行清零处理。

CLR H ; H = 0, 指向bank 0。

B0MOV L, #7FH ; L = 7FH。

CLR_HL_BUF:

CLR @HL ; @HL 清零。

DECMS L ; L - 1, 如果 L = 0, 程序结束。

JMP CLR_HL_BUF

CLR @HL

END_CLR:

...

...

13.1.2 Y/Z（间接寻址寄存器）

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
83H(R/W)	Z	1、通用工作寄存器； 2、RAM 数据寻址指针@HL (91H)。 3、配合指令 MOVC 对 ROM 数据进行查表。							
84H(R/W)	Y								

例：用 Y、Z 作为数据指针，访问bank0中 020H 处的内容。

B0MOV Y, #00H

B0MOV Z, #20H

B0MOV A, @YZ

例：对bank0中的数据进行清零处理。

CLR Y ; Y = 0, 指向bank 0。

B0MOV Z, #7FH ; Z = 7FH。

CLR_YZ_BUF:

CLR @YZ ; @YZ 清零。

```

DECMS Z                ; Z - 1, 如果 Z = 0, 程序结束。
JMP CLR_YZ_BUF
CLR @YZ
END_CLR:
...

```

13.1.3 R 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
82H(r/w)	R	1、作为工作寄存器使用； 2、存储执行查表指令后的高字节数据。（执行MOVC 指令，指定ROM单元的高字节数据会被存入R 寄存器而低字节数据则存入ACC。） 3、存储乘法指令的高8位数据（执行MUL 指令，运算结果的高字节数据会被存入R 寄存器而低字节数据则存入ACC。）							

例：查找 ROM 地址为“TABLE1”的值。

```

B0MOV Y, #TABLE1$M      ; 设置 TABLE1 地址高字节。
B0MOV Z, #TABLE1$L      ; 设置 TABLE1 地址低字节。
MOVC                     ; 查表, R = 00H, ACC = 35H。
INCMS Z                  ; 查找下一地址。
JMP @F                   ; Z 没有溢出。
INCMS Y                  ; Z 溢出 ( FFH -> 00 ), -> Y=Y+1
NOP
@@: MOVC                  ; 查表, R = 51H, ACC = 05H。
... ;

```

TABLE1:

```

DW 0035H                ; 定义数据表 ( 16 位 ) 数据。
DW 5105H
DW 2012H
...

```

13.1.4 X (X 寄存器)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
85H(r/w)	X	通用工作寄存器使用							

13.1.5 W (W 寄存器)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
88H(r/w)~ 8FH(r/w)	W0~ W7	通用工作寄存器使用							

13.1.6 P_CTRL 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
92H(r/w)	P_CTRL	-	LVD_sel<1>	LVD_sel<0>	R_SEL1	R_SEL0	P5_CTRL	P1_CTRL	P0_CTRL

P0_ctrl: P0 扫描口使能位

=1: 使能

=0: 禁止P1_ctrl:

P1 扫描口使能位

=1: 使能

=0: 禁止P5_ctrl:

P5 扫描口使能位

=1: 使能

=0: 禁止

R_SEL<1:0>: 学习模式电阻选择位

=00: 5K

=01: 10K

=10: 20K

=11: 30K

LVD_S<1:0>: LVD 变换选择位

=00: 2.0v 电压检测

=01: 2.2v 电压检测

=10: 2.4v 电压检测

=11: 2.6v 电压检测

13.1.7 PC（程序计数器）与堆栈

程序计数器PC是一个13位二进制程序地址寄存器，分高4位和低8位。专门用来存放下一条需要执行的指令的内存地址。通常程序计数器会随程序中指令的执行自动增加。

若程序执行CALL和JMP 指令时，PC 指向特定的地址。下图描述了其结构。

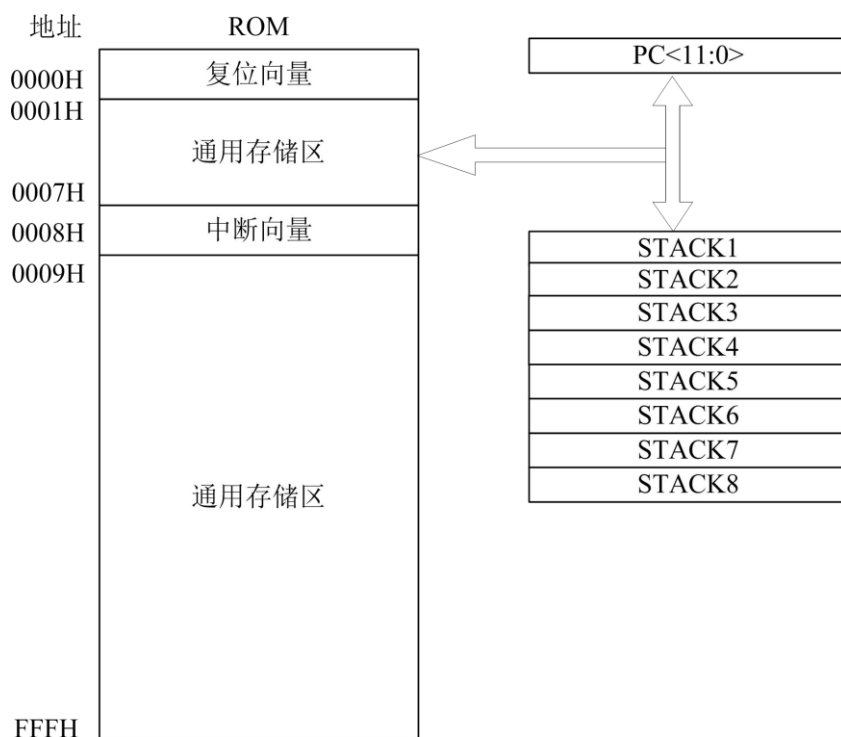


图 5-3 程序计数器结构

13.1.7.1 PCH 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
94H(r/w)	PCH	-	-	-	-	PC11	PC10	PC9	PC8

13.1.7.2 PCL 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
93H(r/w)	PCL	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0

产生 8K Byte ROM 地址以获取对应程序指令代码。

复位后 PCL 的所有位均清零

JMP 指令直接装载 PC 的 12 位。因此，“JMP”指令跳转范围为 8K 程序区间

CALL 指令加载 PC 的 12 位，然后 PC+1 进入堆栈。

RET、RETI、RETLW 指令将栈顶数据装入 PC。

ADD PCL,A (ADC PCL,A) 允许“A”的值加到当前 PC，PC 的高 4 位将自然进位。

MOV PCL,A 允许将寄存器“A”的值装入 PC 的低 8 位，同时 PC 的高 4 位保持不变。

SUB PCL,A (SBC PCL,A) 允许“A”的值被减到当前 PC，但 PC 的高 4 位保持不变。

改变 PCL 内容的指令需要 2 个指令周期，除此之外，所有的指令均只需 1 个指令周期。

13.1.8 PFLAG（状态寄存器）

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
86H(r/w)	PFLAG	/TO	/PD	-	-	-	Z	C	DC

寄存器包含 ALU 的算术状态，RESET 状态。

C: 进位/借位（加减法指令）

=1, 有进位/无借位

=0, 无进位/有借位

DC: 半进位/半借位（加减法指令）

=1, 有低 4 位进位/无低 4 位借位

=0, 无低 4 位进位/有低 4 位借位

Z: 零标志位

=1, 逻辑操作结果为 0

=0, 逻辑操作结果不为 0

/PD: 掉电标志位

=1, 系统上电或执行 CLRWDWT 指令后

=0, 执行 SLEEP 指令后

/TO: 时间溢出标志位

=1, 系统上电或执行 CLRWDWT 或执行 SLEEP 指令后

=0, 看门狗溢出后

13.1.9 RBANK（RAM 存储区选择寄存器）

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
87H(r/w)	RBANK	-	-	-	-	-	-	RBANKS1	RBANKS0

RBANKS<1: 0>: RAM 页面控制位

=00: 选择 BANK0

=01: 选择 BANK1

=10: 选择 BANK2

=11: 选择 BANK1

注：执行带 B0 开头的指令时(如 B0MOV)时，无论在哪个 BANK，都执行 BANK0 区间的 RAM。

13.1.10 OPTION

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0CCH(r/w)	OPTION	WDT_EN	WDT_pre2	WDT_pre1	WDT_pre0	IE_SEL	P5_scan_w	P1_scan_w	P0_scan_w

WDT_EN: 看门狗使能位

=0: 有效

=1: 无效WDT_pre<2:0>:

WDT 溢出时间选择位

WDT_pre2	WDT_pre1	WDT_pre0	WDT 的分频比
0	0	0	1:1
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

IE_SEL: 内外置三极管选择位 (P54 发码脚)

=1: 内置三极管

=0: 外置三极管P5_scan_w:

P5 扫描端口唤醒使能位

=0, 使能

=1, 禁止

P1_scan_w: P1 扫描口唤醒使能位

=0, 使能

=1, 禁止 P0_scan_w:

P0 扫描口唤醒使能位

=0, 使能

=1, 禁止

13.1.11 WDTR 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
96H(w)	WDTR	看门狗清零寄存器							

例：如下是对看门狗定时器的操作，在主程序开头对看门狗清零。

```

MOV      A,#5AH
B0MOV    WDTR,A
. . .
CALL     SUB1
CALL     SUB2
. . .
JMP      MAIN

```

清门狗定时器应注意：

- 1、对看门狗定时器清零之前，检查 I/O 口的状态和 RAM 的内容可增强程序的可靠性；
- 2、不能在中断中对看门狗清零，否则无法检测到主程序跑飞的情况。
- 3、程序中应该只在主程序中有一次清看门狗的动作，这种架构能够最大限度地发挥看门狗的保护功能。

13.1.12 OSCM 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
95H(r/w)	OSCM	-	-	-	CPUM1	CPUM0	CLKMD	STPHX	-

CPUM<1:0>: CPU 工作模式控制位

00: 普通模式 01: 睡眠模式 10: 绿色模式 11: 系统保留

CLKMD: 系统时钟模式控制位

=0: 普通（双时钟）模式，高速时钟作为系统时钟

=1: 低速模式，低速时钟作为系统时钟

STPHX: 高速振荡器控制位

=0: 运行

=1: 停止，内部低速 RC 振荡器仍然运行

例：停止高速振荡器

B0BSET FSTPHX ; 停止外部高速振荡器

注：STPHX 为内部高速 RC 振荡器和外部高速振荡器的控制位。当 STPHX=0 时内部高速 RC 振荡器和外部高速振荡器正常运行；当 STPHX=1 时内部高速 RC 振荡器和外部高速振荡器停止运行。不同的高速时钟选项决定不同的 STPHX 功能。

13.1.13 REM_REG

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
9AH(r/w)	REM_REG	-	-	-	-	ELE_EN	DRV_S2	DRV_S1	DRV_S0

ELE_EN: 在使用内置三极管时，电源是否加电解的控制位

=0: 省电解

=1: 必须加电解 DRV_S<2:0>:

REM 端内置三极管的驱动能力选择

=000: 最大，约 400mA

=011: 约 340mA

=100: 最小，约 150mA

13.1.14 REV0_TOL

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
----	----	----	----	----	----	----	----	----	----

0EAH(r)	REV0_TOL	REV0_TOL<7:0>
---------	----------	---------------

学习定时缓存低位，学习信号下降沿锁存，只读。

13.1.15 REV0_T0H

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0EBH(r)	REV0_T0H	REV0_T0H<7:0>							

学习定时缓存高位，学习信号下降沿锁存，只读。

13.1.16 REV1_T0L

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0ECH(r)	REV1_T0L	REV1_T0L<7:0>							

学习定时缓存低位，学习信号上升沿锁存，只读。

13.1.17 REV1_T0H

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0EDH(r)	REV1_T0H	REV1_T0H<7:0>							

学习定时缓存高位，学习信号上升沿锁存，只读。

13.2 TC0（16 定时/计数器寄存器）

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0B9H(w/r)	TC0C	定时器 TC0 低 8 位							
0B8H(w/r)	TC0M	模式选择							
0BAH(w/r)	TC0R	自动重装载寄存器							
0BBH(w/r)	TC0H/TC0D	PWM0 高电平占空比选择/定时器高 8 位							

13.2.1 概述

定时/计数器 TC0 具有多时钟源，可根据实际需要选择内部时钟 F_{cpu} 和 F_{osc} 或者外部时钟，寄存器 TC0M 控制时钟源的选择。当 TC0C 从 FFFFH 溢出到 0000H 时，TC0C 在继续计数的同时产生一个溢出信号 TC0IF，TC0IE=1，触发 TC0 中断请求。

TC0 的主要作用如下：

- 16 位可编程定时器：根据选定的时钟频率在特定时间产生中断信号(非 PWM 模式)；
- 外部事件计数：对外部事件计数；
- PWM 输出（TC0 为 8 位）；
- 蜂鸣器输出（TC0 为 8 位）。

13.2.2 TC0CL

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0B9H(r/w)	TC0C	TC0 的定时寄存器的低 8 位							

13.2.3 TC0M

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0B8H(r/w)	TC0M	TC0TR	TC0S2	TC0S1	TC0S0	TC0I1	TC0I0	BUZOUT	PWMOUT

TC0TR: TC0 启动控制位。

=0 : 禁止 TC0 定时器;

=1 : 开启 TC0 定时器。

TC0S[2:0]: TC0 分频选择位。

TC0S[2:0]	选择Fcpu	选择Fosc
000	Fcpu / 256	Fosc / 128
001	Fcpu / 128	Fosc / 64
010	Fcpu / 64	Fosc / 32
011	Fcpu / 32	Fosc / 16
100	Fcpu / 16	Fosc / 8
101	Fcpu / 8	Fosc / 4
110	Fcpu / 4	Fosc / 2
111	Fcpu / 2	Fosc / 1

TC0I1: TC0I0: TC0 时钟信号控制位。

=00: Fcpu

=01: 外部时钟

=10: Fosc

=11: Fcarry(红外学习载波输入)

BUZOUT:

=0: 禁止 BUZ 输出

=1: 使能 BUZ 输出

PWMOUT:

=0: 禁止 PWM 输出

=1: 使能 PWM 输出

注: 当 PWM 和 BUZ 都无效时, TC0 为 16 位定时/计数器。TC0H 为高 8 位, TC0C 为低 8 位。

13.2.4 TC0R

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0BAH(r/w)	TC0R	自动重装载寄存器							

当 TC0 溢出时, TC0R 的值自动装入 TC0 中。这样, 用户在使用的过程中就不需要在中断中重新赋值。TC 为双重缓存器结构。若程序对 TC0R 进行了修改, 那么修改后的 TC0R 值先被暂存在 TC0R

的第一个缓存器中，直到当前 TC0 溢出后，才被真正存入 TC0R 缓存器中，从而避免 TC 中断时间出错以及 PWM 和蜂鸣器误动作。

TC0R 初始值计算公式如下：

$$\text{TC0R 初始值} = N - (\text{TC 中断间隔时间} \times \text{输入时钟})$$

上式中，N 为 TC 的最大计数范围 256。

例：TC 中断间隔时间设置为 10ms，时钟源选 Fcpu (TnCKS = 0)，使能 PWM 输出 (PWMEN = 1)，高速时钟为 4MHz，Fcpu = Fosc/4，TPS = 010 (Fcpu/64)。

$$\begin{aligned}\text{TC0R 初始值} &= N - (\text{TC 中断间隔时间} \times \text{输入时钟源}) \\ &= 256 - (10\text{ms} \times 4\text{MHz} / 4 / 64) \\ &= 256 - (10 \times 2 \times 4 \times 106 / 4 / 64) = 100 = 64\text{H}\end{aligned}$$

注：只有在 PWM、BUZ 模式下，系统自动开启自动装载功能。

13.2.5 TC0D

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0BBH(r/w)	TC0D	PWM 高电平占空比控制寄存器							

TC0R 控制 PWM 的周期及分辨率，TC0D 控制 PWM 高电平占空比，当 TC0C=TC0D 时，PWM 翻转为低，完成高电平的占空比。

TC0D 初始值计算公式如下：

$$\text{TC0D 初始值} = \text{TC0R} + (\text{高电平周期} \times \text{输入时钟})$$

例：为了获得 1/3 占空比的 PWM 信号，时钟源选 Fcpu=8M/2, TC0rate=111(Fcpu/2)

TC0R=CAH, PWM 周期为 26.5us (38K), 高电平宽度为 8.8us

$$\begin{aligned}\text{TC0D 初始值} &= \text{CAH} + (8.8\text{us} \times 8\text{MHz} / 2 / 2) \\ &= \text{CAH} + 12\text{H} \\ &= \text{DCH}\end{aligned}$$

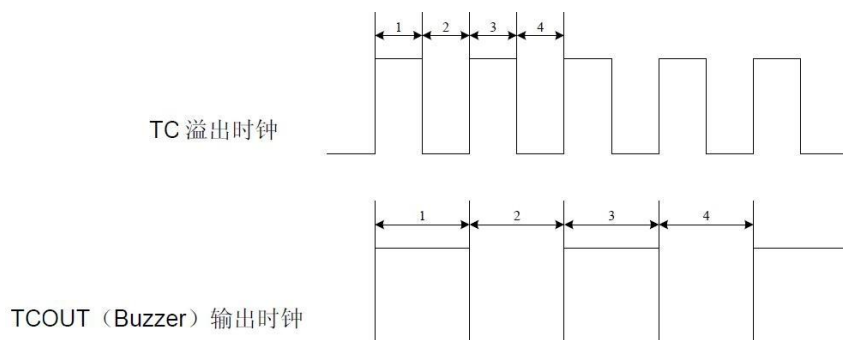
例：为了获得 1/3 占空比的 PWM 信号，时钟源选 Fcpu=8M/1, TC0rate=111(Fcpu/2)

TC0R=97H, PWM 周期为 26.5us (38K), 高电平宽度为 8.8us

$$\begin{aligned}\text{TC0D 初始值} &= 97\text{H} + (8.8\text{us} \times 8\text{MHz} / 1 / 2) \\ &= 97\text{H} + 23\text{H} \\ &= \text{BAH}\end{aligned}$$

13.2.6 BUZ 波形

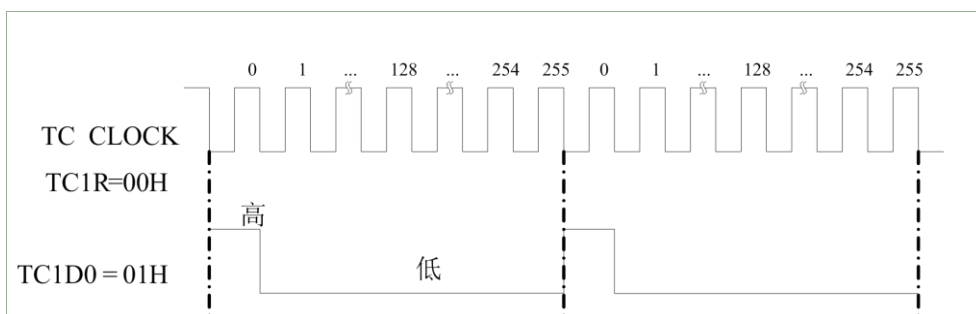
对 TC 时钟频率进行适当设置可得到特定频率的蜂鸣器输出 (TCOUT)，并通过 PWM 引脚输出。单片机内部设置 TC 的溢出频率经过 2 分频后作为 TCOUT 的频率，即 TC 每溢出 2 次 TCOUT 输出一个完整的脉冲，此时 I/O 功能自动被禁止。TCOUT 输出波形如下：



若时钟选择4MHz，系统时钟源采用外部时钟 $F_{osc}/4$ ，程序中设置 $TPS = 110$ ， $TC = TCOR = 131$ ，则TC的溢出频率为2KHz，TCOUT 的输出频率为1KHz。

13.2.7 PWM 波形

PWM信号靠TC0C，TC0R，TC0D的比较产生，当TC0溢出时，PWM（P54）输出低电平，为PWM的初始态，此时TC0C重新载入TC0R寄存器的值，TC0R决定PWM的周期及分辨率。TC0开始计数后，当TC0C等于TC0D时，PWM（P54）翻转为高电平，TC0继续计数，当TC0溢出时，一个PWM周期完成，PWM重新自动载入TC0R的值并且输出低电平，开始下一个周期。TC0R决定了PWM的分辨率，如果TC0R为0x00，则分辨率为1/256，如果TC0R为0x80，则分辨率为1/128，可参看下图示。



注：如果 TC0D0 小于 TC0R，则 PWM 一直输出高电平。

13.3 T1（定时/计数器寄存器）

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0B6H(w/r)	T1C_L	定时器 T1 低8 位							
0B5H(w/r)	T1M	T1 模式控制寄存器							
0B7H(w/r)	T1C_H	定时器 T1 高8 位							
0E9H(w/r)	T1M1	T1 模式控制寄存器 1							

13.3.1 概述

16位普通定时/计数器，内部时钟来自 F_{cpu} 或低速时钟，当T1从FFFFH 溢出到0000H时，T1在继续计数的同时产生一个溢出信号T1IF，中断使能TOIE有效时，触发T1中断请求。

13.3.2 T1M 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0B5H(r/w)	T1M	TR1	TP1S2	TP1S1	TP1S0	TP1SEL	T1CK	T1GATE	TC0GN

TR1: TR1 使能控制。

=0 : 禁止;

=1 : 使能。

TP1S[2:0]: T1 分频选择位。

TP1S[2:0]	T1CK=0	T1CK=1
000	Fcpu / 256	Fosc / 128
001	Fcpu / 128	Fosc / 64
010	Fcpu / 64	Fosc / 32
011	Fcpu / 32	Fosc / 16
100	Fcpu / 16	Fosc / 8
101	Fcpu / 8	Fosc / 4
110	Fcpu / 4	Fosc / 2
111	Fcpu / 2	Fosc / 1

TP1SEL: T1 预分频是否使用;

=0 : 不使用。T1 的时钟源为 1 个 Fcpu, 预分频位不再有效。

=1 : 使用

T1CK: T1 时钟源;

=0 : T1 的时钟源为 Fcpu。

=1 : T1 的时钟源为低速 RC 振荡器

T1GATE: T1 使能选择位。

=0 : 选择 TR1 为 T1 使能位。当 TR1 为 1 时, T1 开始计数。

=1 : 选择红外学习时, 载波下降沿信号为使能。不论 TR1 为何值, 都会计数。

TC0GN: TC0 绿色模式溢出唤醒使能控制。

=0 : 禁止;

=1 : 使能。

13.3.3 T1C 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0B6H(r/w)	T1C_L	T1 定时寄存器的低 8 位							
0B7H(r/w)	T1C_H	T1 定时寄存器的高 8 位							

13.3.4 T1M1 寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0E9H(r/w)	T1M1	-	-	-	PWMW2	PWMW1	INT0W	T1OVS	T1GN

PWMW<2:1>PWM2 上升沿唤醒延时。

=00: 2us

=01: 4us

=10: 8us

=11: 12us

INT0W: INT 绿色模式唤醒选择控制位。

=0 : 禁止 PWM 上升沿在绿色模式唤醒

=1 : 允许 PWM 上升沿在绿色模式唤醒

注: PWM 上升沿作为外中断信号, 触发绿色模式唤醒。(P1M1.5 需置 0)

T1OVS: T1 绿色模式溢出唤醒选择位。

=0 : 选择 T0 的 16 位为溢出位

=1 : 选择 T0 的低 8 位为溢出位

T1GN: T1 绿色模式溢出唤醒使能控制。

=0 : 禁止

=1 : 使能

13.4 带死区互补的 12 位 PWM

13.4.1 PWM 特性

- 1 组带死区互补 PWM 或 3 路独立 PWM 输出
- 提供每个 PWM 周期溢出中断, 但中断共用同一个向量入口
- 输出极性可选择
- PWM 工作时钟源可设定时钟分频比
- PWM 可做定时器/计数器使用(TM1)

13.4.2 PWM 输出模式

PWM 模块包含 1 个独立的波形发生模块, 对应的 3 个 PWM 输出为 PWM0X、PWM1X、PWM2X 通过控制相关寄存器可使没对 PWM 输出配置成互补模式或独立输出模式。

13.4.2.1 PWM 互补输出模式

当 PWM0M 置 0: PWM 将工作在互补输出模式, 互补模式时, 通常使能 PWMx&PWMx1 输出, 此时可以控制对应的周期寄存器、占空比寄存器及死区时间寄存器, 从而控制互补波形的输出。互补输出时可选择 PWM0 &PWM01 输出极性, 方便用户各种电平驱动需求。

13.4.2.2 PWM 独立输出模式

当 PWM0M 置 1: PWM 将工作在独立输出模式, 独立输出模式时, 可以控制相关寄存器使能对应 PWM 端口单一输出或同时输出, 同时让 PWMx&PWMx1 输出时, 其周期相同但占空比可单独设置。当为独立模式时占空比寄存器将控制 PWMx 的占空比, 死区时间控制寄存器将控制 PWMx1 的占空比, 独立输出

时也可控制 PWMx&PWMx1 输出极性，方便用户各种电平驱动需求。(x=0)

13.4.3 PWM 相关寄存器

13.4.3.1 PWM0 使能寄存器 PWM0EN

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0C4H(r/w)	PWM0EN	TM1EN	TM1_GN	TM1SEL8	PWM1_OEN	PWM0M	PWM01_OEN	PWM0_OEN	PWM0_EN

TM1EN: TM1 定时器总使能

=0: 禁止

=1: 使能 TM1_GN:

TM1 绿色模式唤醒使能

=0: 禁止

=1: 使能

TM1SEL8: 8/ 12 位定时器选择位（定时器时 PWM0_EN 必须无效）

=0: 8 位

=1: 12 位

PWM1_OEN: PWM1 输出控制位

=0: 禁止 PWM1 输出

=1: 允许 PWM1 输出

PWM0M: PWM0 工作模式选择位

=0: PWM0&PWM01 工作于互补输出模式

=1: PWM0&PWM01 工作于独立输出模式

注: 修改 PWM0 工作模式时建议先关闭 PWM0 模块。

PWM01_OEN: PWM01 输出控制位

=0: 禁止 PWM01 输出

=1: 允许 PWM01 输出

PWM0_OEN: PWM0 输出控制位

=0: 禁止 PWM0 输出

=1: 允许 PWM0 输出

注: PWM0 允许输出, 必须在 PWM0_EN 置 1 下才有效, 否则为 PWM0 输出关闭状态（输出时对应端口必须设为输出模式）；即使都禁止输出, 只有 PWM0_EN 位被使能, PWM0 都可以溢出中断, 即此时 PWM0 可以作为定时器使用, 此控制位修改立即生效。

PWM0_EN: PWM0 模块使能控制位

=0: 关闭 PWM0 模块

=1: 打开 PWM0 模块（重新计数）

注: 关闭时, PWM0 计数停止, 输出立即关闭。打开时, PWM0 计数器都重新从 1 开始计数, 输出受 PWM0_OEN 和 PWM01_OEN 控制。修改 PWM0 工作模式时建议先关闭 PWM0 模块。

13.4.3.2 PWM0 控制寄存器 PWM0C

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0C5H(r/w)	PWM0C	CK0[3]	CK0[2]	-	-	PWM0S[1:0]		CK0[1]	CK0[0]

CK0[3:2]: PWM0 时钟源选择位 (高 2 位)

PWM0S[1:0]: PWM0 和 PWM01 输出模式选择位

=00: PWM0 和 PWM01 均为高有效

=01: PWM0 为高有效, PWM01 为低有效

=10: PWM0 为低有效, PWM01 为高有效

=11: PWM0 和 PWM01 均为低有效

注: PWM 高低有效指 PWM 占空比为高或低。

CK0[1:0]: PWM0 时钟源选择位 (低 2 位)

CK0[2:0]	CK0[3]=0	CK0[3]=1
111	Fcpu/2	Fosc/1
110	Fcpu/4	Fosc/2
101	Fcpu/8	Fosc/4
100	Fcpu/16	Fosc/8
011	Fcpu/32	Fosc/16
010	Fcpu/64	Fosc/32
001	Fcpu/128	Fosc/64
000	Fcpu/256	Fosc/128

13.4.3.3 PWM0 周期寄存器 PWM0PL、PWM0PH

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0C2H(r/w)	PWM0PL	PWM0PL[7:0]							
0C3H(r/w)	PWM0PH	-	-	-	-	PWM0PH[3:0]			

PWM 周期=[0xFFF-PWM0PH:PWM0PL]*PWM0 工作时钟源周期。

注: 修改 PWM0 周期寄存器, 须先修改高位后修改低位, 且修改都在下一个周期才有效。

注: PWM0PH: PWM0PL 组成 12 位数据。

13.4.3.4 PWM0 占空比寄存器 PWM0DL、PWM0DH

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0C0H(r/w)	PWM0DL	PWM0DL[7:0]							
0C1H(r/w)	PWM0DH	-	-	-	-	PWM0DH[3:0]			

PWM0 占空比=[PWM0DH:PWM0DL-PWM0PH:PWM0PL]*PWM 工作时钟源周期。

注: 修改 PWM0 占空比寄存器, 操作类似修改 PWM 周期寄存器, 都是必须先修改高位后修改低位, 且修改都在下一个周期才有效。

注: 定时器 TM1 计数到 FFF 溢出后重载 PWM0PH/L 的值, 计数到 PWM0DH/L 时 PWM0 翻转 (溢出到翻转这段时间为占空比), 计数到 FFF 溢出再翻转。

13.4.3.5 PWM0 死区时间寄存器 PWM0DTL、PWM0DTH

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0C8H(r/w)	PWM0DTL	PWM0DTL[7:0]							
0C9H(r/w)	PWM0DTH	-	-	-	-	PWM0DTH[3:0]			

独立模式下，PWM01 占空比= $[PWM0DTH:PWM0DTL-PWM0PH:PWM0PL]*PWM$ 工作时钟源周期

互补模式下：PWM01 死区时间= $[PWM0DTH:PWM0DTL]*PWM$ 工作时钟周期。

注：互补模式下：PWM01 死区时间必须小于 PWM0 占空比时间，死区时间与占空比时间的和必须小于 PWM 周期，互补模式下死区时间由 PWM0DTH:PWM0DTL 控制。

注：修改 PWM01 占空比寄存器，操作类似修改 PWM 周期寄存器，都是必须先修改高位后修改低位，且修改都在下一个周期才有效。

注：独立模式下定时器 TM1 计数到 FFF 溢出后重载 PWM0PH/L 的值，计数到 PWM0DTH/L 时 PWM01 翻转(溢出到翻转这段时间为占空比)，计数到 FFF 溢出再翻转。

13.4.3.6 PWM1 占空比寄存器 PWM1DL、PWM1DH

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0D0H(r/w)	PWM1DL	PWM1DL[7:0]							
0D1H(r/w)	PWM1DH	-	-	-	-	PWM1DH[3:0]			

PWM1 占空比= $[T3PWM1DH:T3PWM1DL-T3PWM0PH:T3PWM0PL]*PWM$ 工作时钟源周期。

注：修改 PWM1 占空比寄存器，操作类似修改 PWM 周期寄存器，都是必须先修改高位后修改低位，且修改都在下一个周期才有效。

注：定时器 TM1 计数到 FFF 溢出后重载 PWM0PH/L 的值，计数到 PWM1DH/L 时 T3PWM1 翻转(溢出到翻转这段时间为占空比)，计数到 FFF 溢出再翻转。

13.4.3.7 PWM0 计数器/定时器时间寄存器 PWM0TL、PWM0TH

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0C6H(r/w)	PWM0TL	PWM0TL[7:0]							
0C7H(r/w)	PWM0TH	-	-	-	-	PWM0TH[3:0]			

PWM0 模块也可以做定时器，无自动装载功能，配置 TM1SEL8 (PWM0EN.5) 位：清 0 可选择 8 位计数器模式；置 1 可选 12 位计数器模式。做计数器模式下，需要把 PWM0 模块使能控制位 PWM0_EN (PWM0EN.0) 清零。

13.4.3.8 PWM0 输出端口选择寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0CAH(r/w)	PWMPSCON	-	-	PWM1S[1:0]		PWM01S[1:0]		PWM0S[1:0]	

PWM0S[1:0]: PWM0 输出端口选择

=00: 无输出

=01: P0.1 输出 PWM0

=10: P1.2 输出 PWM0

=11: P0.1、P1.2 输出 PWM0

PWM01S[1:0]: PWM01 输出端口选择

=00: 无输出

=01: P0.2 输出 PWM01

=10: P1.3 输出 PWM01

=11: P0.2、P1.3 输出 PWM01

PWM1S[1:0]: PWM1 输出端口选择

=00: 无输出

=01: P0.3 输出 PWM1

=10: P1.4 输出 PWM1

=11: P0.3、P1.4 输出 PWM1

13.5 中断

13.5.1 概述

本芯片提供7个中断源：TC0、T0、TM1、P0_P1端口变化、P54端口变化、P50端口变化外部中断INT0。外部中断等可以将系统从睡眠模式中唤醒进入高速模式，在返回到高速模式前，中断请求被锁定。一旦程序进入中断，寄存器STKP的位GIE被硬件自动清零以避免响应其它中断。系统退出中断后，硬件自动将GIE置“1”，以响应下一个中断。中断请求存放在寄存器INTRQ中。

注：程序响应中断时，必须开启全局中断控制位GIE。

13.5.2 中断控制寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
99H(r/w)	INTEN	-	TM1IE	TC0IE	T1IE	P0_P1IE	P54IE	P50IE	INT0IE

TM1IE: 12bitPWM 定时器中断控制位

=0: 无效

=1: 有效TC0IE:

TC0 中断控制位

=0: 无效

=1: 有效

T1IE: T1 中断控制位

=0: 无效

=1: 有效

P0_P1IE: P0_P1 电平变化中断使能（需使能 PnW 寄存器）

=0: 无效

=1: 有效 P54IE:

P54 电平变化中断使能

=0: 无效

=1: 有效 P50IE:

P50 电平变化中断使能

=0: 无效

=1: 有效

INT0IE:: INT0 外部中断 (P1.5) 控制位

=0: 无效

=1: 有效

中断请求寄存器 INTRQ 中存放各中断请求标志。一旦有中断请求发生, 则 INTRQ 中对应位将被置“1”, 该请求被响应后, 程序应将该标志位清零。根据 INTRQ 的状态, 程序判断是否有中断发生, 并执行相应的中断服务。

13.5.3 INTRQ

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
97H(r/w)	INTRQ	-	TM1IF	TC0IF	T1IF	P0_P1IF	P54IF	P50IF	INT0IF

TM1IF: TM1 中断请求标志

=0: TM1 无中断请求

=1: TM1 有中断请求

TC0IF: TC0 中断请求标志

=0: TC0 无中断请求

=1: TC0 有中断请求

T1IF: T1 中断请求标志

=0: T1 无中断请求

=1: T1 有中断请求

P0_P1IF: P0_P1 电平变化中断请求标志

=0: 无中断请求

=1: 有中断请求P54IF:

P54 电平变化中断请求标志

=0: 无中断请求

=1: 有中断请求P50IF:

P50 电平变化中断请求标志

=0: 无中断请求

=1: 有中断请求INT0IF::

INT0 外部中断请求标志

=0: INT0 无中断请求

=1: INT0 有中断请求

13.5.4 PEDGE

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
9FH(r/w)	PEDGE	-	IRXEN	IRXO	-	INTS	-	P15G1	P15G0

IRXEN 学习模式下 P5.4 学习载波输入控制位

=0: 无效

=1: 有效, P5.4 置为输入模式, 输入学习的载波给运放。

IRXO 学习模式下 P50 作为载波输出控制位

=0: 无效

=1: 有效(必须 IRXEN 同时有效), P50 输出的和学习到的波形一致。

INTS: 外中断源选择位。

=0 : 选择 P1.5 作为外中断源;

=1 : 选择红外学习的载波信号作为外中断源。

P15G<1: 0>: 外中断边沿控制位

=00: 保留

=01: 上升沿

=10: 下降沿

=11: 上升/下降沿

13.5.5 STKP

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0EFH(r/w)	STKP	GIE	-	LVD2X	LVD1D8	-	STKPB2	STKPB1	STKPB0

只有当全局中断控制位 GIE 置 1 时, 程序才能响应中断。一旦发生中断, PC 指向中断向量地址 008H, 堆栈加 1。

GIE: 全局中断控制位

=0: 禁止全局中断

=1: 允许全局中断

LVD2X: 低压检测标志位(2.0v、2.2v、2.4v、2.6v)

=0: 标志无效

=1: 标志有效 LVD1D8:

低压检测标志位(1.85v)

=0: 标志无效

=1: 标志有效

注: LVD1D8 具有复位功能, 如果 config 选中, 则在电源电压低于 1.85v 时, 系统会复位。

STKPB<2: 0>: 堆栈指针

入栈时, 将 PC 存入当前指针的堆栈寄存器, 然后 STKP-1(初始为全 1); 出栈时 STKP+1, 然后将堆栈寄存器的内容写入 PC。

执行程序调用指令 CALL 和响应中断服务时，堆栈指针 STKP 的值减 1，指针指向下一个堆栈缓存器。同时，对程序计数器 PC 的内容进行入堆栈保存。

堆栈层数	STKP 寄存器			堆栈缓存器		注释
	STKPB2	STKPB1	STKPB0	高字节	低字节	
0	1	1	1	Free	Free	——
1	1	1	0	STK0H	STK0L	——
2	1	0	1	STK1H	STK1L	——
3	1	0	0	STK2H	STK2L	——
4	0	1	1	STK3H	STK3L	——
5	0	1	0	STK4H	STK4L	——
6	0	0	1	STK5H	STK5L	——
7	0	0	0	STK6H	STK6L	——
8	1	1	1	STK7H	STK7L	——
>8	1	1	0	——	——	堆栈溢出，出错

对应每个入栈操作，都有一个出栈操作来恢复程序计数器 PC 的值。RETI 指令用于中断服务程序中，RET 用于子程序调用。出栈时，STKP 加 1 并指向下一个空闲堆栈缓冲器。堆栈恢复操作如下表所示：

堆栈层数	STKP 寄存器			堆栈缓存器		注释
	STKPB2	STKPB1	STKPB0	高字节	低字节	
8	1	1	1	STK7H	STK7L	——
7	0	0	0	STK6H	STK6L	——
6	0	0	1	STK5H	STK5L	——
5	0	1	0	STK4H	STK4L	——
4	0	1	1	STK3H	STK3L	——
3	1	0	0	STK2H	STK2L	——
2	1	0	1	STK1H	STK1L	——
1	1	1	0	STK0H	STK0L	——
0	1	1	1	Free	Free	——

13.6 MTP

MTP 控制寄存器包括 MTP 状态寄存器，地址寄存器(高位、低位)，数据寄存器，在对 MTP 进行读写操作时，首先将地址寄存器，数据寄存器，状态寄存器设置好。

13.6.1 MTPEN (MTP 控制寄存器)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0D8H(r/w)	MTPEN	-	-	MTP_CPEN	-	MTP_CERS	MTP_ERA	MTP_PRO	MTP_RD

MTP_CPEN: MTP 烧写擦除使能位

=0: 禁止

=1: 使能

MTP_CERS: MTP 擦除，整体擦除

=0: 禁止

=1: 使能，当擦除标志有效时硬件清零该使能

MTP_ERA: MTP 页擦除，一页 32BYTE

=0: 禁止

=1: 使能，当页擦除标志有效时硬件清零该使能

MTP_PRO: MTP 写

=0: 禁止

=1: 使能，当烧写标志有效时硬件清零该使能

MTP_RD: MTP 读

=0: 禁止

=1: 使能

13.6.2 MTPQ (MTP 标志位)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0D6H(r/w)	MTPQ	-	-	-	-	MTP_CERS_FLAG	MTP_ERA_FLAG	MTP_PRO_FLAG	-

MTP_CERS_FLAG: 整体擦除标志位

=0: 标志无效

=1: 擦除完成标志位，擦除完成自动置 1，需要软件清 0.

MTP_ERA_FLAG: MTP 页除标志位

=0: 标志无效

=1: 擦除完成标志位，页擦除完成自动置 1，需要软件清 0.

MTP_PRO_FLAG: MTP 写标志位

=0: 标志无效

=1: 写完成标志位, 写完成自动置 1, 需要软件清 0.

13.6.3 MTPC (MTP 擦除烧写频率控制)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0DBH(r/w)	MTPC	-	Pro_clks1	Pro_clks0	Pro_sel4	Pro_sel3	Pro_sel2	Pro_sel1	Pro_sel0

Pro_clks<1:0>: CPEN 到 PROG 有效的延时选择

=00: 64u

=01: 128u

=1x: 256u

Pro_sel4:

=0: MTP 烧写时不选择 4BIT

=1: MTP 烧写时选择一次烧写 4BIT

Pro_sel3:

=0: MTP 烧写时不选择 1BIT

=1: MTP 烧写时选择一次烧写 1BIT

Pro_sel<2:0>:

选择一位烧写时的控制地址。

注: 当 Pro_sel4 及 Pro_sel3 都无效时默认为一次烧写 8bit, 当 Pro_sel4 及 Pro_sel3 都有效时执行 1bit 的烧写。1bit 的烧写由 Pro_sel2~0 控制具体那一位, 4bit 的烧写由 Pro_sel2 控制选择高低四位。

13.6.4 MTPAH/L (MTP 地址低高 2 位/低 8 位)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0DAH(r/w)	MTPAH	-	-	-	-	-	-	MTP 地址高 2 位	
0DCH(r/w)	MTPAL	MTP 地址低 8 位							

13.6.5 MTPD (MTP8 位数据)

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0DEH(r/w)	MTPD	MTP8 位数据							

14 IO 端口

14.1 端口模式控制寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0A0H(r/w)	P0M0	-	-	P05M0	P04M0	P03M0	P02M0	P01M0	P00M0
0A1H(r/w)	P1M0	-	-	P15M0	P14M0	P13M0	P12M0	P11M0	P10M0
0A4H(r/w)	P5M0	-	-	-	P54M0	-	-	-	P50M0

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0AAH(r/w)	P0M1	-	-	P05M1	P04M1	P03M1	P02M1	P01M1	P00M1
0ABH(r/w)	P1M1	-	-	P15M1	P14M1	P13M1	P12M1	P11M1	P10M1
0AEH(r/w)	P5M1	-	-	-	P54M1	-	-	-	P50M1

PnM1: PnM0, Pn 模式控制位 (n=0、1、5)

= 11: Pn 为输出

= 10: Pn 为按键扫描脚(弱 1 强 0)

= 01: Pn 为输入带上拉

= 00: Pn 为纯输入

14. 2端口数据寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
0A5H(r/w)	P0	-	-	P05	P04	P03	P02	P01	P00
0A6H(r/w)	P1	-	-	P15	P14	P13	P12	P11	P10
0A9H(r/w)	P5	-	-	-	P54	-	-	-	P50

14. 3端口唤醒控制寄存器

地址	名称	B7	B6	B5	B4	B3	B2	B1	B0
9BH(r/w)	P0W	-	-	P05W	P04W	P03W	P02W	P01W	P00W
9CH(r/w)	P1W	-	-	P15W	P14W	P13W	P12W	P11W	P10W

PnW[7:0]: Pn 唤醒控制位 (n=0、1)

0 = 唤醒无效;

1 = 唤醒有效。

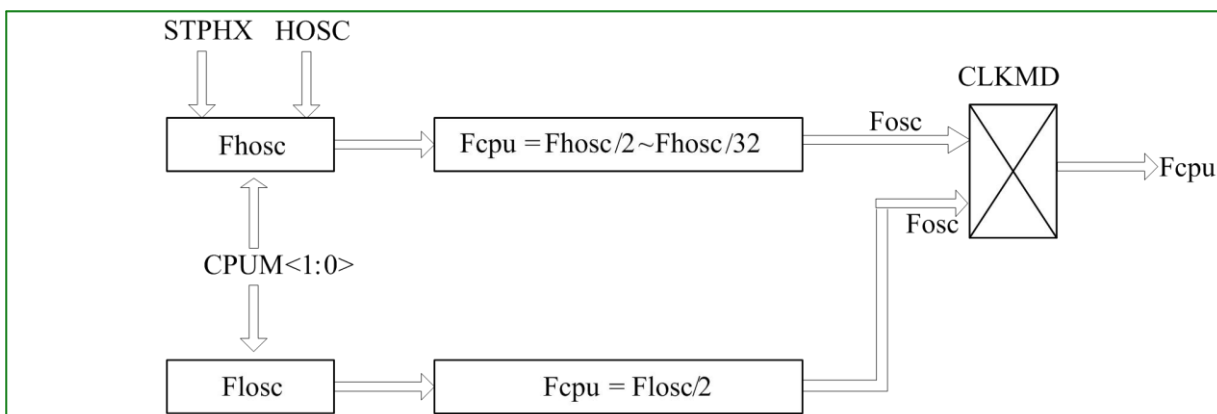
15 系统时钟

15.1 概述

本芯片内置双时钟系统：高速时钟和低速时钟。低速时钟由内部低速振荡器提供，由OSCM寄存器的CLKMD位控制，高、低速时钟都可以作为系统时钟源。 内

部高速振荡器：8MHz，称为IHRC；

内部低速振荡器：32KHz，称为ILRC。



HOSC: High_Clk 编译选项。

Fhosc: 外部高速/内部 RC 振荡器时钟频率。

Fosc: 内部低速 RC 时钟频率。

Fosc: 系统时钟频率。

Fcpu: 指令执行频率。

15.2 指令周期 Fcpu

系统时钟速率，即指令周期（Fcpu），从系统时钟源分离出来，决定系统的工作速率。Fcpu的速率由Fcpu编译选项决定，正常模式下， $F_{cpu} = F_{hosc}/2 \sim F_{hosc}/32$ 。若高速时钟源为内部8MHz振荡器，则Fcpu编译选项选择 $F_{hosc}/4$ ，则Fcpu频率为 $8\text{MHz}/4 = 2\text{MHz}$ 。低速模式下， $F_{cpu} = F_{osc}/2$ ，即 $32\text{KHz}/2 = 16\text{KHz}$ 。

高干扰环境下，强烈建议设置 code option 的 Fcpu 选项在 $F_{hosc}/4$ 以下，以减少干扰的影响。

16 系统工作模式

16.1 概述

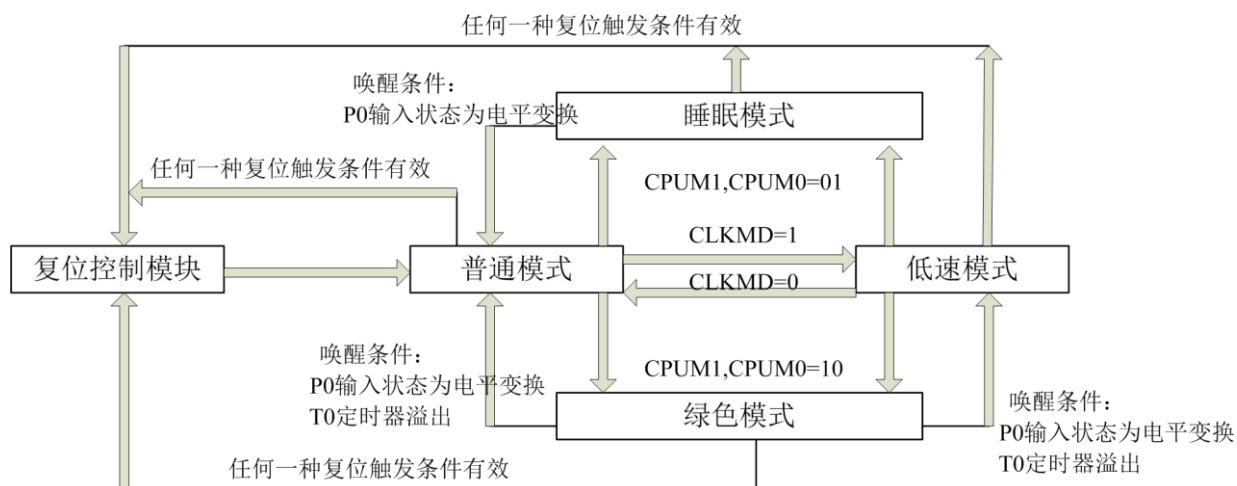
本芯片可以在4种工作模式下以不同的时钟频率工作，这些模式可以控制振荡器的工作、程序的执行以及模拟电路的功耗。

普通模式：系统高速工作模式；

低速模式：系统低速工作模式；

省电模式：系统省电模式（睡眠模式）；

绿色模式：系统理想模式。



工作模式	普通模式	低速模式	绿色模式	睡眠模式
IHRC	运行	STPHX 控制	STPHX 控制	停止
ILRC	运行	运行	运行	运行
CPU 指令	执行	执行	停止	停止
TC0	TC0TR 控制	TC0TR 控制	TC0TR 控制	停止
T1	T1TR 控制	T1TR 控制	T1TR 控制	停止
TM1	TM1EN 控制	TM1EN 控制	TM1EN 控制	停止
看门狗定时器	WDT 控制	WDT 控制	WDT 控制	WDT 控制
中断	全部有效	全部有效	全部有效	定时器 TC0、T1 无效， 其余有效
唤醒功能	——	——	端口变化、按键扫描、TC0 溢出唤醒、T1 溢出唤醒、 TM1 溢出唤醒、外中断、 WDT 溢出复位唤醒	端口变化、按键扫描、 外中断、WDT 溢出唤醒

16.2 普通模式

普通模式是系统高速时钟正常工作模式，系统时钟源由高速振荡器提供，程序执行。上电复位或任意一种复位触发后，系统进入普通模式执行程序。当系统从睡眠模式被唤醒后进入普通模式。普通模式下，高速振荡器正常工作，功耗最大。

程序被执行，所有的功能都可控制。

系统速率为高速。

高速振荡器和内部低速RC振荡器都正常工作。

通过OSCM寄存器，系统可以从普通模式切换到其它任何一种工作模式。

系统从睡眠模式唤醒后进入普通模式。

低速模式可以切换到普通模式。

从普通模式切换到绿色模式，唤醒后返回到普通模式。

普通模式工作时，可以实现学习模式。

学习的具体步骤：

1. 上电复位完成，开启学习模式，P5.4设为输入学习模式模式，TC0设为载波输入计数模式
2. 检测到输入载波下降沿的到来，硬件开启T0定时，同时T0定时的当前值存入缓存寄存器1
3. 检测到输入载波上升沿时，TC0开始+1(TC0上升沿加)，同时将T0定时的当前值存入缓存寄存器2，带软件读取。
4. 为了获得载波占空比，需要程序能在开始计数的几个周期，将T0的计时结果读取，便于计算占空比。
5. TC0计数载波过程中，软件要在一段时间内检测有无载波，如果TC0不增减(30us-50us),则将缓存寄存器的定时结果写入RAM,同时将TC0计数的载波个数写入RAM.
6. 一直学习至RAM溢出，停止学习
7. 将RAM中存取的数据读出，判断码型，通常一个码型占4个RAM,存取两帧
8. 将RAM转换后码型信息写入MTP.

16.3 低速模式

低速模式为系统低速时钟正常工作模式。系统时钟源由内部低速RC振荡器提供。低速模式由OSCM寄存器的CLKMD位控制。当CLKMD=0时，系统为普通模式；当CLKMD=1时，系统进入低速模式。切换进入低速模式后，不能自动禁止高速振荡器，必须通过SPTHX位来禁止以减少功耗。低速模式下，系统速率被固定为Fosc/4（Fosc为内部低速RC振荡器频率）。

程序被执行，所有的功能都可控制。

系统速率为低速，可选分频系数。

内部低速RC振荡器正常工作，高速振荡器由STPHX=1控制。低速模式下，强烈建议停止高速振荡器，但必须先进入低速模式后，再STPHX=1。

通过OSCM寄存器，低速模式可以切换进入其它的工作模式。

从低速模式切换到睡眠模式，唤醒后返回到低速模式或者正常模式。普通模式可以切换进入低速模式。

从低速模式切换到绿色模式，唤醒后返回到低速模式。

16.4 睡眠模式

睡眠模式是系统的理想状态，不执行程序，振荡器也停止工作。整个芯片的功耗低于1uA（ADC 关闭）。睡眠模式的唤醒源为P0/P1/P54/P50电平变换触发，外中断、按键扫描、WDT溢出复位。由 OSCM寄存器的CPUM0位控制是否进入睡眠模式，当CPUM0=1，系统进入睡眠模式。当系统从睡眠模式被唤醒后，CPUM0被自动禁止（0状态）。

程序停止执行。

外部高速振荡器、内部高速振荡器都停止工作。

功耗低于1uA。

如果需要，内部低速振荡器可工作

系统从睡眠模式被唤醒后进入普通模式。

16.5 绿色模式

绿色模式是另外一种理想状态。在睡眠模式下，所有的功能和硬件设备都被禁止，但在绿色模式下，系统时钟保持工作，绿色模式下的功耗大于睡眠模式下的功耗。绿色模式下，不执行程序，但具有唤醒功能的定时器仍正常工作，定时器的时钟源为仍在工作的系统时钟。绿色模式的唤醒源和睡眠模式一样。

由OSCM寄存器CPUM1位决定是否进入绿色模式，当CPUM1=1，系统进入绿色模式。当系统从绿色模式下被唤醒后，自动禁止CPUM1（0状态）。

程序停止执行，所有的功能被禁止。

具有唤醒功能的定时器正常工作。

作为系统时钟源的振荡器正常工作，其它的振荡器工作状态取决于系统工作模式的配置。由普通模式切换到绿色模式，被唤醒后返回到普通模式。

由低速模式切换到绿色模式，被唤醒后返回到低速模式。

绿色模式下的唤醒方式比睡眠模式多了TC0、T1、TM1溢出唤醒。

绿色模式下PWM和Buzzer功能仍然有效，但是定时器溢出时不能唤醒系统。

16.6 系统唤醒

16.6.1 概述

睡眠模式和绿色模式下，系统并不执行程序。唤醒触发信号可以将系统唤醒进入普通模式或低速模式。

从睡眠模式唤醒后返回到普通工作模式，唤醒触发信号包括：

- 1、端口电平变化（P0/P1/P50/P54的电平变换，P0和P1口对应的PnW需要置1）
- 2、INT中断唤醒
- 3、按键扫描唤醒
- 4、WDT溢出唤醒（不复位）

从绿色模式唤醒后返回到上一个工作模式（普通模式或低速模式），唤醒触发信号包括：

- 1、端口电平变化（P0/P1/P50/P54的电平变换，P0和P1口对应的PnW需要置1）

- 2、TC0定时器溢出
- 3、T1定时器溢出
- 4、INT中断唤醒
- 5、按键扫描唤醒
- 6、WDT溢出复位唤醒
- 7、TM1溢出唤醒

16.6.2 唤醒时间

系统进入睡眠模式后，高速时钟振荡器停止运行。把系统从睡眠模式唤醒时，单片机需要等待一段时间以等待振荡电路稳定工作，等待的这段时间就称为唤醒时间。唤醒时间结束后，系统进入普通模式。

$$\text{唤醒时间} = 1/F_{\text{osc}} * 8$$

17 复位

17.1 概述

本芯片有以下几种复位方式：

上电复位；

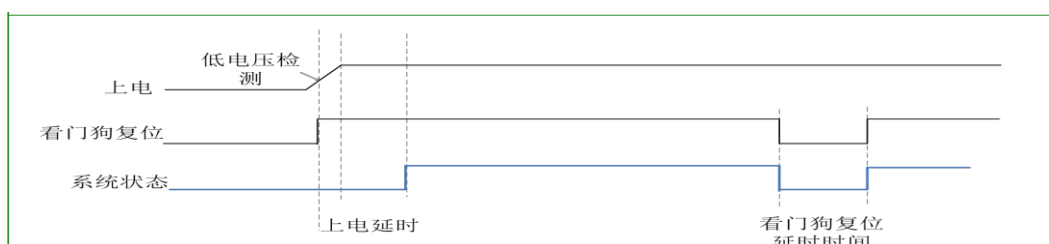
看门狗复位；

掉电复位；

上述任一种复位发生时，所有的系统寄存器恢复默认状态，程序停止运行，同时程序计数器PC清零。复位结束后，系统从向量0000H处重新开始运行。PFLAG寄存器的/T0和/PD两个标志位能够给出系统复位状态的信息。用户可以编程控制/T0和/PD，从而控制系统的运行路径。

/T0	/PD	复位情况	说明
0	0	看门狗复位	看门狗溢出
0	1	保留	——
1	0	上电及 LVD 复位	电源电压低于 LVD 检测值

任何一种复位情况都需要一定的响应时间，系统提供完善的复位流程以保证复位动作的顺利进行。对于不同类型的振荡器，完成复位所需要的时间也不同。因为，VDD 的上升速度和不同晶振的起振时间都不固定。RC 振荡器的起振时间最短，晶体振荡器的起振时间则较长。在用户终端使用的过程中，应考虑主机对上电复位时间的要求。



17.2 上电复位

上电复位与LVD操作密切相关。系统上电的过程呈逐渐上升的曲线形式，需要一定时间才能达到正常电平值。下面给出上电复位的正常时序：

上电：系统检测到电源电压上升并等待其稳定；

系统初始化：所有的系统寄存器被置为初始值；

振荡器开始工作：振荡器开始提供系统时钟；

执行程序：上电结束，程序开始运行。

17.3 看门狗复位

看门狗复位是系统的一种保护设置。在正常状态下，由程序将看门狗定时器清零。若出错，系统处于未知状态，看门狗定时器溢出，此时系统复位。看门狗复位后，系统重启进入正常状态。看门狗复位的时序如下：

看门狗定时器状态：系统检测看门狗定时器是否溢出，若溢出，则系统复位；

系统初始化：所有的系统寄存器被置为默认状态；

振荡器开始工作：振荡器开始提供系统时钟；

执行程序：上电结束，程序开始运行。

看门狗定时器应用注意事项：

对看门狗清零之前，检查I/O口的状态和RAM的内容可增强程序的可靠性；

不能在中断中对看门狗清零，否则无法检测到主程序跑飞的状况；

程序中应该只在主程序中有一次清看门狗的动作，这种架构能够最大限度的发挥看门狗的保护功能。

注：睡眠模式下，WDT溢出唤醒而不复位，其余模式下WDT溢出均复位。

17.4 寄存器复位状态表

17.4.1 上电复位状态表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	1011 100x	1111 1100	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
9	xxxx	xxxx	0000 0000	0000 0000	0000 0000	1110 0001	Write only	1000 0000		1000 0000	0000 0000	0000 0000	0000 0000			1001 0110
A	0000 0000	0000 0000			0000 0000	00xx xxxx	11xx xxxx			000x 000x	0000 0000	0000 0000			0000 0000	
B						0000 0000	0000 0000	0000 0000	0000 0000	0000 0000	0000 0000	0000 0000				
C	0000 0000	0000 0000	0000 0000	0000 0000	0000 0000	0000 0000	0000 0000	1111 0000	0000 0000	0000 0000	0000 0000		0000 0000			
D	0000 0000	0000 0000					0000 0000		0000 0000		0000 0000	0000 0000	0000 0000		0000 0000	

E										0000	0000	0000	0000	0000		0111
										0000	0000	0000	0000	0000		1111
F	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx
	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

17.4.2 WDT 复位状态表

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	uuuu	uuuu	uuuu	uuuu	uuuu	uuuu	0011	1111								
							1000	1100								
9	uuuu	uuuu	0000	0000	0000	1110	Write	1000		1000	0000	0000	0000			1001
			0000	0000	0000	0001	only	0000		0000	0000	0000	0000			0110
A	0000	0000			0000	00xx	11xx			000x	0000	0000			0000	
	0000	0000			0000	xxxx	xxxx			000x	0000	0000			0000	
B						0000	0000	0000	0000	0000	0000	0000				
						0000	0000	0000	0000	0000	0000	0000				
C	0000	0000	0000	0000	0000	0000	0000	1111	0000	0000	0000		0000			
	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000		0000			
D	0000	0000					0000		0000		0000	0000	0000		0000	
	0000	0000					0000		0000		0000	0000	0000		0000	
E										0000	0000	0000	0000	0000		0111
										0000	0000	0000	0000	0000		1111
F	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx	0000	xxxx
	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000

说明：u = 未改变；x = 未知；- = 未使用

18 指令集

序号	助记符号	描述	C	DC	Z	CYC
1	MOV A, M	$A \leftarrow M$	-	-	√	1
2	MOV M, A	$M \leftarrow A$	-	-	-	1
3	B0MOV A, M	$A \leftarrow M$ (bank0)	-	-	√	1
4	B0MOV M, A	M (bank0) $\leftarrow A$	-	-	-	1
5	MOV A, I	$A \leftarrow I$	-	-	-	1
6	B0MOV M, I	$M \leftarrow I$, (M= R,X,Y,Z,H,L,RBANK,PFLAG)	-	-	-	1
7	XCH A, M	$A \leftrightarrow M$	-	-	-	1+N
8	B0XCH A, M	$A \leftrightarrow M$ (bank0), M 不能是 80--87H 的系统寄存器	-	-	-	1+N
9	MOVC	$R, A \leftarrow ROM[Y, Z]$	-	-	-	2

10	ADC A, M	$A \leftarrow M + A + C$, 如果发生进位, 则 C=1, 否则 C=0	√	√	√	1
11	ADC M, A	$M \leftarrow M + A + C$, 如果发生进位, 则 C=1, 否则 C=0	√	√	√	1+N
12	ADD A, M	$A \leftarrow M + A$ 如果发生进位, 则 C=1, 否则 C=0	√	√	√	1
13	ADD M, A	$M \leftarrow M + A$ 如果发生进位, 则 C=1, 否则 C=0	√	√	√	1+N
14	B0ADD M, A	M (bank0) $\leftarrow M$ (bank0) + A if 进位 C=1, else C=0	√	√	√	1+N
15	ADD A, I	$A \leftarrow A + I$, 如果发生进位, 则 C=1, 否则 C=0	√	√	√	1
16	SBC A, M	$A \leftarrow A - M / C$, 如果发生进位, 则 C=0, 否则 C=1	√	√	√	1
17	SBC M, A	$M \leftarrow A - M / C$, 如果发生进位, 则 C=0, 否则 C=1	√	√	√	1+N
18	SUB A, M	$A \leftarrow A - M$, 如果发生进位, 则 C=0, 否则 C=1	√	√	√	1
19	SUB M, A	$M \leftarrow A - M$, 如果发生进位, 则 C=0, 否则 C=1	√	√	√	1+N
20	SUB A, I	$A \leftarrow A - I$, 如果发生进位, 则 C=0, 否则 C=1	√	√	√	1
21	DAA	调整 ACC 十六进制为十进制	√	-	-	1
22	MUL A, M	$R, A \leftarrow A * M$ 计算结果低位存入 ACC, 高位存入 R 寄存器	-	-	√	2
23	AND A, M	$A \leftarrow A \& M$	-	-	√	1
24	AND M, A	$M \leftarrow A \& M$	-	-	√	1+N
25	AND A, I	$A \leftarrow A \& I$	-	-	√	1
26	OR A, M	$A \leftarrow A M$	-	-	√	1
27	OR M, A	$M \leftarrow A M$	-	-	√	1+N
28	OR A, I	$A \leftarrow A I$	-	-	√	1
29	XOR A, M	$A \leftarrow A \oplus M$	-	-	√	1
30	XOR M, A	$M \leftarrow A \oplus M$	-	-	√	1+N
31	XOR A, I	$A \leftarrow A \oplus I$	-	-	√	1
32	COM M	$A \leftarrow M$ 取反	-	-	√	1
33	COMM M	$M \leftarrow M$ 取反	-	-	√	1

34	CLR	M	$M \leftarrow 0$	-	-	-	1
35	SWAP	M	$A(b3 \sim b0, b7 \sim b4) \leftarrow M(b7 \sim b4, b3 \sim b0)$	-	-	-	1
36	SWAPM	M	$M(b3 \sim b0, b7 \sim b4) \leftarrow M(b7 \sim b4, b3 \sim b0)$	-	-	-	1+N
37	RRC	M	$A \leftarrow RRC\ M$	√	-	-	1
38	RRCM	M	$M \leftarrow RRC\ M$	√	-	-	1+N
39	RLC	M	$A \leftarrow RLC\ M$	√	-	-	1
40	RLCM	M	$M \leftarrow RLC\ M$	√	-	-	1+N
41	BCLR	M.b	$M.b \leftarrow 0$	-	-	-	1+N
42	BSET	M.b	$M.b \leftarrow 1$	-	-	-	1+N
43	B0BCLR	M.b	$M(\text{bank}0).b \leftarrow 0$	-	-	-	1+N
44	B0BSET	M.b	$M(\text{bank}0).b \leftarrow 1$	-	-	-	1+N
45	CMPRS	A, I	$ZF, C \leftarrow A-I$, if $A=I$, 则跳过下一条程序	√	-	√	1+S
46	CMPRS	A, M	$ZF, C \leftarrow A-M$, if $A=M$, 则跳过下一条程序	√	-	√	1+S
47	INCS	M	$A \leftarrow M+1$, if $A=0$, 则跳过下一条程序	-	-	-	1+S
48	INCMS	M	$M \leftarrow M+1$, if $M=0$, 则跳过下一条程序	-	-	-	1+S+N
49	INC	M	$A \leftarrow M+1$	-	-	√	1
50	INCM	M	$M \leftarrow M+1$	-	-	√	1+N
51	DECS	M	$A \leftarrow M-1$, if $A=0$, 则跳过下一条程序	-	-	-	1+S
52	DECMS	M	$M \leftarrow M-1$, if $M=0$, 则跳过下一条程序	-	-	-	1+S+N
53	DEC	M	$A \leftarrow M-1$	-	-	√	1
54	BTS1	M.b	if $M.b=1$, 则跳过下一条程序	-	-	-	1+S
55	B0BTS0	M.b	If $M(\text{bank}0).b=0$, 则跳过下一条程序	-	-	-	1+S
56	B0BTS1	M.b	If $M(\text{bank}0).b=1$, 则跳过下一条程序	-	-	-	1+S
57	TS0M	M	If $M=0, Z=1$ 否则 $Z=0$	-	-	√	1
58	JMP	d	$PC\ 15/14 \leftarrow ROMPAGES1/0, PC13 \sim PC0 \leftarrow d$	-	-	-	2
59	CALL	d	$Stack \leftarrow PC15 \sim PC0$ $15/14 \leftarrow RomPAGES1/0, PC13 \sim PC0 \leftarrow d$	-	-	-	2
60	CALLHL		$Stack \leftarrow PC15 \sim PC0, PC\ 15 \sim 8 \leftarrow H, PC7 \sim PC0 \leftarrow L$	-	-	-	2
61	CALLYZ		$Stack \leftarrow PC15 \sim PC0, PC\ 15 \sim 8 \leftarrow Y, PC7 \sim PC0 \leftarrow Z$	-	-	-	2
62	RET		$PC \leftarrow Stack$	-	-	-	2
63	RETI		$PC \leftarrow Stack$, 退出中断, 并打开全局中断	-	-	-	2
64	RETLW I		$PC \leftarrow Stack$, 写 I 到 ACC	-	-	-	2
65	NOP		空操作	-	-	-	1

注:

- 1、在条件跳转指令的条件为真, 则 $S=1$, 否则 $S=0$ 。
- 2、在“B0MOV M, I”中, M 的范围是 $0x80 \sim 0x87$
- 3、执行 DAA 指令的前提是操作数的高 4 位和低 4 位都必须小于 A。即它们都是 BCD 码。否则, 结果会出错。

DAA 应用实例: $A=1001\ 1001=0x99$,

[0x20H]=1001 1001=0x99 ;

ADDAR [0x20H],0 ; 结果=0x132 ,C=1,DC=1

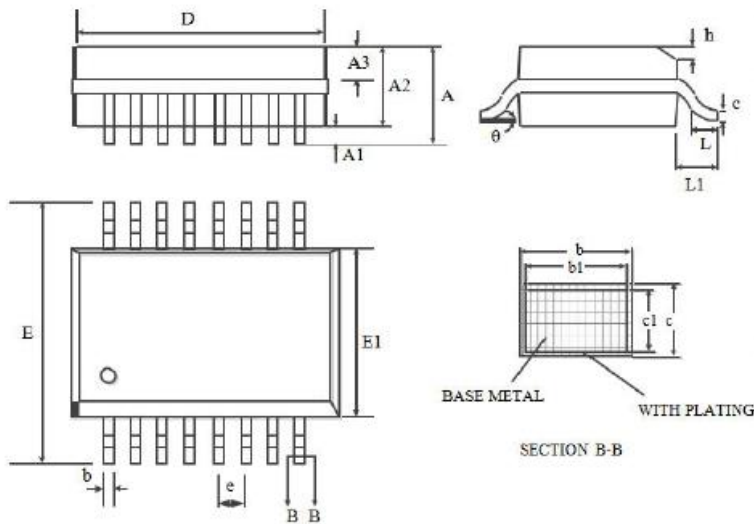
DAA ; 结果=0x198, 98 正是需要的BCD 码值。

操作原理:

当加的结果的低 4bit 大于 9 或者有进位 DC=1, 则低 4 位+6, 高四位也是大于 9 或者 C=1, 则高 4 位+6。即根据结果判断是加上 06h, 60h 还是 66h。

19 封装形式

SOP16



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	--	--	1.75
A1	0.10	--	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	--	0.48
b1	0.38	0.41	0.43
c	0.21	--	0.26
c1	0.19	0.20	0.21
D	9.70	9.90	10.10
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27BSC		
h	0.25	--	0.50
L	0.50	--	0.80
L1	1.05BSC		
θ	0	--	8°