

产品目录

OB5510CGP

OB5510CIP

描述

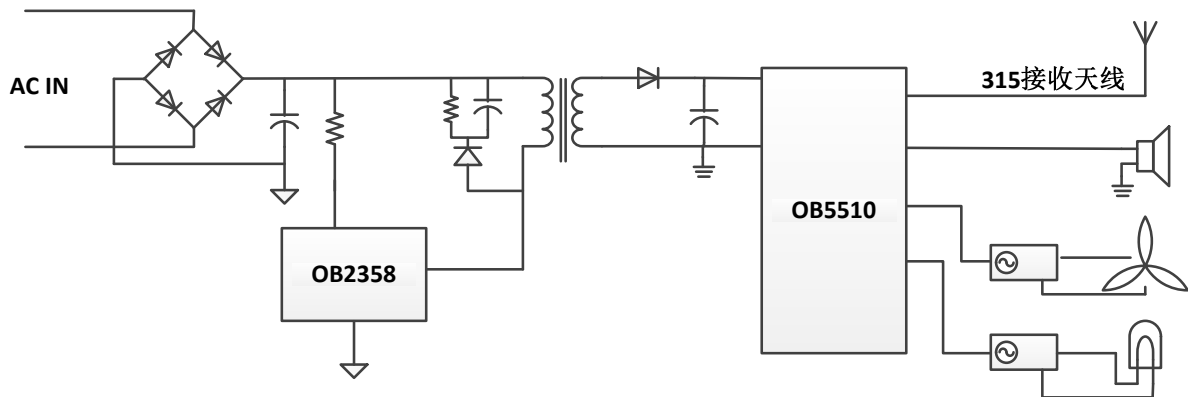
OB5510 是一集成高压工艺与数模混合技术的主控芯片，具备快速的单周期 8051 内核，8KB 闪存，512B SRAM，能执行所有 ASM51 兼容指令，与 MCS-51 相同的指令设置。片内集成有 DC Regulator, ADC, MOSFET 等模拟模块，可直接驱动 5 个继电器或可控硅，芯片能操作在系统电源 12V 以下。

特征

- 指令设置兼容 MCS-51
- 8K 字节的片上闪存程序存储器
- 512 字节的标准的 8052 RAM
- 双 16-bit 数据指针 (DPTR0&DPTR1)
- 3 个 16-bit 的定时器/计数器 (计时器 0,1,2, 3)

- 最多 11 个 GPIO
- 外部中断源: INTx 2
- I2C x 1
- UART x 1
- 可编程的看门狗定时器 (WDT)
- 3 路 12 位模拟数字转换 (ADC)
- 片上闪存存储器支持 ICP 及 EEPROM 功能
- 低电压中断 LVI/低电压复位 LVR
- 电源管理单元空闲及掉电模式
- 支持在线仿真(OCD)功能
- 高速 1T 架构，最高可达 16MHz
- 高压模拟单元
 - 输入电压: 7.5V~12V
 - 集成 5VLDO, 输出电压: 5V $\pm 2\%$ @25° C. $\pm 4\%$ @ -40° C~85° C; 最大输出电流: 200mA
 - 集成 5 个 开漏端口, 80mA 灌电流能力, 内建续流二极管
- 封装型态: 16/20-SOP

系统应用图



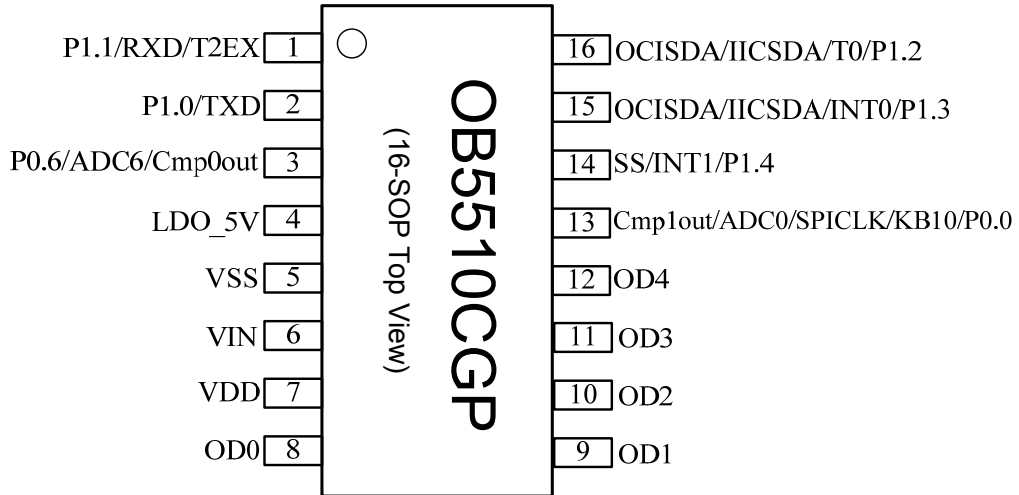
目录

产品目录	1
描述	1
特征	1
系统应用图	1
各封装引脚配置	4
订购信息	5
应用极限参数	5
标识信息	6
管脚描述	6
IC 内部框图	7
高压直流部分电气特性	9
高压模拟模块特殊功能寄存器设定	10
特殊功能寄存器(SFR)	12
功能描述	16
1. 总特征	16
1.1 嵌入式程序存储器	16
1.2 IO 口	16
1.3 指令时钟周期选择	16
1.4 时钟输出选择	16
1.5 复位	18
1.5.1 硬件复位功能	18
1.5.2 软件复位功能	18
1.5.3 Reset status	19
1.5.4 Time Access Key register (TAKEY)	19
1.5.5 软件复位寄存器(SWRES)	19
1.5.6 Example of software reset	20
1.6 系统时钟源	20
2. 指令设置	21
3. 存储器结构	25
3.1 程序存储器	25
3.2 数据存储器	26
3.3 数据内存-低 128 字节(00h 到 7Fh)	26
3.4 数据存储器-高 128 字节(80h 到 FFh)	26
3.5 存储器-扩展的 256 字节(\$00 到 \$FF)	26
4. CPU 结构	27
4.1 累加器	27
4.2 B 寄存器	27
4.3 程序状态字	28
4.4 堆栈指针	28
4.5 数据指针	29
4.6 数据指针 1	29
4.7 时钟控制寄存器	30
4.8 接口控制寄存器	30
5. GPIO 管脚型态	31
6. 定时器 0、定时器 1	32
6.1 定时器/计数器模式控制寄存器(TMOD)	32
6.2 定时器/计数器控制寄存器(TCON)	33

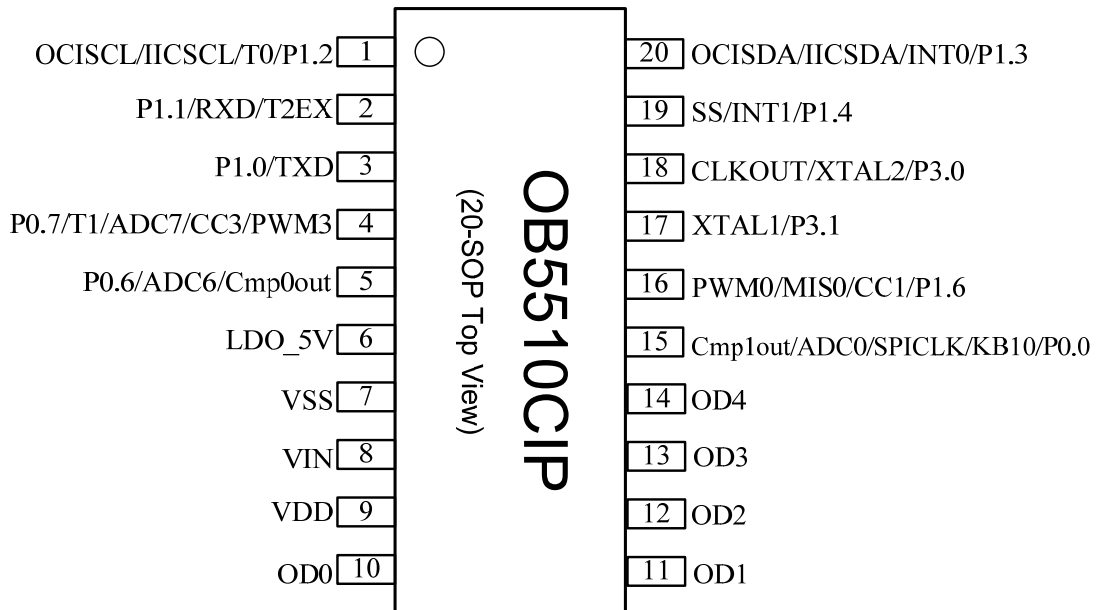
6.3	定时器输入频率控制寄存器	33
6.4	模式 0 (13 位定时/计数).....	34
6.5	模式 1 (16 位定时/计数).....	34
6.6	模式 2 (8 位自动重载定时/计数)	36
6.7	模式 3 (两个独立 8 位定时/计数(仅定时器 0)).....	36
7.	定时器 2 以及捕捉/比较单元(CCU).....	37
7.1	定时器 2 功能	40
7.1.1	定时器模式	40
7.1.2	外部信号计数模式	40
7.1.3	外部信号定时器模式	41
7.1.4	定时器 2 的重载.....	41
7.2	比较功能.....	41
7.2.1	比较模式 0.....	41
7.2.2	比较模式 1.....	42
7.3	捕获功能.....	43
7.3.1	捕捉模式 0.....	43
7.3.2	捕捉模式 1.....	43
8.	串行接口.....	44
8.1	串行接口由以下 4 种模式可以设置	45
8.1.1	模式 0	45
8.1.2	模式 1.....	46
8.1.3	模式 2.....	46
8.1.4	模式 3.....	46
8.2	串行接口的多重机通讯	47
8.3	输入频率控制寄存器.....	47
8.4	波特率发生器.....	47
8.4.1	串行接口的模式 1 和 3	47
9.	看门狗定时器	49
10.	中断.....	53
10.1	优先权配置	57
11.	电源管理单元	59
11.1	待机模式(空闲模式).....	59
11.2	停止模式.....	59
12.	脉宽调制器(PWM)	60
13.	IIC 功能.....	63
14.	键盘接口(KBI).....	68
15.	LVI – 低压侦测中断.....	71
16.	12 位模拟数字转换器(ADC).....	73
17.	在系统编程(INTERNAL ISP)与 EEPROM	77
17.1	ISP 服务程序	77
17.2	锁定位(N).....	77
17.3	对 ISP 服务程序编程.....	78
17.4	启动 ISP 服务程序	78
17.5	ISP 寄存器– TAKEY, IFCON, ISPFAL, ISPFAL, ISPFAL, ISPFAL and ISPFAL	79
极限参数		84
电气特性		84
封装尺寸		86
SOP16		86

各封装引脚配置

16 Pin SOP



20 Pin SOP



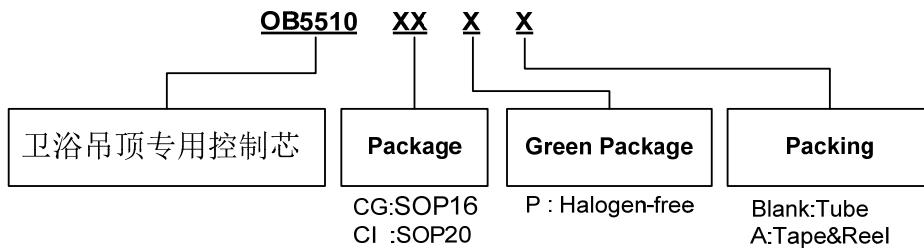
订购信息

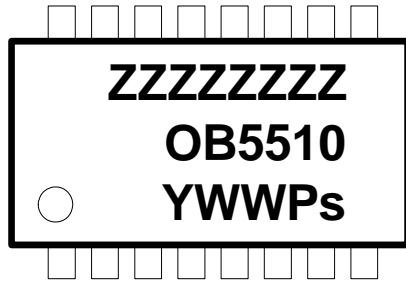
品名	描述
OB5510CGP	SOP16, Halogen-free in Tube
OB5510CGPA	SOP16, Halogen-free in T&R
OB5510CIP	SOP20, Halogen-free in Tube
OB5510CIPA	SOP20, Halogen-free in T&R

应用极限参数

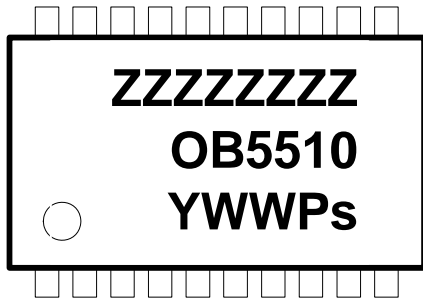
参数	数值
VDD Voltage	-0.3V to 20V
VIN Voltage	-0.3V to 20V
OD0, OD1, OD2, OD3, OD4	-0.3V to 40V
VDO_5V	-0.3V to 7V
All the other pin	-0.3V to 7V
Min/Max Operating Junction Temperature T _J	-40 to 150 °C
Operating Ambient Temperature T _A	-40 to 85 °C
Min/Max Storage Temperature T _{stg}	-55 to 150 °C
Lead Temperature (Soldering, 10secs)	260 °C

备注: 芯片的实际应用条件超出规定的“应用极限值”将会对芯片造成永久性损伤。以上应用极限值标志了芯片可承受应力等级, 但并不建议芯片在此极限条件或超出“推荐工作条件”下工作。芯片长时间处于最大额定工作条件, 会影响芯片的可靠性。



标识信息


ZZZZZZZZ:Lot Code
 Y:Year Code
 WW:Week Code(01-52)
 P:Halogen-free Package
 S:Internal Code



ZZZZZZZZ:Lot Code
 Y:Year Code
 WW:Week Code(01-52)
 P:Halogen-free Package
 S:Internal Code

管脚描述

SOP20	SOP16	Symbol	I/O	Description
1	16	OCISCL/IIC_SCL/T0/P1.2	I/O	OCISCL/IIC_SCL/T0/P1.2
2	1	P1.1/RXD/T2EX	I/O	P1.1/RXD/T2EX
3	2	P1.0/TXD	I/O	P1.0/TXD
4	/	P0.7/T1/ADC7/CC3/PWM3	I/O	P0.7/T1/ADC7/CC3/PWM3
5	3	P0.6/ADC6/Cmp0out	I/O	P0.6/ADC6/Cmp0out
6	4	LDO_5V	O	DC-5V output
7	5	VSS	I	Ground
8	6	VIN	I	Regulator input
9	7	VDD	I	7.5V~12V Power supply
10	8	OD0	O	Open drain output 0
11	9	OD1	O	Open drain output 1
12	10	OD2	O	Open drain output 2
13	11	OD3	O	Open drain output 3

14	12	OD4	O	Open drain output 4
15	13	Cmp1out/ADC0/SPICLK/KB10/P0.0	I/O	Cmp1out/ADC0/SPICLK/KB10/P0.0
16	/	PWM0/MIS0/CC1/P1.6	I/O	PWM0/MIS0/CC1/P1.6
17	/	XTAL1/P3.1	I/O	XTAL1/P3.1
18	/	CLKOUT/XTAL2/P3.0	I/O	CLKOUT/XTAL2/P3.0
19	14	SS/INT1/P1.4	I/O	SS/INT1/P1.4
20	15	OCISDA/IICSDA/INT0/P1.3	I/O	OCISDA/IICSDA/INT0/P1.3

IC 内部框图

OB5510为专门用于卫浴吊顶系统的控制处理器，它由模拟和数字两部分构成，模拟部分提供LDO输出供电模块，LDO供电模块有高达200mA的电流输出能力，并且拥有休眠模式，进入休眠模式后LDO消耗电流减小以适应低功耗要求。OB5510还有5路Open Drain驱动电路，大大简化了系统设计的难度。

寄存器	地址	重置值	描述
ADCC2	ACh	0Fh	SADC Control 2 Register
ADCDH	ADh	00h	SADC data high byte
ADC DL	AEh	00h	SADC data low byte
ADCCS	AFh	00h	SADC clock select
ADCSH	EFh	00h	SADC Sample and Hold Time
WDT			
RSTS	A1h	00h	Reset status register
WDTC	B6h	04h	Watchdog timer control register
WDTK	B7h	00h	Watchdog timer refresh key.
TAKEY	F7h	00h	Time Access Key register
PWM			
PWMC	B5h	00h	PWM control register
PWMD0H	BCh	00h	PWM channel 0 data high byte
PWMD0L	BDh	00h	PWM channel 0 data low byte
PWMD3H	B3h	00h	PWM channel 3 data high byte
PWMD3L	B4h	00h	PWM channel 3 data low byte
PWMMDH	CEh	00h	PWM Max Data Register, high byte.
PWMMDL	CFh	FFh	PWM Max Data Register, low byte.
TIMER0/TIMER1			
TCON	88h	00h	Timer/Counter Control
TMOD	89h	00h	Timer Mode Control
TL0	8Ah	00h	Timer 0, low byte
TL1	8Bh	00h	Timer 1, low byte
TH0	8Ch	00h	Timer 0, high byte
TH1	8Dh	00h	Timer 1, high byte
PFCON	D9h	00h	Peripheral Frequency control register
PCA(TIMER2)			
CCEN	C1h	00h	Compare/Capture Enable Register
CCL1	C2h	00h	Compare/Capture Register 1, low byte
CCH1	C3h	00h	Compare/Capture Register 1, high byte
CCL2	C4h	00h	Compare/Capture Register 2, low byte
CCH2	C5h	00h	Compare/Capture Register 2, high byte
CCL3	C6h	00h	Compare/Capture Register 3, low byte
CCH3	C7h	00h	Compare/Capture Register 3, high byte
T2CON	C8h	00h	Timer 2 Control
CCCON	C9h	00h	Compare/Capture Control
CRCL	CAh	00h	Compare/Reload/Capture Register, low byte
CRCH	CBh	00h	Compare/Reload/Capture Register, high byte
TL2	CCh	00h	Timer 2, low byte

寄存器	地址	重置值	描述
TH2	CDh	00h	Timer 2, high byte
CCEN2	D1h	00h	Compare/Capture Enable 2 register
GPIO			
P0	80h	FFh	Port 0
P1	90h	FFh	Port 1
P3	B0h	FFh	Port 3
P0M0	D2h	00h	Port 0 output mode 0
P0M1	D3h	00h	Port 0 output mode 1
P1M0	D4h	00h	Port 1 output mode 0
P1M1	D5h	00h	Port 1 output mode 1
P3M0	DAh	00h	Port 3 output mode 0
P3M1	DBh	00h	Port 3 output mode 1
ISP/IAP/EEPROM			
IFCON	8Fh	00h	Interface control register
ISPF AH	E1h	FFh	ISP Address-High register
ISPF AL	E2h	FFh	ISP Address-Low register
ISPF DL	E3h	FFh	ISP Data High register
ISPF DH	E3h	FFh	ISP Data Low register
ISPF C	E4h	00h	ISP control register
TAKEY	F7h	00h	Time Access Key register
LVI/LVR/SOFTRESET			
RSTS	A1h	00h	Reset status register
LVC	E6h	20h	Low voltage control register
SWRES	E7h	00h	Software Reset register
TAKEY	F7h	00h	Time Access Key register
IIC			
AUX	91h	00h	Auxiliary register
IICS	F8h	00h	IIC status register
IICCTL	F9h	04h	IIC control register
IICA1	FAh	A0h	IIC channel 1 Address 1 register
IICA2	FBh	60h	IIC channel 1 Address 2 register
IICRWD	FCh	00h	IIC channel 1 Read / Write Data buffer
IICEBT	FDh	00h	IIC Enable Bus Transaction register

功能描述

1. 总特征

OB5510内核是一个8位的微处理器,它的所有功能以及特殊功能寄存器（SFR）的详细定义将在以下章节给出.

1.1 嵌入式程序存储器

可通过编程器或在线编程(ICP)将程序加载到8KB的嵌入式闪存体中,其高品质的闪存体具有重复可擦写编程并记忆数据,如EEPROM。

1.2 IO口

OB5510有11个I/O, 四种型态: 准双向口(标准 8051输出口), 推挽, 开漏, 与仅输入。在章节5中有详细说明。

1.3 指令时钟周期选择

传统的52 系列单片机时钟周期是12T, 即12 个振荡器时钟为1个机器周期。OB5510为1T~8T的微控制器, 即机器周期为1个时钟周期~8个时钟周期。换句话说, 执行一条指令可是1个时钟~8个时钟。

符号: CKCON					地址: 8Eh		
7	6	5	4	3	2	1 0	
-	ITS[2:0]			-	-	CLKOUT[1:0]	Reset
							10H

ITS: 指令时钟周期选择.

ITS [2:0]	指令时钟周期
000	1T 模式
001	2T 模式(默认)
010	3T 模式
011	4T 模式
100	5T 模式
101	6T 模式
110	7T 模式
111	8T 模式

默认为2T模式, 在任何时候,如CKCON [6:4] (地址为8Eh) 被改变时, 每条指令并不是都能在一个机器周期内被执行的。所有指令的确切机器周期将在下一章节中给出。

1.4 时钟输出选择

在任何时候CKCON [1:0] (地址为8Eh) 被改变时, OB5510能在P3.0产生时钟输出, 主系统时钟源设置为使用振荡器(时钟由晶振输入脚输入)或片内RC振荡器皆可。

CLKOUT: 时钟输出除频选择.

CKCON [1:0]	Mode.
00	GPIO(默认)
01	主系统时钟频率
10	主系统时钟频率/2
11	主系统时钟频率/4

1.5 复位

1.5.1 硬件复位功能

OB5510提供了片上硬件复位机制, 片上硬件复位的时间长度可以通过编程器设置。

片上硬件复位的时间长度
200ms
100ms
50ms
25ms(default)
16ms
8ms
4ms

1.5.2 软件复位功能

OB5510提供一种软件复位机制来实现整个芯片的复位.要实现软件复位,于程序中需把3个特殊值 55h,AAh和5Ah按顺序写到TAKEY寄存器来使能软件复位寄存器(SWRES)之写入. 在软件复位寄存器获得可写权后,可以对SWRES寄存器写入FFh. 硬件将解码出复位讯号它是与其它硬件复位讯号作“OR”处理. 软件复位寄存器在软件复位过程的最后会进行自复位.

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
Software Reset function											
RSTS	Reset status register	A1h	-	LVRLP INTF	LVRL PF	PDRF	WDTF	SWRF	LVRF	PORF	00H
TAKEY	Time Access Key register	F7h	TAKEY [7:0]								00H
SWRES	Software Reset register	E7h	SWRES [7:0]								00H

1.5.3 Reset status

符号: RSTS								地址: A1h
7	6	5	4	3	2	1	0	Reset
-	LVRLP INTF	LVRL PF	PDRF	WDTF	SWRF	LVRF	PORF	00H

LVRLPINTF: “内部” 低电压复位旗标.

当 MCU 复位信号由“内部”低电压复位产生时, LVRLPINTF 旗标将由硬件置高.此旗标需由软件清零.

LVRLPF: 低电压复位旗标.(低功率)

当 MCU 复位信号由低电压复位(低功率)产生时, LVRLPF 旗标将由硬件置高.此旗标需由软件清零.

PDRF: 复位脚复位旗标.

当 MCU 复位信号由复位脚产生时, PDRF 旗标将由硬件置高.此旗标需由软件清零.

WDTF: 看门狗定时器复位旗标.

当 MCU 复位信号由看门狗产生时, WDTF 旗标将由硬件置高.此旗标需由软件清零.

SWRF: 软件复位旗标.

当 MCU 复位信号由软件复位产生时, SWRF 旗标将由硬件置高.此旗标需由软件清零.

LVRF: 低电压复位旗标.

当 MCU 复位信号由低电压复位产生时, LVRF 旗标将由硬件置高.此旗标需由软件清零.

PORF: 上电复位旗标.

当 MCU 复位信号由上电复位产生时, PORF 旗标将由硬件置高.此旗标需由软件清零.

1.5.4 Time Access Key register (TAKEY)

符号: TAKEY								地址: F7H
7	6	5	4	3	2	1	0	Reset
TAKEY [7:0]								00H

软件复位寄存器(SWRES) 默认为只读; 软件把 3 个特殊值按顺序写到 TAKEY 寄存器来使能软件复位寄存器(SWRES)可写. 它们是

```
MOV TAKEY, #55h
MOV TAKEY, #0AAh
MOV TAKEY, #5Ah
```

1.5.5 软件复位寄存器(SWRES)

符号: SWRES								地址: E7H
7	6	5	4	3	2	1	0	Reset
SWRES [7:0]								00H

SWRES[7:0]: 软件复位寄存器. 在软件复位过程的最后会进行自重置.

SWRES [7:0] = FFh, 产生软件复位.

SWRES [7:0] = 00h ~ FEh, 不产生复位动作.

1.5.6 Example of software reset

```
MOV TAKEY, #55h
MOV TAKEY, #0AAh
MOV TAKEY, #5Ah ; enable SWRES write attribute
MOV SWRES, #0FFh ; software reset MCU
```

1.6 系统时钟源

默认时钟来自于内部16MhzOSC时钟信号，其时钟应用于初始化阶段，其主要的工作是确定时钟源使用的正常运行。

内部时钟源来源于不同分频的内部OSC如表 1-1所示，时钟源可在编程器或ICP中设置.

表 1-1: Selection of clock source

Clock source
external crystal (use XTAL1 and XTAL2 pins)
external crystal/2 (use XTAL1 and XTAL2 pins)
external crystal/4 (use XTAL1 and XTAL2 pins)
external crystal (only use XTAL1, the XTAL2 define as I/O)
16MHz from internal OSC
8MHz from internal OSC
4MHz from internal OSC
2MHz from internal OSC
1MHz from internal OSC

对于来源于内部OSC的频率，这将有些许误差，在应用程序中需要精确的频率时，须注意使用. 最大误差如表 1- 2

表 1- 2: Temperature with variance

Temperature	Max Variance
25°C	±2%

2. 指令设置

所有OB5510的指令都是兼容的二进制码，且具有标准的8051一样的功能，以下表格列出了以OB5510微控器为核心的指令集的周期的总结，这里的周期表示为机器周期。

表 2- 1: Arithmetic operations

符号	描述	代码	字节	周期
ADD A,Rn	Add register to accumulator	28-2F	1	1
ADD A,direct	Add direct byte to accumulator	25	2	2
ADD A,@Ri	Add indirect RAM to accumulator	26-27	1	2
ADD A,#data	Add immediate data to accumulator	24	2	2
ADDC A,Rn	Add register to accumulator with carry flag	38-3F	1	1
ADDC A,direct	Add direct byte to A with carry flag	35	2	2
ADDC A,@Ri	Add indirect RAM to A with carry flag	36-37	1	2
ADDC A,#data	Add immediate data to A with carry flag	34	2	2
SUBB A,Rn	Subtract register from A with borrow	98-9F	1	1
SUBB A,direct	Subtract direct byte from A with borrow	95	2	2
SUBB A,@Ri	Subtract indirect RAM from A with borrow	96-97	1	2
SUBB A,#data	Subtract immediate data from A with borrow	94	2	2
INC A	Increment accumulator	04	1	1
INC Rn	Increment register	08-0F	1	2
INC direct	Increment direct byte	05	2	3
INC @Ri	Increment indirect RAM	06-07	1	3
INC DPTR	Increment data pointer	A3	1	1
DEC A	Decrement accumulator	14	1	1
DEC Rn	Decrement register	18-1F	1	2
DEC direct	Decrement direct byte	15	2	3
DEC @Ri	Decrement indirect RAM	16-17	1	3
MUL AB	Multiply A and B	A4	1	5
DIV	Divide A by B	84	1	5
DA A	Decimal adjust accumulator	D4	1	1

表 2- 2: Logic operations

符号	描述	代码	字节	周期
ANL A,Rn	AND register to accumulator	58-5F	1	1
ANL A,direct	AND direct byte to accumulator	55	2	2
ANL A,@Ri	AND indirect RAM to accumulator	56-57	1	2
ANL A,#data	AND immediate data to accumulator	54	2	2
ANL direct,A	AND accumulator to direct byte	52	2	3
ANL direct,#data	AND immediate data to direct byte	53	3	4
ORL A,Rn	OR register to accumulator	48-4F	1	1
ORL A,direct	OR direct byte to accumulator	45	2	2
ORL A,@Ri	OR indirect RAM to accumulator	46-47	1	2
ORL A,#data	OR immediate data to accumulator	44	2	2
ORL direct,A	OR accumulator to direct byte	42	2	3
ORL direct,#data	OR immediate data to direct byte	43	3	4
XRL A,Rn	Exclusive OR register to accumulator	68-6F	1	1
XRL A,direct	Exclusive OR direct byte to accumulator	65	2	2
XRL A,@Ri	Exclusive OR indirect RAM to accumulator	66-67	1	2
XRL A,#data	Exclusive OR immediate data to accumulator	64	2	2
XRL direct,A	Exclusive OR accumulator to direct byte	62	2	3
XRL direct,#data	Exclusive OR immediate data to direct byte	63	3	4
CLR A	Clear accumulator	E4	1	1
CPL A	Complement accumulator	F4	1	1
RL A	Rotate accumulator left	23	1	1
RLC A	Rotate accumulator left through carry	33	1	1
RR A	Rotate accumulator right	03	1	1
RRC A	Rotate accumulator right through carry	13	1	1
SWAP A	Swap nibbles within the accumulator	C4	1	1

表 2- 3: Data transfer

符号	描述	代码	字节	周期
MOV A,Rn	Move register to accumulator	E8-EF	1	1
MOV A,direct	Move direct byte to accumulator	E5	2	2
MOV A,@Ri	Move indirect RAM to accumulator	E6-E7	1	2
MOV A,#data	Move immediate data to accumulator	74	2	2
MOV Rn,A	Move accumulator to register	F8-FF	1	2
MOV Rn,direct	Move direct byte to register	A8-AF	2	4
MOV Rn,#data	Move immediate data to register	78-7F	2	2
MOV direct,A	Move accumulator to direct byte	F5	2	3
MOV direct,Rn	Move register to direct byte	88-8F	2	3
MOV direct1,direct2	Move direct byte to direct byte	85	3	4
MOV direct,@Ri	Move indirect RAM to direct byte	86-87	2	4
MOV direct,#data	Move immediate data to direct byte	75	3	3
MOV @Ri,A	Move accumulator to indirect RAM	F6-F7	1	3
MOV @Ri,direct	Move direct byte to indirect RAM	A6-A7	2	5
MOV @Ri,#data	Move immediate data to indirect RAM	76-77	2	3
MOV DPTR,#data16	Load data pointer with a 16-bit constant	90	3	3
MOVC A,@A+DPTR	Move code byte relative to DPTR to accumulator	93	1	3
MOVC A,@A+PC	Move code byte relative to PC to accumulator	83	1	3
MOVX A,@Ri	Move external RAM (8-bit addr.) to A	E2-E3	1	3
MOVX A,@DPTR	Move external RAM (16-bit addr.) to A	E0	1	3
MOVX @Ri,A	Move A to external RAM (8-bit addr.)	F2-F3	1	4
MOVX @DPTR,A	Move A to external RAM (16-bit addr.)	F0	1	4
PUSH direct	Push direct byte onto stack	C0	2	4
POP direct	Pop direct byte from stack	D0	2	3
XCH A,Rn	Exchange register with accumulator	C8-CF	1	2
XCH A,direct	Exchange direct byte with accumulator	C5	2	3
XCH A,@Ri	Exchange indirect RAM with accumulator	C6-C7	1	3
XCHD A,@Ri	Exchange low-order nibble indir. RAM with A	D6-D7	1	3

表 2- 4: Program branches

符号	描述	代码	字节	周期
ACALL addr11	Absolute subroutine call	xxx11	2	6
LCALL addr16	Long subroutine call	12	3	6
RET	from subroutine	22	1	4
RETI	from interrupt	32	1	4
AJMP addr11	Absolute jump	xxx01	2	3
LJMP addr16	Long jump	02	3	4
SJMP rel	Short jump (relative addr.)	80	2	3
JMP @A+DPTR	Jump indirect relative to the DPTR	73	1	2
JZ rel	Jump if accumulator is zero	60	2	3
JNZ rel	Jump if accumulator is not zero	70	2	3
JC rel	Jump if carry flag is set	40	2	3
JNC	Jump if carry flag is not set	50	2	3
JB bit,rel	Jump if direct bit is set	20	3	4
JNB bit,rel	Jump if direct bit is not set	30	3	4
JBC bit,direct rel	Jump if direct bit is set and clear bit	10	3	4
CJNE A,direct rel	Compare direct byte to A and jump if not equal	B5	3	4
CJNE A,#data rel	Compare immediate to A and jump if not equal	B4	3	4
CJNE Rn,#data rel	Compare immed. to reg. and jump if not equal	B8-BF	3	4
CJNE @Ri,#data rel	Compare immed. to ind. and jump if not equal	B6-B7	3	4
DJNZ Rn,rel	Decrement register and jump if not zero	D8-DF	2	3
DJNZ direct,rel	Decrement direct byte and jump if not zero	D5	3	4
NOP	No operation	00	1	1

表 2- 5: Boolean manipulation

符号	描述	代码	字节	周期
CLR C	Clear carry flag	C3	1	1
CLR bit	Clear direct bit	C2	2	3
SETB C	Set carry flag	D3	1	1
SETB bit	Set direct bit	D2	2	3
CPL C	Complement carry flag	B3	1	1
CPL bit	Complement direct bit	B2	2	3
ANL C,bit	AND direct bit to carry flag	82	2	2
ANL C,/bit	AND complement of direct bit to carry	B0	2	2
ORL C,bit	OR direct bit to carry flag	72	2	2
ORL C,/bit	OR complement of direct bit to carry	A0	2	2
MOV C,bit	Move direct bit to carry flag	A2	2	2
MOV bit,C	Move carry flag to direct bit	92	2	3

3. 存储器结构

OB5510存储器结构和通用的8051结构相同，它们是作为程序存储器的8KB的嵌入式闪存。

3.1 程序存储器

OB5510有8KB的嵌入式可编程存储器中,如以下图 3-1,可做为通用的程序存储.其它还包括最大为1K的特定ISP服务程序存储空间. 这8K的地址从\$0000到\$1FFF. ISP服务程序的地址从\$1C00到\$1FFF. ISP服务程序空间可以被分割N块128字节(N=0到8). 当N=0时,意味着没有ISP复位程序空间可用. 全部的8KB字节空间都被用来做程序存储. 当N=1时,意味着地址\$1F80到 \$1FFF保留为ISP服务程序空间. 当N=2意味着内存地址\$1F00到\$1FFF保留为ISP服务程序空间等等... 数值N可以通过编程器设置.

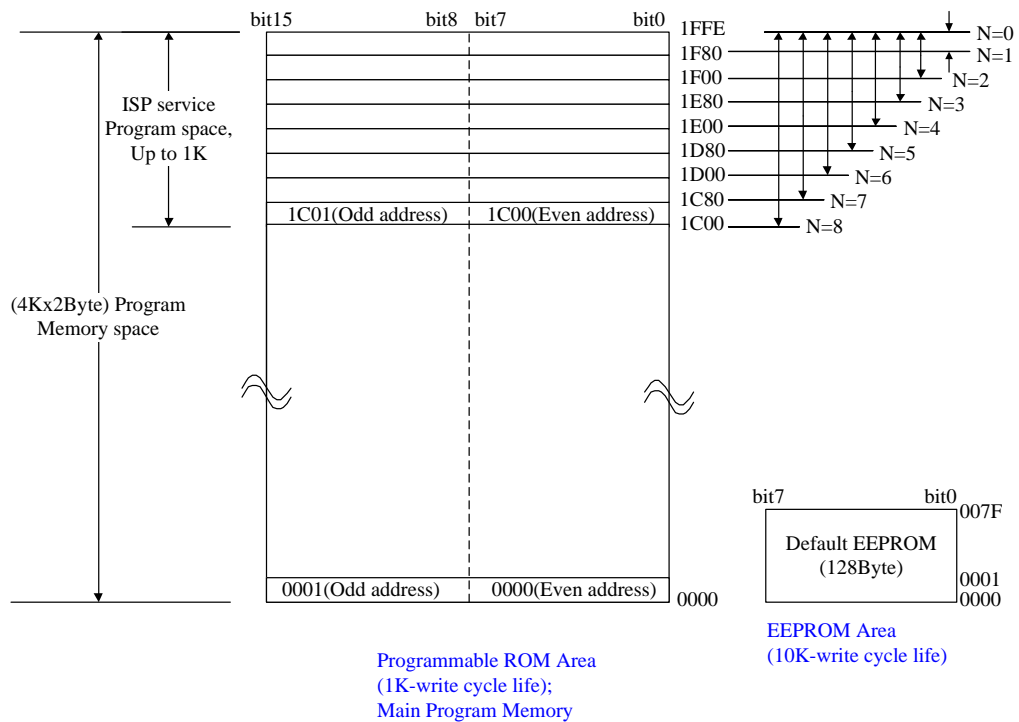


图 3-1: OB5510 ROM

3.2 数据存储

OB5510具有512B Bytes的片上SRAM, 如以下 图 3- 2, 其中256 Bytes和通用的8052内部存储器结构一样。

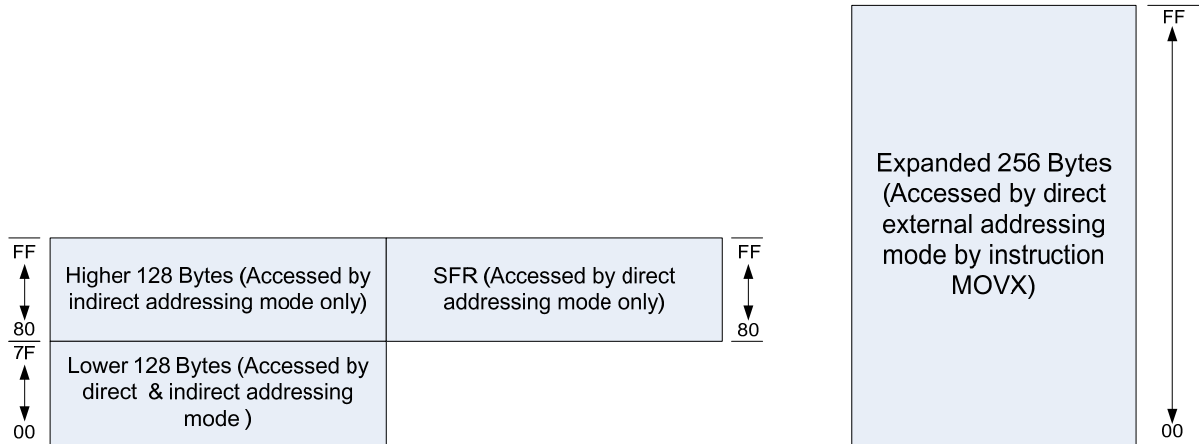


图 3- 2: RAM architecture

3.3 数据内存-低 128 字节(00h 到 7Fh)

数据存储器从 00h 到 FFh 的地址和在 8052 中的定义是一样的。
 00h 到 7Fh 的地址可通过直接或者间接寻址方式访问。
 00h 到 1Fh 是寄存器的空间,
 20h 到 2Fh 是位寻址空间,
 30h 到 7Fh 是通用的数据存储区。

3.4 数据存储器-高 128 字节(80h 到 FFh)

80h 到 FFh 的地址仅仅能从间接寻址的方式访问,它是一个数据区。

3.5 存储器-扩展的 256 字节(\$00 到 \$FF)

从00h到FFh是片内扩展的SRAM区域,共256字节, 该空间地址只能通过外部直接寻址的方式进行访问。(利用MOVX指令)

4. CPU结构

OB5510结构由以下四部分组成:

- (1) 控制单元
- (2) 算法-逻辑单元
- (3) 存储器控制单元
- (4) RAM 和 SFR 控制单元

OB5510结构允许接受来自程序存储器的指令并与RAM或SFR做数据处理，以下各段详细叙述了主要功能寄存器。

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
8051 Core											
ACC	Accumulator	E0h	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00H
B	B register	F0h	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00H
PSW	Program status word	D0h	CY	AC	F0	RS[1:0]		OV	F1	P	00H
SP	Stack Pointer	81h	SP[7:0]								07H
DPL	Data pointer low 0	82h	DPL[7:0]								00H
DPH	Data pointer high 0	83h	DPH[7:0]								00H
DPL1	Data pointer low 0	84h	DPL1[7:0]								00H
DPH1	Data pointer high 0	85h	DPH1[7:0]								00H
AUX	Auxiliary register	91h	BRGS	-	-	-	-	-	-	DPS	00H
CKCON	Clock control register	8Eh	-	ITS[2:0]			-	-	CLKOUT[1:0]		10H
IFCON	Interface control register	8Fh	-	CDPR	-	-	-	-	-	ISPE	00H

4.1 累加器

符号: **ACC** 地址: **E0h**

7	6	5	4	3	2	1	0	Reset
ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00h

ACC[7:0]: ACC 是一个累加器，大部分单操作指令的一个操作数取自累加器。

4.2 B 寄存器

符号: **B** 地址: **F0h**

7	6	5	4	3	2	1	0	Reset
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00h

B[7:0]: B 寄存器被用于乘法或除法指令，也可作为一般寄存器以存储临时数据。

4.3 程序状态字

符号: PSW							地址: D0h	
7	6	5	4	3	2	1	0	Reset
CY	AC	F0	RS [1:0]		OV	F1	P	00h

CY: 进位标志位

0: 没有进位或借位发生

1: 有进位或借位发生

AC: 为 BCD 辅助进位标志位

0: 没有辅助进位或借位发生

1: 有辅助进位或借位发生

F0: 用户设置标志位 0

RS: R0~R7 寄存器库选择位

RS[1:0]	Bank Selected	Location
00	Bank 0	00h – 07h
01	Bank 1	08h – 0Fh
10	Bank 2	10h – 17h
11	Bank 3	18h – 1Fh

OV: 溢出标志位

0: 没有溢出发生

1: 有溢出发生

F1: 用户设置标志位 1

P: 奇偶校验位, 受硬件影响, 显示累加器中的奇偶的 1 位, 即奇偶校验

0: 累加器 A 为 1 的位数为偶数

1: 累加器 A 为 1 的位数为奇数

4.4 堆栈指针

堆栈指针是一个1字节的寄存器,在复位后初始化为07h。此寄存器在执行PUSH和CALL指令之前增值,使得堆栈指针在08h开始执行

符号: SP							地址: 81h	
7	6	5	4	3	2	1	0	Reset
SP [7:0]								07h

SP[7:0]: 堆栈指针储存了暂时寄存器的地址,该地址是堆栈指针的起始位置.换言之,它总是指向堆栈指针的顶端。

4.5 数据指针

数据指针为2字节.低位为DPL.高位为DPH.它可以作为一个2字节的寄存器(MOV DPTR,#data16)来使用,或者作为两个寄存器(例如,MOV DPL,#data8),它通常被用作是运行外部程序或者是数据空间(如,MOVC A,@ A+DPTR或者各自的MOV A, @ DPTR).

符号: DPL							地址: 82h	
7	6	5	4	3	2	1	0	Reset
DPL [7:0]								00h

DPL[7:0]: Data pointer Low 0

符号: DPH							地址: 83h	
7	6	5	4	3	2	1	0	Reset
DPH [7:0]								00h

DPH [7:0]: Data pointer High 0

4.6 数据指针 1

双数据指针加速了块状数据的运行速度.标准的DPTR 是一个16位的寄存器,是用来寻址外部空间或者外部存储器.在OB5510中,标准的数据指针称为DPTR,第二个数据指针称为DPTR1.选择位用来选择使用的数据指针.数据指针的选择位位于AUX寄存器的LSB(DPS).

用户通过触发AUX寄存器中的LSB位进行切换.所有有关的DPTR指令都将为当前选用的DPTR所操作。

符号: DPL1							地址: 84h	
7	6	5	4	3	2	1	0	Reset
DPL1 [7:0]								00h

DPL1[7:0]: Data pointer Low 1

符号: DPH1							地址: 85h	
7	6	5	4	3	2	1	0	Reset
DPH1 [7:0]								00h

DPH1[7:0]: Data pointer High 1

符号: AUX							地址: 91h	
7	6	5	4	3	2	1	0	Reset
BRGS	-	-	-	-	-	-	DPS	00H

DPS: 数据指针选择位.

DPS = 0 选择数据指针 0.

DPS = 1 选择数据指针 1.

4.7 时钟控制寄存器

符号: CKCON						地址: 8Eh		
7	6	5	4	3	2	1	0	Reset
-	ITS[2:0]			-	-	CLKOUT[1:0]		10H

ITS[2:0]: 指令时钟周期选择.

ITS [2:0]	指令时钟周期
000	1T 模式
001	2T 模式(默认)
010	3T 模式
011	4T 模式
100	5T 模式
101	6T 模式
110	7T 模式
111	8T 模式

4.8 接口控制寄存器

符号: IFCON						地址: 8Fh		
7	6	5	4	3	2	1	0	Reset
-	CDPR	-	-	-	-	-	ISPE	00H

CDPR: 程序码已加密指示位(仅读)

ISPE: ISP 功能使能位

ISPE = 1, 允许使用 ISP 功能

ISPE = 0, 禁止使用 ISP 功能

5. GPIO管脚型态

OB5510的I/O口是: 准双向口 (标准的8051端口输出),推挽电路,开漏, 与只输入. 两种寄存器的配置为每个端口的每个位选择输出方式.OB5510的所有端口可以通过软件配置四种型号的一种.如下表显示:

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
I/O port function register											
P0M0	Port 0 output mode 0	D2h					P0M0[7:0]				00H
P0M1	Port 0 output mode 1	D3h					P0M1[7:0]				00H
P1M0	Port 1 output mode 0	D4h					P1M0[7:0]				00H
P1M1	Port 1 output mode 1	D5h					P1M1[7:0]				00H
P3M0	Port 3 output mode 0	DAh					P3M0[7:0]				00H
P3M1	Port 3 output mode 1	DBh					P3M1[7:0]				00H

PxM1.y	PxM0.y	Port output mode
0	0	Quasi-bidirectional (standard 8051 port outputs) (pull-up) 准双向口 (标准的8051端口输出)
0	1	Push-pull 推挽
1	0	Input only (high-impedance) 仅输入 (high-impedance)
1	1	Open drain 开漏

一般的应用,每个管脚都可独立的置高或置低.如下表显示:

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
Ports											
Port 3	Port 3	B0h	-	-	-	-	-	-	P3.1	P3.0	FFh
Port 1	Port 1	90h	-	P1.6	-	P1.4	P1.3	P1.2	P1.1	P1.0	FFh
Port 0	Port 0	80h	P0.7	P0.6	-	-	-	-	-	P0.0	FFh

符号: **P0** 地址: **80h**

7	6	5	4	3	2	1	0	Reset
P0.7	P0.6	-	-	-	-	-	P0.0	FFh

符号: **P1** 地址: **90h**

7	6	5	4	3	2	1	0	Reset
-	P1.6	-	P1.4	P1.3	P1.2	P1.1	P1.0	FFh

符号: **P3** 地址: **B0h**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	P3.1	P3.0	FFh

6. 定时器0、定时器1

OB5510有二个16bit的定时/计数寄存器: 定时器0, 定时器1. 所有这些都可以通过被设置为定时或计数操作.

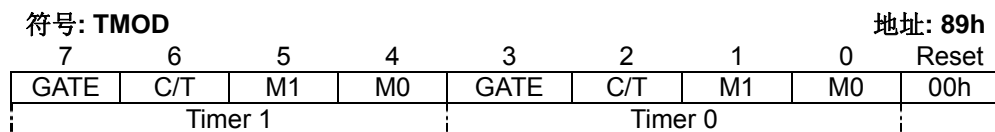
在计时的模式中, 定时器0和定时器1寄存器的递增频率, 可由寄存器PFCON中选择为与振荡器频率相同或为振荡器频率的1/12或为振荡器频率的1/96.

在计数的模式中, 当检测到相应的输入脚T0/T1有下降沿产生时, 寄存器便得到递增, 由于它需要2个机器周期来识别由1到0的跳变, 其最大输入的计数频率为振荡器频率的1/2, 从而使得这里没有限制的占空比, 以确定适当的识别为0或1的状态, 因此, 一个输入信号至少要稳定在1个机器周期.

定时器0和定时器1都有四种操作模式选择, 在选择操作模式中, 两个特殊功能寄存器 (TMOD and TCON) 被应用其中.

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
Timer 0 and 1											
TL0	Timer 0, low byte	8Ah	TL0[7:0]								00H
TH0	Timer 0, high byte	8Ch	TH0[7:0]								00H
TL1	Timer 1, low byte	8Bh	TL1[7:0]								00H
TH1	Timer 1, high byte	8Dh	TH1[7:0]								00H
TMOD	Timer Mode Control	89h	GATE	C/T	M1	M0	GATE	C/T	M1	M0	00H
TCON	Timer/Counter Control	88h	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H
PFCON	Peripheral Frequency control register	D9h	-	-	SRELPS[1:0]		T1PS[1:0]		T0PS[1:0]		00H

6.1 定时器/计数器模式控制寄存器(TM0D)



GATE: 该位被置位时为门控时, 仅当‘INT0 或者 INT1’脚为高时, 且‘TRx’控制位被置位 (参考 TCON 寄存器) 时使能, 计数器在每个 T0 或 T1 输入脚处于下减沿触发时得到加强.

C/T: 门控定时器或计数器选择器. 该位被置位时用作计数器功能, 该位被清零时用作定时器功能.

M[1:0]: 定时/计数器 0 或定时/计数器 1 的选择模式

M1	M0	Mode	Function
0	0	Mode0	13bit计数器/定时器, 包含TL0/TL1寄存器的低5

			位及TH0/TL1寄存器的全部8位，其TL0/TL1寄存器的高3位可设置为0.
0	1	Mode1	16 位计数器/定时器.
1	0	Mode2	8 位自动重载的计数器/定时器,自动重载的值保留在TH0 和TH1.同时TL0 或者TL1 在每个机器周期内都会递增.当溢出时，将THx 存放的值装入T x.
1	1	Mode3	如定时器1 的M1 和M2 位被设置为1，定时器2 停止计数。如定时器0 的M1 和M0 位被设置为1，定时器0 作为两个独立的8 位定时器/计数器.

6.2 定时器/计数器控制寄存器(TCON)

符号: TCON								地址: 88h
7	6	5	4	3	2	1	0	Reset
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00h

TF1: 定时器 1 溢出标志。定时器/计数器溢出时由硬件置位。中断执行时硬件清零，或由软件清零

TR1: 定时器 1 运行控制位。如置位/清零, 关闭定时器/计数器 1.

TF0: 定时器 0 溢出标志。定时器/计数器溢出时由硬件置位。中断执行时硬件自动清零，或由软件清零

TR0: 定时器 0 运行控制位。如置位/清零,关闭定时器/计数器 0

IE1: 中断 1 边沿标志。当检测到外部中断 1 边沿/低电平时由硬件置位该标志。中断处理时由硬件清零，或通过软件清零

IT1: 中断 1 类型控制位，通过软件置位/清零该位来选择外部中断 1 以下降沿/低电平方式触发

IE0: 中断 0 边沿标志。当检测到外部中断 0 边沿/低电平时由硬件置位该标志。中断处理时由硬件清零，或通过软件清零

IT0: 中断 0 类型控制位。通过软件置位/清零该位来选择外部中断 0 以下降沿/低电平方式触发

6.3 定时器输入频率控制寄存器

符号: PFCON						地址: D9h		
7	6	5	4	3	2	1	0	Reset
-	-	SRELPS[1:0]		T1PS[1:0]		T0PS[1:0]		00h

T1PS[1:0]: 定时器 1 分频器选择位

T1PS[1:0]	分频器
00	Fosc/12
01	Fosc
10	Fosc/96
11	reserved

T0PS[1:0]: 定时器 0 分频器选择位

T0PS[1:0]	分频器
00	Fosc/12
01	Fosc
10	Fosc/96
11	reserved

6.4 模式 0 (13 位定时/计数)

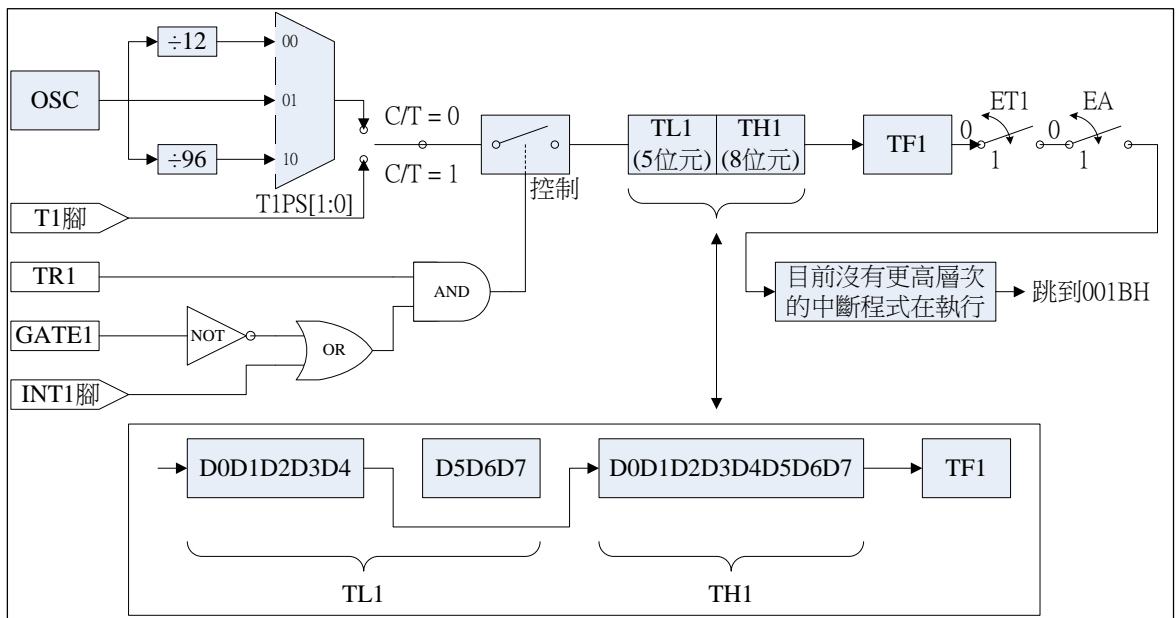


图 6-1: 模式 0 - 13 位定时器/计数器操作

6.5 模式 1 (16 位定时/计数)

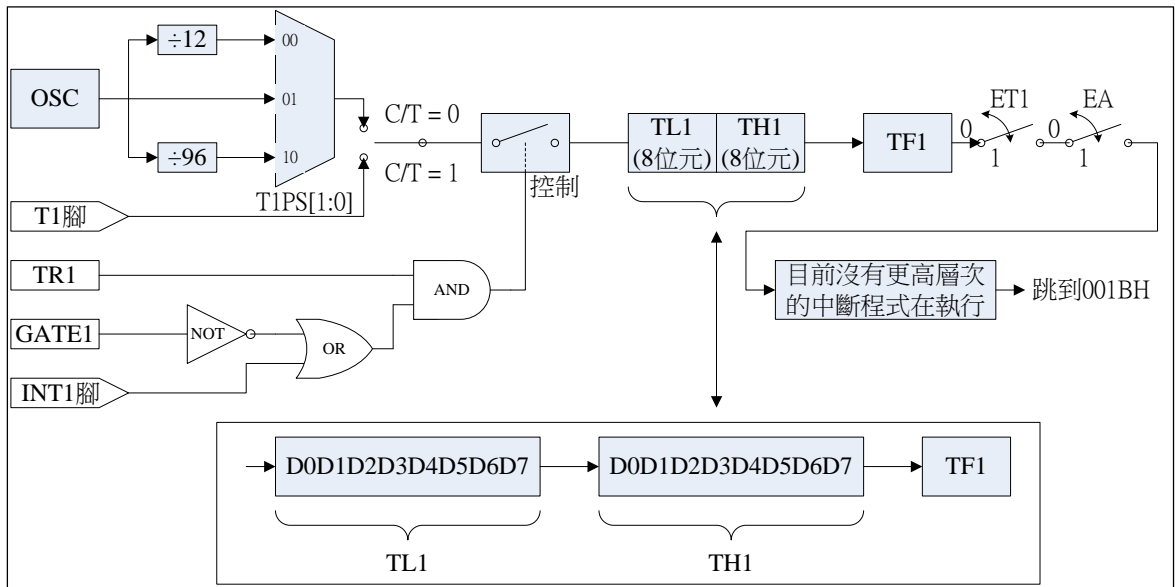


图 6-2: 模式 1 -16 位定时器/计数器操作

6.6 模式 2 (8 位自动重载定时/计数)

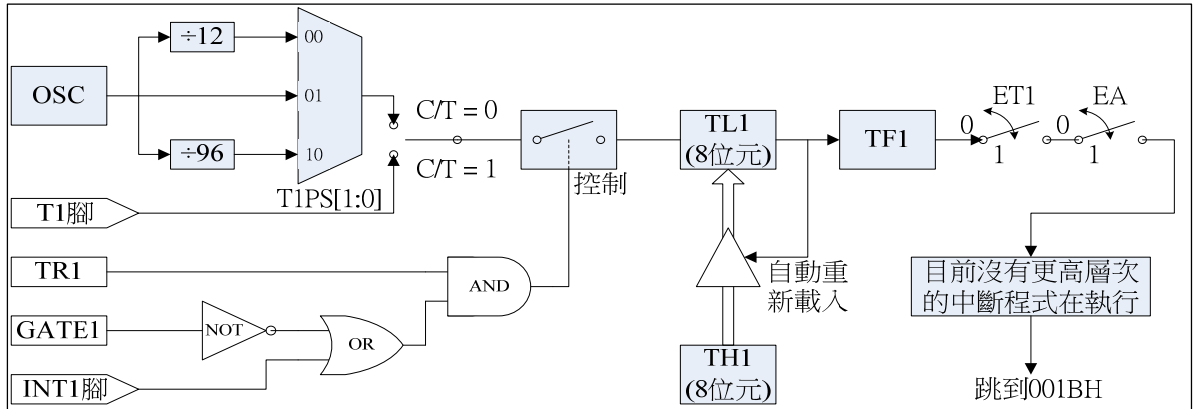


图 6-3: 模式 2 -8 位自动重载定时/计数

6.7 模式 3 (两个独立 8 位定时/计数(仅定时器 0))

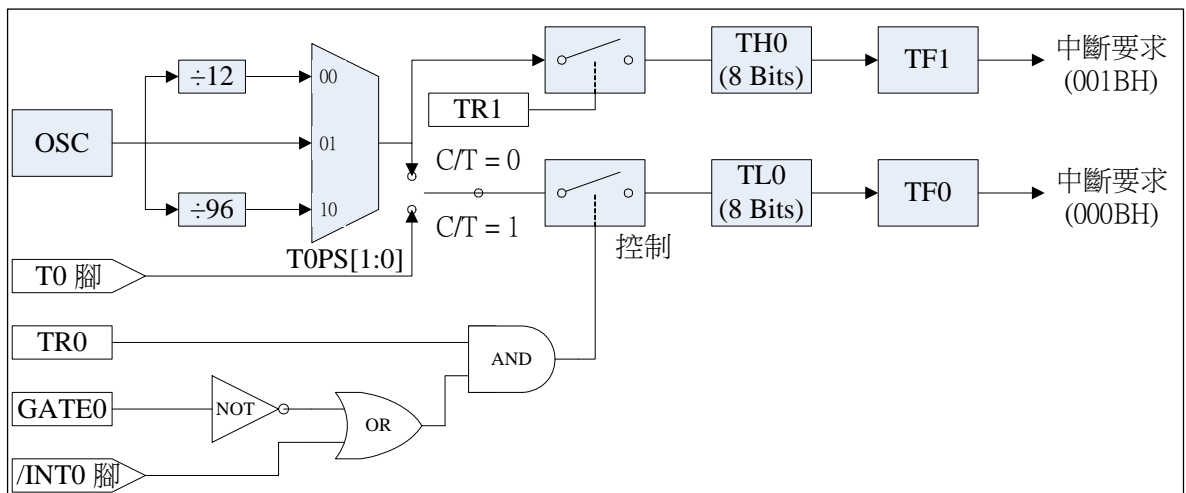


图 6-4: 模式 3 -两个独立 8 位定时/计数(仅定时器 0)

7. 定时器2 以及捕捉/比较单元(CCU)

定时器2不仅仅是一个16位的定时器，也是一个带有4个通道比较,捕获及重载功能，这是非常相似在其它一些微控制器的可编程计数器阵列（PCA），除脉冲宽度调制（PWM）

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
Timer 2 and Capture Compare Unit											
T2CON	Timer 2 control	C8h	T2PS[2:0]			T2R[1:0]		-	T2I[1:0]		00H
CCCON	Compare/Capture Control	C9h	CCI3	CCI2	CCI1	CCI0	CCF3	CCF2	CCF1	CCF0	00H
CCEN	Compare/Capture Enable register	C1h	-	COCAM1[2:0]			-	COCAM0[2:0]			00H
CCEN2	Compare/Capture Enable 2 register	D1h	-	COCAM3[2:0]			-	COCAM2[2:0]			00H
TL2	Timer 2, low byte	CCh	TL2[7:0]								00H
TH2	Timer 2, high byte	CDh	TH2[7:0]								00H
CRCL	Compare/Reload/Capture register, low byte	CAh	CRCL[7:0]								00H
CRCH	Compare/Reload/Capture register, high byte	CBh	CRCH[7:0]								00H
CCL1	Compare/Capture register 1, low byte	C2h	CCL1[7:0]								00H
CCH1	Compare/Capture register 1, high byte	C3h	CCH1[7:0]								00H
CCL2	Compare/Capture register 2, low byte	C4h	CCL2[7:0]								00H
CCH2	Compare/Capture register 2, high byte	C5h	CCH2[7:0]								00H
CCL3	Compare/Capture register 3, low byte	C6h	CCL3[7:0]								00H
CCH3	Compare/Capture register 3, high byte	C7h	CCH3[7:0]								00H

符号: **T2CON**

地址: **C8h**

7	6	5	4	3	2	1	0	Reset
T2PS[2:0]			T2R[1:0]		-	T2I[1:0]		00H

T2PS[2:0]: 分频器选择位:

T2PS = 000 –定时器 2 的时钟为振荡频率.

T2PS = 001 –定时器 2 的时钟为振荡频率的 1/2.

- T2PS = 010 –定时器 2 的时钟为振荡频率的 1/4.
- T2PS = 011 –定时器 2 的时钟为振荡频率的 1/6.
- T2PS = 100 –定时器 2 的时钟为振荡频率的 1/8.
- T2PS = 101 –定时器 2 的时钟为振荡频率的 1/12.
- T2PS = 110 –定时器 2 的时钟为振荡频率的 1/24.

T2R[1:0]: 定时器 2 重载模式选择位

- T2R[1:0] = 0X –重载无效
- T2R[1:0] = 10 –模式 0:自动重载
- T2R[1:0] = 11 –模式 1:依 T2EX 脚下降缘重载

T2I[1:0]: 定时器 2 输入选择位

- T2I[1:0] = 00 –定时器 2 计数停止
- T2I[1:0] = 01 –输入频率依 T2PS[2:0]分频器选择
- T2I[1:0] = 10 –定时器 2 在 T2 脚的外部信号递增
- T2I[1:0] = 11 –门控定时器 2 可为内部时钟输入

符号: CCCON								地址: C9h
7	6	5	4	3	2	1	0	Reset
CCI3	CCI2	CCI1	CCI0	CCF3	CCF2	CCF1	CCF0	00H

CCI3: 捕捉/比较信道 3 中断功能控制位.

“1” 中断功能使能.

CCI2: 捕捉/比较信道 2 中断功能控制位.

“1” 中断功能使能.

CCI1: 捕捉/比较信道 1 中断功能控制位.

“1” 中断功能使能.

CCI0: 捕捉/比较信道 0 中断功能控制位.

“1” 中断功能使能.

CCF3: 捕捉/比较通道 3 中断旗标位.可由软件清零.

CCF2: 捕捉/比较通道 2 中断旗标位.可由软件清零.

CCF1: 捕捉/比较通道 1 中断旗标位.可由软件清零.

CCF0: 捕捉/比较通道 0 中断旗标位.可由软件清零.

捕捉/比较中断与定时器 2 中断共享同一个中断向量.

符号: CCEN								地址: C1h
7	6	5	4	3	2	1	0	Reset
-	COCAM1[2:0]			-	COCAM0[2:0]			00H

COCAM1[2:0]: 000: 禁止比较/捕获功能

如以下图 7-4 数字阐述了比较模式0的功能

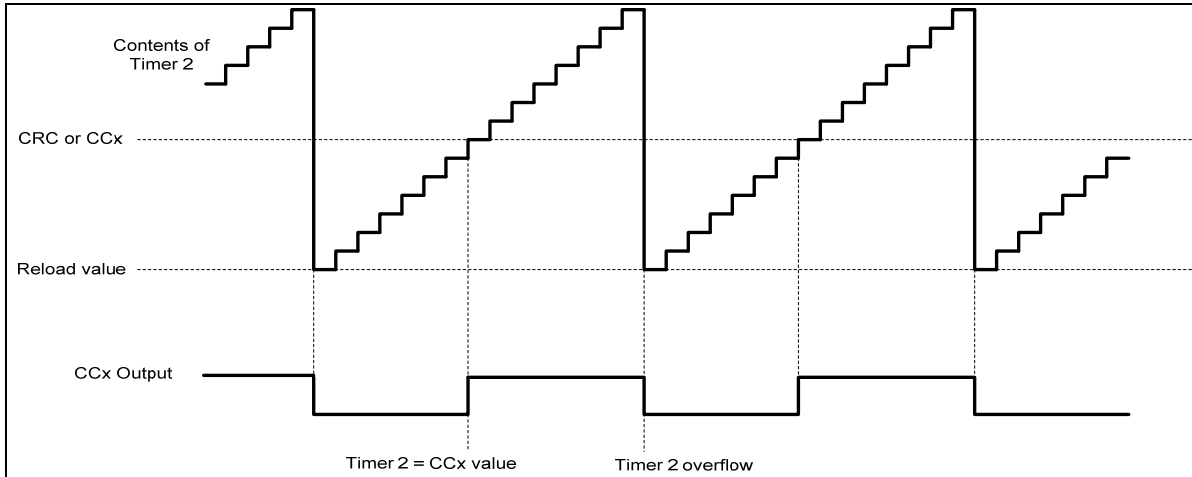


图 7-4: Compare mode 0 function

7.2.2 比较模式 1

在比较模式1中，其输出信号的跳变是由软件来决定的。定时器2的溢出不会导致输出的改变。在此模式中，两种信号的跳变都是可以控制的，如图 7-5显示了在比较模式1中的寄存器/端口结构菜单。

在比较模式1中，其值将首先被写入到“阴影寄存器”中，当比较信号被激活时，其值被传递到输出寄存器

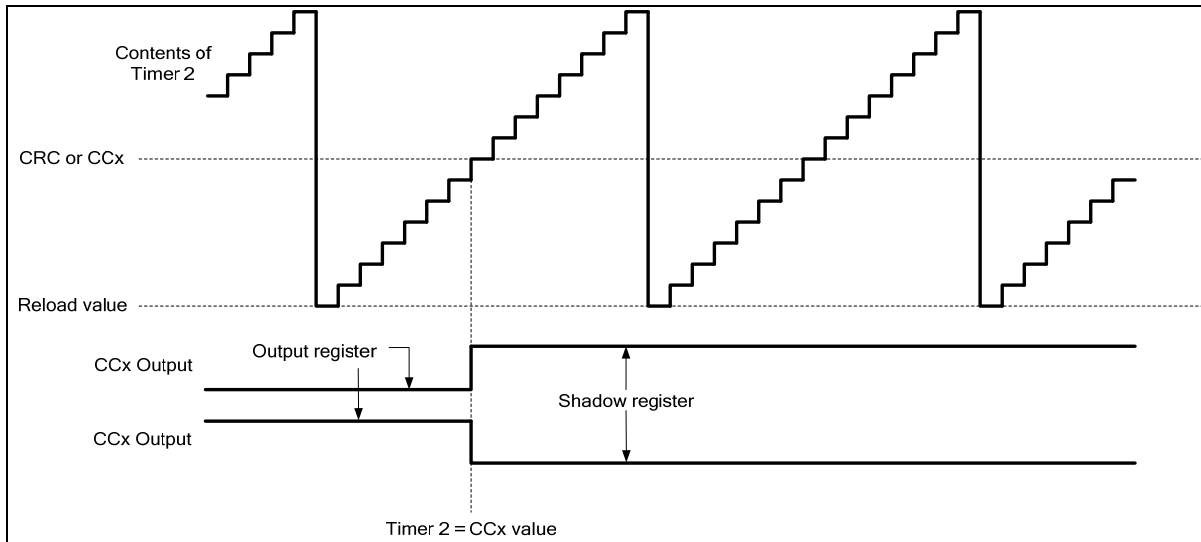


图 7-5: 比较模式 1 功能

7.3 捕获功能

在一个外部事件（模式0）或一个软件写操作（模式1）上，实际的定时器/计数器的值可以被保存在寄存器CCx或CRC中。

7.3.1 捕捉模式 0

如以下 图 7-6，在模式0中，定时器2的值的捕捉在以下情况进行：

- (5) 上升沿输入 CC0-CC3。
- (6) 下降沿输入 CC0-CC3。
- (7) 上升沿及下降沿输入 CC0-CC3。

定时器2的内容将被对应的捕捉寄存器锁存。

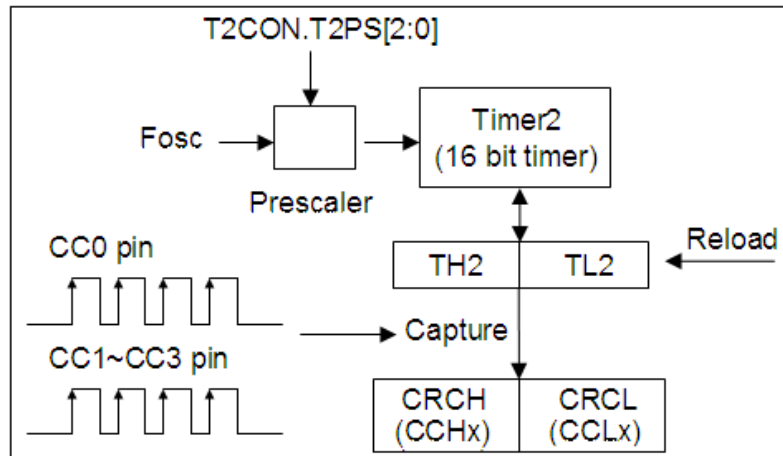


图 7-6: 捕捉模式 0 功能

7.3.2 捕捉模式 1

如以下 图 7-7，在模式1中，定时器2的值的捕捉将导致写入到捕捉寄存器中的低字节无任何价值，其捕捉寄存器的写入值与改功能无关，定时器2的内容将被对应的捕捉寄存器锁存。

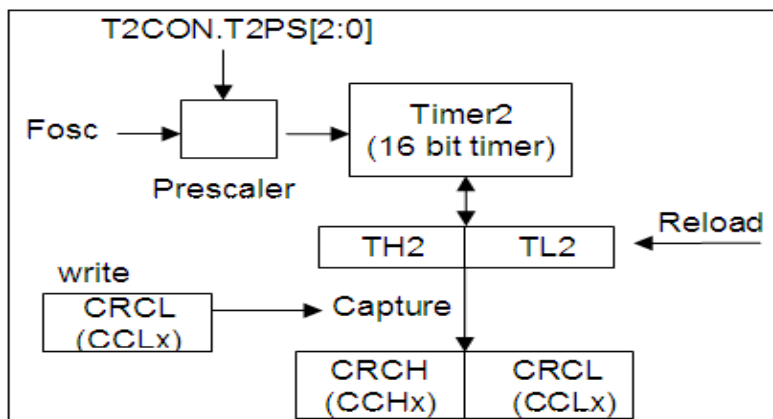


图 7-7: 捕捉模式 1 功能

8. 串行接口

两个串行缓冲器组成两个单独寄存器，即一个传输缓冲区和一个接收缓冲区。

写入数据到特殊功能寄存器（SFR）SBUF并设置这些数据在串行输出缓冲，并开始传输，来自SBUF的读取及从串行接收缓冲区读取数据，串行口可同时传输和接收数据，它也可在接收时缓存1字节，如CPU在第一个字节传输完成之前读取第二个字节，以防接收数据丢失。

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
Serial interface 0 and 1											
PCON	Power control	87H	SMOD	-	-	-	-	-	STOP	IDLE	00H
AUX	Auxiliary register	91h	BRGS	-	-	-	-	-	-	DPS	00H
SCON	Serial Port control register	98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SRELL	Serial Port reload register low byte	AAH	SREL.7	SREL.6	SREL.5	SREL.4	SREL.3	SREL.2	SREL.1	SREL.0	00H
SRELH	Serial Port reload register high byte	BAH	-	-	-	-	-	-	SREL.9	SREL.8	00H
SBUF	Serial Port data buffer	99H	SBUF[7:0]								00H
PFCON	Peripheral Frequency control register	D9h	-	-	SRELPS[1:0]		T1PS[1:0]		T0PS[1:0]		00H

符号: **AUX** 地址: **91h**

7	6	5	4	3	2	1	0	Reset
BRGS	-	-	-	-	-	-	DPS	00H

BRGS: BRGS = 0 –波特率产生器使用定时器 1 TH1 寄存器.

BRGS = 1 –波特率产生器使用 SREL 寄存器.

符号: **SCON** 地址: **98h**

7	6	5	4	3	2	1	0	Reset
SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00h

SM0,SM1: 串行口模式选择.

SM0	SM1	Mode
0	0	0
0	1	1
1	0	2
1	1	3

在 UART 的 4 种模式中，模式 0~3 稍后解释

SM2: 多处理机通信使能位

REN: 如置位，串行接收使能，软件清除禁止接收

TB8: 在模式 2 和 3 中, 发送的第 9 位数据位, 置位或清零取决于它执行的功能, 如奇偶校验功能, 多处理机通信等。

RB8: 在模式 2 和 3 中, RB8 为接收的第 9 位数据位。在模式 1 中, 如 SM2=0, RB8 为停止位。在模式 0 中, 此位不被使用。须由软件清除。

TI: 发送中断标志位。在完成串行传输后由硬件置位, 须由软件清除。

RI: 接收中断标志。在完成串行传输后由硬件置位, 须由软件清除。

8.1 串行接口由以下 4 种模式可以设置

SM0	SM1	Mode	描述	Board Rate
0	0	0	Shift register	Fosc/12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	Fosc/32 or Fosc/64
1	1	3	9-bit UART	Variable

这里的Fosc是晶体或振荡器的频率。

8.1.1 模式 0

引脚RXD充当输入和输出。TXD输出时钟。每次发送或接收以LSB最低位作首位, 每次8位。波特率固定为晶体的频率1/12,接收通过以下的在SCON中的设置标志在模式0中初始化: RI = 0及REN = 1。在其它模式中, 当REN=1时, 开始从起始位接收串行数据。



图 8-1: 发送模式 0

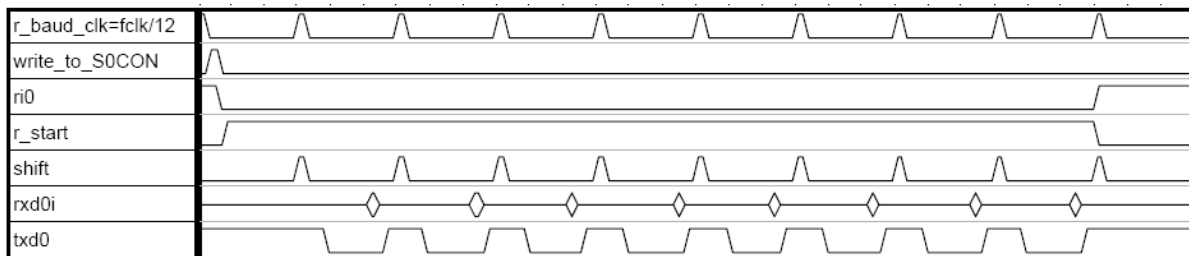


图 8-2: 接收模式 0

8.1.2 模式 1

引脚RXD充当输入，TXD充当串行输出，无任何外部时钟被使用，每次数据为10位：一个起始位(=0)，8个数据位（LSB位在前），及一个停止位(=1)。在接收数据时，起始位将被同步传输，8个数据位可通过SBUF来读取，一个停止位存于特殊功能寄存器SCON的设置标志RB8内，在模式1中，无论是内部的波特率发生器或定时器1可以用来指定波特率。



图 8-3: 发送模式 1

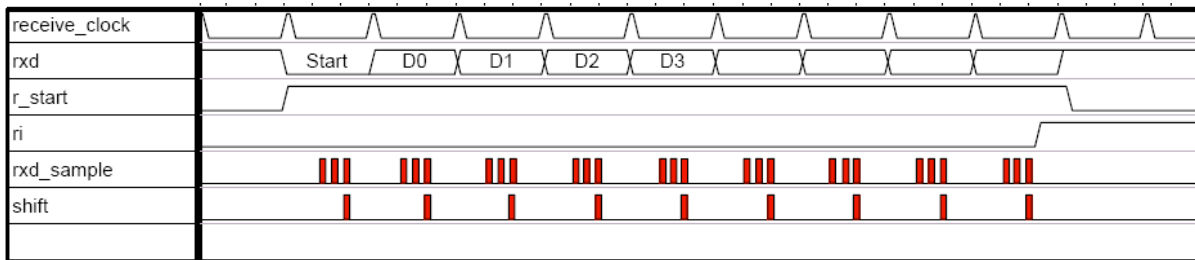


图 8-4: 接收模式 1

8.1.3 模式 2

该模式和模式1类似，但有两点不同。波特率被固定在振荡器的频率的1/32（SMOD=1）或1/64（SMOD=0），且有11位数据被传输或接收：1个起始位（=0），8个数据位（LSB在前），一个可编程的第9位及一个停止位（=1），9位可以用来控制串行接口的奇偶性。在传输中，SCON中的TB8输出第9位，在接收中，SCON中的RB8将被影响。

8.1.4 模式 3

模式2和3的唯一不同之处在于：在模式3中，无论是内部的波特率发生器或定时器1可以用来指定波特率。



图 8-5: 传输模式 2 和模式 3

图 8-6: 接收模式 2 和 3 中

8.2 串行接口的多重机通讯

在串行接口的模式2和3,接收9位的功能,可用于多处理器的通讯。在这种情况下,从机在SCON中的位SM2被置位为1。当主机输出从机的地址时,Bit9将被置1,从而在所有的从机中导致串行口接收中断。从机将接收到的字节和它们的网络地址比较,如匹配,其从机将清除SM2,并接收其余的信息,其它的从机将远离SM2无效,并忽略此信息。在解决从机后,其主机在Bit9清0时将输出剩余部分的信息,因此,没有串口接收中断会产生在未选中的处理机中。

8.3 输入频率控制寄存器

符号: PFCON						地址: D9h	
7	6	5	4	3	2	1	0
-	-	SRELPS[1:0]		T1PS[1:0]		T0PS[1:0]	
							Reset
							00H

SRELPS[1:0]: SREL 分频器选择位

SRELPS[1:0]	Prescaler
00	Fosc/64
01	Fosc /32

T1PS[1:0]: Timer1 分频器选择位

T1PS[1:0]	Prescaler
00	Fosc/12
01	Fosc
10	Fosc/96
11	reserved

8.4 波特率发生器

8.4.1 串行接口的模式 1 和 3

8.4.1.1 当 BRGS = 0 (在 AUX 寄存器):

(8) 当 T1PS[1:0] = 00

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{osc}}}{32 \times 12 \times (256 - \text{TH1})}$$

(2) 当 T1PS[1:0] = 01

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{osc}}}{32 \times (256 - \text{TH1})}$$

(3) 当 T1PS[1:0] = 10

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{osc}}}{32 \times 96 \times (256 - \text{TH1})}$$

8.4.1.2 当 BRGS = 1 (在 AUX 寄存器)

(4) 当 SRELPS[1:0] = 00

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{osc}}}{64 \times (2^{10} - \text{SREL})}$$

(5) 当 SRELPS[1:0] = 01

$$\text{Baud Rate} = \frac{2^{\text{SMOD}} \times F_{\text{osc}}}{32 \times (2^{10} - \text{SREL})}$$

9. 看门狗定时器

看门狗定时器是一个8位的计数器并会在计数器溢出时产生复位讯号.WDT在噪音.电源干扰,或断电等导致软件死循环或跑飞程序等情况下非常有用.WDT功能可以帮助拥护的软件从不正常的软件状态中恢复正常.WDT不同于通用的8052的计时器0,1,2.为了防止WDT复位,可以通过软件定时清除WDT计数器.当不可预料的复位发生时,用户应该检查WDTC寄存器的WDTF位.在一外部复位后,该看门狗定时器将无效且所有寄存器都清零.

看门狗定时器是一个自由运行的片上RC振荡器(约20KHz). WDT将保持运行就算是系统时钟被关闭(例如,在睡眠状态).在正常的运行或睡眠状态,一个WDT超时(如使能)将导致MCU复位.WDT在正常状态下可随时被使能或失效.请注意WDTC寄存器的WDTE位.默认的WDT超时时间为接近178.0ms (WDTM [3:0] = 0100b).

WDT有可选择的分频.要选择分频,需要正确设置看门控制寄存器(WDTC)的位3~位0(WDTM [3:0])

$$WDTCLK = \frac{20KHz}{2^{WDTM}}$$

$$\text{Watchdog reset time} = \frac{256}{WDTCLK}$$

表 9-1: WDT 超时时间

WDTM [3:0]	Divider (20 KHz RC oscillator in)	Time period @ 20KHz
0000	1	12.8ms
0001	2	25.6ms
0010	4	51.2ms
0011	8	102.4ms
0100	16	204.8ms (default)
0101	32	409.6ms
0110	64	819.2ms
0111	128	1.6384s
1000	256	3.2768s
1001	512	6.5536s
1010	1024	13.10s
1011	2048	26.21s
1100	4096	52.42s
1101	8192	104.85s
1110	16384	209.71s
1111	32768	419.43s

注: RC 振荡器(20 KHz), 大约有 ± 20 % 误差

看门狗一旦开始工作将无法停止.当WDTK寄存器用户可通过对看门狗定时器重置密钥(WDTK)写入55h实现WDT计时器清0.这将会清除8位计数器内容并让计数器重启.看门狗定时器必须定时刷新以防来自可变的重置复位请求信号.

当看门狗定时器溢出时, WDTF标志位将被置1并自动重置复位MCU.该标志位可被软件或外部复位清除.

对WDTE位写入1能使WDT功能使能.在WDTE设为1后.8位的计数器用通过WDTM[3:0]设置好的分频进行计数.

范例 2:

看门狗定时器使能并选择周期中断信号产生时间为 204.8ms

第一步,先确认烧录时看门狗功能已选择使用

第二步,如下

```
MOV TAKEY, #55h
```

```
MOV TAKEY, #0AAh
```

```
MOV TAKEY, #5Ah ; enable WDTC write attribute.
```

```
MOV WDTC, #64h ; Set WDTM [3:0] = 0100b. Set WDTE =1 to enable WDT function  
; and Set CWDTR =1 to enable period interrupt function
```

10. 中断

OB5510提供12个中断源并带有4级优先权。每一中断源都在特殊寄存器(SFR)中有自身的请求标志，每个中断请求信号通过相应的标志及特殊寄存器（SFR）中的IEN0及IEN1中的使能位来独自允许或禁止。

当中断发生时，CPU将会跳转预先设定的地址，如表 10-1 所示，一旦中断开始执行，就只能被更高优先级的中断终止，其中断服务会被来自指令RETI的返回所终结。当一RETI被执行时且中断发生时，处理器将返回此指令，并执行下一条指令。

当中断条件发生时，该处理器通过设置一个标志位来表示，无论中断使能或禁止，每个中断标志都将在每个机器周期中采样一次，随后采样由硬件来检测，当中断被使能时，且采样指出相应中断时，中断请求标志被设置。在随后的指令周期中，中断将被硬件所确定。从而迫使一个LCALL 指向相应的地址向量。

当中断发生时，中断响应将需要不同的时间，这取决于相关的处理器。如一处理器执行的中断服务程序同等或更优先，新的中断将不会启用，在其它情况下，响应时间将取决于当前的指令。以最快的速度回应一个中断是需要7个机器周期，这包括一个检测中断的机器周期和6周期执行LCALL周期。

表 10-1: 中断向量

	Interrupt Request Flags	Interrupt Vector Address	Interrupt Number *(use Keil C Tool)
1	IE0 – 外部中断0	0003h	0
2	TF0 – 定时器0 中断	000Bh	1
3	IE1 – 外部中断 1	0013h	2
4	TF1 – 定时器1 中断	001Bh	3
5	RI/TI – 串行口中断	0023h	4
6	TF2/EXF2 – 定时器2 中断	002Bh	5
7	PWMIF – PWM 中断	0043h	8
8	ADCIF – A/D 转换 中断	0053h	10
9	KBIIF – 键盘 中断	005Bh	11
10	LVIIF – 低压侦测 中断	0063h	12
11	IICIF – IIC 中断	006Bh	13
12	WDTIF– 看门狗 中断	008Bh	17

*参考Keil C用户指南中的有关中断功能使用说明

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
Interrupt											
IEN0	Interrupt Enable 0 register	A8H	EA	-	ET2	ES	ET1	EX1	ET0	EX0	00H
IEN1	Interrupt Enable 1 register	B8H	EXEN2	-	IEIIC	IELVI	IEKBI	IEADC	-	IEPWM	00H
IEN2	Interrupt Enable 2 register	9AH	-	-	-	-	-	-	EWDT	-	00H
IRCON	Interrupt request register	C0H	EXF2	TF2	IICIF	LVIIF	KBIIF	ADCIF	-	PWMIF	00H
IRCON2	Interrupt request register 2	97H	-	-	-	-	-	-	WDTIF	-	00H
IP0	Interrupt priority level 0	A9H	-	-	IP0.5	IP0.4	IP0.3	IP0.2	IP0.1	IP0.0	00H
IP1	Interrupt priority level 1	B9H	-	-	IP1.5	IP1.4	IP1.3	IP1.2	IP1.1	IP1.0	00H
ENHIT	ENHance Interrupt Register Type	E5H	-	ENHIT1	-	ENHIT0	-	-	-	-	07H
INTDEG	External Interrupt Deglitch Register	EEh					INT1DEG[1:0]		INT0DEG[1:0]		00H

符号: IEN0

地址: A8h

7	6	5	4	3	2	1	0	Reset
EA	-	ET2	ES0	ET1	EX1	ET0	EX0	00h

- EA: EA=0 –禁能所有中断
EA=1 –使能所有中断
- ET2: ET2=0 –禁能定时器 2 中断
ET2=1 –使能定时器 2 中断
- ES: ES=0 –禁能串行口中断
ES=1 –使能串行口中断
- ET1: ET1=0 –禁能定时器 1 中断
ET1=1 –使能定时器 1 中断
- EX1: EX1=0 –禁能外部中断 1
EX1=1 –使能外部中断 1
- ET0: ET0=0 –禁能定时器 0 中断
ET0=1 –使能定时器 0 中断
- EX0: EX0=0 –禁能外部中断 0
EX0=1 –使能外部中断 0

符号: **IEN1** 地址: **B8h**

7	6	5	4	3	2	1	0	Reset
EXEN2	-	IEIIC	IELVI	IEKBI	IEADC	IESPI	IEPWM	00H

EXEN2: 定时器 2 重载中断使能位

EXEN2 = 0 –禁能定时器 2 外部重载中断

EXEN2 = 1 –使能定时器 2 外部重载中断

IEIIC: IIC 中断使能位

IEIIC = 0 –禁能 IIC 中断

IEIIC = 1 –使能 IIC 中断

IELVI: 低压侦测中断使能位

IELVI = 0 –禁能低压侦测中断

IELVI = 1 –使能低压侦测中断

IEKBI: KBI 中断使能位

IEKBI = 0 –禁能 KBI 中断

IEKBI = 1 –使能 KBI 中断

IEADC: A/D 转换中断使能位

IEADC = 0 –禁能 ADC 中断

IEADC = 1 –使能 ADC 中断

IESPI: SPI 中断使能位

IESPI = 0 –禁能 SPI 中断

IESPI = 1 –使能 SPI 中断

IEPWM: PWM 中断使能位

IEPWM = 0 –禁能 PWM 中断

IEPWM = 1 –使能 PWM 中断

符号: **IEN2** 地址: **9Ah**

7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	EWDT	-	00H

EWDT: WDT 看门狗中断使能位

EWDT = 0 –禁能 WDT 中断

EWDT = 1 –使能 WDT 中断

符号: IRCON								地址: C0h	
7	6	5	4	3	2	1	0	Reset	
EXF2	TF2	IICIF	LVIIIF	KBIIF	ADCIF	-	PWMIF	00H	

EXF2: 定时器 2 重载中断旗标位. 必须由软件清零

TF2: 定时器 2 中断旗标位. 必须由软件清零

IICIF: IIC 中断旗标位

LVIIIF: 低压侦测中断旗标位

KBIIF: KBI 中断旗标位

ADCIF: A/D 转换中断旗标位

PWMIF: PWM 中断旗标位. 必须由软件清零

符号: IRCON2								地址: 97h	
7	6	5	4	3	2	1	0	Reset	
-	-	-	-	-	-	WDTIF	-	00H	

WDTIF: 看门狗中断旗标位

符号: ENHIT								地址: E5h	
7	6	5	4	3	2	1	0	Reset	
-	ENHIT1	-	ENHIT0	-	-	-	-	07H	

注: 此功能只支持 H 版本以后的 MCU 设置才有效

ENHIT1: 中断 1 触发边沿控制位。

当 ENHIT1 设置为 0 且 IT1 设置为 1 时, 则开启中断 1 触发选项为下降沿触发

当 ENHIT1 且 IT1 皆设置为 1 时, 则开启中断 1 触发选项为上升沿触发

	ENHIT1=0	ENHIT1=1
IT1=0	INT1低电平触发	INT1 低电平触发
IT1=1	INT1下降沿触发	INT1上升沿触发

ENHIT0: 中断 0 触发边沿控制位。

当 ENHIT0 设置为 0 且 IT0 设置为 1 时, 则开启中断 0 触发选项为下降沿触发

当 ENHIT0 且 IT0 皆设置为 1 时, 则开启中断 0 触发选项为上升沿触发

	ENHIT0=0	ENHIT0=1
IT0=0	INT0低电平触发	INT0 低电平触发

IT0=1	INT0下降沿触发	INT0上升沿触发
-------	-----------	-----------

符号: INTDEG						地址: EEh		
7	6	5	4	3	2	1	0	Reset
-	-	-	-	INT1DEG[1:0]	INT0DEG[1:0]			00H

INT1DEG[1:0]: 选择 INT1 deglitch 时间.

00: no deglitch.

01: 5us

10: 10us

11: 15us

INT0DEG[1:0]: 选择 INT0 deglitch 时间.

00: no deglitch.

01: 5us

10: 10us

11: 15us

10.1 优先级配置

所有中断来源相组合于以下组中:

表 10-2: Priority level groups

Groups		
外部中断 0	-	PWM 中断
定时器 0 中断	看门狗中断	SPI 中断
外部中断 1	比较器中断	ADC 中断
定时器 1 中断	-	KBI 中断
串行口中断	-	低压侦测中断
定时器 2 中断	-	IIC 中断

通过对SFR中的IP0或IP1置位或清零，每一组中断都可被独自编程为四个优先级中的一个，如要求相同的优先级被同时接收，一内部轮询序列将依次确定哪个请求被优先服务。

符号: IP0						地址: A9h		
7	6	5	4	3	2	1	0	Reset
-	-	IP0.5	IP0.4	IP0.3	IP0.2	IP0.1	IP0.0	00h

符号: IP1						地址: B9h		
7	6	5	4	3	2	1	0	Reset
-	-	IP1.5	IP1.4	IP1.3	IP1.2	IP1.1	IP1.0	00h

12. 脉宽调制器(PWM)

OB5510提供-1个通道的PWM输出。
中断值为43h.

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
PWM											
PWMC	PWM Control register	B5h	PWMCS[2:0]			-	PWM3 EN	PWM2 EN	PWM1 EN	PWM0 EN	00H
PWMD0H	PWM 0 Data register high byte	BCh	PWMP 0	-	-	-	-	-	PWMD0[9:8]		00H
PWMD0L	PWM 0 Data register low byte	BDh	PWMD0[7:0]								00H
PWMMDH	PWM Max Data register high byte	CEh	-	-	-	-	-	-	PWMMD[9:8]		00H
PWMMDL	PWM Max Data register low byte	CFh	PWMMD[7:0]								FFH

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
PWM											
PWMC	PWM Control register	B5h	PWMCS[2:0]			-	PWM3 EN	PWM2 EN	PWM1 EN	PWM0 EN	00H
PWMD0H	PWM 0 Data register high byte	BCh	PWMP 0	-	-	-	-	-	PWMD0[9:8]		00H
PWMD0L	PWM 0 Data register low byte	BDh	PWMD0[7:0]								00H
PWMD3H	PWM 3 Data register high byte	B3h	PWMP 3	-	-	-	-	-	PWMD3[9:8]		00H
PWMD3L	PWM 3 Data register low byte	B4h	PWMD3[7:0]								00H
PWMMDH	PWM Max Data register high byte	CEh	-	-	-	-	-	-	PWMMD[9:8]		00H
PWMMDL	PWM Max Data register low byte	CFh	PWMMD[7:0]								FFH

符号: **PWMC**

地址: **B5h**

7	6	5	4	3	2	1	0	Reset
PWMCS[2:0]			-	PWM3EN	-	-	PWM0EN	00H

PWMCS[2:0]: PWM 时钟源选择.

PWMCS [2:0]	Mode
000	Fosc
001	Fosc/2
010	Fosc/4
011	Fosc/6
100	Fosc/8
101	Fosc/12
110	Timer 0 overflow

111	Timer 0 external input (P1.2/T0)
-----	----------------------------------

PWM0EN: PWM 通道 0 使能位.

PWM0EN = 1 – PWM 通道 0 使能.

PWM0EN = 0 – PWM 通道 0 使能.

符号: PWMD0H							地址: BCh	
7	6	5	4	3	2	1	0	Reset
PWMP0	-	-	-	-	-	PWMD0[9:8]		00H

符号: PWMD0L							地址: BDh	
7	6	5	4	3	2	1	0	Reset
PWMD0[7:0]								00h

PWM3EN PWM 通道 3 使能位.

PWM3EN = 1 – PWM 通道 3 使能.

PWM3EN = 0 – PWM 通道 3 禁能.

PWMP0: PWM 通道 0 空闲极性选择.

“0” – PWM 通道 0 于空闲时置低.

“1” – PWM 通道 0 于空闲时置高.

PWMD0[9:0]: PWM 通道 0 数值寄存器.

符号: PWMD3H							地址: B3h	
7	6	5	4	3	2	1	0	Reset
PWMP3	-	-	-	-	-	PWMD3[9:8]		00H

符号: PWMD3L							地址: B4h	
7	6	5	4	3	2	1	0	Reset
PWMD3[7:0]								00H

PWMP3: PWM 通道 3 空闲极性选择.

“0” – PWM 通道 3 于空闲时置低.

“1” – PWM 通道 3 于空闲时置高.

PWMD3[9:0]: PWM 通道 3 数值寄存器.

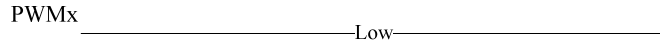
符号: PWMDH							地址: CEh	
7	6	5	4	3	2	1	0	Reset
-	-	-	-	-	-	PWMD[9:8]		00H

符号: PWMDL							地址: CFh	
7	6	5	4	3	2	1	0	Reset
PWMD[7:0]								FFH

PWMMD[9:0]: PWM 最大数值寄存器.

PWM 从 0000h 计数至最大数值寄存器 PWMMD[9:0].当 PWM 计数至与最大数值寄存器数值相等时 PWMMD[9:0]产生溢出.

PWMP_x = 0 & PWMD_x = 00h



PWMP_x = 0 & PWMD_x ≠ 00h



PWMP_x = 1 & PWMD_x = 00h



PWMP_x = 1 & PWMD_x ≠ 00h



$$\text{PWM period} = \frac{\text{PWMMD} + 1}{\text{PWM clock}}$$

$$\text{Leader pulse} = \frac{\text{PWMD}_x}{\text{PWM clock}}$$

13. IIC 功能

这些IIC模块使用SCL（时钟）和SDA（数据）线来联系其它的IIC界面，其速度可以由软件设置特殊寄存器（SFR）中的IICBR[2: 0]，从而使其高达到400KBpS（最大值）。IIC模块可以是主机也可以是从机，提供两个中断（RXIF,TXIF），并有两个地址以作数据传输，它将产生开始，在主机模式中，反复启动和停止信号，在从机模式中，它将检测开始，反复启动和停止信号，最大沟通的长度及连接设备的数目被一个最大400pF的电容所限制

中断向量为6Bh.

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
IIC function											
IICCTL	IIC control register	F9h	IICEN	MSS	MAS	AB_EN	BF_EN	IICBR[2:0]			04H
IICS	IIC status register	F8h	-	MPIF	LAIF	RXIF	TXIF	RXAK	TXAK	RW or BB	00H
IICA1	IIC Address 1 register	FAh	IICA1[7:1]							MATC H1 or RW1	A0H
IICA2	IIC Address 2 register	FBh	IICA2[7:1]							MATC H2 or RW2	60H
IICRWD	IIC Read/Write register	FCh	IICRWD[7:0]								00H
IICEBT	IIC Enaable Bus Transaction	FDh	FU_EN	-	-	-	-	-	-	-	00H

符号: IICCTL

地址: F9h

7	6	5	4	3	2	1	0	Reset
IICEN	MSS	MAS	AB_EN	BF_EN	IICBR[2:0]			04h

IICEN: IIC 模式使能

IICEN = 1, 使能

IICEN = 0, 禁止

MSS: 主\从机模式选择

MSS = 1, 选择主机模式

MSS = 0, 选择从机模式

软件必须在设置其它寄存器之前置位

MAS: 主模式的地址选择（仅在主机模式中）

MAS = 0, 使用 IICA1

MAS = 1, 使用 IICA2

AB_EN: 仲裁失去了使能位. (仅主机模式)

当 AB_EN 位使能, 硬件将检查仲裁丢失位. 一旦发生丢失仲裁, 硬件将返回到空闲状态. 如果此位被清除, 硬件也不会理会仲裁丢失情况. 当多主机与从机连接时需设置此位. 于单主机与从机时清

除此位.

BF_EN: 总线忙碌使能位. (仅主机模式)

当 **BF_EN** 位使能,硬件将不会产生开始信号至总在线直到 **BF** 已置低.此位若被清零将永远产生开始信号至总线当 **MStart** 置高.当多主机与从机连接时需设置此位.于单主机与从机时清除此位.

IICBR[2:0]: 波特率选择 (仅在主机模式) 这里的 **Fosc** 为外部晶体或振荡器的频率, 系统默认为 **Fosc/512** 以使用户选择

IICBR[2:0]	Baud rate
000	$Fosc/32$
001	$Fosc/64$
010	$Fosc/128$
011	$Fosc/256$
100	$Fosc/512$
101	$Fosc/1024$
110	$Fosc/2048$
111	$Fosc/4096$

符号: **IICS**

地址: **F8H**

7	6	5	4	3	2	1	0	Reset
-	MPIF	LAIF	RXIF	TXIF	RXAK	TXAK	RW or BB	00H

MPIF: 停止条件中断旗标

当停止条件发生此位将置高.此位需由软件清零

LAIF: 仲裁失去中断旗标. (只主机模式)

当仲裁失去条件发生此位将置高.此位需由软件清零

RxIF: 数据接收中断标志位, 在 **IICRWD** (IIC 读写数据缓存) 载入一个新的接收数据时被置位; 软件清零该位后, **IIC** 中断标志位 (**IICIF**) 将自动清零。

TxIF: 数据传输中断标志位, 当所有位于转换寄存器中的位数据被传输时, 该位被置位, 来自 **IICRWD**(IIC 读/写数据缓存)的 8 位数据被下载至转换寄存器中; 软件清零该位后, **IIC** 中断标志位 (**IICIF**) 将自动清零。

RxAK: 接收确认位. 置零,这意味着一个确认信号已经收到在完成 8 位数据传输总线上下后.

TxAK: 传输确认位. 当收到完整的 8 位数据, 此位将设置(**NoAck**) 或清除(**Ack**) 并传输到主机显示接收状态. 事实上, 它是被作为一个字节的传输在第 9 位如图 13-1.

RW or BB: 主机模式:

BB:总线忙碌表示位

当侦测到 **scl=0** 或 **sda=0** 或总线产生开始信号,此位将置高.当侦测到停止信号,此位将清零. 此位可由软件清零以使系统回到就绪状态.

从机模式:

RW:从机模式的读取 (接收) 或写(传输) 在 **IIC** 总线. 当此位被清除, 表示从机模式在 **IIC** 总线接收数据.(只从机模式)

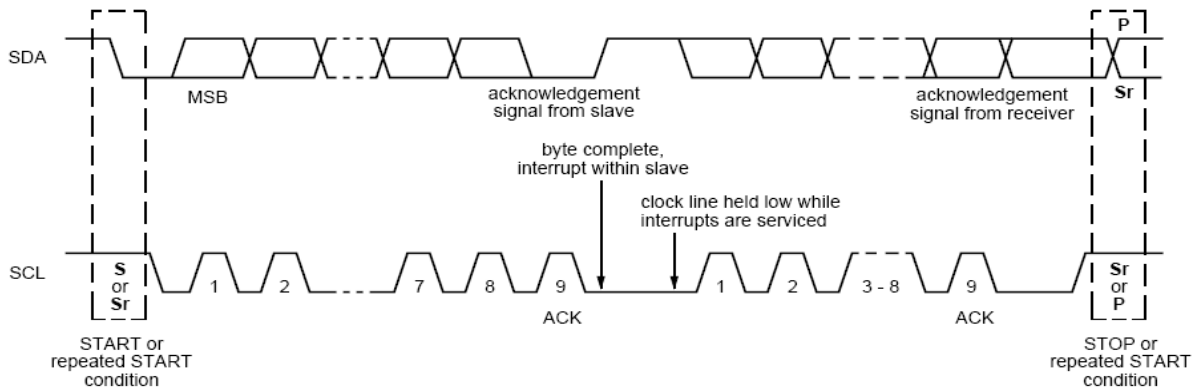
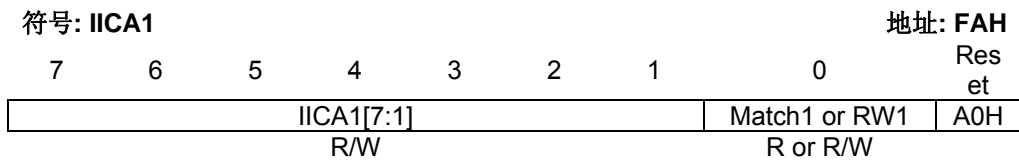


图 13-1: Acknowledgement bit in the 9th bit of a byte transmission



从机模式:

IICA1[7:1]: IIC 地址寄存器

这是为从机模式的第一个 7-bit 的地址，它在一个地址（来自主机模式）接收后被检测

Match1: 当 IICA1 和来自主机模式方的接收地址匹配时，该位将被硬件置位，当 IIC 总线读或写第一笔资料时，该位将被自动清除。

主机模式:

IICA1[7:1]: IIC 地址寄存器

显示要与它通讯的从机的 7 位地址

RW1: 如模块被设置为 Mstart 或 Rstart bit，该位将被作为从机方的 RW 发送，它在 IIC 地址之后的抵 8th 位显示，如图 13-2。它用来告诉从属模式中将来的通信的方向。如设置为 1，模块在主模式中为接收方，如是 0，模块在主模式中为传输方。

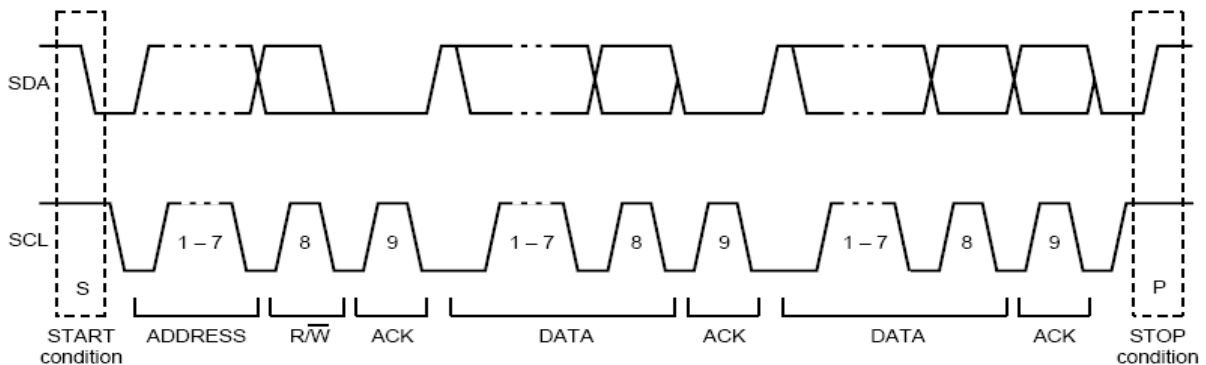
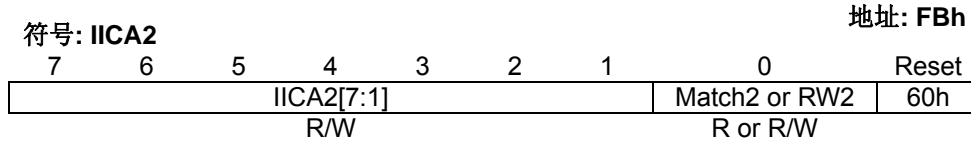


图 13-2: RW bit in the 8th bit after IIC address

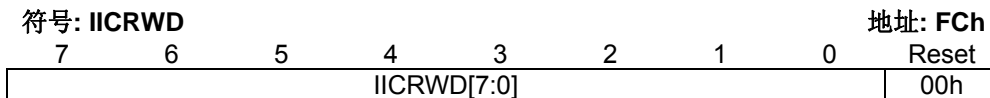

从机模式:
IICA2[7:1]: IIC 地址寄存器

这是为从机模式的第二个 7-bit 的地址, 它在一个地址 (来自主机模式) 接收后被检测

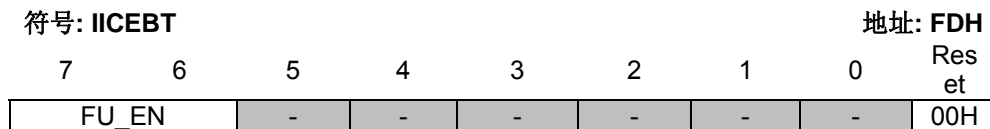
Match2: 当 IICA2 和来自主机模式方的接收地址匹配时, 该位将被硬件置位, 当 IIC 总线读或写第一笔资料时, 该位将被自动清除。

主机模式:
IICA2[7:1]: IIC 地址寄存器

显示要与它通讯的从机的 7 位地址

RW2: 如模块被设置为 **Mstart** 或者 **Rstart bit**, 该位将被作为从机方的 **RW** 发送, 它用来告诉从机模式中将来的通信的方向。如设置为 1, 模块在主模式中为接收方, 如是 0, 模块在主模式中为传输方。

IICRWD[7:0]: IIC 读写缓存

在接收 (读) 模式中, 接收的字节存储于此

 在传输模式中, 该字节通过该 **SDA** 被转换

主机模式:

00: 保留

 01: IIC 模块将致能以便由 **SDA** 及 **SCL** 读写数据。

 10: IIC 模块将于 **SDA** 及 **SCL** 发出开始信号,再送出储存于 **IICA1** 或 **IICA2**(由 **MAS** 位选择)的地址

 11: IIC 模块将于 **SDA** 及 **SCL** 发出停止信号。

FU_EN[7:6] 此两位将自动清零,软件需重复写入。

从机模式:

 01: **FU_EN[7:6]** 此两位于从机模式仅能写入 01.其它值是无效的。

注意:

FU_EN[7:6] 当总线空闲时,于读写数据前此两位需先写入 01;否则,SCL 将被锁住(置低).

FU_EN[7:6] 当读写数据后并收到主机之停止信号时此两位需接着再写入 01. 在传送数据时(从机模式),于此两位写入 01 前待传送数据需先写入 IICRWD.

FU_EN[7:6] 此两位将自动清零,软件需重复写入.

14. 键盘接口(KBI)

键盘接口(KBI) 可被连接一个 $4 \times n$ 矩阵键盘或任何普通的设备. 具有4路输入高或低的可编程的中断能力. 该4路接口可通过外部中断从IDLE和STOP模式中退出. 且输出相互彼此独立, 但共享同一中断向量5Bh.

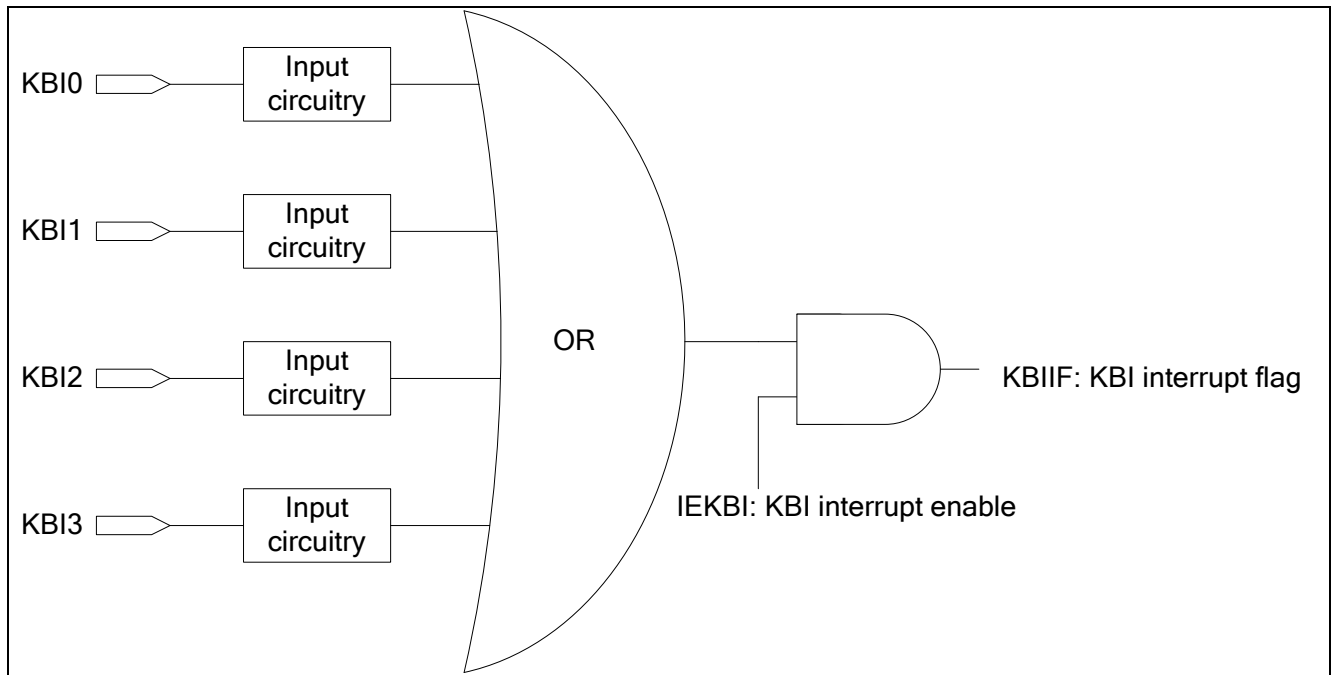


图 14-1: 键盘接口框图

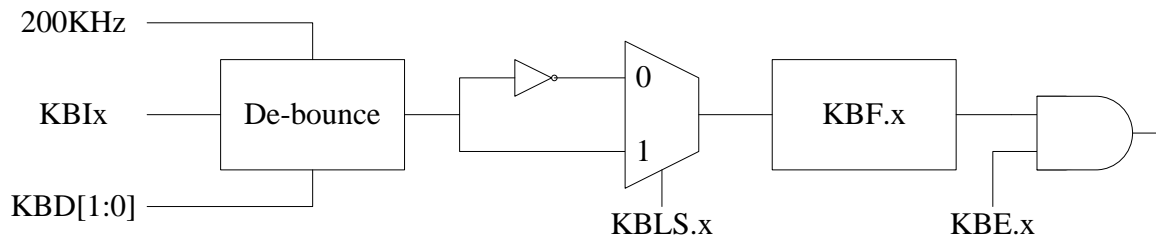


图 14-2: 键盘输入电路

KBI	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
KBI function											
KBLS	KBI level selection	93h	-	-	-	-	KBLS3	KBLS2	KBLS1	KBLS0	00H
KBE	KBI input enable	94h	-	-	-	-	KBE3	KBE2	KBE1	KBE0	00H
KBF	KBI flag	95h	-	-	-	-	KBF3	KBF2	KBF1	KBF0	00H
KBD	KBI De-bounce control register	96h	KBDE N	-	-	-	-	-	KBD1	KBD0	00H

符号: KBLS
地址: 93h

7	6	5	4	3	2	1	0	Reset
-	-	-	-	KBLS.3	KBLS.2	KBLS.1	KBLS.0	00h

KBLS.3: KBI3 电位选择位

0: 使能 KBI3 低电平检测.

1: 使能 KBI3 高电平检测.

KBLS.2: KBI2 电位选择位

0: 使能 KBI2 低电平检测.

1: 使能 KBI2 高电平检测.

KBLS.1: KBI1 电位选择位

0: 使能 KBI1 低电平检测.

1: 使能 KBI1 高电平检测.

KBLS.0: KBI0 电位选择位

0: 使能 KBI0 低电平检测.

1: 使能 KBI0 高电平检测.

符号: KBE
地址: 94h

7	6	5	4	3	2	1	0	Reset
-	-	-	-	KBE.3	KBE.2	KBE.1	KBE.0	00h

KBE.3: KBI3 使能位

0: 使能标准 I/O 口.

1: 使能 KBF 寄存器中的 KBF.3 位产生一中断请求.

KBE.2: KBI2 使能位

0: 使能标准 I/O 口.

1: 使能 KBF 寄存器中的 KBF.2 位产生一中断请求.

KBE.1: KBI1 使能位

0: 使能标准 I/O 口.

1: 使能 KBF 寄存器中的 KBF.1 位产生一中断请求.

KBE.0: KBI0 使能位

0: 使能标准 I/O 口.

1: 使能 KBF 寄存器中的 KBF.0 位产生一中断请求.

符号: KBF
地址: 95h

7	6	5	4	3	2	1	0	Reset
-	-	-	-	KBF.3	KBF.2	KBF.1	KBF.0	00h

KBF.3: KBI3 标志位

当 KBI3 检测到一编程水平线时, 该位由硬件置位.

如 KBE.3 同样被置位时将产生一个 KBI 中断请求, 该位须软件清零.

KBF.2: KBI2 标志位

当 KBI2 检测到一编程水平线时, 该位由硬件置位.

如 KBE.2 同样被置位时将产生一个 KBI 中断请求, 该位须软件清零.

KBF.1: KBI1 标志位

当 KBI1 检测到一编程水平线时, 该位由硬件置位.

如 KBE.1 同样被置位时将产生一个 KBI 中断请求, 该位须软件清零.

KBF.0: KBI0 标志位

当 KBI0 检测到一编程水平线时, 该位由硬件置位.

如 KBE.0 同样被置位时将产生一个 KBI 中断请求, 该位须软件清零.

符号: KBD							地址: 96H	
7	6	5	4	3	2	1	0	Reset
KBDEN	-	-	-	-	-	KBD.1	KBD.0	00H

KBDEN: KBI 去反弹使能位. 默认使能.

KBDEN = 0, 使能去反弹功能. 去反弹时间于 KBD [1:0] 选择.

KBDEN = 1, 禁能去反弹功能. KBI 输入口不需要去反弹机制.

KBD[1:0]: KBI 去反弹时间选择. 当 KBDEN = "0", 默认去反弹时间为 320 ms.

KBD[1:0] = 00, 去反弹时间为 320 ms.

KBD[1:0] = 01, 去反弹时间为 160 ms.

KBD[1:0] = 10, 去反弹时间为 80 ms.

KBD[1:0] = 11, 去反弹时间为 40 ms.

15. LVI – 低压侦测中断

中断向量为63h.

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
LVI function											
RSTS	Reset status register	A1h	-	LVRLP INTF	LVRLP F	PDR F	WDTF	SWRF	LVRF	PORF	00H
LVC	Low voltage control	E6h	LVI_EN	-	LVRE	LVIF	-	-	LVIS[1:0]		20H

符号: **RSTS**

地址: **A1h**

7	6	5	4	3	2	1	0	Reset
-	LVRLP INTF	LVRLPF	PDRF	WDTF	SWRF	LVRF	PORF	00H

LVRLPINTF: “内部” 低电压复位旗标.

当 MCU 复位信号由“内部”低电压复位产生时, LVRLPINTF 旗标将由硬件置高.此旗标需由软件清零.

LVRLPF: 低电压复位旗标.(低功率)

当 MCU 复位信号由低电压复位(低功率)产生时, LVRLPF 旗标将由硬件置高.此旗标需由软件清零.

PDRF: 外部复位脚复位旗标.

此旗标于芯片之复位信号是由外部复位脚产生时由硬件自动设置.此旗标可由软件清零

LVRF: 低电压复位旗标.

此旗标于芯片之复位信号是由低电压复位产生时由硬件自动设置.此旗标可由软件清零

PORF: 上电复位旗标.

此旗标于芯片之复位信号是由上电复位产生时由硬件自动设置.此旗标可由软件清零

符号: **LVC**

地址: **E6h**

7	6	5	4	3	2	1	0	Reset
LVI_EN	-	LVRE	LVIF	-	-	LVIS[1:0]		20H

LVI_EN: 低电压中断功能使能位

0: 禁能低电压检测中断功能

1: 使能低电压检测中断功能

LVRE: 外部低电压重置功能使能位.

0: 禁能外部低电压重置功能.

1: 使能外部低电压重置功能.

LVIF: 低电压中断旗标位

LVIS: 低电压中断电压位阶选择:

00: 2.4V

01: 2.6V

10: 3.2V

11: 4.0V

16. 12位模拟数字转换器(ADC)

OB5510提供了12位ADC，数字输出采样的模拟信号放入ADCD[11:0]中，而ADC的中断向量则为53h。

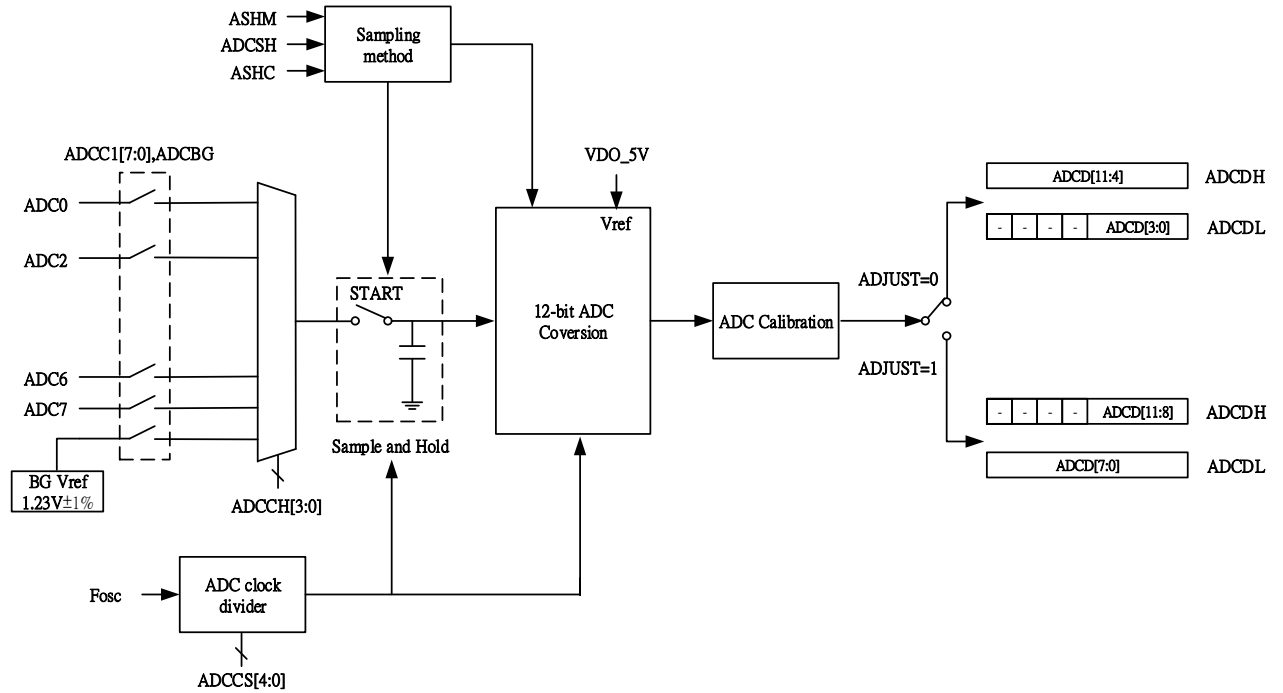


图 16-1: ADC 模拟到数字转换器的操作设置

ADC SFRs 如下所示:

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值	
ADC												
ADCC1	ADC Control 1	ABH	ADC7EN	ADC6EN	-	-	-	ADC2EN	-	ADC0EN	00H	
ADCC2	ADC Control 2	ACH	START	ADJUST	ASHM	ASHC	ADCC[3:0]				00H	
ADCDH	ADC Data High	ADH	ADCDH[7:0]									00H
ADCDL	ADC Data Low	AEL	ADCDL[7:0]									00H
ADCCS	ADC Clock Select	AFH	ADCBGE	-	-	ADCCS[4:0]				00H		
ADCSH	ADC Sample and Hold Time	EFH	ADCSH[7:0]									00H

符号: ADCC1							地址: ABh			
7	6	5	4	3	2	1	0	Reset		
ADC7EN	ADC6EN	-	-	-	ADC2EN	-	ADC0EN	00H		

ADC7EN: ADC 通道 7 使能位。

ADC7EN = 0 –禁能 ADC 通道 7。

ADC7EN = 1 –使能 ADC 通道 7。

ADC6EN: ADC 通道 6 使能位。

ADC6EN = 0 –禁能 ADC 通道 6。

ADC6EN = 1 –使能 ADC 通道 6。

ADC2EN: ADC 通道 2 使能位。

ADC2EN = 0 –禁能 ADC 通道 2。

ADC2EN = 1 –使能 ADC 通道 2。

ADC0EN: ADC 通道 0 使能位。

ADC0EN = 0 –禁能 ADC 通道 0。

ADC0EN = 1 –使能 ADC 通道 0。

符号: ADCC2							地址: ACh	
7	6	5	4	3	2	1	0	Reset
START	ADJUST	ASHM	ASHC	ADCCH[3:0]			00H	00H

START: 当该位被置位时, ADC 将启动单次转换, 转换完成后自动清零.

ADJUST: ADC 数字输出格式调整.

ADJUST = 0: (默认)

ADC 数字输出高字节 ADCD [11:4] = ADCDH [7:0].

ADC 数字输出低字节 ADCD [3:0] = ADCDL [3:0].

ADJUST = 1:

ADC 数字输出高字节 ADCD [11:8] = ADCDH [3:0].

ADC 数字输出低字节 ADCD [7:0] = ADCDL [7:0].

ASHM: ADC 取样与维持时间模式 选择位:

ASHM=0: ADC 取样时间由硬件控制

ASHM=1: ADC 取样时间由软体控制

ASHC: ADC 取样与保持控制位. 该寄存器动作于ASHM=1.

ASHC=0: 禁能ADC取样.

ASHC=1: 使能ADC 取样.

ADCCH[3:0]: ADC 通道选择.

ADCCH [3:0]	通道
0000	0
0010	2
0110	6
0111	7
1000	Bandgap 内部参考电压
Others	-

ADJUST = 0:

符号: **ADCDH**

地址: **ADh**

7	6	5	4	3	2	1	0	Reset
ADCD[11:4]								00H

符号: ADCDL

地址: AEh

7	6	5	4	3	2	1	0	Reset
-	-	-	-	ADCD[3:0]				00H

ADJUST = 1:

符号: ADCDH

地址: ADh

7	6	5	4	3	2	1	0	Reset
-	-	-	-	ADCD[11:8]				00H

符号: ADCDL

地址: AEh

7	6	5	4	3	2	1	0	Reset
ADCD[7:0]								00H

ADCD[11:0]: ADC 数字寄存器.

符号: ADCSH

地址: EFh

7	6	5	4	3	2	1	0	Reset
ADCSH[7:0]								00H

ADCSH [7:0]: ADC 取样与维持时间寄存器, 作用在于调整取样时间。而该寄存器动作于 ASHM=0。

符号: ADCCS

地址: AFh

7	6	5	4	3	2	1	0	Reset
ADCBGE	-	-	ADCCS[4:0]				00H	

ADCBGE: ADC 测量 MCU 内部参考电压使能位。(内部参考电压 1.05V±2%)

0: 禁能。

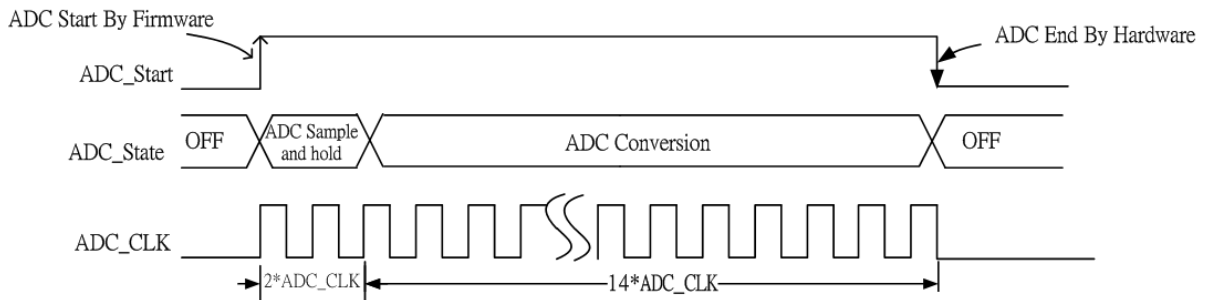
1: 始能。

ADCCS[4:0]: ADC 时钟选择.

$$ADC_Clock = \frac{Fclk}{6 \times (ADCCS[4:0] + 1)}$$

If ASHM=0 , ADCSH=0x00:

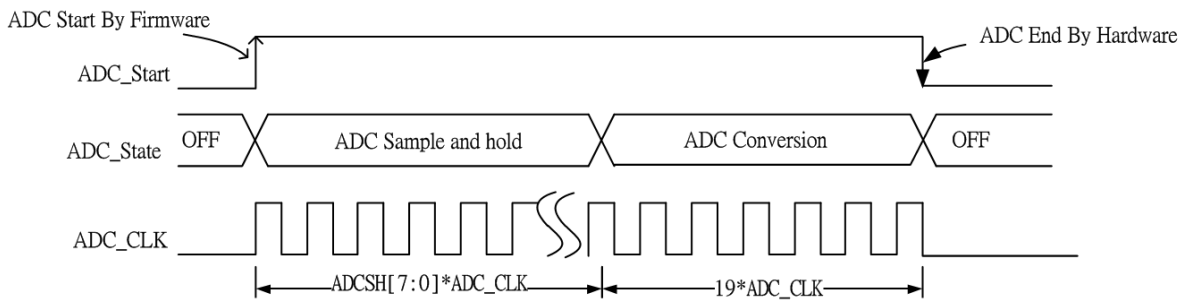
$$ADC_Conversion_Rate = \frac{ADC_Clock}{16}$$



If ASHM=0 , ADCSH≠0x00:

$$ADC_Conversion_Rate = \frac{ADC_Clock}{19 + ADCSH[7:0]}$$

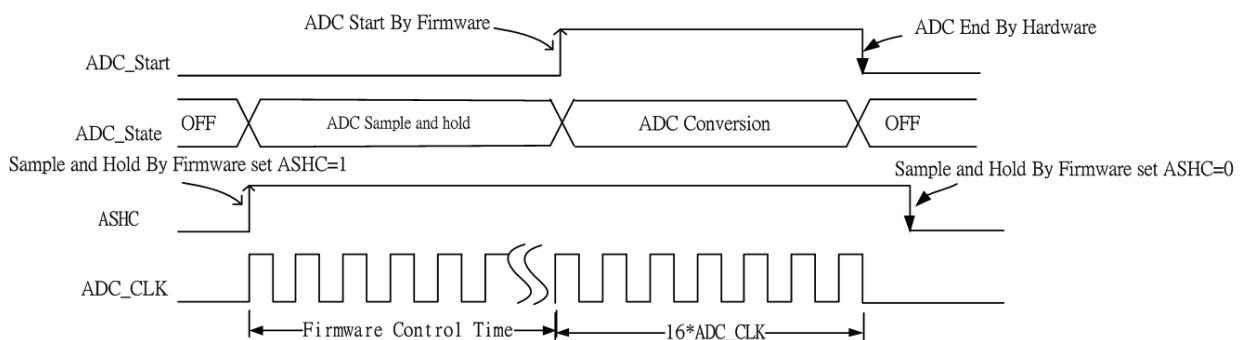
※Sample and hold time by hardware control decisions.



If ASHM=1:

$$ADC_Conversion_Rate = \frac{ADC_Clock}{16} + F/W_Sampling_Time$$

※Sample and hold time by firmware control decisions.



17. 在系统编程(Internal ISP)与EEPROM

OB5510可通过内部硬件电路产生可编程存储器中控制讯号.用户利用可编程存储器中控制寄存器,可编程存储器中地址寄存器和可编程存储器中数据寄存器在不需要将OB5510移出系统的情况下实现ISP功能. OB5510提供了可以实现可编程存储器中编程/芯片擦除/页擦除/保护功能的可编程存储器中控制讯号.用户需要去设计和使用任何OB5510可以输入数据的接口,然后利用ISP服务程序去实现可编程存储器中的编程/芯片擦除/页擦除/保护功能.

17.1 ISP 服务程序

ISP服务程序是由用户自行开发并放置于ISP服务程序区的软件. 用户可依需求决定ISP服务程序的大小. 另外用户需将ISP服务程序烧录至OB5510芯片中才可使用ISP功能.

由于ISP服务程序是由用户自行开发, 它需包括可编程存储器中数据寄存器的烧录及OB5510芯片与上位机的通讯协议. 举例来说, 使用者使用OB5510芯片上之UART接口与上位机作数据之传送及接收, 则OB5510芯片上之ISP服务程序内需有波特率设定, 为避免数据传输错误需有数据总合检查或同位检查或其它错误检查机制.

于正常动作模式或空闲模式可启动 ISP服务程序, 于停止模式则不行.

17.2 锁定位(N)

锁定位N有两个功能:一是配置服务程序的大小,另一个则是通过可编程存储器中擦除功能锁定ISP服务程序的空间.

ISP服务程序地址位从\$1C00到\$1FFF. 可被分为N*128字节(N=0-8). 当N为0时没有ISP功能.所有的8KB字节可编程存储器中都被用来做程序存储.当N为1时,ISP服务程序占用128字节,剩余的7.875K字节可编程存储器中可被用做程序存储.ISP服务程序的最大空间为1K字节(当N为8时). 在这样的配置下,可用的程序存储空间为7K 字节.

在N定下来后,OB5510会从上往下(从最顶端到地址\$1FFF)保存ISP服务程序空间.ISP服务程序的开始地址位于\$1x00 (x为任意数字,以N为准,如表 17-1所示). 请参考章节3.1有关ISP服务程序与程序内存之关系图.

锁定位N功能不同于可编程存储器中保护功能,可编程存储器中擦除功能可以擦除除ISP服务程序空间外的所有可编程存储器中记忆体.如可编程存储器中没有被保护,ISP服务程序的内容一直可被读.如可编程存储器中被保护,所有的可编程存储器中程序空间包括ISP服务程序空间不能被读.

表 17-1 ISP code area.

N	ISP service program address
0	No ISP service program
1	128 bytes (\$1F80h ~ \$1FFFh)
2	256 bytes (\$1F00h ~ \$1FFFh)
3	384 bytes (\$1E80h ~ \$1FFFh)
4	512 bytes (\$1E00h ~ \$1FFFh)
5	640 K bytes (\$1D80h ~ \$1FFFh)
6	768 K bytes (\$1D00h ~ \$1FFFh)
7	896 K bytes (\$1C80h ~ \$1FFFh)
8	1.0 K bytes (\$1C00h ~ \$1FFFh)

ISP 服务程序配置于 N*128 字节 (N= 0 ~ 8)

17.3 对 ISP 服务程序编程

在锁定位N被设置并ISP服务程序被编程后,ISP服务程序记忆体被自动的保护(锁定).锁定位N有它自己的编程/擦除时序.这不同于可编程存储器中记忆体的编程/擦除时序.因此,被锁定的ISP服务程序不可被可编程存储器中擦除功能所擦除.如客户需要擦除锁定的ISP服务程序,他只可以通过烧写器实现.当OB5510在系统时,客户不可改变ISP服务程序.

17.4 启动 ISP 服务程序

要启动ISP服务程序就是要在ISP服务程序的开始地址装载程序计数(PC)并执行它.有四种实现的方法:

- (6) 空复位.用首地址为空(\$0000=#0FFH)的硬件复位会在 ISP 服务程序的开始地址装载 PC.硬件复位包括内部(电源或重启)和外部重置端口复位.
- (7) 执行 JUMP 指令可在 ISP 服务程序的开始地址装载 PC
- (8) 通过硬件设置进入 ISP 服务程序.用户可通过设置当硬件复位时, P1.6'为低', OB5510 会强行进入 ISP 服务程序.硬件复位包括内部(电源或重启)和外部键盘复位.在应用系统设计时,用户因该对 P1.6 在复位时间内防止 OB5510 进入 ISP 服务程序.
- (9) 通过硬件设置进入 ISP 服务程序,P1.1 在硬件复位时会侦测到 2 个时钟讯号. 硬件复位包括内部(电源或重启)和外部阻容复位

在硬件复位周期内, 硬件可以侦测P1.6/P1.1状态.如果他们满足了上面的其中一个条件,芯片将会自动跳转到ISP模式.在ISP服务程序被执行后,用户需要对OB5510进行复位,通过硬件复位或WDT,或是'JUMP'至地址\$0000来实现重启程序.

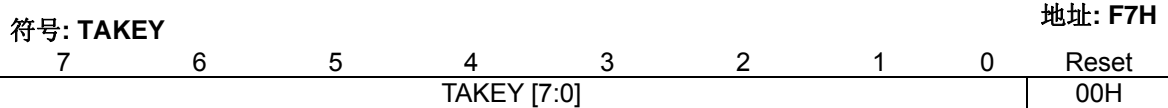
为了用户的不同应用情况,这里有6种不同的进入机制.这个进入方法可以在烧路器或ISP中选取.

- (10) 首地址为空例如\$0000 = 0xFF. 且由内部重置信号触发.
- (11) 首地址为空例如\$0000 = 0xFF. 且由外部重置信号触发.
- (12) P1.6 = 0. 且由内部重置信号触发.
- (13) P1.6 = 0. 且由外部重置信号触发.
- (14) P1.1 输入 2 个时钟. 且由内部重置信号触发.

(15) P1.1 输入 2 个时钟. 且由外部重置信号触发.

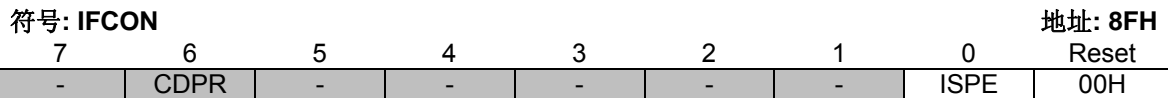
17.5 ISP 寄存器– TAKEY, IFCON, ISPFAH, ISPFAL, ISPFD and ISPFC

符号	描述	地址	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	重置值
ISP function											
TAKEY	Time Access Key register	F7h	TAKEY [7:0]								00H
IFCON	Interface Control register	8Fh	-	CDPR	-	-	-	-	-	ISPE	00H
ISPFAH	ISP Address – High register	E1h	ISPFAH [7:0]								FFH
ISPFAL	ISP Address – Low register	E2h	ISPFAL [7:0]								FFH
ISPFDH	ISP High Data register	EBh	ISPFDH[7:0]								FFH
ISPFDL	ISP Low Data register	E3h	ISPFDL[7:0]								FFH
ISPFC	ISP Control register	E4h	EMF1	-	EMF3	EMF4	-	ISPF.2	ISPF.1	ISPF.0	00H

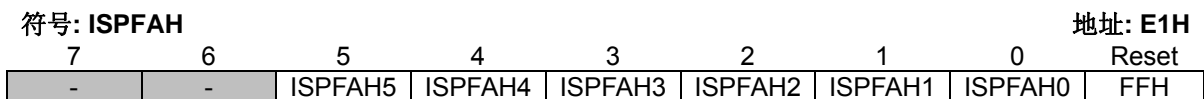


ISP使能位(ISPE)默认为只读,软件必须连续的对TAKEY寄存器写三个特定值55h,AAH,和5Ah,使ISPE位可写.这是:

```
MOV TAKEY, #55h
MOV TAKEY, #0AAh
MOV TAKEY, #5Ah
```



位 0(ISPE) of IFCON 是 ISP 使能位,用户可通过设置 ISPE 位为 1 使能 OB5510 的全部 ISP 功能,通过设置 ISPE 为 0 来禁止全部 ISP 功能.ISPE 的作用像一个加密匙.用户可禁止所有的 ISP 功能以保护软件程序不被意外的擦除.ISP 寄存器 ISPFAH,ISPFAL,ISPFD,与 ISPFC 默认为只读.软件必须将 ISPE 位设为 1 以使上述 4 个寄存器为可写.



ISPFAH [5:0]: 使用 ISP 功能之地址高字节

符号: **ISPFAL** 地址: **E2H**

7	6	5	4	3	2	1	0	Reset
ISPFAL7	ISPFAL6	ISPFAL5	ISPFAL4	ISPFAL3	ISPFAL2	ISPFAL1	ISPFAL0	FFH

ISPFAL [7:0]: 使用 ISP 功能之地址低字节

ISPFAL 与 ISPFALH 提供了 14 位可编程存储器中记忆体地址用做 ISP 功能. 这个可编程存储器中记忆体地址不应该包括 ISP 服务程序空间地址. 若 ISPFALH & ISPFAL 寄存器控制的 14 位可编程存储器中记忆体地址覆盖了 ISP 服务程序空间的地址. 这个可编程存储器中程序/页擦除等 ISP 功能的执行将会无效.

符号: **ISPFDH** 地址: **EBH**

7	6	5	4	3	2	1	0	Reset
ISPFDH7	ISPFDH6	ISPFDH5	ISPFDH4	ISPFDH3	ISPFDH2	ISPFDH1	ISPFDH0	FFH

ISPFDH [7:0]: 使用 ISP 功能之数值.

ISPFDH 寄存器提供 ISP 功能中所需之 8 位数据寄存器.

符号: **ISPFDL** 地址: **E3H**

7	6	5	4	3	2	1	0	Reset
ISPFDL7	ISPFDL6	ISPFDL5	ISPFDL4	ISPFDL3	ISPFDL2	ISPFDL1	ISPFDL0	FFH

ISPFDL [7:0]: ISP data for ISP function.

ISPFDL 寄存器提供 ISP 功能中所需之 8 位数据寄存器.

符号: **ISPF** 地址: **E4H**

7	6	5	4	3	2	1	0	Reset
EMF1	-	EMF3	EMF4	-	ISPF[2]	ISPF[1]	ISPF[0]	00H

EMF1: 进入机制 (1) 标志位, 由复位信号清除. (仅读)

EMF3: 进入机制 (3) 标志位, 由复位信号清除. (仅读)

EMF4: 进入机制 (4) 标志位, 由复位信号清除. (仅读)

ISPF [2:0]: ISP 功能选择位.

ISPF[2:0]	ISP function
000	1-byte program ROM
001	2-byte program ROM
010	Chip protect
011	Write EEPROM
100	Read EEPROM
101	Write option
110	Read option
111	reserved

选择的 ISP 功能在软件对 ISPFC 寄存器写入数据时将会执行一次。

范例. 可编程存储器中地址: \$XYMN

将对地址\$XY00 to \$XYFF 执行页抹除

要实现芯片擦除 ISP 功能,OB5510 将擦除除 ISP 服务程序外的所有可编程存储器中程序记忆体. 要实现芯片保护 ISP 功能,OB5510 内容将被读为#00H.

ISPF[2:0]=000 (1-byte program ROM)

EX1: 将对程序记忆体地址\$1004H 执行字节写, 写入#22H

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah           ; enable ISPE write attribute
MOV IFCON, #01H          ; enable ISP function
MOV ISPFAH, #10H         ; set flash address-high, 10H
MOV ISPFAL, #04H         ; set flash address-low, 04H
MOV ISPFDL, #22H         ; set flash data to be programmed, data = 22H
MOV ISPFC, #00H          ; start to program #22H to the flash address $1004H
```

ISPF[2:0]=000 (1-byte program ROM)

EX2: 将对程序记忆体地址\$1005H 执行字节写, 写入#33H

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah           ; enable ISPE write attribute
MOV IFCON, #01H          ; enable ISP function
MOV ISPFAH, #10H         ; set flash address-high, 10H
MOV ISPFAL, #05H         ; set flash address-low, 05H
MOV ISPFDL, #33H         ; set flash data to be programmed, data = 33H
MOV ISPFC, #00H          ; start to program #33H to the flash address $1005H
```

ISPF[2:0]=001 (2-byte program ROM)

EX3: 将对程序记忆体地址\$1004H 执行字节写, 写入#3322H

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah           ; enable ISPE write attribute
MOV IFCON, #01H          ; enable ISP function
MOV ISPFAH, #10H         ; set flash address-high, 10H
MOV ISPFAL, #04H         ; set flash address-low, 04H
MOV ISPFDL, #22H         ; set flash data to be programmed, data = 22H
MOV ISPFDH, #33H         ; set flash data to be programmed, data = 33H
MOV ISPFC, #01H          ; start to program #3322H to the flash address $1004H
```

ISPF[2:0]=011 (EEPROM write)

EX4: 将对 EEPROM 地址\$0004H 执行字节写, 写入#22H

```
MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah           ; enable ISPE write attribute
MOV IFCON, #01H          ; enable ISP function
```

```

MOV ISPFAH, #00H      ; set flash address-high, 00H
MOV ISPFAL, #04H      ; set flash address-low, 04H
MOV ISPFDL, #22H      ; set flash data to be programmed, data = 22H
MOV ISPFC, #03H       ; start to program #22H to the EEPROM address $0004H
  
```

ISPF[2:0]=011 (EEPROM write)

EX5: 将对 EEPROM 地址\$0005H 执行字节写，写入#33H

```

MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah       ; enable ISPE write attribute
MOV IFCON, #01H       ; enable ISP function
MOV ISPFAH, #00H      ; set flash address-high, 00H
MOV ISPFAL, #05H      ; set flash address-low, 05H
MOV ISPFDL, #33H      ; set flash data to be programmed, data = 33H
MOV ISPFC, #03H       ; start to program #33H to the EEPROM address $0005H
  
```

ISPF[2:0]=100 (EEPROM read)

EX6: 将对 EEPROM 地址\$0004H 执行字节读取

```

MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah       ; enable ISPE write attribute
MOV IFCON, #01H       ; enable ISP function
MOV ISPFAH, #00H      ; set flash address-high, 00H
MOV ISPFAL, #04H      ; set flash address-low, 04H
MOV ISPFC, #04H       ; start to program read the EEPROM address $0004H to
ISPFDL
MOV A, ISPFDL         ; read ISPFDL and store it in Accumulator
  
```

ISPF[2:0]=101 (write option)

EX7: 将对 option 地址\$0004H 执行字节写，写入#22H

```

MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah       ; enable ISPE write attribute
MOV IFCON, #01H       ; enable ISP function
MOV ISPFAH, #00H      ; set flash address-high, 00H
MOV ISPFAL, #04H      ; set flash address-low, 04H
MOV ISPFDL, #22H      ; set flash data to be programmed, data = 22H
MOV ISPFC, #05H       ; start to program #22H to the flash address $0004H
  
```

ISPF[2:0]=101 (write option)

EX8: 将对 option 地址\$0005H 执行字节写，写入#33H

```

MOV TAKEY, #55h
MOV TAKEY, #AAh
MOV TAKEY, #5Ah       ; enable ISPE write attribute
MOV IFCON, #01H       ; enable ISP function
MOV ISPFAH, #00H      ; set flash address-high, 00H
MOV ISPFAL, #05H      ; set flash address-low, 05H
MOV ISPFDL, #33H      ; set flash data to be programmed, data = 33H
MOV ISPFC, #05H       ; start to program #33H to the flash address $0005H
  
```

ISPF[2:0]=110 (read option)

EX9: 将对 option 地址\$0004H 执行字节读取

```

MOV TAKEY, #55h
  
```

```
MOV TAKEY, #AAh
MOV TAKEY, #5Ah           ; enable ISPE write attribute
MOV IFCON, #01H           ; enable ISP function
MOV ISPFAH, #00H          ; set flash address-high, 00H
MOV ISPFAH, #04H          ; set flash address-low, 04H
MOV ISPFC, #06H           ; start to program read the option address $0004H to
ISPFDL
MOV A, ISPFDL              ; read ISPFDL and store it in Accumulator
```

EX10: 将对程序记忆体地址地址\$1004H 执行字节读取

```
CLR A
MOV DPTR, #1004h          ;
MOVC A, @A+DPTR           ; read the data of address $1004 and store it in
Accumulator
```

EX11: 将对程序记忆体地址地址\$1005H 执行字节读取

```
CLR A
MOV DPTR, #1005h          ;
MOVC A, @A+DPTR           ; read the data of address $1005 and store it in
Accumulator
```

极限参数

Symbol	Description	Min.	Typ.	Max.	Unit.	Remarks
VDD	直流供电电压	-0.3V		+20V		
TS	储存温度	-55		150	°C	
TA	工作环境温度	-40	25	85	°C	Ambient temperature under bias
OTP_trig	Over temperature protection		150		°C	
OTP_release	Over temperature protection release		120		°C	

电气特性

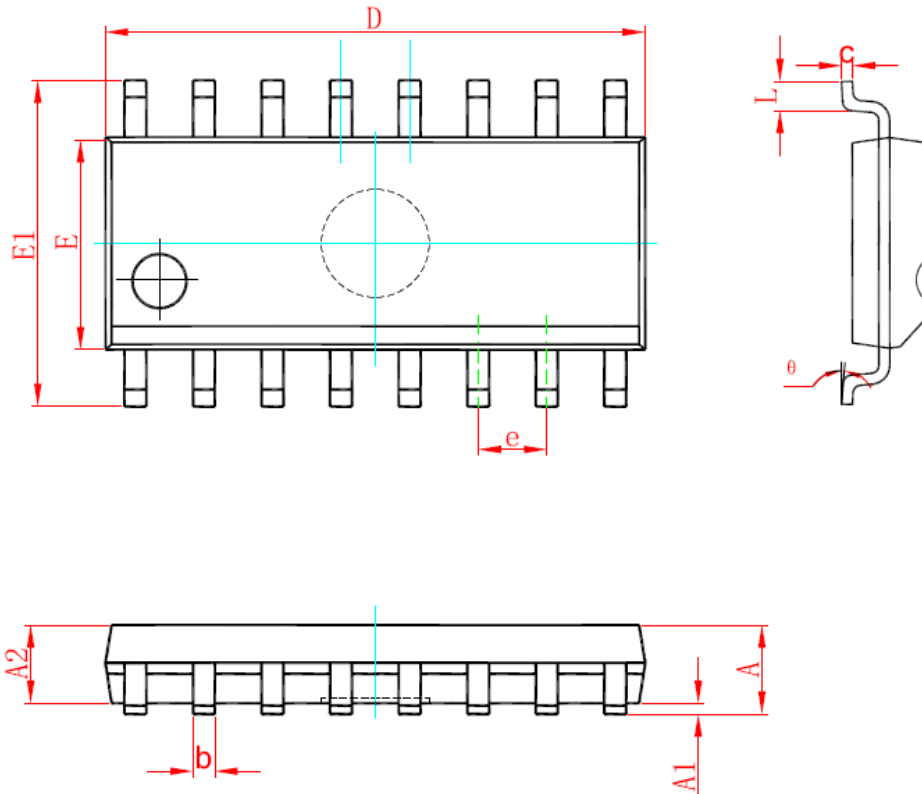
 高压直流部分电气特性 (GND=0V, T_A=25°C)

Symbol	Valid	Min	Typical	Max	Units	Conditions
工作电压	VDD	7.5	9	12	V	
芯片启动电压	VDD(ON)	6.5	7	7.5	V	VDD 电压从0V 开始上升, 芯片开始工作
芯片关闭电压	VDD(OFF)	5.1	5.4	5.7	V	VDD 从9V 开始下降, 芯片停止工作
LDO 输入电压	V _{IN}	6.5		12	V	
LDO 输出电压	V _{LDO}	4.9	5.0	5.1	V	I _{OUT} =200mA, T _A =25°C
LDO 输出电压 (含温度变化)	V _{LDO}	4.8	5.0	5.2	V	I _{OUT} =200mA, T _A = -40°C 至 85°C
LDO 输出电流1	I _{OUT1}	-	-	200	mA	LDO 输入电压大于6.5V
LDO 输出电流2	I _{OUT2}	200	-	-	uA	LDO 进入低功耗模式
LDO 纹波抑制比	RR	-	50	-	db	F=100Hz
LDO 负载调整率	ΔV _{O1}	-	60	-	mV	V _{IN} =9V, 1mA < I _{OUT} < 200mA
LDO 线性调整率	ΔV _{O2}	-	30	-	mV	I _{OUT} =100mA, 8V < V _{IN} < 12V
LDO 静态功耗1	I _{SS1}		300	400	uA	LDO 高功耗模式
LDO 静态功耗2	I _{SS2}		30	40	uA	LDO 低功耗模式
LDO 建立时间	T _{SET}		20	-	mS	LDO 外接100uF 电容, 8V < V _{IN} < 12V
OD0-OD4灌电流能力	I _{OUT}	80	-	-	mA	V _{in} =9V, VOL=GND+0.8V
AN5V/9V 输入分压电阻比值误差	R	-1	0	1	%	标准分压为 V _{IN} /3.6, V _{LDO} /2
最大功耗	ΔP _{MAX}	-	-	0.6	W	LDO 的最大功耗

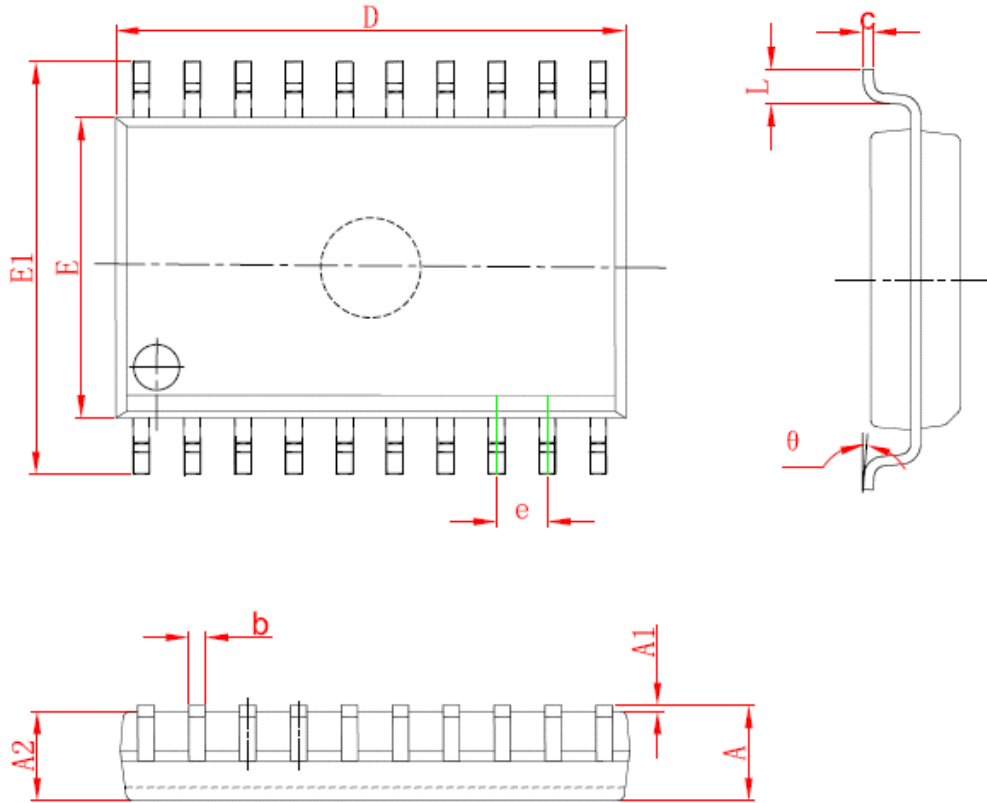
MCU DC电气特性:

TA = -40°C to 85°C; VIN=9V, if not otherwise noted; Vcc = VDO = 5.0V

Symbol	Parameter	Valid	Min	Typ	Max	Units	Conditions
VIL1	Input Low-voltage	Px.x	-0.5		0.2Vcc	V	Vcc=5V
VIH1	Input High-voltage	Px.x	0.8Vcc		Vcc + 0.5	V	
IOL	Sink Current	Px.x	40			mA	
IOL1	Sink Current	COM0~COM6	100			mA	
IOH	Source Current (Pull-Up)	Px.x	0.36			mA	
IOH1	Source Current (Push-Pull)	Px.x	20			mA	
IOH2	Source Current (Push-Pull)	COM0~COM6	100			mA	
IIL	Logic 0 Input Current	Px.x,SCL,SD A			-75	uA	Vi= 0.45V
ITL	Logical Transition Current	Px.x,SCL,SD A			-650	uA	Vi= 2.0V
ILI	Input Leakage Current	Px.x,SCL,SD A			±10	uA	0.45V<Vi<Vcc
CIO	Pin Capacitance				10	pF	1MHz, Ta= 25°C
ICC	MCU Power Supply Current				12	mA	VCC=5V, 25 °C
					11	mA	VCC=5V, 25 °C
					8	uA	VCC=5V, 25 °C

封装尺寸
SOP16


Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	1.350	1.750	0.053	0.069
A1	0.100	0.250	0.004	0.010
A2	1.250	1.650	0.049	0.065
b	0.310	0.510	0.012	0.020
c	0.100	0.250	0.004	0.010
D	9.800	10.400	0.386	0.409
E	3.800	4.040	0.150	0.159
E1	5.800	6.240	0.228	0.246
e	1.270 (BSC)		0.050 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°

SOP20:


Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	2.350	2.650	0.093	0.104
A1	0.100	0.300	0.004	0.012
A2	2.050	2.550	0.081	0.100
b	0.310	0.510	0.012	0.020
c	0.200	0.330	0.008	0.013
D	12.520	13.000	0.493	0.512
E	7.400	7.600	0.291	0.299
E1	10.000	10.610	0.394	0.418
e	1.27 (BSC)		0.05 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°

改版记录

文件名	日期	改版信息

IMPORTANT NOTICE

RIGHT TO MAKE CHANGES

On-Bright Electronics Corp. reserves the right to make corrections, modifications, enhancements, improvements and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

WARRANTY INFORMATION

On-Bright Electronics Corp. warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with its standard warranty. Testing and other quality control techniques are used to the extent it deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

On-Bright Electronics Corp. assumes no liability for application assistance or customer product design. Customers are responsible for their products and applications using On-Bright's components, data sheet and application notes. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

LIFE SUPPORT

On-Bright Electronics Corp.'s products are not designed to be used as components in devices intended to support or sustain human life. On-bright Electronics Corp. will not be held liable for any damages or claims resulting from the use of its products in medical applications.

MILITARY

On-Bright Electronics Corp.'s products are not designed for use in military applications. On-Bright Electronics Corp. will not be held liable for any damages or claims resulting from the use of its products in military applications.