



AD16H02

用户手册

版本号: V1.0.0.8

版权所有©

西安恩狄集成电路有限公司

本资料内容为西安恩狄集成电路有限公司在现有数据资料基础上编制而成，本资料中所记载的实例以正确的试用方法和标准操作为前提，使用方在应用该等实例时应充分考虑外部诸条件，西安恩狄集成电路有限公司不担保或确认该等实例在使用方的适用性、适当性或完整性，西安恩狄集成电路有限公司亦不对使用方使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。文档中所有涉及到第三方软件的，请自行购买正版软件，因第三方软件版权问题涉及到的一切后果，与西安恩狄集成电路有限公司无关。基于使本资料的内容更加完善等原因，西安恩狄集成电路有限公司保留未经预告的修改权。

西安恩狄集成电路有限公司

地 址：陕西省西安市高新区高新一路 19 号思安大厦 501

地 址：深圳市龙岗区坂田国际中心 A 栋 21 层 2112 室

电 话：+ (86 29) 88322766 网 站：www.admicrochip.com

微信号：恩狄 ADUC



版本修订记录

Bin	Version	Change List	Owner	Data
1	1.0.0.0	初版	ZhaoWei	2021.9.29
2	1.0.0.1	增加了 ADC 超时处理	ZhaoWei	2021.11.16
3	1.0.0.2	更新订购信息	ZhaoWei	2021.12.8
4	1.0.0.3	1) 增加 H 桥参数描述 2) 增加烧录引脚说明章节 1.5、在板烧录说明描述 3) 修改 PWOFF 模式唤醒后描述	Aaron	2022.1.19
5	1.0.0.4	增加 SLEEP 模式下的使用注意事项	ZhaoWei	2022.3.3
6	1.0.0.5	1) 更新 SMCR 寄存器读写操作配置 2) 更新引脚排列、引脚说明 3) 更新烧录引脚说明	Aaron	2022.6.1
7	1.0.0.6	1、修改字体格式； 2、修改“FVRPGA”描述； 3、修改 CMP_PCH[3:2]的 00 通道； 4、ADVREF[1:0]的 01 修改为 VDD；10 修改为 Reserved； 5、ADCS[2:0]111 和 011 修改为 Reserved； 6、修改绝对最大额定值描述； 7、修改操作条件描述； 8、修改电气特性 9、用户配置字 RESSEL 说明修改； 10、修改部分笔误；	Hanley	2023.4.14
8	1.0.0.7	1、修改 IO 章节寄存器的描述 2、修改 PWM2CON2 的寄存器描述	Mjw	2023.4.17
9	1.0.0.8	1、修改定时器章节的描述 2、修改中断章节寄存器的描述 3、修改电气参数章节的描述	Mjw	2023.6.16

目 录

版本修订记录	1
1 产品简介	5
1.1 功能特性	5
1.2 系统结构	6
1.3 引脚排列	6
1.4 引脚说明	7
1.5 烧录引脚说明	8
2 中央处理器	9
2.1 处理寄存器	9
2.2 存储器结构	12
2.2.1. 程序存储器	12
2.3 用户配置字	14
2.4 寄存器列表	17
3 系统时钟	19
4 复位和电源电压检测	20
4.1 上电复位计数器(PWRT Power-up Reset Timer)	20
4.2 振荡启动计数器(OST Oscillator Start-up Timer)	20
4.3 复位顺序	20
4.4 LVD 检测	23
5 OTP 操作描述	24
5.1 OTP 的相关寄存器	24
6 GPIO	26
6.1 IO 工作模式	26
6.2 上下拉电阻开漏模式	27
6.3 模拟 IO 寄存器	28
7 定时器	29
7.1 Timer0/WDT & Prescaler/BUZZER	29
7.1.1. Timer0 计数/定时器	29
7.1.2. 使用内部时钟: 定时模式	29
7.1.3. 使用外部时钟/内部 32K 时钟: 计数模式	29
7.1.4. 看门狗定时器 (WDT)	29
7.1.5. Prescaler (预置器)	29
7.1.6. BUZZER (BUZZER 输出)	30
7.1.7. TIMER0 的相关寄存器	30
7.2 Timer1 16 位定时/计数器	32
7.2.1. 使用外部时钟/内部 32K 时钟/内部 256K 时钟/比较器输出: 计数模式	32
7.2.2. TIMER1 特殊事件	33
7.2.3. 1 路 16 位 PWM 输出	33
7.2.4. 定时器 TIMER1 相关的寄存器	33

7.3	TIMER2 定时器.....	36
7.3.1	TIMER2 的工作原理.....	36
8	捕获/比较/脉冲宽度调制模块(CCP).....	39
8.1	CCP 相关寄存器.....	39
8.2	捕捉模式.....	41
8.3	比较模式.....	42
8.4	PWM 模式.....	42
8.5	PWM 相关寄存器.....	43
9	驱动电路 H 桥.....	46
9.1	互补式输出.....	46
9.2	高压控制寄存器.....	46
9.3	PWM 调制.....	49
9.4	死区时间.....	50
9.5	互补式输出控制防呆电路.....	52
9.6	极性控制.....	52
9.7	高压输出驱动.....	52
9.8	高压输出驱动寄存器.....	53
9.8.1	高压输出驱动应用电路.....	54
9.8.2	高压输出驱动防呆电路.....	54
9.8.3	高压输出驱动热保护.....	55
9.8.4	高压输出回读电路.....	55
10	省电模式 (SLEEP).....	59
10.1	睡眠唤醒.....	60
10.1.1	SMCR 的相关寄存器.....	60
11	中断方式.....	62
11.1	外部中断.....	62
11.2	定时器中断.....	62
11.3	Port B 输入改变中断.....	62
11.4	中断的相关寄存器.....	63
12	固定参考电压 (FVR).....	66
12.1	FVR 的相关寄存器.....	66
13	运算放大器和比较器.....	67
13.1	运放 OP0.....	67
13.2	运放 OP1.....	69
13.3	比较器 CMP.....	70
13.3.1	CMP 的相关寄存器.....	71
14	AD 转换.....	72
	ADC 的相关寄存器.....	73
15	DA 转换.....	76
	DAC 的相关寄存器.....	77
16	十六进制转化为十进制 (HeADecimal Convert to Decimal HCD).....	79

17 绝对最大额定值	81
18 典型的电路应用	82
19 操作条件	83
20 电气特性	84
直流特性	84
IO(直流特性)	84
FVR 84	
LVT 85	
WDT OSC(交流特性)	85
ADC 85	
LDO 86	
SLEEP	86
21 封装信息	87
SOP16	87
SOP8	87
22 订购信息	88

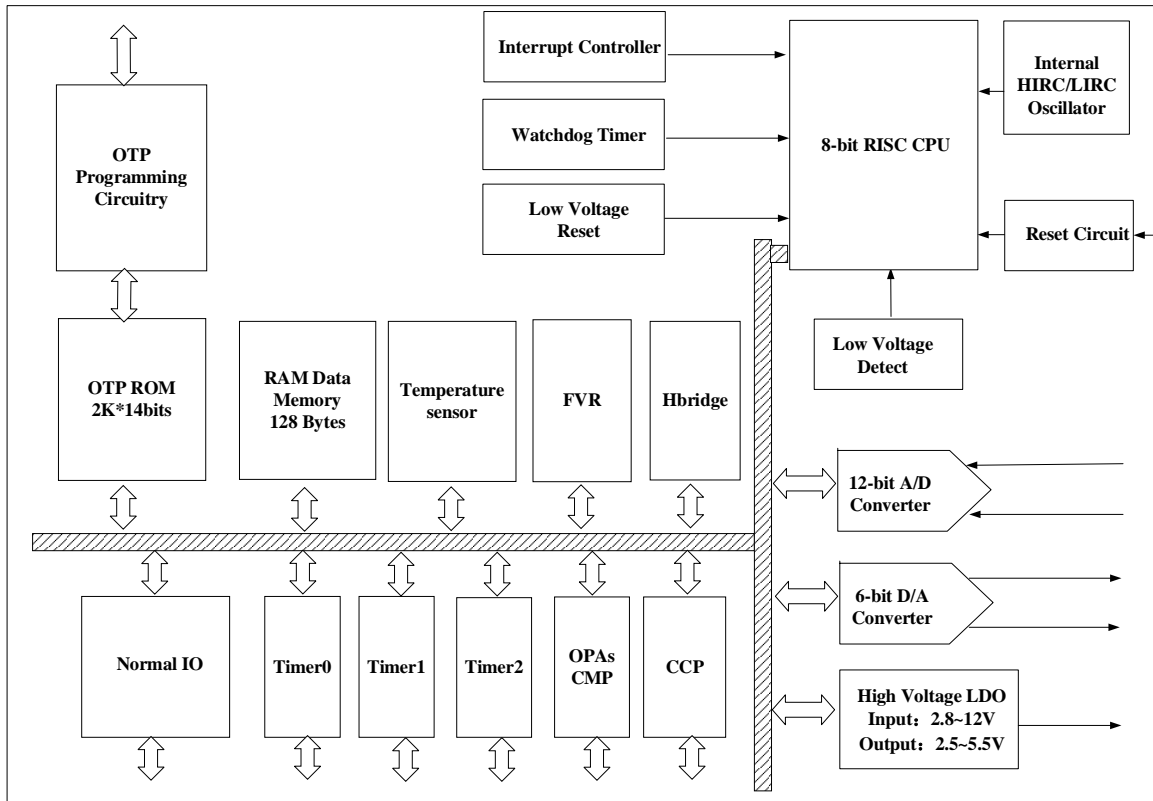
1 产品简介

1.1 功能特性

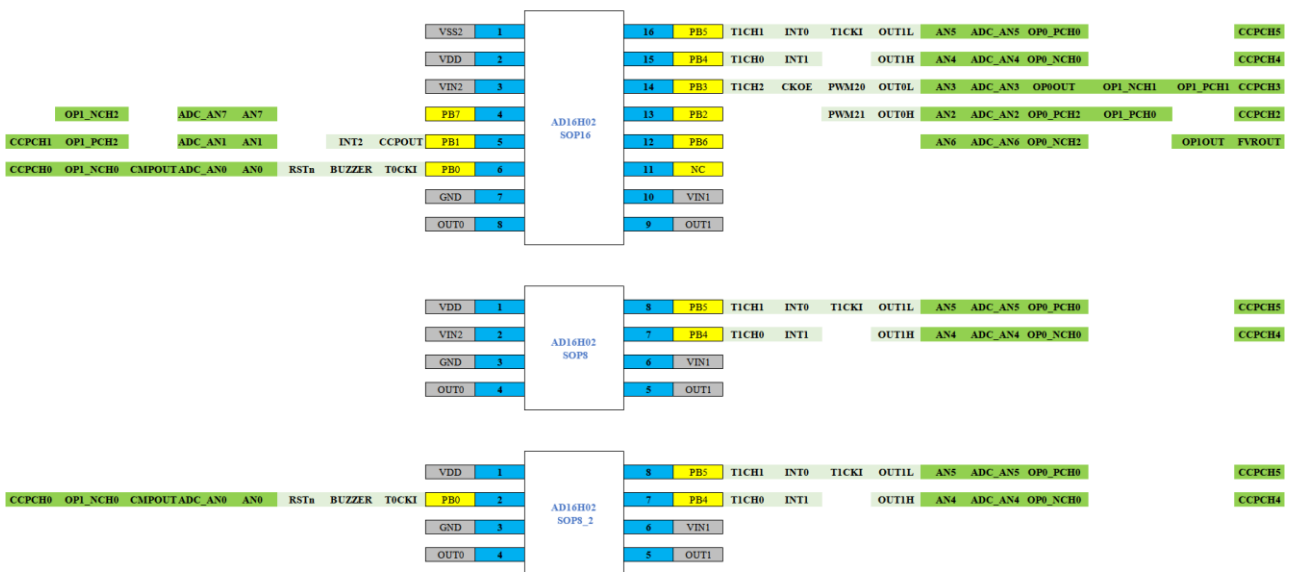
- 程序空间为 2k*14bit (2k Bytes)，SRAM 为 128 字节，8 级硬件堆栈。
- 支持 2T 和 4T 的 CPU 运算模式。
- 内部集成高压的高性能低噪声 LDO 电源，支持 3.3V~12V 电压输入，LDO 可输出 20mA 电流，支持 2.5V、3.0V、3.3V、5.0V 四档电压输出。
- CPU 双时钟模式，高速可选 16M/8M/4M/2M/1M；低速时钟可选 256K/128K/64K/32K/16K，作为 Fcpu 时钟；通过寄存器可实时切换 CPU 时钟；有 ACTIVE、IDLE、PWSAVE、PWOFF 四种工作模式。
- TIMER0，8 位定时/计数器，可选 Fcpu、32K、T0CKI、CMPOUT 为工作时钟，支持 BUZZER 模式。
- TIMER1，16 位定时/计数器，有多个时钟输入选择，可以工作在异步时钟模式，支持捕捉模式的输出。
- TIMER2，12 位定时/计数器，支持中断，时钟源为 Fsys、CCP 模块，支持捕捉、比较、PWM 模式。
- 内部集成驱动电流 1A 的 H 桥，可使用 TIMER2_CCP_PWM 直接驱动，12V/0.1A 时内阻为 600mΩ，内置 H 桥电流传感器，构成电流环调节；支持过温、过压、过流保护。
- 12bit 高精度的 ADC，支持多个 IO 引脚采集，同时支持内部模拟信号采集，内置温度传感器。
- LVR 提供 7 种电压选择，支持低电压中断，同时也支持高压中断，高压中断可选择自动强制 PB0 输出指定电平，可实现电源电压采集。
- 2 组低失调和高增益运算放大器，灵敏度高并能有效抑制误触发，可用于微伏信号放大。
- 通过 IOSR/IOSW 指令实现 IO 寄存器读写。
- 超强程序加密算法，保证芯片程序内容唯一性。
- 有可靠的保证使得程序代码不被读出。
- 只有 41 个单字指令，除跳转指令为两个周期指令以外其余为单周期指令。
- 上电复位计数器 (PWRT) 和振荡启动计数器 (Oscillator Start-up Timer OST)。
- 内部振荡器集成了一个看门狗保证了可靠的操作。
- 一类双向输入输出 I/O 口 PortB；通过编程控制 I/O 端口的上拉/下拉、开漏等状态；上拉电阻支持两类选择；上电默认 PB0 上拉电阻打开；PB1 下拉电阻打开。
- 中断：
 - ✧ 三个内部计数/定时器中断源
 - ✧ 三个外部 IO 管脚中断源：INT 管脚
 - ✧ PortB 的输入改变中断源
 - ✧ CCP 中断源
 - ✧ 低电压、高电压 LVD 中断源
 - ✧ 运放输出电压比较中断源
 - ✧ ADC 中断
 - ✧ OP 中断

- 通过外部中断、PortB 中断、LVD 中断、TIMER 中断、CCP 比较模式、WDT 溢出和外部复位实现睡眠模式唤醒。
- VIN 工作电压范围：3.3V~12V；VDD 工作电压范围：2.5V~ 5.5V，CPU 工作的最高频率为 16MHz。

1.2 系统结构



1.3 引脚排列



1.4 引脚说明

管脚名	功能名	功能描述
PB0	T0CKI	TIMER0 的外部时钟输入
	BUZZER	BUZZ 的输出
	RSTn	芯片外部复位引脚
	CMPOUT	CMP 的输出
	OP1_NCH0	OP1 的负端输入
	ADC0	ADC 的模拟通道
	AN0	模拟通道
	CCPCH0	CCP 的输入通道
PB1	INT2	外部中断
	CCPOUT	CCP 的输出
	ADC1	ADC 的模拟通道
	AN1	模拟通道
	CCPCH1	CCP 的输入通道
	OP1_PCH2	OP1 的正端输入
PB2	OUT0H	OUT0 上臂输出信号
	PWM21	PWM21 的输出引脚
	OP0_PCH2	OP0 的正端输入
	OP1_PCH0	OP1 的正端输入
	CKOE	系统的时钟输出
	ADC2	ADC 的模拟通道
	AN2	模拟通道
	CCPCH2	CCP 的输入通道
PB3	OUT0L	OUT0 下臂输出信号
	PWM20	PWM20 的输出引脚
	T1CH2	TIMER1 的脉宽检测输入信号
	OP0OUT	OP0 的输出
	ADC3	ADC 的模拟输入通道
	AN3	模拟通道
	CCPCH3	CCP 的输入信号
PB4	INT1	外部中断
	T1CH0	TIMER1 的脉宽检测输入信号
	OP0_NCH0	OP 的负端输入
	OUT1H	OUT1 的上臂输出信号
	ADC4	ADC 的模拟通道输入
	AN4	模拟通道
	CCPCH4	CCP 的输入信号
PB5	INT0	外部中断
	T1CKI	TIMER1 的外部时钟输入

管脚名	功能名	功能描述
	T1CH1	TIMER1 的脉宽检测输入信号
	OP0_PCH0	OP0 的正端输入
	OUT1L	OUT1 的下臂输出信号
	ADC5	ADC 的模拟通道输入
	AN5	模拟通道
	CCPCH5	CCP 的输入信号
PB6	FVROUT	FVR 的输出
	OP1OUT	OP1 的输出
	OP0_NCH2	OP0 的负端输入
	ADC6	ADC 的模拟通道输入
	AN6	模拟通道
PB7	ADC7	ADC 的模拟通道输入
	OP1_NCH2	OP1 的负端输入
VSS		芯片电源负极
VDD		LDO 输出(MCU 电源输入)
VIN2		LDO 输入 (HVCC)
OUT0		H 桥左臂输出
OUT1		H 桥右臂输出
GND		H 桥电源负极 (SOP-16)、芯片和 H 桥电源共地 (SOP-8)
VIN1		H 桥电源正极 (HVCC)

1.5 烧录引脚说明

AD16H02烧录引脚说明			
序号	烧录引脚	烧录信号	在板烧录说明
1	VCC、VIN1	VCC_4.5	16H02 在烧录时，烧录器会提供两种电源，一种是 4.5V 电压用于建立连接与参数校准，一种供 9.3V 电压，用于烧录 HEX 程序。因此，要确保电路中，与这两种电源相连的其它器件，要能承受住相应的电压；
2	VSS2、GND	GND	
3	VIN2/VIN1(SOP8-2)	VCC_9.3	
4	PB4	DAT	PB4 Pin 是 Data，该引脚不建议连接 0.1uf 以上的电容。不允许连接强上拉/下拉电阻。
5	PB5	CLK	PB5 Pin 是 CLK，该引脚走线不允许出现电容，不允许出现强上拉/下拉电阻。
6	OUT1	CKOE	OUT1 Pin 是 PWM 输出，频率约 1KHz 左右，该引脚走线不允许出现电容，不允许出现强上拉/下拉电阻。

注：VCC_9.3 注意事项 封装：SOP8-2;接 VIN1; 其余封装接 VIN2。

2 中央处理器

2.1 处理寄存器

INDF (间接寻址寄存器)

Bit	Name	Description	Attribute	Reset
7:6	Reserved			
5:0	FSR [5:0]	通过 FSR 访问数据区(不是一个实际的物理地址)	R/W	00 0000

INDF 不是一个实际的物理地址，间接寻址时 INDF 通过 RAM 选择寄存器（FSR）来访问其所指向的地址。间接寻址读操作直接读地址 00h(FSR=“0”），间接寻址不能对 INDF 直接进行写操作（尽管有些状态会发生改变）。FSR 的 5-0 位可以用来选择 64 个寄存器（地址:00h~3Fh）。

例 2.1:间接寻址

- 地址 38 内容为 10h
- 地址 39 内容为 0Ah
- 将 38 写入 FSR 中
- 通过 A 读 INDF 返回 10h
- FSR 加 1 (@FSR=39h)
- 通过 A 读 INDF 返回 0A h

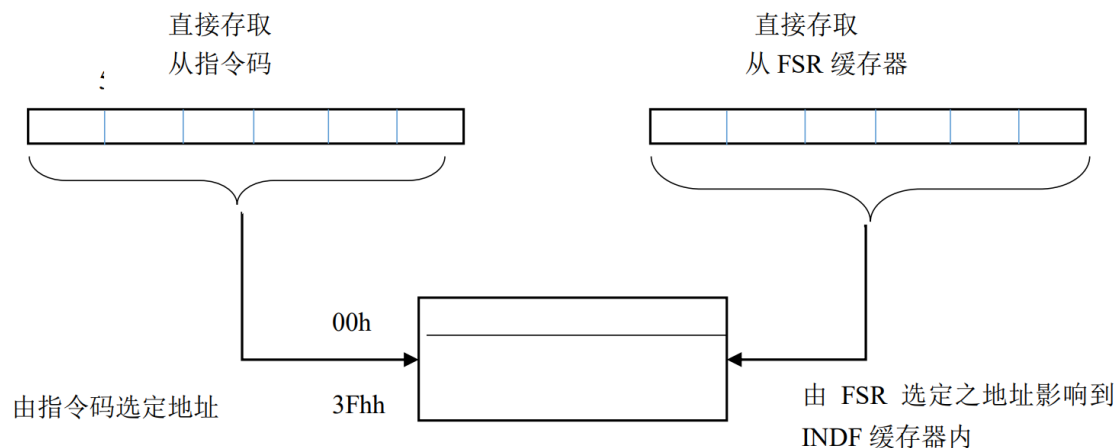


图 2.1 直接/间接存取

PCL (Low Bytes of Program Counter) & Stack
地址: 0x02

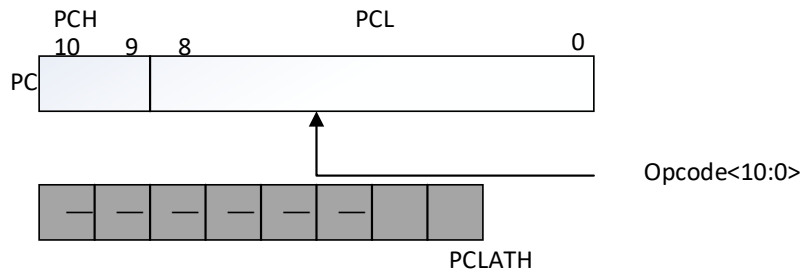
Bit	Name	Description	Attribute	Reset
7:0	PCL [7:0]	PC 低 8 位	R/W	0

AD16H02 的 PC 指针和堆栈的位数为 11 位，堆栈有 8 级，低位的 PC 指针为 PCL 寄存器，该寄存器是可读写的，高位的 PC 指针为 PCH 寄存器，该寄存器包含 PC<10:9>位，该寄存器不能直接读写。PCH 寄存器的改变是通过 PCLATH 寄存器来实现的。每一条指令执行的时候它的 PC 指针包含下一条指令的操作地址。指令没有改变 PC 内容时、在每一个指令周期 PC 指针自动加 1。

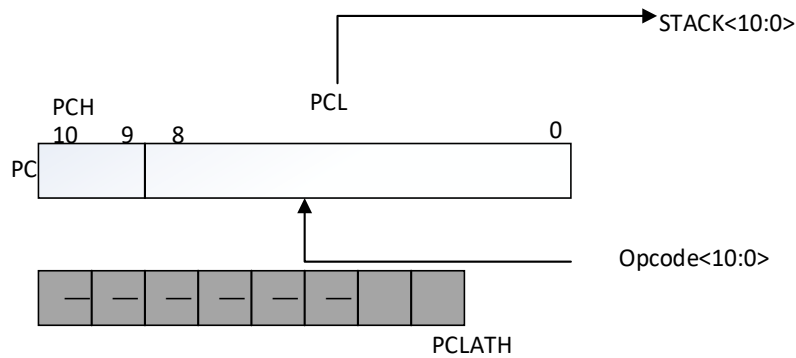
➤ 对于 GOTO 指令有 PC<10:0>，PCL 映射成 PC<8:0>，PCLATH 不变。

- 对于 CALL 指令有 $PC<10:0>$ ，下一条指令地址被推进堆栈，PCL 映射成 $PC<8:0>$ ，PCLATH 不变。
- 对于 RETLW, RETFIE, RETURN 指令有 $PC<10:0>$ ，PC 的内容更改为出栈信息，PCL 映射成 $PC<8:0>$ ，PCLATH 不变。对于其他指令，PCL 就是目标信息， $PC<8:0>$ 的内容就是指令地址或。不管怎样， $PC<10:9>$ 来源于 PCLATH<1:0>位 (PCLATH PCH)，PCLATH 不会改变，从而 PCH 不会改变。

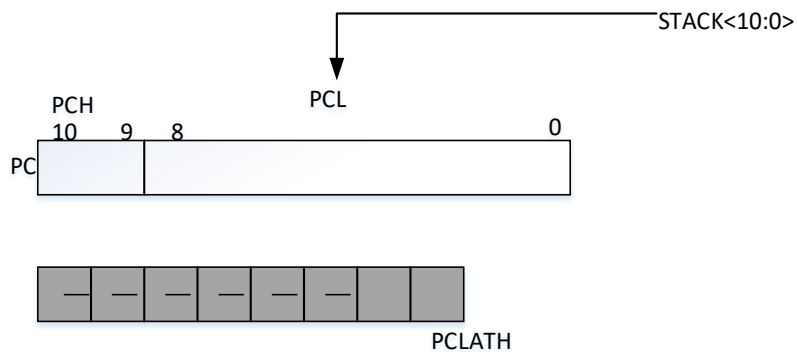
1、GOTO 指令



2、CALL 指令



3、RETLW, RETFIE, RETURN 指令



4、以 PCL 为目的的指令

注释 1. PCLATH 只有在 PCL 内容是目标地址才有效，当 PCL 是运算结果时，PCLATH 不起作用。

STATUS (状态字寄存器)

地址：0x03

Bit	Name	Description	Attribute	Reset
7	RST	定义系统复位类型位. 1:Port B 脚位变化唤醒 SLEEP 0:其他类型唤醒 SLEEP.	R/W	0
6	GPR	通用寄存器读/写位	R/W	0
5	RP0	BANK1 寄存器访问 1: 设定访问 BANK1 寄存器, 也可以通过 IOSR、IOSW 指令访问 BANK1 寄存器 0: 设定访问 BANK0 寄存器	R/W	1
4	TO	看门狗溢出标志位 1: 当系统上电时或执行“CLRWDT”或 SLEEP 指令后 0: 看门狗定时器溢出	R/W	1
3	PD	系统休眠标志位 1: 当系统上电时或执行“CLRWDT”指令后 0: 当执行“SLEEP”指令后	R/W	0
2	Z	零标志位 1: 算术或逻辑运算结果为“0”时 0: 算术或逻辑运算结果不为“0”时	R/W	0
1	DC	辅助进位/借位标志.(低四位向高四位进位/借位标志) ADDWF, ADDLW 1: 低 4 位有进位 0: 低 4 位无进位 SUBWF, SUBLW 1: 低 4 位无借位 0: 低 4 位有借位	R/W	0
0	C	进位标志 ADDWF, ADDLW 1: 最高位有进位 0: 最高位无进位 SUBWF, SUBLW 1: 无借位 0: 有借位 注释：减法是通过将 2 的补第二个操作数的执行。旋转 (RRF, RLF) 指令, 该位装载高或低位源寄存器位。	R/W	0

指令执行以后可能会影响 STATUS 寄存器的 Z、DC、C 标志位, 则不能直接对这三个标志位进行写操作, 这些标志位的设置由 MCU 的逻辑自动完成。同时, TO 和 PD 位也是不能通过指令直接改变写操作。因此, 与 STATUS 作为目标寄存器的指令后, 结果可能会与预期的不同。例如:运行 CLRFS STATUS 将把 STATUS 的高三位置零和 Z 标志位置 1 同时该寄存器的内容如下

0	0	0	u	u	1	u	u
---	---	---	---	---	---	---	---

u 表示为指令执行前后该位没有发生改变

FSR (间接寻址指针)

地址: 0x04

Bit	Name	Description	Attribute	Reset
7:0	FSR [7:0]	用来选择访问间接寻址时目标寄存器地址	R/W	0x00

PCLATH (PC 指针高位缓冲区)

地址: 0x0A

Bit	Name	Description	Attribute	Reset
7:3	Reserved			
2:0	PCLATCH [2:0]	PC 高 3 位	R/W	000

2.2 存储器结构

AD16H02 存储器包含程序存储器和数据存储器。

2.2.1. 程序存储器

AD16H02 有一个 11 位 PC 指针能访问 2K×14 的存储空间。

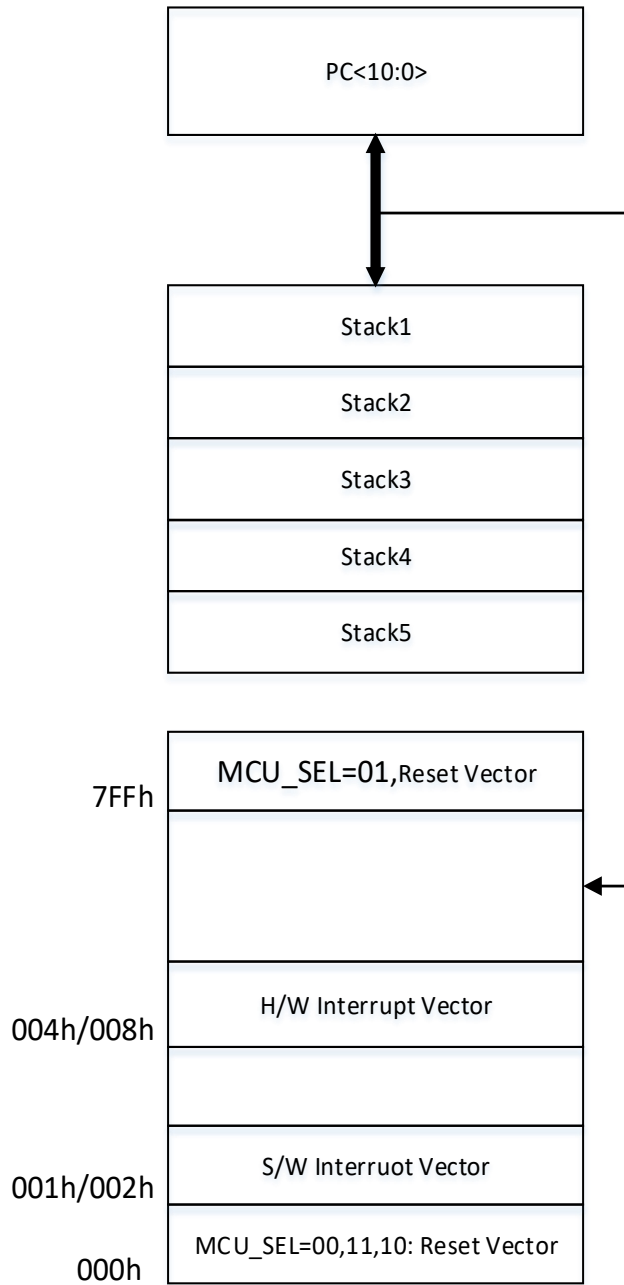
AD16H02 的复位地址根据 MCU_SEL 选择可配置为 000H 和 7FF。

H/W 中断向量地址根据 MCU_SEL 选择可配置为 004H, 008H, S/W 中断向量地址 001H 和 002H。

AD16H02 的 CALL/GOTO 能指向在同一个程序页面（一个程序页面为 2K）的所有存储空间。

程序存储器分布图和堆栈结构

MCU_SEL	11/00	01	10
MCU	ADCPU2	ADCPU1	ADCPU0
RESET	0x0	0xFFF	0x0
H/W	0x4	0x8	0x8
S/W	NONE	0x2	0x1



AD16H02

2.3 用户配置字

注:不经过配置的配置字默认为 1;

配置字 1

位	名称	说明
3:0	LVR [3:0]	LVR 电压选择 0000: Reserved 0001: Reserved 0010: Reserved 0011: 2.3V 0100: 2.6V 0101: 2.8V 0110: 3.2V 0111: 3.4V 1000: 3.6V 1001: 4.1V 其他保留
6:4	FINTOSC	内部 RC 振荡器频率选择 111: 1:1 分频 110: 1:2 分频 101: 1:4 分频 100: 1:8 分频 011: 1:16 分频
7	FCPUS	指令周期选择 1: 1 个指令周期为 4 个机器周期 0: 1 个指令周期为 2 个机器周期
9:8	OTP	11: 芯片配置成 2K 大小, 可以写一次, 地址空间 000-7FF, 00: 芯片配置成 1K 大小, 可以写一次, 地址空间 400-7FF, 01: 芯片配置成 1K 大小, 可以写一次, 地址空间 000-3FF,

配置字 2

位	名称	说明
1	RDPIN	IO 作为输出时，读端口方式控制位 1:从管脚读 0:从寄存器读
2	SMTEN	I/O 输入缓冲控制位 1:通过 Schmitt 触发器 0:不通过 Schmitt 触发器
3	RESSEL	端口上/下拉电阻选择 0: 上拉电阻: 30K 下拉电阻: 30K 1: 上拉电阻: 190K 下拉电阻: 300K
4	OSCM	振荡器模式选择 1: 内部高频 RC 振荡器 0: 内部低频 RC 振荡器
6	PROTECT	代码保护选择位 1:代码不加密 EPROM code protection off 0:代码加密 EPROM code protection on
7	OTPDIS	代码禁止读取 1:代码允许读取 0: 代码禁止读取

配置字 3

位	名称	说明
2~0	TWDT	看门狗溢出时间及上电复位时间选择 111: PWRT = TWDT (no Prescaler) = 32.8ms 110: PWRT = TWDT (no Prescaler) = 4ms 101: PWRT = TWDT (no Prescaler) =262.2ms 100: PWRT = TWDT (no Prescaler) =65.6ms 011: PWRT = 486us;TWDT (no Prescaler) =32.8ms 010: PWRT = 486us;TWDT (no Prescaler) = 4ms 001: PWRT = 486us;TWDT (no Prescaler) =262.2ms 000: PWRT = 486us;TWDT (no Prescaler) =65.6ms
3	WDTE	WDT 使能 1: 使能 WDT(开启看门狗需要寄存器中 WDTE 位同时打开) 0: 关闭 WDT

5~4	MCU_SEL	芯片兼容选择 00/11: ADCPU2 01: ADCPU1 10: ADCPU0
6	LVTEN	低压复位使能 1: 禁止低压复位功能 0: 使能低压复位功能
7	RESETE	外部复位使能 1: 屏蔽外部复位功能 0: 使能外部复位功能

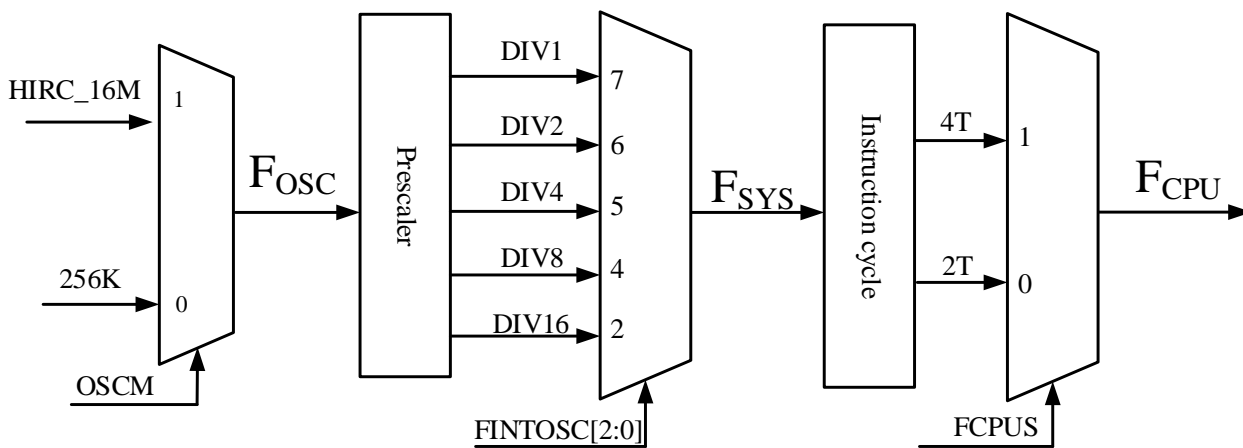
注：使能 WDT 之后，当芯片产生复位时，H 桥模块不复位。

2.4 寄存器列表

	Address	MAPEN=0	Address	MAPEN=1	Address
INDF	00H	INDF	80H	INDF	80H
TMR0	01H		81H		81H
PCL	02H	PCL	82H	PCL	82H
STATUS	03H	STATUS	83H	STATUS	83H
FSR	04H	FSR	84H	FSR	84H
	05H		85H		85H
PORTB	06H	TRISB	86H	TRISB	86H
ANASEL	07H	SMTVB	87H	SMTVB	87H
PCON	08H	WUBCON	88H	WUBCON	88H
PIR1	09H	PIE1	89H	INT1EN	89H
PCLATH	0AH	PCLATH	8AH	PCLATH	8AH
T0CON	0BH	T0CON	8BH	T0CON	8BH
TMR1L	0CH	T1CON0	8CH	CMPCON	8CH
TMR1H	0DH	T1CON1	8DH	OP1CON0	8DH
PIE0	0EH	FVRCON0	8EH	OP1CON1	8EH
PIR0	0FH	FVRCON1	8FH	CLKCFG0	8FH
CCPR1L	10H	CCPCON	90H	IAPTRIG	90H
CCPR1H	11H	PWM2CON0	91H	IAPCTRL	91H
CCPR1LH	12H	PWM2CON1	92H	IAPADDRH	92H
DACON	13H	PWM2CON2	93H	IAPADDRL	93H
OPCON0	14H	PWM2CON3	94H	IAPDATH	94H
OPCON1	15H	T2CON	95H	IAPDATL	95H
TMR2L	16H	PR2L	96H	IAPWAIT	96H
TMR2H	17H	PR2H	97H	HBC3	97H
PMS	18H	HVC	98H		98H
PXC	19H	HBC0	99H		99H
DTC	1AH	HBC1	9AH		9AH
POLS	1BH	HBC2	9BH		9BH
ADRESH	1CH	PDBCON	9CH		9CH
ADRESL	1DH	ODBCON	9DH		9DH
ADCON0	1EH	PUBCON	9EH		9EH
ADCON1	1FH	SMCR	9FH		9FH
96Byte SRAM	20H	32Byte	A0H	32Byte	A0H
	21H				
	...				
	...				
	...				
	...				
	...				
	...		BFH		BFH
	...				
	...				
	...				
	...				
	70H	16byte Common	F0H	16byte Common	F0H
	...				
	7DH				
	7EH				
	7FH		FFH		FFH
BANK0		BANK1		BANK1	

bin	address	name	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	reset value	value					
1	0x00	INDF	Addressing this location use contents of FSR to address data memory (not a physical register)									xxxx xxxx	uuuu uuuu				
2	0x01	TMR0	TMR0[7:0]									0000 0000	uuuu uuuu				
3	0x02	PCL	PCL[7:0]									0000 0000/1111 1111	0000 0000/1111 1111				
4	0x03	STATSU	RST	RP1	PR0	TO	PD	Z	DC	C	0001 1xxx	000q quuu					
5	0x04	FSR	FSR[6:0]									-xxx xxxx	-uuu uuuu				
6	0x05											xxxx xxxx	uuuu uuuu				
7	0x06	PORTB	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0		0000 0000	uuuu uuuu					
8	0x07	ANASEL6	ANASEL6	ANASEL5	ANASEL4	ANASEL3	ANASEL2	ANASEL1	ANASEL0		0000 0000	uuuu uuuu					
9	0x08	PCON	WDTE	PBOST	LVDM[1:0]			LVT[3:0]			1000 0000	1000 0000					
10	0x09	PIR1	HBIF	LVDIF	OVPIF	OCPIF	THIF	INT2IF	INT1IF	INT0IF	--0 0000	--0 0000					
11	0x0a	PCLATH	PCH[2:0]									--- -000/--- -11	--- -000/--- -111				
12	0x0b	TOCON	TOOUT	TOCS1	TOCS0	TOSE	PSA	PS2	PS1	PS0	0111 1111	0111 1111					
13	0x0c	TMR1H	Timer1 Register High Byte									0000 0000	uuuu uuuu				
14	0x0d	TMR1L	Timer1 Register Low Byte									0000 0000	uuuu uuuu				
15	0x0e	PIE0	GIE	T2IE	T1IE	ADIE	OP1E	CCP1IE	PBIE	TOIE	0000 0000	0000 0000					
16	0x0f	PIR0	MAPEN	T2IF	T1IF	ADIF	OP1F	CCP1IF	PBIF	TOIF	0000 0000	0000 0000					
17	0x10	CCPR1H	Capture/Compare/PWM Register 1 High Byte									0000 0000	0000 0000				
18	0x11	CCPR1L	Capture/Compare/PWM Register 1 Low Byte									0000 0000	0000 0000				
19	0x12	CCPR1H	CCPR1L[11:8]									--- 0000	--- 0000				
20	0x13	DACON	DACEN	DACS5	DACS4	DAC[3:0]						-0xx xxxx	uuuu uuuu				
21	0x14	OPCON1	OPOUT	LVDST	AN_AVSS	PCH2	PGAEN	OPPOS	CMPMODE	OUTEN	x--- 400	x--- 400					
22	0x15	OPCON	OPON	EX	A1NS	A1O2N	PCH[1:0]		NCH[1:0]		0xxx xxxx	0uuu uuuu					
23	0x16	TMR2L	TMR2L[7:0]									0000 0000	0000 0000				
24	0x17	TMR2H	TMR2H[3:0]									--- 0000	--- 0000				
25	0x18	PMS							PMS3	PMS2	PMS1	PMS0	--- 0000	--- 0000			
26	0x19	PXC	OUTOEN	OUTOLEN	OUTIHEN	OUTILEN	PCIT	PC1B	PC0T	PC0B	--- 0000	--- 0000					
27	0x1a	DTIC	DTCKS1	DTCKS0	DTEN	DTD4	DTD3	DTD2	DTD1	DTD0	0000 0000	0000 0000					
28	0x1b	POLS	OTPSSEL	PTMODE	OUTASMT	OUTBSMT	POLS1T	POLS1B	POLS0T	POLS0B	--- 0000	--- 0000					
29	0x1c	ADRESH	A/D Result Register High Byte									0000 0000	0000 0000				
30	0x1d	ADRESL	A/D Result Register Low Byte									0000 0000	0000 0000				
31	0x1e	ADCON0	ADFM	ADSP	CHS3	CHS2	CHS1	CHS0	GO_DONE	ADON	0000 0000	0000 0000					
32	0x1f	ADCON1	ADVREF1	ADVREF0	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0	0000 0000	0000 0000					
33	0x80	INDF	not a physical register									N/A	N/A				
34	0x81																
35	0x82	PCL	PCL[7:0]									0000 0000	0000 0000				
36	0x83	STATSU	RST	RP1	PR0	TO	PD	Z	DC	C	0001 1xxx	000q quuu					
37	0x84	FSR	FSR[6:0]									-xxx xxxx	-uuu uuuu				
38	0x85											1111 1111	t20n				
39	0x86	TRISB							TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111	
40	0x87	SMTVB							SMTVB5	SMTVB4	SMTVB3	SMTVB2	SMTVB1	SMTVB0	1111 1111	1111 1111	
41	0x88	WUBCON							WUB6	WUB5	WUB4	WUB3	WUB2	WUB1	WUB0	0000 0000	0000 0000
42	0x89	PIE1	HBIE	LVDIE	HBCIE	OCIE	THIE	INT2IE	INT1IE	INT0IE	--- -00	--- -00					
43	0x8a	PCLATH	PCH[2:0]									--- -00/--- -11	--- -00/--- -11				
44	0x8b	TOCON	TOOUT	TOCS1	TOCS0	TOSE	PSA	PS2	PS1	PS0	0111 1111	0111 1111					
45	0x8c	T1CON0	T1CK1	T1CK0	T1CKPS1	T1CKPS0	T1SE	T1SYNC	TMR1CS0	TMR1ON	0000 0000	0000 0000					
46	0x8d	T1CON1	INT2EDGE	INT1EDGE	INT0EDGE	T1EDGE	T1CH0	T1CH1	T1M1	T1M0	--- -00	--- -00					
47	0x8e	FVRCON0							FVROUTEN	FVRPGA	FVREN	--- -00	--- -00				
48	0x8f	FVRCON1							FVR_SEL4	FVR_SEL3	FVR_SEL2	FVR_SEL1	FVR_SEL0	--0 0000	--0 0000		
49	0x90	CCPCON	CCP1CH[3:0]			CCP1M[3:0]							0000 0000	0000 0000			
50	0x91	PWM2CON0	PWM2ACF	PWM2ASF	PWM21OE	PWM20OE	PWM21DLY[3:0]		PWM21POL		PWM20POL	0000 0000	0000 0000				
51	0x92	PWM2CON1										0000 0000	0000 0000				
52	0x93	PWM2CON2	HBFLTEN	FLTEN	ASTART	ACLOSE	OPFLT	IOFLT	PWM20ADPOS	PWM20ADEN	0000 0000	0000 0000					
53	0x94	PWM2CON3	PWMADDLY[7:0]														
54	0x95	T2CON	PWM2CAEN	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS2	TMR2ON	T2CKPS1	T2CKPS0	0000 0000	0000 0000					
55	0x96	PR2L	PR2L[7:0]									1111 1111	1111 1111				
56	0x97	PR2H	PR2H[3:0]									--- 1111	--- 1111				
57	0x98	HVC	HBEN	HBOVPEN	HBOCPEN	HVBEN	HBOTPEN	HBMPEN	HBCMAEN	HBCMBEN	--0 0000	--0 0000					
58	0x99	HBC0	OCV	OTPV	HVS0	HVSEN	OUT1H	OUT1L	OUT0H	OUT0L	0000 0000	0000 0000					
59	0x9a	HBC1	OCPF[3:0]									--0 0000	--0 0000				
60	0x9b	HBC2	OVPF[2:0]			OVPDLY[4:0]							--- -00	--- -00			
61	0x9c	PDBCON							PDB6	PDB5	PDB4	PDB3	PDB2	PDB1	PDB0	--11 1110	--11 1110
62	0x9d	ODBCON							ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	--0 0000	--0 0000
63	0x9e	PUBCON							PUB6	PUB5	PUB4	PUB3	PUB2	PUB1	PUB0	--11 1101	--11 1101
64	0x9f	SMCR	IDLE	OSCM	OPSM	OPPGA1	OPPGA0	CKOE	OSCF	PWOFF	-010 00x0	-010 00x0					
		WREG	WREG[7:0]									xxxx xxxx	uuuu uuuu				
			MAPEN=1														
0x8c	CMPCON	CMPEN	CMPOUT	CMP_D2A_	CMPPPOS	CMPPCH[1:0]			CMPNCH[1:0]		0000 0000	0000 0000					
0x8d	OP1CON0	OP1OUT	AN_AVSS	OP1PGA[1:0]	PGAEN1	OP1_SW_EN	CPMODE1	OP1_D2A_EN				--- -00	--- -00				
0x8e	OP1CON1	OP1ON	OP1_SW	A1NS1	A1O2N1	OP1PCH[1:0]		OP1NCH[1:0]			--- -00	--- -00					
0x8f	CLKCFG							CCPCLKEN	T2CLKEN	T1CLKEN	TOCLKEN	--- 0000	--- 0000				
0x90	IAPTRIG	0X16,0X02,0X96									0000 0000	0000 0000					
0x91	IAPCTRL										LOCK	PG	--- -00	--- -00			
0x92	IAPADDRH										IAPADDR[10:8]		--- -00	--- -00			
0x93	IAPADDRL	IAPADDR[7:0]									0000 0000	0000 0000					
0x94	IAPDATH	IAPDAT[13:8]									--0 0000	--0 0000					
0x95	IAPDATH	IAPDAT[7:0]									0000 0000	0000 0000					
0x96	IAPWAIT	IAPWAIT[7:0]									0000 0000	0000 0000					
0x97	HBC3	HBMPF[3:0]			HBMCMEN		OCPEN	OVPEN	OTPCEN		0000 0000	0000 0000					

3 系统时钟



系统时钟有 16MHz 和 256KHz 两种选择。

当系统工作在 256KHz 的时候，不支持的功能或模块有：OP0、OP1、CMP、LVR、H 桥、ADC、FVR、BGR 模块。

4 复位和电源电压检测

AD16H02 单片机能通过以下方式复位：

- 上电复位(POR)
- 掉电复位(Brown-out Reset BOR)
- RSTn 管脚复位
- 看门狗 WDT 溢出复位

一些寄存器在某些复位条件下没有影响，在上电和其他复位情况下它们的状态是未知的，大多数寄存器会回到复位状态。在上电复位、RSTn 管脚复位、看门狗 WDT 溢出复位时，对 VDD 上升信号检测告之芯片是否加上上电复位脉冲信号。要使用这个特点，用户需要把 RSTn 管脚连接到 VDD。RSTn 或 WDT 睡眠唤醒也导致芯片复位，其复位操作的不会在睡眠之前。根据不同的复原状态设置对 /TO 和 /PD 位 (STATUS<4 :3>)置 1 或清零。

4.1 上电复位计数器(PWRT Power-up Reset Timer)

上电复位计数器提供一个 32.8ms、4ms、262.2ms、65.6ms 延迟时间（该延迟时间由 TWDT<1:0>设置）（或 486us，基于不同的振荡源和复位条件）在 Power-on Reset (POR), Brown-out Reset (BOR), RSTn Reset 或看门狗溢出复位。只要 PWRT 在运行，设备就一直保持在复位状态。VDD、温度和其他变化而会影响 PWDT 控制的设备延迟时间。

4.2 振荡启动计数器(OST Oscillator Start-up Timer)

在 HIRC 或 LIRC 振荡模式下 PWRT 延迟 (32.8/4/262.2/65.6ms) 之后振荡启动计数器会再提供一个 64 个 clock 的延迟。这种延迟晶体谐振器能提供稳定的振荡源，这段时间内只要 OST 在工作，设备就一直保持在复位状态。

4.3 复位顺序

AD16H02 复位时序如下：

1. 复位锁存器置 1，PWRT & OST 清零。
2. 当内部的 POR, BOR, RSTn 复位或 WDT 溢出复位脉冲加载完成后，PWRT 开始计数。
3. PWRT 溢出以后，OST 开始计数延迟。
4. OST 延迟完成以后，复位锁存器清零后芯片得到一个复位信号。

在高频或低频振荡模式下机器复位延迟时间为 32.8ms、4ms、262.6ms、65.6ms 加上 64 个振荡周期。在 OSC 振荡模式下单片机会在 Power-on Reset (POR), Brown-out Reset (BOR), 或 RSTn 复位以后再延迟 486us，看门狗溢出复位后再延迟 32.8ms、4ms、262.6ms、65.6ms 的时间。

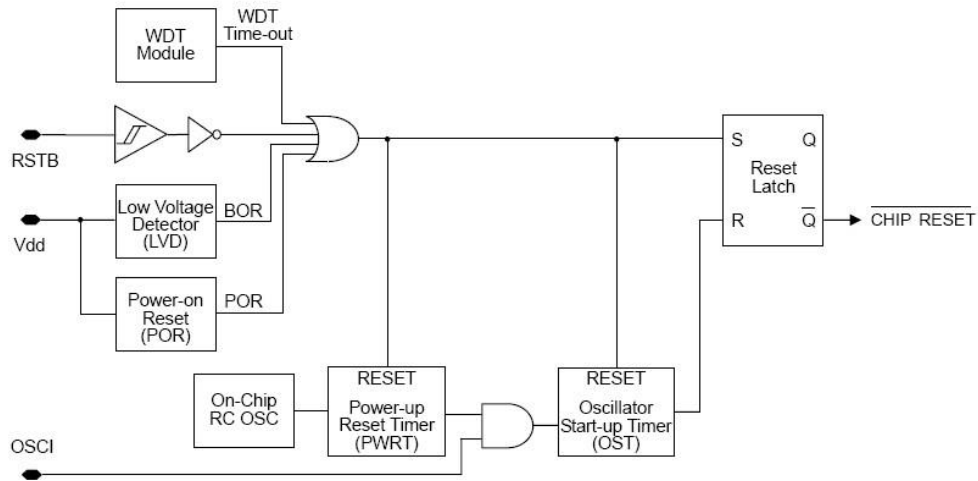


图 4.3 电路结构图

复位以后各个寄存器状态列表

寄存器	地址	上电、掉电复位	RSTn、WDT 复位
W	N/A	xxxxxxxx	uuuu uuuu
T0CON	8bh/0bh	0011 1111	0011 1111
TRISB	86h	--11 1111	--11 1111
PWMCON	11h	xxxx 0000	uuuu 0000
SMCR	9Fh/1Fh	-010 00x0	-010 00x0
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	0Bh	xxxx xxxx	uuuu uuuu
PCL	02h	1111 1111/0000 0000	1111 1111/0000 0000
STATUS	03h	0001 1xxx	000# #uuu
FSR	04h	--xx xxxx	--uu uuuu
PORTB	06h	--xx xxxx	--uu uuuu
PCON	08h	1000 0000	1000 0000
WUBCON	98h/18h	--00 0000	--00 0000
PCLATH	8Ah/0Ah	---- --00	---- --00
PDBCON	9Ch/1Ch	--11 1010	--11 1010
ODBCON	9Dh/1Dh	--00 0000	--00 0000
INTEN	0Eh	0--0 0000	0--0 0000
INTF	0Fh	0000 0000	0000 0000
INT1EN	1Ah	0000 0000	0000 0000
IN1TF	1Bh	0--0 0000	0--0 0000
General Purpose Registers	10 ~ 4Fh	xxxx xxxx	uuuu uuuu
ADRESH	1ch	0000 0000	0000 0000
ADRESL	1dh	0000 0000	0000 0000
ADCON	1eh	0000 0000	0000 0000

Legend: u = 不变, x = 未知, -= 不起作用, # = 参见下表的值

RST/ TO / PD 复位和唤醒后的状态

RST	TO	PD	复位方式
0	1	1	Power-on Reset
0	1	1	Brown-out reset
0	u	u	RSTn Reset during normal operation
0	1	0	RSTn Reset during SLEEP
0	0	1	WDT Reset during normal operation
0	0	0	WDT Wake-up during SLEEP
1	1	0	Wake-up on pin change during SLEEP

Legend: u =不变

TO /PD 状态位影响事件

事件	TO	PD
Power-on	1	1
WDT Time-Out	0	u
SLEEP instruction	1	0
CLRWDT instruction	1	1

Legend: u =不变

4.4 LVD 检测

LVD 可通过 PCON[3:0]配置电压阈值。电压检测电路有一定的回滞特性，通常回滞电压为 0.05V 左右。例如，如果选择了 3.6V 的 LVD 电压，则当电源电压下降到约 3.6V 时复位有效，而电压需要上升到约 3.65V 时 LVD 复位才会解除。

PCON (电源控制寄存器)

地址：0x08

Bit	Name	Description	Attribute	Reset
7	WDTE	看门狗使能位(需要配置字和 WDTE 都置 1 才可能使能看门狗) 0: 看门狗关闭 1: 看门狗开启	R/W	1
6	PB0ST	LVDM=11 时, PB0 输出值	R/W	0
5:4	LVDM [1:0]	电压比较中断 00: 禁止电压比较器 01: VDD 低于阈值电压产生中断 10: VDD 高于阈值电压产生中断 11: VDD 高于阈值电压产生中断, 且强制 PB0 输出为 PB0ST 值	R/W	00
3:0	LVT [3:0]	VDD 电压阈值选择 0000: Reserved 0001: Reserved 0010: Reserved 0011: 2.3V 0100: 2.6V 0101: 2.8V 0110: 3.2V 0111: 3.4V 1000: 3.6V 1001: 4.1V 其余未使用	R/W	1001

注：PCON 寄存器中设置的 LVT 电压值的选择用于产生中断，配置字中设置的 LVR 电压选择用于产生复位，两者相互独立，复位的优先级高于中断。

5 OTP 操作描述

OTP 可以自编程的地址空间从 000H 到 7FFH，IAPADDR[10:0]（由 IAPADDRH 高 3 位和 IAPADDRL 组成）对应 OTP 的地址 0H。IAPDATH[13:8]和 IAPDATL[7:0]组成 14 位数据。

OTP 写入数据操作步骤：

1. 写入前 VIN2 电压供电为 9.3V。
2. 对 IAPTRIG 连着写入 0x16、0x02、0x96，进入到 IAP mode。注意：如果往 IAPTRIG 写入其他任何值，则进入 IAPLOCK 状态。进行重新上电来解除锁定，解除后需要重新开始进行 IAP 写操作。
3. 设置 IAPADDRH（高位地址）和 IAPADDRL（低位地址），设置数据要写入 OTP 对应 000H-7FFH 地址内的某个地址。
4. 设置 IAPDATH（高位数据）和 IAPDATL（低位数据），即把需要写入数据寄存器的数据暂放在 IAPDAT 寄存器中。
5. CPU 将 IAPDAT[13:0]数据写入 IAPADDRH 和 IAPADDRL 对应的 OTP 地址中。
6. 如果要继续写，则回到步骤 2 开始。如果要退出写入操作，则到步骤 7。
7. 对 IAPTRIG 写入 0，便可以退出写操作。

5.1 OTP 的相关寄存器

IAPTRIG (MAPEN=1)

地址：0x90

Bit	Name	Description	Attribute	Reset
7:0	IAPTRIG [7:0]	IAP 的触发寄存器	R/W	0x00

IAPCTRL (MAPEN=1, 寄存器读写)

地址：0x91

Bit	Name	Description	Attribute	Reset
7:2	Reserve			
1	LOCK	IAP 锁定状态位 1: 锁定 0: 未锁定	R	0
0	PG	写入控制位 1: 写入（需要 100us） 0: 写完成	R/W	0

IAPADDRH (MAPEN=1,寄存器读写)

地址：0x92

Bit	Name	Description	Attribute	Reset
7:6	Reserve			
5:0	IAPADDRH [13:8]	IAP 地址高 6 位	R/W	000000

IAPADDR (MAPEN=1, 寄存器读写)

地址: 0x93

Bit	Name	Description	Attribute	Reset
7:0	IAPADDR [7:0]	IAP 地址低 8 位	R/W	0x00

IAPDATA (MAPEN=1, 寄存器读写)

地址: 0x94

Bit	Name	Description	Attribute	Reset
7:6	Reserve			
5:0	IAPDATA [13:8]	IAP 数据高六位	R/W	0 0000

IAPDATA (MAPEN=1, 寄存器读写)

地址: 0x95

Bit	Name	Description	Attribute	Reset
7:0	IAPDATA [7:0]	IAP 数据低八位	R/W	0x00

IAPWAIT (MAPEN=1, 寄存器读写)

地址: 0x96

Bit	Name	Description	Attribute	Reset
7:0	IAPWAIT [7:0]	IAP 延时位	R/W	0x00

$((IAPWAIT / Fcpu) * 2) > 100us$

注意: 该寄存器若无特殊情况时一般填写 0x66

6 GPIO

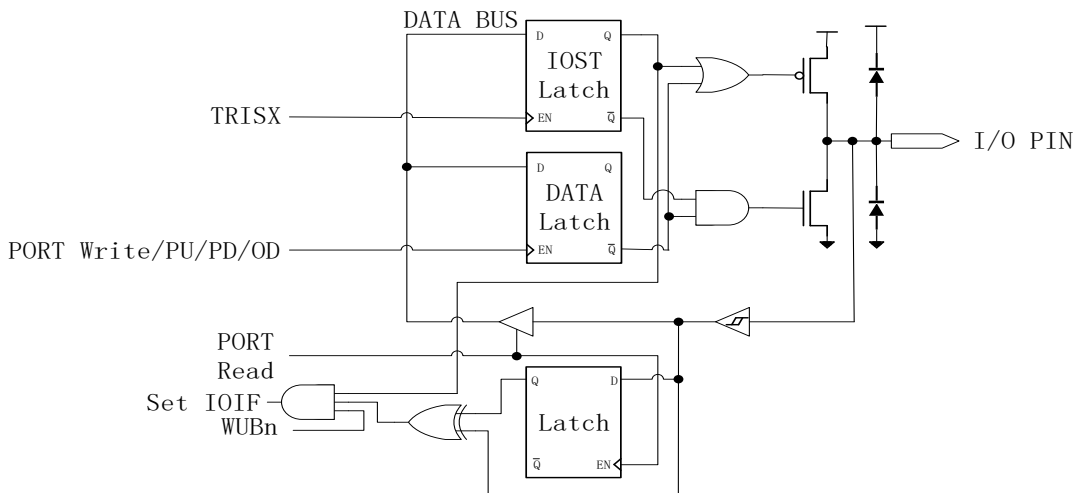
AD16H02 的 PortB 口输入/输出方式由 I/O 控制寄存器(TRISB)设置。

PB<7:0>有相应的上拉控制位(PUBCON 寄存器)来设置使能内部上拉。

PB<7:0>有相应的下拉控制位(PDBCON 寄存器)来设置使能内部下拉。

PB<7:0>有相应的开漏控制位(ODBCON 寄存器)来设置使能开漏来设置输出为开漏输出。

PB<7:0>有输入改变中断/唤醒功能，通过 WUBCON 寄存器可以分别设置每个引脚是否具有该功能。



6.1 IO 工作模式

PORTB 寄存器

地址: 0x06

Bit	Name	Description	Attribute	Reset
7:0	PORTB [7:0]	PB I/O 引脚 1: 端口引脚电平 > V _{IH} 0: 端口引脚电平 < V _{IL}	R/W	0x00

TRISB (I/O 口方向控制寄存器)

地址: 0x86

Bit	Name	Description	Attribute	Reset
7:0	TRISB [7:0]	PB 口的数据方向控制选择 1: IO 的输入模式 0: IO 的输出模式	R/W	0xFF

6.2 上下拉电阻开漏模式

PB 口可通过 PBU 和 PBD 选择是否接上拉/下拉电阻。上拉电阻在端口置为输入状态时有效，置为输出状态时无效。通过 ODB 控制 PB 端口是否使能开漏输出。

PDBCON (I/O 下拉控制寄存器)

地址：0x9C

Bit	Name	Description	Attribute	Reset
7:0	PDB [7:0]	PB 口的下拉使能选择 1: 关闭内部下拉 0: 使能内部下拉	R/W	11111101

PUBCON (I/O 上拉控制寄存器)

地址：0x9E

Bit	Name	Description	Attribute	Reset
7:0	PUB [7:0]	PB 口的上拉使能选择 1: 关闭内部上拉 0: 使能内部上拉	R/W	11111110

ODBCON (I/O 开漏控制寄存器)

地址:0x9D

Bit	Name	Description	Attribute	Reset
7:0	ODB [7:0]	PB 口的开漏使能选择 1: 使能内部开漏 0: 关闭内部开漏	R/W	0x00

WUBCON (Port B 输入改变中断控制寄存器)

地址：0x88

Bit	Name	Description	Attribute	Reset
7:0	WUB [7:0]	PB 口的电平变化中断控制允许位 1: 使能 PortB 端口电平变化中断允许位 0: 禁止 PortB 端口电平变化中断允许位	R/W	0x00

6.3 模拟 IO 寄存器

SMTVB (B 口施密特寄存器)

地址: 0X87

Bit	Name	Description	Attribute	Reset
7:0	SMTVB [7:0]	PB 口的施密特控制位 1: $0.3 \cdot V_{DD} / 0.7 \cdot V_{DD}$ (IO 低电平变为高电平时为 $0.7 \cdot V_{DD}$, 高电平变为低电平时为 $0.3 \cdot V_{DD}$) 0: $0.2 \cdot V_{DD} / 0.4 \cdot V_{DD}$ (IO 低电平变为高电平时为 $0.4 \cdot V_{DD}$, 高电平变为低电平时为 $0.2 \cdot V_{DD}$)	R/W	0xFF

ANASEL (IO 的模拟通道寄存器)

地址: 0x07

Bit	Name	Description	Attribute	Reset
7:0	PBnAEN [7:0]	PB 口的模拟通道使能位 1: PB 口打开模拟通道 0: PB 口关闭模拟通道	R/W	0x00

7 定时器

7.1 Timer0/WDT & Prescaler/BUZZER

7.1.1. Timer0 计数/定时器

Timer0 为 8 位定时/计数器, Timer0 的时钟源可以是内部时钟(Fcpu)、外部时钟源(T0CKI pin)、内部 32K 时钟和 CMPOUT 比较器的输出。

7.1.2. 使用内部时钟: 定时模式

T0CS[1:0]=0 为定时模式, 定时模式在没有预置器的情况下, 定时寄存器每个指令周期自动加 1, 设置 TMR0 以后, 定时器将在两个时钟周期以后开始自增。

7.1.3. 使用外部时钟/内部 32K 时钟: 计数模式

T0CS[2:0]=1,2,3,4 为计数模式, 是通过选择 T0CKI 上升或下降沿/32K 时钟, 触发 Timer0 寄存器的增加, T0CKI 由 T0SE 位(T0CON<4>)决定上升下降触发, 外在时钟要求与内部时钟(Tosc)同步。同步以后, Timer0 实际增加有一个延迟。

在没有预置器的情况下, 外部时钟输入同样也可以作为预置器输出; T0CKI 与内部时钟同步时能方便处理在 T2 和 T4 周期上的预分频。因此 T0CKI 为高或低电平必须要保持两个以上时钟周期才有效。

有预置分频时器, 外部时钟输入被异步分频器平分, 这种常用来计算波形。因此 T0CKI 的一个波形周期至少 $4T_{osc}$ 才能被预置器平分。

7.1.4. 看门狗定时器 (WDT)

看门狗定时器 (WDT) 的运行依赖于芯片里的 RC 振荡器, 无需任何额外的电路就能工作。在一般操作或睡眠模式情况下, 看门狗定时器的溢出都会导致 MCU 复位, 同时 TO (STATUS<4>)位被清零。

如 WDT 位(PCON<7>)清零, 看门狗定时器不能工作。

在没有预置器时看门狗的溢出为 32.8ms、4ms、262.6ms、65.6ms 这个时间可以通过 TWDT<2:0>设置。

需要看门狗的溢出周期变长, 可以通过设置 T0CON 寄存器的看门狗定时器分频。最大 128 分频, 即最长的看门狗溢出周期为 33.6 秒。

CLRWD 指令能使 WDT 和预置器清零, 启用看门狗可以防止超时, 如果超时 MCU 会复位。

SLEEP 指令重置 WDT 和预置器, 启用看门狗就给机器分派了一个最大睡眠时间。

7.1.5. Prescaler (预置器)

有一个 8 位的向下计数器作为 Timer0 和看门狗定时器(WDT)的预置器。注意该预置器只能分配给 Timer0 或 WDT 使用, 不能两者同时使用。PSA 位(T0CON<3>) 决定预置器是指派给 Timer0 还是 WDT。PS<2:0>位(T0CON<2:0>)可配置分频。当作为 Timer0 的预置器的时候, TMR0 会被预置器清零。当作为 WDT 的预置器的时候, CLRWD 指令会清除预置器内容。预置器不能读写, 机器复位, 预置器各位全为 1。

为了避免机器非正常复位, 当 Timer0 或 WDT 的预置器发生改变的时候, 需要执行 CLRWD 或 CLRF TMR0 指令, 反之亦然。

注：4T 模式下定时： $Time = (PR0+1) * \{PS2:PS0\} * 4 / F_{sys}$ ；//TMR0 的初值为 0 时
 $Time = (0xFF - [TMR0] + 1) * \{PS2:PS0\} * 4 / F_{sys}$ ；//TMR0 设定初值
 2T 模式下定时： $Time = (PR0+1) * \{PS2:PS0\} * 2 / F_{sys}$ ；//TMR0 的初值为 0 时
 $Time = (0xFF - [TMR0] + 1) * \{PS2:PS0\} * 2 / F_{sys}$ ；//TMR0 设定初值

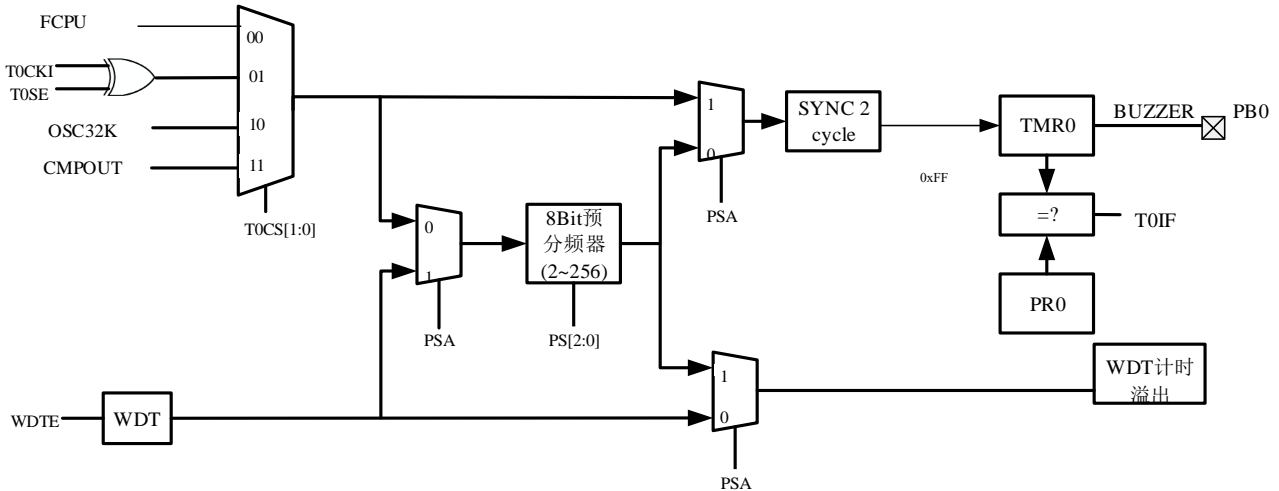


图 7.1.5: Timer0/WDT Prescaler 结构图

7.1.6. BUZZER (BUZZER 输出)

Buzzer 输出的是一个简单的 1/2 占空比信号，由 T0IF 产生。当 T0IF 溢出时，Buzzer 开始输出一个方波，中断间隔时间频率 2 分频后作为 Buzzer 输出的频率。

TMR0 溢出后，Buzzer 输出时，TMR0IF 有效，且当 TMR0IE=1 时，可使能 TMR0 中断功能。Buzzer 输出引脚与 GPIO 引脚共用，T0OUT=1 时，该引脚自动设为 Buzzer 输出引脚。如清 T0OUT 位后禁止 Buzzer 输出后，该引脚自动返回到最后一个 GPIO 模式。

7.1.7. TIMER0 的相关寄存器

TMR0 (定时/计数器)

地址：0x01

Bit	Name	Description	Attribute	Reset
7:0	TMR0[7:0]	8 位定时/计数器	R/W	0x00

T0CON (TMR0 控制寄存器)

地址：0x0B

Bit	Name	Description	Attribute	Reset
7	T0OUT	PB0 溢出输出 BUZZER 1: 启动 BUZZER 模式 0: 禁止 BUZZER 模式	R/W	0

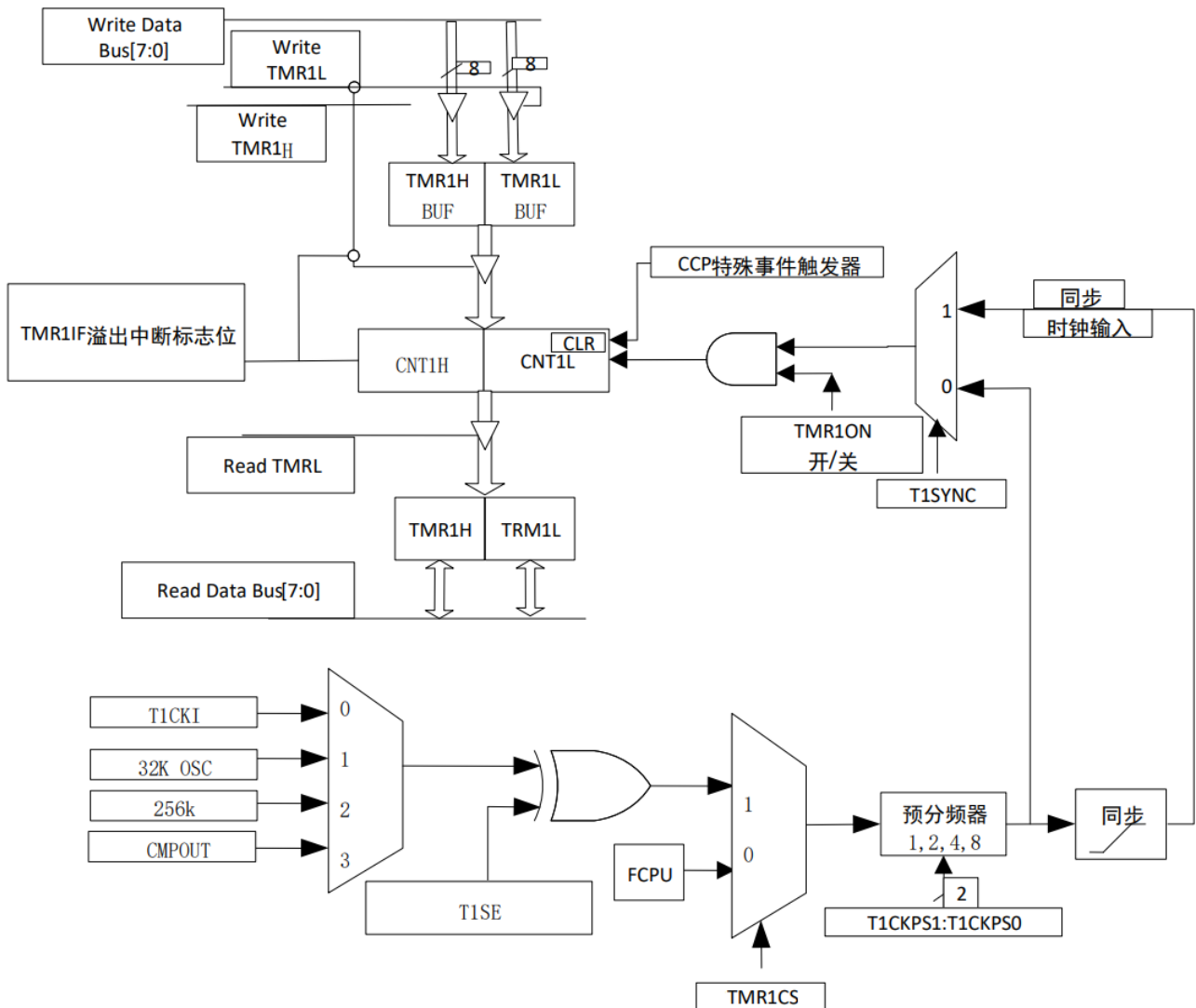
6:5	T0CS [1:0]	<p>T0 的时钟源选择</p> <p>00: T0 时钟源为 CPU 运行时钟</p> <p>01: T0 时钟源为 T0CKI</p> <p>10: T0 时钟源为内部 32K OSC</p> <p>11: T0 时钟源是 CMPOUT 比较器输出</p>	R/W	11																											
4	T0SE	<p>TMR0 触发方式控制位</p> <p>1: T0CKI 脚下降沿触发计数</p> <p>0: T0CKI 脚上升沿触发计数</p>	R/W	1																											
3	PSA	<p>分频器选择位.</p> <p>1: WDT (看门狗定时器)</p> <p>0: TMR0 (Timer0)</p>	R/W	1																											
2:0	PS [2:0]	<p>分频率选择控制位</p> <table border="1"> <thead> <tr> <th></th> <th>Timer0 Rate</th> <th>WDT Rate</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>1:2</td> <td>1:1</td> </tr> <tr> <td>001</td> <td>1:4</td> <td>1:2</td> </tr> <tr> <td>010</td> <td>1:8</td> <td>1:4</td> </tr> <tr> <td>011</td> <td>1:16</td> <td>1:8</td> </tr> <tr> <td>100</td> <td>1:32</td> <td>1:16</td> </tr> <tr> <td>101</td> <td>1:64</td> <td>1:32</td> </tr> <tr> <td>110</td> <td>1:128</td> <td>1:64</td> </tr> <tr> <td>111</td> <td>1:256</td> <td>1:128</td> </tr> </tbody> </table>		Timer0 Rate	WDT Rate	000	1:2	1:1	001	1:4	1:2	010	1:8	1:4	011	1:16	1:8	100	1:32	1:16	101	1:64	1:32	110	1:128	1:64	111	1:256	1:128	R/W	111
	Timer0 Rate	WDT Rate																													
000	1:2	1:1																													
001	1:4	1:2																													
010	1:8	1:4																													
011	1:16	1:8																													
100	1:32	1:16																													
101	1:64	1:32																													
110	1:128	1:64																													
111	1:256	1:128																													

7.2 Timer1 16 位定时/计数器

TIMER1 为 16 位定时/计数器，内部拥有写入缓冲区。当溢出时，自动从写入缓冲区装载到 TIMER1 计数器作为初值，然后每个 TIMER1 时钟周期，计数递增。当 TIMER1 计数值从 0xFFFF 变为 0x0000 时产生溢出中断，并自动重载写入缓冲区数值。TIMER1 的时钟源可以是内部时钟 (Fcpu)、外部时钟源(TICKI pin)、内部 256K 时钟、内部 32K 时钟和 CMPOUT 比较器的输出。

写入 16 位初值时，首先写入 TMR1H 寄存器，然后写入 TMR1L，此时硬件自动将写入缓冲区值重载到 TIMER1 的计数器中；当读取 TMR1L 时，自动将当前 TIMER1 的计数值读取 TMR1H 和 TMR1L 寄存器，保证读取时间准确。

同时 TIMER1 可以与 CCPR1H, CCPR1L 采用比较器模式，产生 1 路 16 位精度的 PWM 输出，输出引脚为 CCPOUT(PB1)。



7.2.1. 使用外部时钟/内部 32K 时钟/内部 256K 时钟/比较器输出: 计数模式

T1CH[1:0]==1,2,3 为计数模式，是通过选择 TICKI 上升或下降沿/32K 时钟/内部 256K 时钟/比较器输出，触发 Timer1 寄存器的增加，触发位由 T1SE 位(T1CON<3>)决定上升下降触发，可以选择时钟要求与内部时钟(Tosc)是否同步。同步以后，Timer1 实际增加有一个延迟。当不选择同步计数时，可以在低功耗模式

下继续工作，低功耗模式会自动关闭内部高速时钟。

预置分频器时，外部时钟输入被异步分频器平分，这种常用来计算波形。

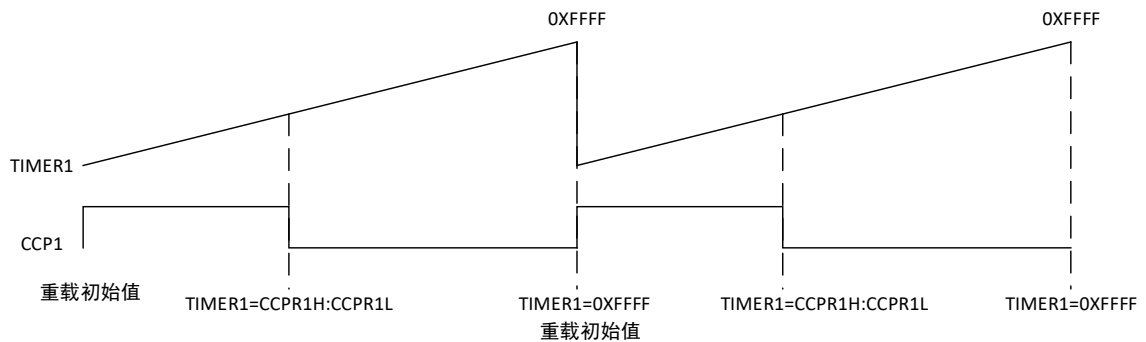
7.2.2. TIMER1 特殊事件

如果 CCP1 模块被配置成为特殊事件触发模式 (CCP1M=1011) 中的比较模式时，该位触发信号将复位 Timer1。如果使能了 AD 模式 (ADON=1)，则来自 CCP1 的触发信号还将启动 A/D 转换。

要使用这一功能，必须将 Timer1 配置为定时器或同步计数器。在这种情况下，CCPR1H:CCPR1L 这对寄存器实际上变成了 Timer1 的周期寄存器。如果 Timer1 在异步计数器模式下运行，复位操作可能不起作用。如果 Timer1 的写操作和特殊事件触发信号同时发生，则写操作优先。

7.2.3. 1 路 16 位 PWM 输出

如果 CCP1 模块被配置成为比较模式，同时 Timer1 配置位定时器或同步计数器，CCPR1H:CCPR1L 这对寄存器实际上变成了 Timer1 的占空比控制寄存器，实现 16 位 PWM 控制输出。



7.2.4. 定时器 TIMER1 相关的寄存器

TMR1L (Timer1 16 位低 8 位寄存器)

地址: 0x0C

Bit	Name	Description	Attribute	Reset
7:0	TMR1L [7:0]	16 位定时/计数器低 8 位	R/W	0x00

TMR1H (Timer1 16 位高 8 位寄存器)

地址: 0x0D

Bit	Name	Description	Attribute	Reset
7:0	TMR1H [7:0]	16 位定时/计数器高 8 位	R/W	0x00

T1CON0 (Timer1 控制寄存器)

地址: 0x8C

Bit	Name	Description	Attribute	Reset
7:6	T1CK [1:0]	T1 的时钟选择 00: T1CKI 作为时钟 01: 内部 32K OSC 作为时钟 10: 内部 256K OSC 作为时钟 11: CMPOUT 作为时钟	R/W	00
5:4	T1CKPS [1:0]	T1 的输入时钟分频 00: TIMER1 输入时钟 1:1 分频 01: TIMER1 输入时钟 1:2 分频 10: TIMER1 输入时钟 1:4 分频 11: TIMER1 输入时钟 1:8 分频	R/W	00
3	T1SE	TIMER1 的时钟异或输入选项	R/W	0
2	T1SYNC	1: 选择同步分频后时钟作为 TIMER1 时钟 0: 选择分频时钟作为 TIMER1 时钟	R/W	0
1	TMR1CS	1: 选择 T1CK[1:0]作为外设 TIMER1 的时钟 0: 选择 FCPU 时钟外设 TIMER1 的时钟	R/W	0
0	TMR1ON	1: 使能 Timer1 定时计数器 0: 关闭 Timer1 定时计数器	R/W	0

T1CON1 (Timer1 控制寄存器)

地址: 0x8D

Bit	Name	Description	Attribute	Reset
7:5	INTnEDGE [2:0]	外部中断边沿选择位 (n=0、1、2) 1: 下降沿触发中断 0: 上升沿触发中断	R/W	0
4	T1EDGE	在 T1M[1:0]在脉宽测量模式时 1: 在上升沿启动计数, 下降沿停止计数 0: 在下降沿启动计数, 上升沿停止计数	R/W	0
3:2	T1CH [1:0]	TIMER1 脉宽信号输入选择 00: T1CH0 作为脉宽检测输入信号(PB4) 01: T1CH1 作为脉宽检测输入信号(PB5) 10: T1CH2 作为脉宽检测输入信号(PB3) 11: CLK32K 作为脉宽检测输入信号	R/W	00
1:0	T1M [1:0]	TIMER1 工作模式 00: TIMER1 工作在普通模式, 溢出时间产生中断标志位。 01: 脉宽测量模式 0, 测量上升沿与下降沿之间的时间, T1EDGE=1 时, 在脉冲的上升沿开始计数, 脉冲下降沿停止计数, 在脉冲下降沿触发中断。T1EDGE=0 时, 在脉冲的下降沿开始计数, 脉冲的上升沿停止计数, 在脉冲上升沿触发中断。产生中断标志后, 可以直接读取 TIMER1 内容, 就可以得到测量值。 10: 脉宽测量模式 1, 测量上升沿与下降沿之间的时间, T1EDGE=1 时在脉冲的上升沿 TIMER1 数据寄存器发生复位从 0x0000 开始计数, 并在下降沿沿触中断。T1EDGE=0 时, 在脉冲的下降沿 TIMER1 数据寄存器发生复位从 0x0000 开始计数, 在脉冲的上升沿触发中断。产生中断标志后, 可以直接读取 TIMER1 内容, 就可以得到测量值。 11: 模式 2, TIMER1 工作在定时模式, 在定时产生中断时, 启动 ADC 采集。此模式使用时, 首先要将 ADC 配置完成, 才能使用。	R/W	00

7.3 TIMER2 定时器

Timer2 定时器模块具有以下特征：

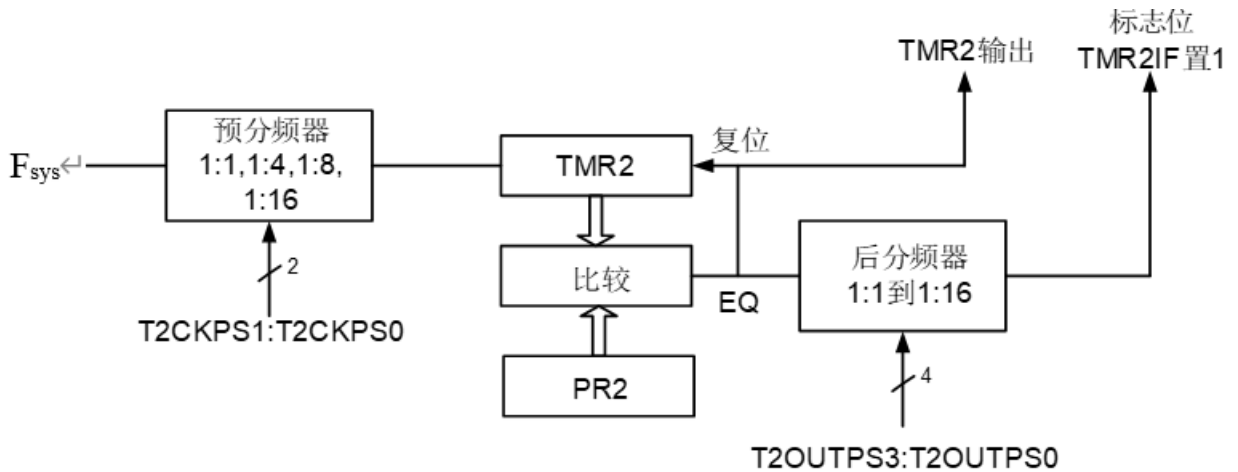
- 12 位定时器和周期寄存器（分别为 TMR2L、TMR2H 和 PR2L、PR2H）
- 可读写（以上四个寄存器）
- 可软件编程的预分频器（分频比为 1:1、1:4、1:8、1:16）
- 可软件编程的后分频器（分频比为 1:1 至 1:16）
- 当 TMR2（TMR2H, TMR2L）与 PR2（PR2H, PR2L）匹配时产生中断
- 采用系统时钟 F_{sys} 控制

7.3.1. TIMER2 的工作原理

TIMER2 模块的时钟输入是系统时钟(F_{sys})，时钟被输入到 TIMER2 预分频器，有如下几个分频比可供选择：1:1、1:4、1:8 和 1:16。预分频器的输出随后用于使 TMR2 寄存器递增。

持续将 TMR2 和 PR2 的值做比较确定它们是否匹配。TMR2 将从 00H 开始递增直至与 PR2 中的值匹配，匹配下时会发生以下三个事件：

- TMR2 在下一递增周期被复位为 00H
- TIMER2 后分频器递增
- TIMER2 与 PR2 比较器的匹配输出随后输入给 TIMER2 的后分频器。后分频器的输出用于产生 TIMER2 的中断标志位置 1



定时器 TIMER2 相关的寄存器

PR2L (timer2 的周期寄存器)

地址：0x96

Bit	Name	Description	Attribute	Reset
7:0	PR2L [7:0]	周期寄存器的低 8 位	R/W	0XFF

PR2H (timer2 的周期寄存器)

地址：0x97

Bit	Name	Description	Attribute	Reset
-----	------	-------------	-----------	-------

7:4	Reserved			
3:0	PR2H [3:0]	周期寄存器的高 4 位	R/W	0X1F

T2CON(timer2 控制寄存器)

地址: 0x95

Bit	Name	Description	Attribute	Reset
7	Reserved		R/W	0
6:3	T2OUTPS [3:0]	Timer2 输出后分频选择位 0000: 1:1 后分频值 0001: 1:2 后分频值 0010: 1:3 后分频值 0011: 1:4 后分频值 0100: 1:5 后分频值 0101: 1:6 后分频值 0110: 1:7 后分频值 0111: 1:8 后分频值 1000: 1:9 后分频值 1001: 1:10 后分频值 1010: 1:11 后分频值 1011: 1:12 后分频值 1100: 1:13 后分频值 1101: 1:14 后分频值 1110: 1:15 后分频值 1111: 1:16 后分频值	R/W	0000
2	TMR2ON	Timer2 使能位 1: 使能 Timer2 0: 禁止 Timer2	R/W	0
1:0	T2CKPS [1:0]	Timer2 时钟预分频选择位 00: 预分频值为 1 01: 预分频值为 4 10: 预分频值为 8 11: 预分频值为 16	R/W	00

TMR2L (Timer2 的低 8 位寄存器)

地址: 0x16

Bit	Name	Description	Attribute	Reset
-----	------	-------------	-----------	-------

7:0	TMR2L [7:0]	Timer2 定时/计数器的低 8 位	R/W	0X00
-----	-------------	---------------------	-----	------

TMR2H (Timer2 的高 4 位寄存器)

地址: 0x17

Bit	Name	Description	Attribute	Reset
7:4	Reserved			
3:0	TMR2H [11:8]	Timer2 定时/计数器的高 4 位	R/W	0000

注: 配置 TMR2 和 PR2 寄存器时, 先写高字节再写低字节;

8 捕获/比较/脉冲宽度调制模块(CCP)

捕捉/比较/PWM (CCP) 模块包含一个 16 位寄存器，它可被用作：1 个 16 位捕捉寄存器、1 个 16 位比较寄存器、1 个 PWM 主/从占空比寄存器。捕捉/比较寄存器 (CCPR1) 由两个 8 位寄存器组成：CCPR1L (低字节) 和 CCPR1H (高字节)。CCPCON 寄存器控制 CCP 的操作。捕获模式是通过 TIMER1 的相关寄存器控制。比较匹配将产生特殊事件触发信号，该信号会使 TMR1H 和 TMR1L 寄存器清零。PWM 模式通过 TIMER2 的相关寄存器配置。

8.1 CCP 相关寄存器

CCPR1H(CCP1 高字节寄存器)

地址：0x10

Bit	Name	Description	Attribute	Reset
7:0	CCPR1H [7:0]	CCPR1 寄存器高字节，用于捕获、比较	R/W	0X00

CCPR1L (CCP1 低字节寄存器)

地址：0x11

Bit	Name	Description	Attribute	Reset
7:0	CCPR1L [7:0]	CCPR1 寄存器低字节，用于捕获、比较、PWM1 的占空比的低 8 位	R/W	0X00

CCPR1LH (CCP1 低字节中的高字节寄存器)

地址：0x12

Bit	Name	Description	Attribute	Reset
7:4	Reserved			
3:0	CCPR1LH [11:8]	CCPR1 寄存器低字节中的高 4 字节，用于 PWM1 的高 4 位	R/W	0000

CCPCON0(CCP 控制寄存器)

地址：0x90

Bit	Name	Description	Attribute	Reset
7:4	CCP1CH [3:0]	CCP1 捕获触发信号选择 0000: 外部 CCP1CH0 引脚输入信号 0001: 外部 CCP1CH1 引脚输入信号 0010: 外部 CCP1CH2 引脚输入信号 0011: 外部 CCP1CH3 引脚输入信号 0100: CMPOUT 输出信号 0101: 内部 32K 输入信号	R/W	0000

Bit	Name	Description	Attribute	Reset
		0110: 外部 CCP1CH4 引脚输入信号 0111: 外部 CCP1CH5 引脚输入信号		
3:0	CCP1M [3:0]	<p>CCP1 模式选择位</p> <p>0000: 禁止捕捉/比较/PWM (复位 CCP1 模块) ;</p> <p>0100: 比较模式, 选择 CCPR1 匹配时将输出置为高电平 (CCPIF 位置 1), 输出引脚为 PB1, TIMER1 溢出时, PB1 为低电平;</p> <p>0101: 比较模式, 选择 CCPR1 匹配时将输出置为低电平 (CCPIF 位置 1), 输出引脚为 PB1, TIMER1 溢出时, PB1 为高电平;</p> <p>0110: 比较模式, 选择 CCPR1 匹配时将产生软件中断 (CCPIF 位置 1, 而 PB1 引脚不受影响) ;</p> <p>0111: 比较模式, 选择 CCPR1 触发特殊事件 (CCPIF 位置 1, PB1 引脚不受影响) ; CCP1 清零 Timer1; 并启动 ADC 采集 (如果 ADCON=1) ;</p> <p>1000: 捕捉模式, 在每个下降沿发生, 捕捉值存入 CCPR1 寄存器, 并产生中断标志 CCPIF;</p> <p>1001: 捕捉模式, 在每个上升沿发生, 捕捉值存入 CCPR1 寄存器, 并产生中断标志 CCPIF;</p> <p>1010: 捕捉模式, 在每 4 个上升沿发生一次, 捕捉值存入 CCPR1 寄存器, 并产生中断标志 CCPIF;</p> <p>1011: 捕捉模式, 在每 16 个上升沿发生一次, 捕捉值存入 CCPR1 寄存器, 并产生中断标志 CCPIF;</p> <p>1100: 捕捉模式, 下降沿, 复位 TIMER1, 捕捉值存入 CCPR1 寄存器, 并产生中断标志 CCPIF;</p> <p>1101: 捕捉模式, 上升沿, 复位 TIMER1, 捕捉值存入 CCPR1 寄存器, 并产生中断标志 CCPIF;</p> <p>1110: 捕捉模式, 下降沿, 捕捉值存入 CCPR1 寄存器, 不产生中断标志 CCPIF;</p> <p>1111: 捕捉模式, 上升沿, 捕捉值存入 CCPR1 寄存器, 不产生中断标志 CCPIF;</p> <p>001X: PWM 模式, 启动 CCP1 PWM 模式;</p>	R/W	0000

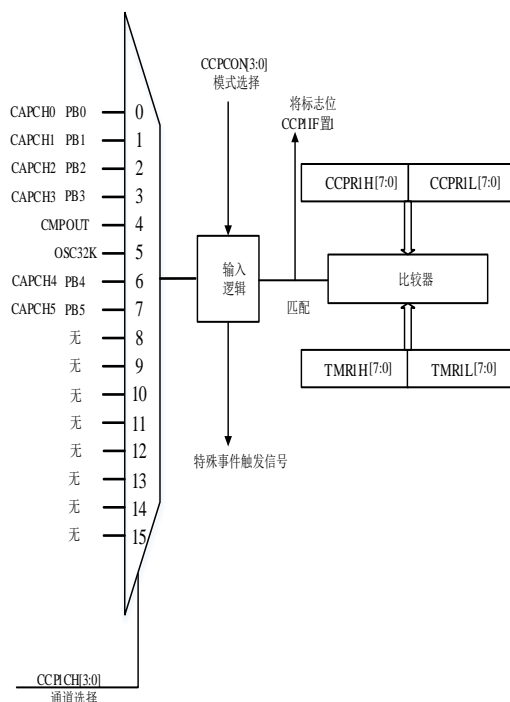
8.2 捕捉模式

在捕捉模式下，当引脚 CCP1 发生事件时，CCPR1H:CCPR1L 将捕捉 TMR1 寄存器的 16 位值。事件定义如下，由 CCPCON[3:0] 进行配置：

- 1000：捕捉模式，在每个下降沿发生，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF
- 1001：捕捉模式，在每个上升沿发生，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF
- 1010：捕捉模式，在每 4 个上升沿发生一次，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF
- 1011：捕捉模式，在每 16 个上升沿发生一次，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF
- 1100：捕捉模式，下降沿，复位 TIMER1，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF
- 1101：捕捉模式，上升沿，复位 TIMER1，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF
- 1110，1111：捕捉模式，上升沿和下降沿，捕捉值存入 CCPR1 寄存器，并产生中断标志 CCPIF

进行捕捉后，中断请求标志位 CCP1IF 被置 1。该中断标志位必须用软件清零。如果当 CCPR1H 和 CCPR1L 寄存器中的值被读出之前又发生另一次捕捉，那么原来的捕捉值会被新捕捉值覆盖。输入捕获通道由 CCP1CHI[3:0] 进行配置：

- 0000：外部 CAPCH0(PB0) 引脚输入信号
- 0001：外部 CAPCH1(PB1) 引脚输入信号
- 0010：外部 CAPCH2(PB2) 引脚输入信号
- 0011：外部 CAPCH3(PB3) 引脚输入信号
- 0100：CMPOUT 的输出信号
- 0101：内部 32K 输入信号
- 0110：外部 CAPCH4(PB4) 引脚输入信号
- 0111：外部 CAPCH5(PB5) 引脚输入信号



CCP1 捕捉模式工作原理图

8.3 比较模式

在比较模式下,CCPR1[15:0]寄存器的 16 位值不断与 TMR1[15:0]寄存器的值进行比较。事件定义如下,由 CCPCON[3:0]进行配置:

- ◇ 0100: 比较模式, 选择 CCPR1 匹配时将输出置为高电平 (CCPIF 位置 1), 输出引脚为 PB1; TIMER1 溢出时, PB1 为低电平; 可以实现 16 位 PWM 控制
- ◇ 0101: 比较模式, 选择 CCPR1 匹配时将输出置为低电平 (CCPIF 位置 1), 输出引脚为 PB1; TIMER1 溢出时, PB1 为高电平; 可以实现 16 位 PWM 控制
- ◇ 0110: 比较模式, 选择 CCPR1 匹配时将产生软件中断 (CCPIF 位置 1, 而 PB1 引脚不受影响)
- ◇ 0111: 比较模式, 选择 CCPR1 触发特殊事件 (CCPIF 位置 1, PB1 引脚不受影响) CCP1 清零 TIMER1; 并启动 ADC 采集 (如果 ADON=1)

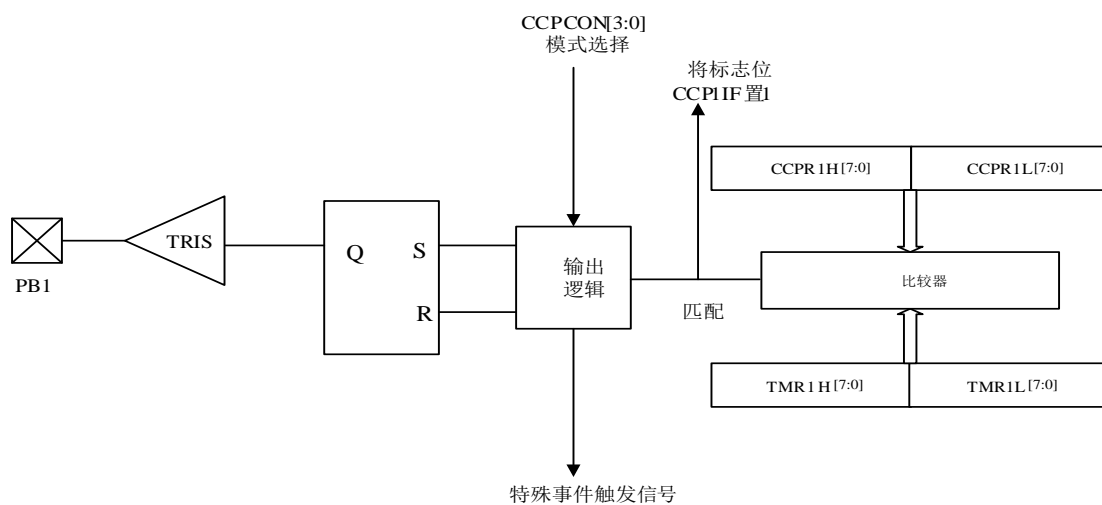


图 8.3 比较模式工作原理图

8.4 PWM 模式

PWM 信号的占空比、周期和分辨率由以下寄存器决定:

- ◇ PR2L、PR2H
- ◇ T2CON
- ◇ CCPR1L, CCPCON0
- ◇ PWM2CON2, PWM2CON3

在脉宽调制 (PWM) 模式下, CCP 模块会产生最大 12 位分辨率的 PWM 输出信号。

可通过写 PR2 寄存器来指定 PWM 周期。可以用以下公式计算 PWM 周期:

当 FCPUS=1(4T)时, PWM 周期= $[(PR2[11:0])+1] \times (TMR2 \text{ 预分频值})$

当 TMR2 和 PR2 相等时, 在下一个递增计数周期中将发生以下三个事件:

- ◇ TMR2 被清零
- ◇ PWM 信号被置 1 (例外情况: 如果 PWM 占空比=0%, 将不会被置 1。)
- ◇ PWM 占空比从 CCPR1L 锁存到 CPR1L (芯片内部寄存器)

通过写入 CCPR1L 寄存器和 CCPR1LH 位来指定 PWM 的占空比, 最高分辨率可达 12 位。CCPR1L 包

含八个 LSB，CCPR1LH 包含四个 MSB。由 CCPR1LH:CCPR1L 表示这个 12 位值。为了 PWM 转换期间不出现毛刺，每个 PWM 周期，CCPR1LH:CCPR1L 的值会被装载进芯片内部寄存器 CCPR1H[11:0]里，作为 PWM 产生占空比波形的寄存器使用。计算 PWM 占空比的公式如下：

$$\text{PWM 占空比} = (\text{CCPR1LH}:\text{CCPR1L}) / (\text{PR2H}:\text{PR2L})$$

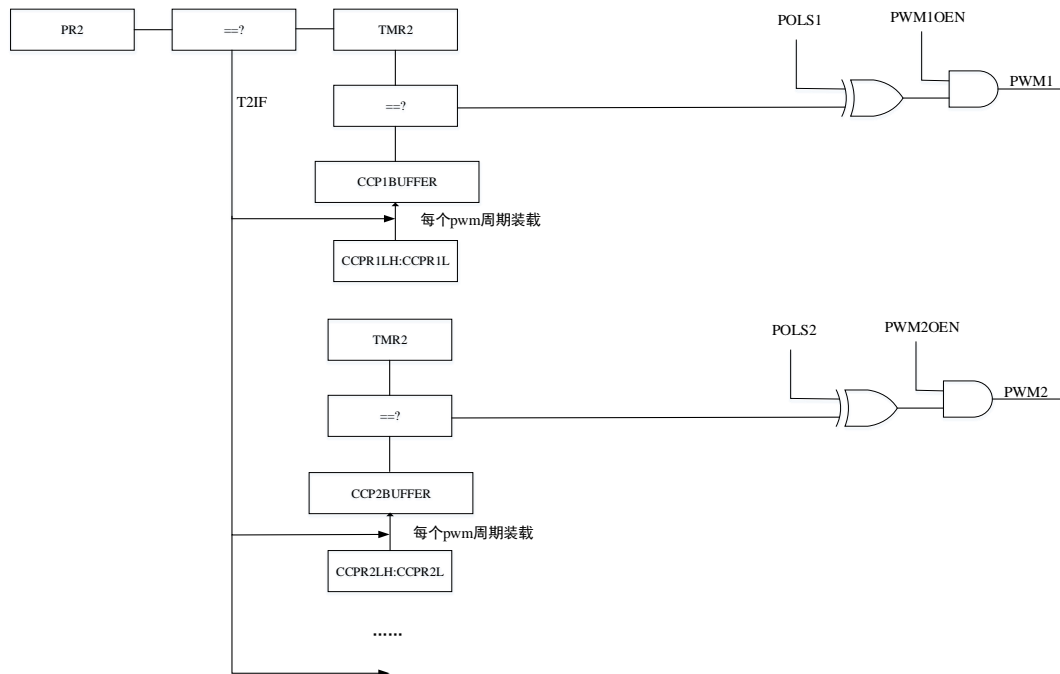


图 8.4 PWM 模块的简化框图

8.5 PWM 相关寄存器

PWM2CON0 (PWM 控制寄存器 0)

地址：0x91

Bit	Name	Description	Attribute	Reset
7	PWM2ACF	PWM2 发生自动关闭 1: 发生自动关闭 0: 未发生自动关闭	R/W	0
6	PWM2ASF	PWM2 发生自动启动 1: 发生自动启动 0: 未发生自动启动	R/W	0
5	PWM21OE	PWM21 输出使能 1: 使能输出 0: 禁止输出	R/W	0
4	PWM20OE	PWM20 输出使能 1: 使能输出 0: 禁止输出	R/W	0

Bit	Name	Description	Attribute	Reset
3:2	Reserved			
1	PWM21OL	PWM21 出极性 1: 正向 0: 反向	R/W	0
0	PWM20POL	PWM20 输出极性 1: 正向 0: 反向	R/W	0

PWM2CON1 (PWM 控制寄存器 1)

地址: 0x92

Bit	Name	Description	Attribute	Reset
7:0	PWM2DLY[7:0]	PWM20 和 PWM21 的死区控制 死区时间计数器死区时间 = (PWM2DLY[7:0])/Fsys	R/W	0x00

PWM2CON2 (PWM 控制寄存器 2)

地址: 0x93

Bit	Name	Description	Attribute	Reset
7	HBFLTEN	当桥产生保护, 根据 ASTART 和 ACLOSE 关闭开启 PWM 1: 使能 0: 禁止	R/W	0
6	FLTEN	Reserved		
5	ASTART	发生故障后(ACLOSE=1), 当使能的故障都解除后, 自动启动 PWM 输出 1: 使能启动 PWM 输出 0: 禁止启动 PWM 输出	R/W	0
4	ACLOSE	发生故障后, 自动关闭 PWM 输出 1:使能关闭 PWM 输出 0:禁止关闭 PWM 输出	R/W	0
3	CMPFLT	CMPOUT 为故障输入, 当从 0 变为 1 后, 禁止 PWM 输出 1: 使能 CMPOUT 作为故障输入 0: 禁止 CMPOUT 作为故障输入	R/W	0
2	IOFLT	PB0 映射为故障 IO, 当从 0 变为 1 后, 禁止 PWM 输出 1: 使能 PB0 作为故障输入 0: 禁止 PB0 作为故障输入	R/W	0
1	PWM20ADPOS	PWM20 跳变沿使能 ADC 采集 1: 使能 PWM20 由 0 变为 1 采集	R/W	0

Bit	Name	Description	Attribute	Reset
		0: 使能 PWM20 由 1 变为 0 采集		
0	PWM20ADEN	使能 PWM20 输出跳变时, 自动启动 ADC 采集功能 1: 使能 ADC 采集 0: 禁止 ADC 采集	R/W	0

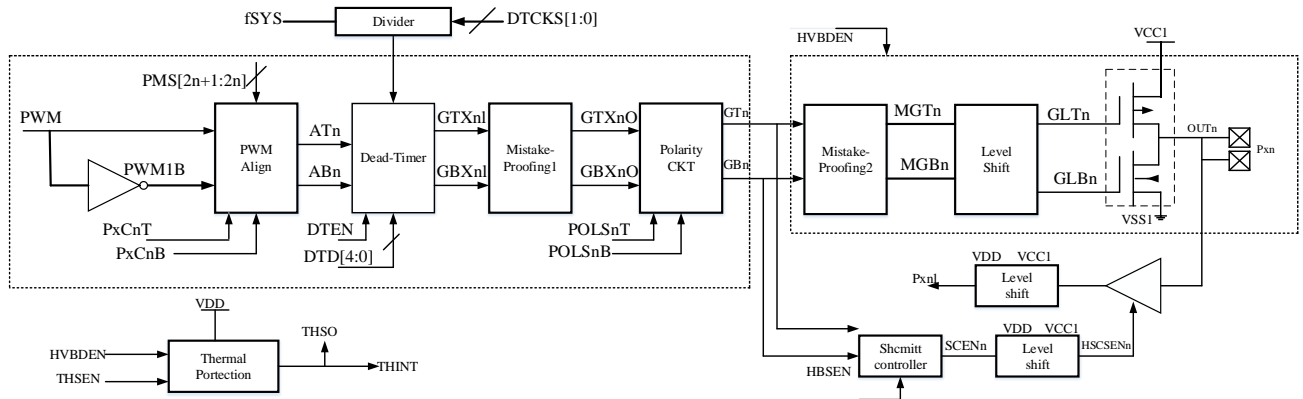
PWM2CON3 (PWM 控制寄存器 3)

地址: 0x94

Bit	Name	Description	Attribute	Reset
7:0	PWMMADDLY[7: 0]	PWM 启动 ADC 采集延时时间 $T = \{PWMMADDLY[7:0]\} * F_{sys}$ F_{sys} 为系统时钟	R/W	0x00

9 驱动电路 H 桥

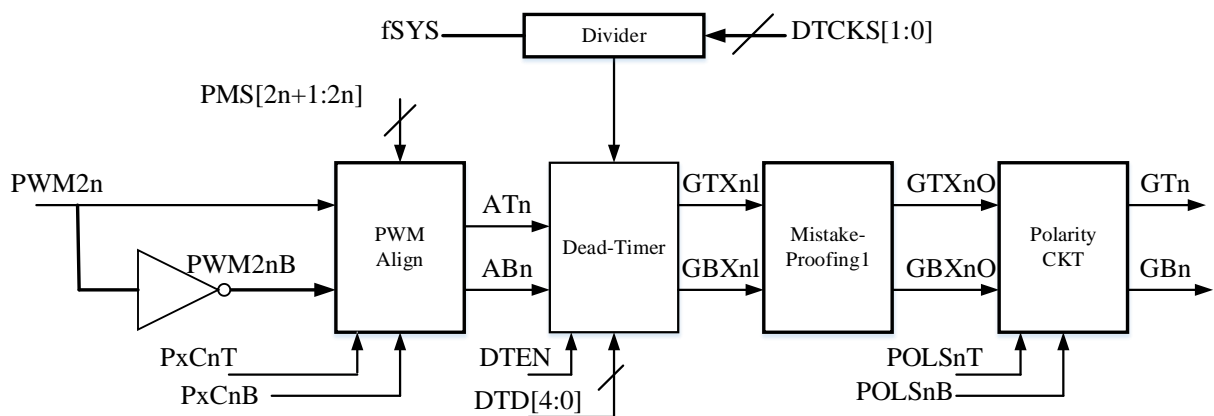
AD16H02 提供片上 H-Bridge 驱动功能，整个结构主要分为互补式输出控制与高压输出驱动两个大区块。



H-Bridge 驱动器结构图 (n=0 代表 H 桥左臂, n=1 代表 H 桥右臂)

9.1 互补式输出

AD16H02 内含两组互补式输出控制电路。每一组由四个主要电路组成即 PWM 调制电路、死区时间控制电路、防呆电路及极性控制电路。



注: PWM2n 表示 CCP 的 PWM 输出信号, PWM21 与 PWM20 同相

9.2 高压控制寄存器

互补式输出控制是通过一系列寄存器实现的。这些寄存器可用于选择 PWM 调制模式、死区时间设置以及输出极性控制等。

PXC 寄存器

地址: 0x19

Bit	Name	Description	Attribute	Reset
7	OUT0HEN	OUT0 上臂控制信号 PB2 输出使能 1: 允许使能 0: 禁止使能	R/W	0

Bit	Name	Description	Attribute	Reset
6	OUT0LEN	OUT0 下臂控制信号 PB3 输出使能 1: 允许使能 0: 禁止使能	R/W	0
5	OUT1HEN	OUT1 上臂控制信号 PB4 输出使能 1: 允许使能 0: 禁止使能	R/W	0
4	OUT1LEN	OUT1 下臂控制信号 PB5 输出使能 1: 允许使能 0: 禁止使能	R/W	0
3:2	PxC1T:PxC1B	OUT1 高压电平转换驱动器的上/下臂输出选择位 00: 上/下臂输出都关闭 01: 上臂输出关闭/下臂输出导通 10: 上臂输出导通/下臂输出关闭 11: 上/下臂输出都关闭（防止上/下臂同时导通） PMS 寄存器的 PMS3~PMS2 位设置为“11”时，OUT1 高压电平转换驱动器的上/下臂输出由 PxC1T:PxC1B 这两个位控制。	R/W	00
1:0	PxC0T:PxC0B	OUT0 高压电平转换驱动器的上/下臂输出选择位 00: 上/下臂输出都关闭 01: 上臂输出关闭/下臂输出导通 10: 上臂输出导通/下臂输出关闭 11: 上/下臂输出都关闭（防止上/下臂同时导通） PMS 寄存器的 PMS1~PMS0 位设置为“11”时，OUT0 高压电平转换驱动器的上/下臂输出由 PxC0T:PxC0B 这两个位控制。	R/W	00

PMS 寄存器

地址：0x18

Bit	Name	Description	Attribute	Reset
7:4	CMPFILTER [3:0]	CMP 的滤波时间 = CMPFILTER[3:0]/F _{cpu} 备注：与 H 桥无关，只与运放章节的 CMP 有关	R/W	0000
3:2	PMS3:PMS2	OUT1 高压电平转换驱动器的 PWM21 调制模式选择位 00: 互补式控制 01: 非互补式上臂调制	R/W	00

Bit	Name	Description	Attribute	Reset
		10: 非互补式下臂调制 11: OUT1 控制（由 PxC1T 和 PxC1B 位分别控制上/下臂输出）		
1:0	PMS [1:0]	OUT0 高压电平转换驱动器的 PWM20 调制模式选择位 00: 互补式控制 01: 非互补式上臂调制 10: 非互补式下臂调制 11: OUT0 控制（由 PxC0T 和 PxC0B 位分别控制上/下臂输出）	R/W	00

DTC 寄存器

地址: 0x1A

Bit	Name	Description	Attribute	Reset
7:6	DTCKS [1:0]	选择死区时间时钟源 f_{DT} （死区时间的时钟源和各种保护的时钟源一致，为避免触发其他保护，此处须设置为 $f_{SYS}/8$ ） 00: $f_{DT}=f_{SYS}$ 01: $f_{DT}=f_{SYS}/2$ 10: $f_{DT}=f_{SYS}/4$ 11: $f_{DT}=f_{SYS}/8$	R/W	11
5	DTEN	死区时间使能/除能控制 1: 允许使能 0: 禁止使能	R/W	0
4:0	DTD [4:0]	死区时间计数器死区时间= $(DTD[4:0]+1)/f_{DT}$ 用于设置上下臂的死区时间	R/W	00000

注：需要先配置寄存器，最后使能死区时间以及整个桥的使能。（桥启动后不要随意更改死区时间）

POLS 寄存器

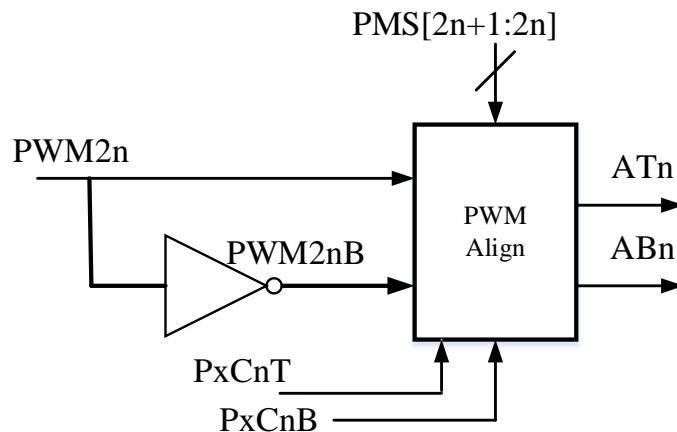
地址: 0x1B

Bit	Name	Description	Attribute	Reset
7	OTPSEL	过温保护阈值 1: Reserved（禁止配置） 0: 111℃	R/W	0
6	PTMODE	保护模式的选择（触发保护后同时控制 H 桥下臂两个	R/W	0

Bit	Name	Description	Attribute	Reset
		MOS 的开关状态) 1: 刹车模式 (H 桥下臂两个 MOS 导通, 上臂两个 MOS 关断) 0: 输出禁止 (H 桥四个 MOS 关断)		
5	OUT0SMT	OUT0 高压输出状态读回信号 1: 高 0: 低	R	0
4	OUT1SMT	OUT1 高压输出状态读回信号 1: 高 0: 低	R	0
3	POLS1T	OUT1 高压电平转换驱动器上臂输出极性控制 1: 反相输出 0: 同相输出	R/W	0
2	POLS1B	OUT1 高压电平转换驱动器下臂输出极性控制 1: 反相输出 0: 同相输出	R/W	0
1	POLS0T	OUT0 高压电平转换驱动器上臂输出极性控制 1: 反相输出 0: 同相输出	R/W	0
0	POLS0B	OUT0 高压电平转换驱动器下臂输出极性控制 1: 反相输出 0: 同相输出	R/W	0

9.3 PWM 调制

用户可选择由单端 PWM 信号、互补式 PWM 信号对或软件设置来驱动 H-Bridge, 可通过 PMS 和 PXC 寄存器中的相关位控制, 如下图及表格说明所示。



注：PWM20 表示 **OUT0** 的控制信号，PWM20B 为 PWM20 的反相
 PWM21 表示 **OUT1** 的控制信号，PWM21B 为 PWM21 的反相
 （n=0 代表 H 桥左臂，n=1 代表 H 桥右臂）

PWM 调制方框图 (n=0 或 1)

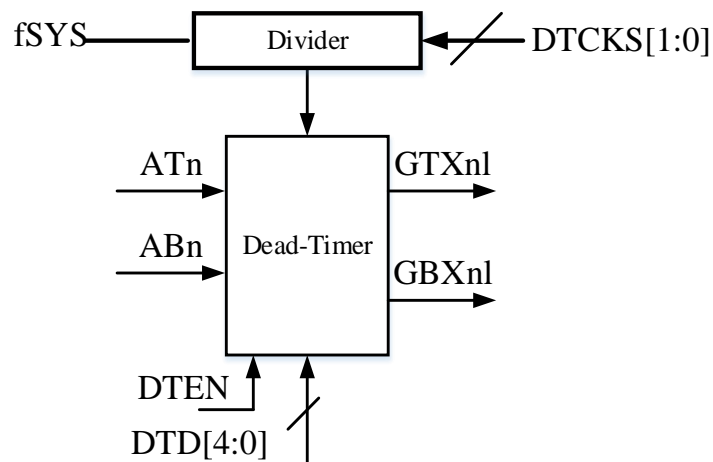
PMS [1:0]	PWM 调制模式	AT0	AB0
00	互补式控制	PWM20	PWM20B
01	非互补式上臂调制	PWM20	0
10	非互补式下臂调制	0	PWM20
11	由 PXC 寄存器控制上下臂输出	PxCnT	PxCnB

PMS [3:2]	PWM 调制模式	AT1	AB1
00	互补式控制	PWM21	PWM21B
01	非互补式上臂调制	PWM21	0
10	非互补式下臂调制	0	PWM21
11	由 PXC 寄存器控制上下臂输出	PxCnT	PxCnB

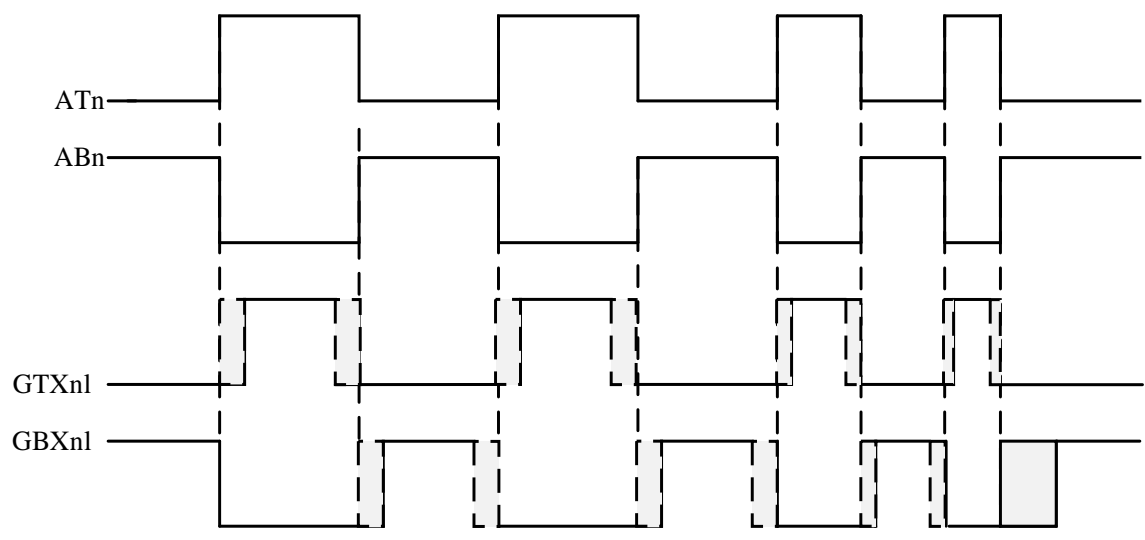
9.4 死区时间


死区时间电路设计的目的为：插入死区时间可确保外部驱动电路晶体管对的上下臂在转态时不会瞬间导通（上下臂 MOS 皆开启）而产生短路电流。为了消除这种危险，设计了一段死区时间，确保输出转态的过程中，两个晶体管不会同时导通。死区时间插入使能或除能由 DTC 寄存器的 DTEN 位控制。死区时间要控制在 $0.3\mu\text{s}\sim 5\mu\text{s}$ 左右，可通过 DTCKS1~DTCKS0 位选择死区时钟源，并通过 DTD4~DTD0 位对插入的死区时间进行调整。

下图为死区时间方框图和插入死区时间时序图。



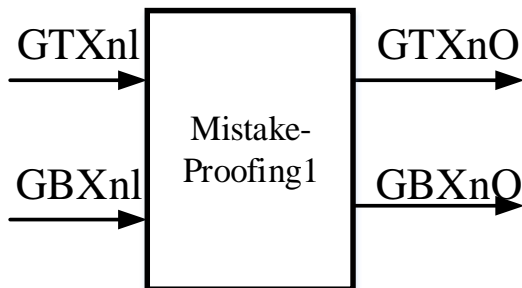
死区时间方框图 (n=0 代表 H 桥左臂，n=1 代表 H 桥右臂)



Note:  Dead-Timer

9.5 互补式输出控制防呆电路

此防呆电路设计的目的为：当软件有误写动作发生，或是因外力因素如 ESD 发生时，导致方向控制的寄存器被打乱，造成外部驱动晶体管对上臂与下臂的输出 MOS 皆为开启的状态，此时防呆电路则强迫输出 MOS 皆为关闭，以保护马达。



防呆电路 (n=0 代表 H 桥左臂, n=1 代表 H 桥右臂)

GTXnl	GBXnl	GTXnO	GBXnO
0	0	0	0
0	1	0	1
1	0	1	0
1	1	0	0

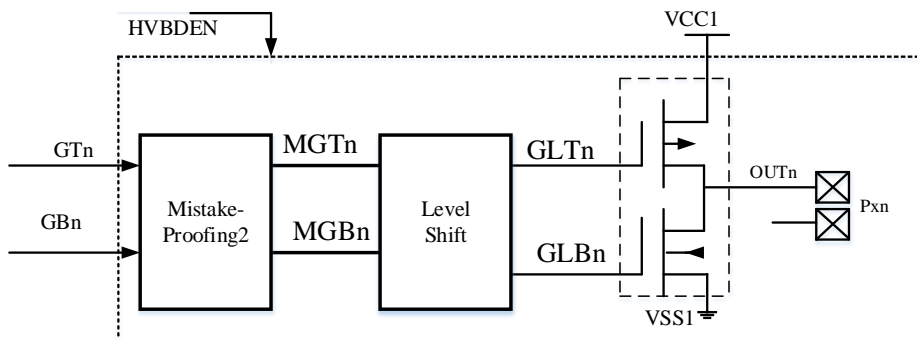
注：表格中的 0 表示 MOS 关闭，1 表示 MOS 开启。

9.6 极性控制

外部 MOS Gate 开关输出是否反相由 POLS 寄存器的 POLSnT 和 POLSnB 位控制。(n=0 代表 H 桥左臂, n=1 代表 H 桥右臂)

9.7 高压输出驱动

共有两组高压输出驱动电路，每组的主要电路为一条防呆电路和一条 2V 高压制程的电平转换电路。经过电平转换后的两条高压线路并联连接至一条高压输出 I/O，可提供直流马达驱动，并依电流需求控制一或两条高压，以支持多种外部驱动电路组合。整个高压输出驱动电路的开启与关闭是由 HVC 的 HVB DEN 位进行控制。



高压输出方框图 (n=0 代表 H 桥左臂, n=1 代表 H 桥右臂)

9.8 高压输出驱动寄存器

高压输出驱动的功能是通过寄存器 HVC 控制。主要进行高压输出驱动电路的开启与关闭、热保护功能设置以及驱动电流控制。

HVC 寄存器

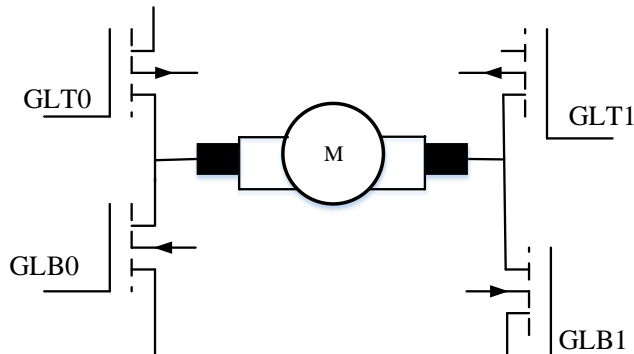
地址：0x98

Bit	Name	Description	Attribute	Reset
7	HBEN	H 桥工作使能 1: 允许使能 0: 禁止使能	R/W	0
6	HBOVPEN	过压保护使能、滤波使能与过压保护延时使能 1: 允许使能 0: 禁止使能	R/W	0
5	HBOCPEN	过流保护及滤波使能 1: 允许使能 0: 禁止使能	R/W	0
4	HVBDEN	高压输出驱动电路开/关控制 (OUT0, OUT1 输出与设置不匹配产生保护总控位, 为防止输出异常建议打开) 1: 开启高压输出电路 0: 关闭整个高压输出电路	R/W	0
3	HBOTPEN	热保护及滤波使能 (默认: 111°C) 1: 允许使能 0: 禁止使能设置 THSEN=0, 热保护功能关闭。 若高压输出驱动电路也需要关闭, 则必须先设置 HBOTPEN =0 再设置 HVBDEN=0。 热保护功能开启时, 需先设置 HBOTPEN =1 再设置 HVBDEN=1。	R/W	0
2	HBMPEN	OUT0, OUT1 输出与设置不匹配产生保护及滤波使能 1: 允许使能 0: 禁止使能	R/W	0
1	HBCMAEN	OUT0L 电流采集使能 1: 允许使能 0: 禁止使能	R/W	0
0	HBCMBEN	OUT1L 电流采集使能 1: 允许使能 0: 禁止使能	R/W	0

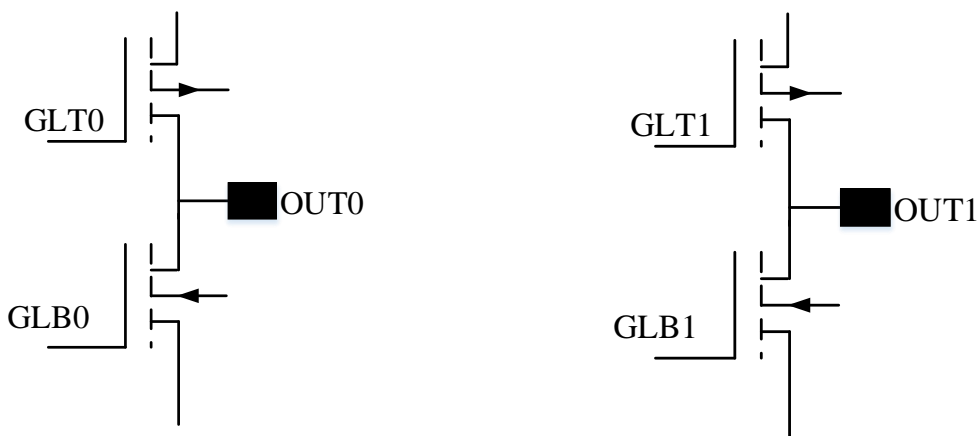
9.8.1. 高压输出驱动应用电路

高压输出驱动电路可针对不同产品应用需求提供各种不同的使用，依据不同电流大小（最大支持 1A）来驱动外部元件，且 PMOS/NMOS 独立开关可有多样性的组合。以下列举两种应用组合：

H-Bridge 组合：（直驱 DC Motor）

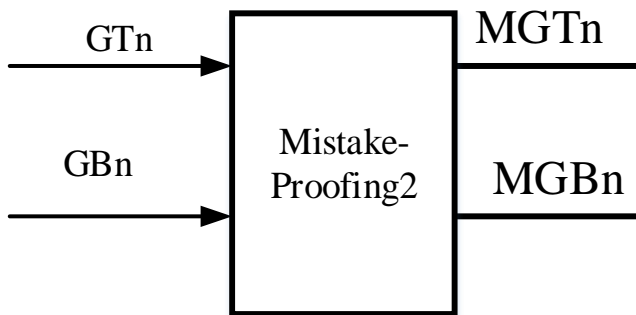


PMOS/NMOS 独立使用



9.8.2. 高压输出驱动防呆电路

此防呆电路设计的目的为，防止当软件有误写动作发生或是因外力因素，如 ESD 发生时，导致方向控制的寄存器被打乱，造成外部驱动晶体管对上臂与下臂的输出 MOS 皆为开启的状态。



防呆电路图（n=0 或 1）

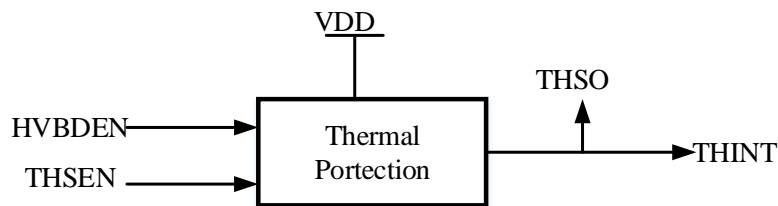
GTn	GBn	MGTn	MGBn
0	0	0	0
0	1	0	1
1	0	1	0
1	1	0	1

注：表格中的 0 表示 MOS 关闭，1 表示 MOS 开启。

9.8.3. 高压输出驱动热保护

该高压输出驱动电路内建热保护功能。THSEN 位用于控制热保护功能的开启与关闭。为避免热保护功能异常操作，当热保护功能开启时，需先设置 THSEN 位为 1，再设置 HVBDEN 为 1；当高压输出驱动电路关闭时，需先将 THSEN 位清零，再清零 HVBDEN 位。

若使能热保护功能，THIF 位用于表明是否有过温情况发生。当温度超过设定的温度值时，此位将从 0 变为 1，表示过温情况发生。此外，还会置位中断控制寄存器内相应的热保护中断请求标志位。如果中断使能，就会产生对应的内部中断信号，以通知 MCU 进行处理。



热保护方框图

9.8.4. 高压输出回读电路

高压输出回读电路主要是由施密特触发器与电平转换器所组成，其目的是通 HBC 寄存器的 PxnI 位将实际输出的 Pxn 状态读取回来与驱动信号 GLTn 与 GLBn 做对比。当 GLTn 与 GLBn 为关闭的状态时 Pxn 会呈现浮空的状态，造成了施密特触发器与电平转换器产生漏电流。为了改善这个问题，设置了施密特控制器。当施密特控制器除能时，施密特触发器始终使能，此时上面所述的状况可能发生。若将施密特控制器使能，在 GLTn 与 GLBn 都关闭的状态时可将施密特触发器关闭，以防止漏电流的发生。

详见高压输出驱动寄存器描述。

HBC0 寄存器

地址：0x99

Bit	Name	Description	Attribute	Reset
7	OCPV	过流保护状态位 1: 触发过流保护 0: 未触发过流保护	R	0

Bit	Name	Description	Attribute	Reset
6	OTPV	过温保护状态位 1: 触发过温保护 0: 未触发过温保护	R	0
5	HVS0	HVS 输入分压, 分压后结果应该小于 1/2 VDD 电压 (可以使用 ADC 采集 H 桥分压后的电源) 1: 1/32 分压 0: 1/16 分压	R/W	0
4	HVSEN	HVSEN 输入分压使能 1: 使能 0: 禁止	R/W	0
3:2	OUT1H:OUT1L	OUT1 上下管的输出状态位 1: 开启 0: 关闭	R	00
1:0	OUT0H:OUT0L	OUT0 上下管的输出状态位 1: 开启 0: 关闭	R	00

如上图所示, 施密特控制器输入源为极性控制电路所输出的 GT_n 与 GB_n 。施密特控制器的控制开关为 HVC 寄存器中的 HBSEN 位。而输出信号 $SCEN_n$ 经过电平转换器后得到 $HSCEN_n$, 再使用 $HSCEN_n$ 作为施密特触发器的开关。由下表可以得知施密特控制器使能或除能时其输入与输出的关系。

HBSEN	GT_n	GB_n	$SCEN_n$ ($HSCEN_n$)
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	x	x	1

HBC1 寄存器

地址: 0x9A

Bit	Name	Description	Attribute	Reset
7:4	OTPF [3:0]	过温保护滤波时间 = $OTPF[3:0]/F_{DT}$	R/W	0
3:0	OCPF [3:0]	过流保护滤波时间 = $OCPF[3:0]/F_{DT}$	R/W	0

HBC2 寄存器

地址: 0x9B

Bit	Name	Description	Attribute	Reset
7:5	OVPF [2:0]	过压保护滤波时间 = $OVPF[2:0]/F_{DT}$	R/W	0
4:0	OVPDLY [4:0]	过压保护时间延时 = $OVPDLY[4:0]/F_{DT}$	R/W	0

HBC3 寄存器 (MAPEN=1)

地址: 0x97

Bit	Name	Description	Attribute	Reset
7:4	HBMPF[3:0]	OUT0, OUT1 输出与设置不匹配产生保护滤波时间 = HBMPF[3:0]/F _{DT}	R/W	0000
3	HBMCEN	OUT0, OUT1 输出与设置不匹配, 使能保护开关 1: 允许使能 0: 禁止使能	R/W	0
2	OCPEN	过流保护开关使能 1: 允许使能 0: 禁止使能	R/W	0
1	OVPEN	过压保护开关使能 1: 允许使能 0: 禁止使能	R/W	0
0	OTPCEN	过温保护开关使能 (默认: 111°C) 1: 允许使能 0: 禁止使能	R/W	0

注:

1. 需要先配置 HVC 寄存器中对应的保护与滤波使能, 再使能 HBC3 中对应的保护功能才能启动过压, 过温, 过流, OUT0, OUT1 输出与设置不匹配保护;

2. 在用到 H 桥的应用中, 为将电机产生的反向电动势对 MCU 上电过程产生的干扰降到最小使得 MCU 正常启动, 建议电机快速转动前先让电机转一下占空比 1%, 10ms 即可 (仅上电时需要);

例:

如果要启动过流保护, 先对 HVC 寄存器中的 HBOVPEN 置位, 再设置 HBC1 寄存器中的 OCPF[3:0], 最后对 HBC3 寄存器中的 OCPEN 置位, 正确启动 H 桥过流保护

H 桥启停推荐程序:

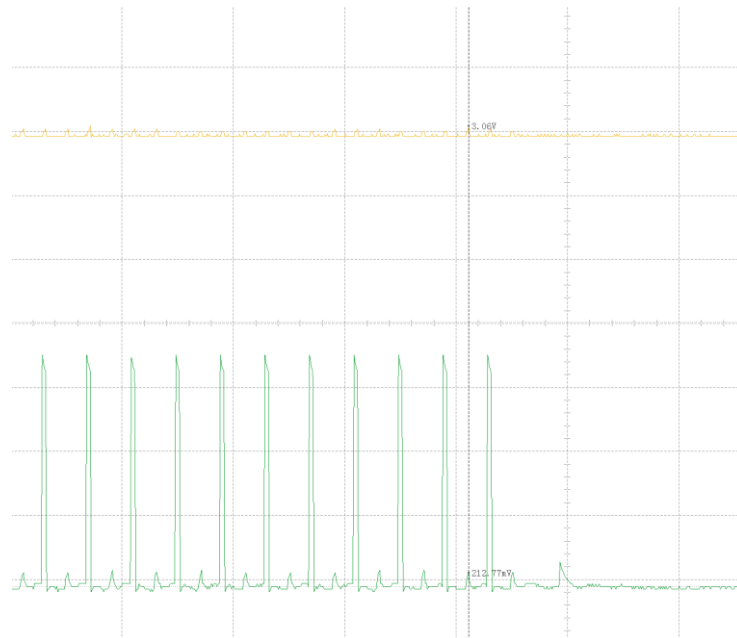
启动 H 桥程序:

```
void StartMotor(void)
{
    PMS = 0X0F;
    PXC = 0X05;    //上臂关闭, 下臂导通
    T2IF = 0;
    TMR2ON = 1;   //启动 T2
    HVC = 0X90;   //使能桥输出
    while(!T2IF);T2IF = 0;
    while(!T2IF);T2IF = 0;
    T2IE = 1;
}
```

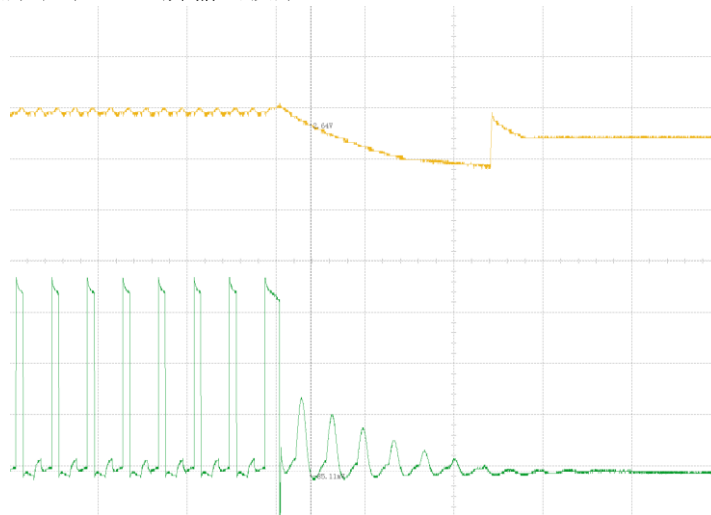
停止 H 桥程序：

```
void StopMotor(void)
{
    T2IE = 0; T2IF = 0;
    PMS = 0X0F;
    PXC = 0X05;    //上臂关闭，下臂导通
    while(!T2IF);T2IF = 0;
    while(!T2IF);T2IF = 0;
    delay_100us(100);
    TMR2ON = 0;    //关闭 T2
    HVC = 0X00;    //关闭桥
}
```

启停 H 桥时请参考上述程序，输出波形如下图，图中黄色波形表示 VDD，绿色波形表示 OUT 脚输出波形：



如果未使用上述程序，直接启停会导致 VDD 掉电，芯片发生复位。输出波形如下图所示，图中黄色波形表示 VDD，绿色波形表示 OUT 脚输出波形：



10 省电模式 (SLEEP)

在三种睡眠模式下，单片机能通过以下方式唤醒：如下图所示

注：√表示可唤醒的方式

模块	睡眠模式		
	IDLE	PWSAVE	PWOFF
CCP	√	√	
TIMER0	√		
TIMER1	√	√	
TIMER2	√		
WDT	√	√	√
RESET	√	√	√
INT	√	√	√
PB	√	√	√
LVD	√	√	√

下图为各模块在正常模式和睡眠模式下的工作状态。

注：√表示在该模式可以正常运行。

模块	正常模式	睡眠模式		
	ACTIVE	IDLE	PWSAVE	PWOFF
OSC16M	√	√		
OSC256K	√	√	√	√ (WDTON)
CPU	√			
OTP	√	√	√	
SRAM	√	√	√	√
TIMER0	√	√		
TIMER1	√	√	√	
TIMER2	√	√		
CCP	√	√		
WDT	√	√	√	√
INT	√	√	√	√
LDO	√	√	√	
BGR	√	√	√	
LVD	√	√	√	
ADC	√	√		
DAC	√	√		
POR	√	√	√	√
IO	√	√	√	√
RESET	√	√	√	√

AD16H02 拥有三种睡眠模式：

00: 正常工作模式

01: IDLE 模式, CPU 进入睡眠, 外设继续运行, 保存 SRAM 数据, 唤醒后打开中断会进入中断, 否则顺序执行

10: PWSAVE 模式, 进入睡眠, 保存 SRAM 数据, 唤醒后打开中断会进入中断, 否则顺序执行

11: PWOFF 模式, 进入深度睡眠, 保存 SRAM 数据, 唤醒后顺序执行

执行 SLEEP 指令以后机器进入省电模式, I/O 维持原状, /PD 位清零(STATUS<3>), /TO 位置 1, 看门狗清零同时保持运行状态, 高速晶体停振。

10.1 睡眠唤醒

在睡眠状态下, 单片机能通过以下方式唤醒:

1. RSTn 管脚复位
2. 看门狗复位 (机器设置了看门狗)。
3. INT 管脚中断, 或 PORTB 输入改变中断。
4. TIMER0 中断唤醒, TIMER1 中断唤醒
5. 在睡眠状态下, PWOFF 模式, 单片机能通过以下方式唤醒:
6. RSTn 管脚复位

外部的 RSTn 管脚和看门狗溢出都能使机器复位, 通过查看/PD 和/TO 位可以检测芯片产生的是那种复位, /PD 位置 1 为上电复位, 置 0 为执行 SLEEP, /TO 位置 0 为看门狗溢出复位。芯片通过中断唤醒, 该中断屏蔽位置 1, 中断唤醒与 GIE 是否使能无关。当 GIE 位被清零, 机器唤醒以后执行 SLEEP 指令以后的指令; 当 GIE 位被置 1, 机器唤醒以后跳转到中断复位地址 (008h)。

注: 1.睡眠前, 请再次将唤醒源配置使能。

2.所有空闲的 IO 应改为固定状态, 如果外部无上下拉, IO 不能设为悬空状态, 唤醒 IO 除外。

3.进入睡眠前, 请定义全局变量读取所有 IO 口进行锁存。

10.1.1. SMCR 的相关寄存器

SMCR (状态控制寄存器)

地址: 0x9F

Bit	Name	Description	Attribute	Reset
7	IDLE	Bit[0]: Bit[7] PWOFF, IDLE 00: 不支持 01: CPU 进入睡眠, 外设继续运行 10: 进入睡眠, 保存 SRAM 数据 11: 进入深度睡眠, 保存 SRAM 数据	R/W	0
6	OSCM	系统时钟选择 1: 选择内部高速时钟 16MHz 作为系统时钟	R/W	1

Bit	Name	Description	Attribute	Reset
		0: 选择内部低速时钟 256KHz 作为系统时钟		
5	Reserved			
4:3	OP0PGA [1:0]	内部放大倍数选择 00: 6.25 倍 01: 12.5 倍 10: 25 倍 11: 50 倍	R/W	00
2	CKOE	PB3 输出当前系统时钟 1: 使能系统时钟输出 0: 禁止系统时钟输出	R/W	0
1	OSCF	内部工作时钟状态位 1: 表示高速时钟工作 (16MHz) 0: 表示低速时钟工作 (256KHz)	R	1
0	PWOFF	Bit[0]: Bit[7] PWOFF, IDLE 00: 不支持 01: CPU 进入睡眠, 外设继续运行 10: 进入睡眠, 保存 SRAM 数据 11: 进入深度睡眠, 保存 SRAM 数据	R/W	0

11 中断方式

AD16H02 系统具备有五种中断方式:

1. INT 管脚的外部中断
2. TMR0 溢出中断
3. TMR1 溢出中断
4. TMR2 溢出中断
5. Port B 输入改变中断 (PB7:PB0 脚)

PIR0、PIR1 为中断标志寄存器，决定该寄存器机器所发生的中断状态。

中断允许总控位 GIE (PIE0<7>), 能使后所有中断被开放 (GIE=1) 或屏蔽后所有中断被关闭(GIE=0), 每个中断能否启用通过 PIE0、PIE1 寄存器决定，同时保证 GIE 置 1。

中断发生时 GIE 位 (在中断发生前 GIE 位和该中断相关的中断屏蔽位置 1) 被硬件清零从而禁止进一步中断 (AD16H02 不区分中断优先级别), 同时下条指令跳到(004h/008h, 根据 MCUSEL 配置位决定)后开始执行。中断标志位在中断允许总控位 GIE 重新置 1 的时候需要被软件清零以防止重复中断。一个中断标志位 (PBIF 除外的) 会被它的中断事件置 1, 而不管与它相关的中断屏蔽位是否启用。通过 PIR0 和 PIE0 的相应中位来判断是否发生中断以及中断类型。当通过 INT 指令发生软中断时, 下条指令跳到(001/002, 根据 MCUSEL 配置位决定)后开始执行。

11.1 外部中断

外部中断 INT 管脚上升沿还是下降沿触发由 INTEDG 位 (在 TICON1 寄存器)决定, 当一个有效的跳变发生时标志位 INTIF 置 1。当 INTIE 位清零, 外部中断被关闭。

在睡眠之前 INTIE 位已被置 1, INT 管脚可以作为系统睡眠条件。在睡眠之前 GIE 位已被置 1 机器唤醒以后会执行中断服务程序, 否则会运行睡眠以后的下一条指令。

11.2 定时器中断

定时器 T0、T1、T2 在计数溢出时会产生中断请求 T0IF、T1IF、T2IF。它们有各自的使能位 T0IE、T1IE、T2IE, 当其被置 0 时, 其对应的中断功能将被关闭。

11.3 Port B 输入改变中断

输入改变中断触发时 PB<7:0> PBIF 标志位置 1 (PIR0<1>)。PBIE 位(PIE0<1>)清零, 则关闭该中断功能。

在输入改变中断发生之前, 必须读取 Port B 信息与 PortB 的管脚相对应的 WUBn 位 (WUBCON<7:0>) 清零或设置为输出。

PBIE 在睡眠之前置 1, Port B 输入脚改变中断也可以作为睡眠唤醒条件。在睡眠之前 GIE 位已被置 1 机器唤醒以后会执行中断服务程序, 否则会运行睡眠以后的下一条指令。

11.4 中断的相关寄存器

PIE0 (中断屏蔽寄存器)

地址: 0x0E

Bit	Name	Description	Attribute	Reset
7	GIE	<p>中断允许总控位</p> <p>0: 禁止所有中断. 对于睡眠唤醒模式的中断事件, MCU 将执行 SLEEP 后的指令。</p> <p>1: 使能所有没有屏蔽的中断. 对于睡眠唤醒模式的中断事件, MCU 将跳转到中断地址 (004h/008h, 根据 MCU_SEL 配置位选择)。</p>	R/W	0
6	T2IE	<p>Timer2 中断屏蔽位。</p> <p>0: 禁止 TIMER2 溢出中断</p> <p>1: 使能 TIMER2 溢出中断</p>	R/W	0
5	T1IE	<p>Timer1 中断屏蔽位。</p> <p>1: 使能 TIMER1 溢出中断</p> <p>0: 禁止 TIMER1 溢出中断</p>	R/W	0
4	ADIE	<p>AD 外部中断屏蔽位</p> <p>1: 使能外部中断</p> <p>0: 禁止外部中断.</p>	R/W	0
3	CMPIE	<p>CMP 外部中断屏蔽位</p> <p>1: 使能外部中断</p> <p>0: 禁止外部中断.</p>	R/W	0
2	CCPIE	<p>CCP 中断屏蔽位。</p> <p>1: 使能外部中断</p> <p>0: 禁止外部中断.</p>	R/W	0
1	PBIE	<p>Port B 输入改变中断屏蔽位</p> <p>1: 使能 Port B 输入改变中</p> <p>0: 禁止 Port B 输入改变中</p>	R/W	0
0	TOIE	<p>Timer0 溢出中断屏蔽位。</p> <p>1: 使能 Timer0 溢出中断</p> <p>0: 禁止 Timer0 溢出中断</p>	R/W	0

PIR0 (中断标志寄存器)

地址: 0x0F

Bit	Name	Description	Attribute	Reset
7	MAPEN	访问 BANK1 寄存器。 MAPEN=0, BANK1 成为 BANK10 寄存器组, MAPEN=1, BANK1 称为 BANK11 寄存器组,	R/W	0
6	T2IF	溢出中断标志, 发生 Timer2 溢出中断置 1, 软件设置清零	R/W	0
5	T1IF	溢出中断标志, 发生 Timer1 溢出中断置 1, 软件设置清零	R/W	0
4	ADIF	ADC 外部中断标志, 发生中断置 1, 软件设置清零	R/W	0
3	CMPIF	比较器中断标志, 当 CMP 从 0 变为 1 时, 标志位置 1, 软件设置清零	R/W	0
2	CCPIF	CCP 中断标志, 发生中断置 1, 软件设置清零	R/W	0
1	PBIF	PB 口的中断标志位 1:至少一个 PORTB 引脚的电平状态发生了改变 (必须用软件清零, 写 0 清 0) 0:没有一个 PORTB 引脚的电平状态发生改变	R/W	0
0	T0IF	溢出中断标志, 发生 Timer0 溢出中断置 1, 软件设置清零	R/W	0

PIE1(中断屏蔽寄存器)

地址: 0x89

Bit	Name	Description	Attribute	Reset
7	HBIE	桥的输出与逻辑输出是否相等产生的中断允许位 1: 允许中断 0: 禁止中断	R/W	0
6	LVDIE	低电压中断允许位 1: 允许中断 0: 禁止中断	R/W	0
5	OVPIE	过压保护中断允许位 1: 允许中断 0: 禁止中断	R/W	0
4	OCPIE	过流保护中断允许位 1: 允许中断 0: 禁止中断	R/W	0
3	THIE	过温中断允许位 1: 允许中断 0: 禁止中断	R/W	0
2	INT2IE	外部中断 2 允许位 1: 允许 INT2 中断 0: 禁止 INT2 中断	R/W	0
1	INT1IE	外部中断 1 允许位 1: 允许 INT1 中断 0: 禁止 INT1 中断	R/W	0
0	INT0IE	外部中断 0 允许位 1: 允许 INT0 中断 0: 禁止 INT0 中断	R/W	0

PIR1 (中断标志寄存器)

地址: 0x09

Bit	Name	Description	Attribute	Reset
7	HBIF	桥的输出与逻辑输出是否相等产生的中断请求标志位 1: 相等, 中断请求 0: 不相等, 无请求	R	0
6	LVDFIF	LVD 的中断标志 当 LVDM[1:0]=2'b01 时 1: LVD 检测电压低于设置阈值电压(必须由软件清零) 0: LVD 检测电压高于设置阈值电压, 或已经由软件清 0 当 LVDM[1:0]=2'b10 时 1: LVD 检测电压高于设置阈值电压(必须由软件清零) 0: LVD 检测电压低于设置阈值电压, 或已经由软件清 0 LVDM[1:0]在 PCON 寄存器中	R	0
5	OVPIF	过压保护中断请求标志位 1: 中断请求 0: 无请求	R	0
4	OCPIF	过流保护中断请求标志位 1: 中断请求 0: 无请求	R	0
3	THIF	热保护中断请求标志位 1: 中断请求 0: 无请求	R	0
2	INT2IF	INT2 外部中断标志位 1: 产生 INT2 外部中断标准位 0: 未产生 INT2 外部中断标志位	R	0
1	INT1IF	INT1 外部中断标志位 1: 产生 INT1 外部中断标准位 0: 未产生 INT1 外部中断标志位	R	0
0	INT0IF	INT0 外部中断标志位 1: 产生 INT0 外部中断标准位 0: 未产生 INT0 外部中断标志位	R	0

12 固定参考电压 (FVR)

固定参考电压或 FVR 是稳定的参考电压，独立于 VDD，可选择 1.0V（2.0V）两种输出级别。可配置 FVR 的输出为以下各项提供参考电压：

- ADC 输入通道
- ADC 正参考电压
- 运放 0、1 正反向输入

12.1 FVR 的相关寄存器

FVRCON0 寄存器

地址：0x8E

Bit	Name	Description	Attribute	Reset
7:3	Reserved			
2	FVROUTEN	FVR 输出 IO 使能 (PB6) 1: 使能 FVR 输出 0: 禁止 FVR 输出	R/W	0
1	FVRPGA	FVR 输出电压放大两倍控制位 1: 使能放大 0: 禁止放大	R/W	0
0	FVREN	FVR 使能 1: 使能 FVR 0: 禁止 FVR	R/W	0

FVRCON1 寄存器

地址：0x8F

Bit	Name	Description	Attribute	Reset
7:5	Reserved			
4:0	FVR_SEL [4:0]	FVR 电压选择位 00000: 1.0V 00001: DACOUT 00010: NTC 00011: VIN_DIV(VIN 分压 1/16 或 1/32) 00100: HCOOUT(电流采集 1:1 瞬态输出) 00101: HCOOUT_AVG(电流采集 1:1 平均输出) 其他未使用 VIN 分压 1/16 或 1/32 需要配置 HBC0[5]	R/W	0

13 运算放大器和比较器

AD16H02 提供 2 个轨到轨运放，可以选取多个输入作为运放/比较器输入。

13.1 运放 OP0

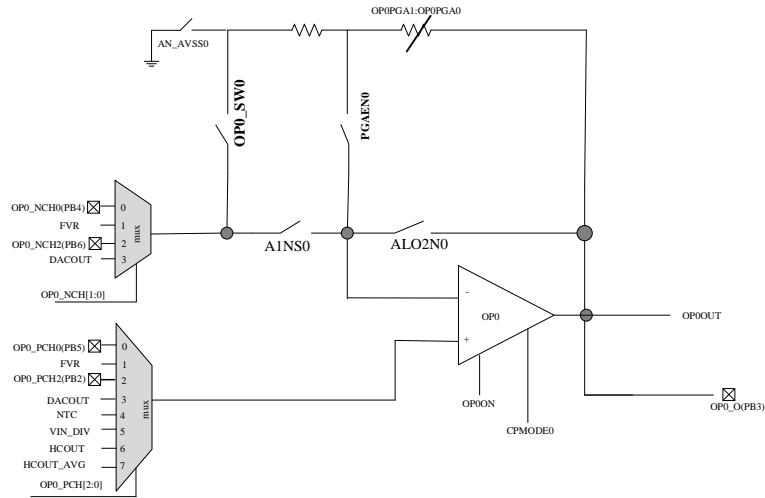


图 13.1 运放 OP0 结构图

OP0CON0(OP 控制寄存器)

地址：0x14

Bit	Name	Description	Attribute	Reset
7	OP0OUT	运放输出信号 1: 运放输出为 0 0: 运放输出为 1	R	0
6	LVDST	LVR 比较器输出值，与 OP 模块无关，与 LVD 模块相关 1: 电源电压高于设定值 0: 电源电压低于设定值	R	0
5	AN_AVSS0	是否接地 1: 接地 0: 不接地	R/W	0
4	OP0_PCH2	与 OP0_PCH[1:0]组成 OP0 的正端输入	R/W	0
3	PGAEN0	OP0 的内部使能放大 1: 使能 0: 禁止	R/W	0
2	Reserved			
1	CPMODE0	运放切换为比较器模式 1: 切换为比较器模式 0: 切换为运放模式	R/W	0
0	OP0_A2D_EN	OP0 使能输出位 (OP0OUT) 1: 运放输出至寄存器中 0: 运放输出不在寄存器保存	R/W	0

注：运放 OP0 的放大倍数在 SMCR 寄存器的比特位中

OP0CON1 (运放控制寄存器)

地址: 0x15

Bit	Name	Description	Attribute	Reset
7	OP0ON	运放使能 1: 使能运放 0: 关闭运放	R/W	0
6	OP0_SW0	运放内部开关 1: 闭合开关 0: 断开开关	R/W	0
5	A1NS0	运放连接 1: 负端与 NCH 选择连接 0: 断开	R/W	0
4	A1O2N0	运放 buffer 模式 1: 负端与输出短接, 形成 BUFFER 0: 禁止	R/W	0
3:2	OP0_PCH [1:0]	运放正端输入选择, 正端由 OP0CON0 的 BIT[3:2]以及 OP0CON0 的 BIT4 组成。 000: OP0_PCH0 (PB5) 001: FVROUT 010: OP0_PCH2 (PB2) 011: DACOUT 100: NTC 101: VIN_DIV (VIN 分压 1/16 或 1/32) 110: HCOUT (电流采集 1:1 瞬态输出) 111: HCOUT_AVG (电流采集 1:1 平均输出)	R/W	000
1:0	OP0_NCH [1:0]	运放负端输入选择 00: OP0_NCH0 (PB4) 01: FVROUT 10: OP0_NCH2 (PB6) 11: DACOUT	R/W	00

13.2 运放 OP1

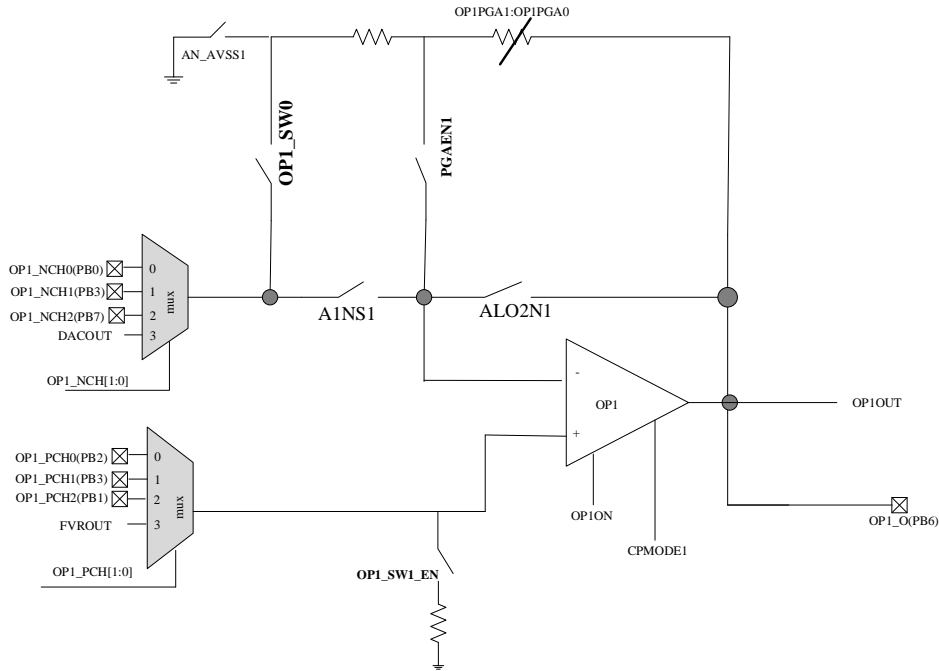


图 13.2 运放 OP1 结构图

OP1CON0 (OP1 控制寄存器, MAPEN=1)

地址: 0x8D

Bit	Name	Description	Attribute	Reset
7	OP1OUT	运放输出信号 1: 运放输出为 0 0: 运放输出为 1	R	0
6	AN_AVSS1	是否接地 1: 接地 0: 不接地	R/W	0
5:4	OP1PGA [1:0]	内部放大倍数选择 00: 6.25 倍 01: 12.5 倍 10: 25 倍 11: 50 倍	R/W	0
3	PGAEN1	OP1 的内部使能放大 1: 使能 0: 禁止	R/W	0
2	OP1_SW_EN	OP1 的下拉电阻连接的开关是否导通 1: 导通 0: 不导通	R/W	0
1	CPMODE1	运放切换为比较器模式 1: 切换为比较器模式 0: 切换为运放模式	R/W	0
0	OP1_A2D_EN	OP1 使能输出位(OP1OUT) 1: 允许使能 0: 禁止使能	R/W	0

OPICON1 (OP1 运放控制寄存器, MAPEN=1)

地址: 0x8E

Bit	Name	Description	Attribute	Reset
7	OP1ON	运放使能 1: 使能运放 0: 关闭运放	R/W	0
6	OP1_SW	运放内部开关 1: 闭合开关 0: 断开开关	R/W	0
5	A1NS1	运放连接 1: 负端与 NCH 选择连接 0: 断开	R/W	0
4	A1O2N1	运放 buffer 模式 1: 负端与输出短接, 形成 BUFFER 0: 禁止	R/W	0
3:2	OP1_PCH [1:0]	运放正端输入选择 00: OP1_PCH0(PB2) 01: OP1_PCH1(PB3) 10: OP1_PCH2(PB1) 11: FVROUT	R/W	0
1:0	OP1_NCH [1:0]	运放负端输入选择 00: OP1_NCH0(PB0) 01: OP1_NCH1(PB3) 10: OP1_NCH2(PB7) 11: DACOUT	R/W	0

13.3 比较器 CMP

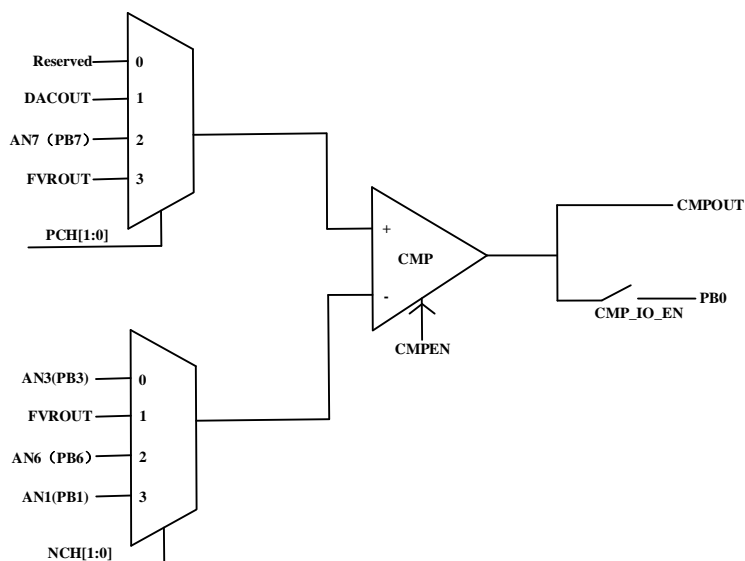


图 13.3 比较器结构图

13.3.1. CMP 的相关寄存器

CMPCON (MAPEN=1, 比较器控制寄存器)

地址: 0x8C

Bit	Name	Description	Attribute	Reset
7	CMPEN	比较器的使能 1: 使能 CMP 0: 禁止 CMP	R/W	0
6	CMPOUT	CMP_IO_EN 使能之后, 通过 PB0 输出比较器滤波后的波形;	R	0
5	CMP_IO_EN	比较器对应的 IO 使能(PB0) 1: 使能 IO 输出 0: 禁止使能	R/W	0
4	CMMPOS	比较器输出信号是否取反 1: 取反 0: 同向	R/W	0
3:2	CMP_PCH [3:2]	比较器的正端输入 00: Reserved 01: DACOUT 10: AN7(PB7) 11: FVROUT	R/W	00
1:0	CMP_NCH [3:2]	比较器的负端输入 00: AN3 (PB3) 01: FVROUT 10: AN6(PB6) 11: AN1(PB1)	R/W	00

备注: CMP 的滤波比特位在 2.7 小节的 PMS 寄存器中

14 AD 转换

AD16H02 包含一个 16 通道输入的 12 位 ADC，能够将一个模拟输入转换成 12 位数字信号。在根据需要配置好 A/D 模块后，必须在转换开始之前对选定的通道进行采样。采集时间一结束，即可启动 A/D 转换。当 A/D 转换完成之后，转换结果被装入 ADRESH:ADRESL 寄存器对，GO/DONE 位被清零且 A/D 中断标志位 ADIF 被置 1。

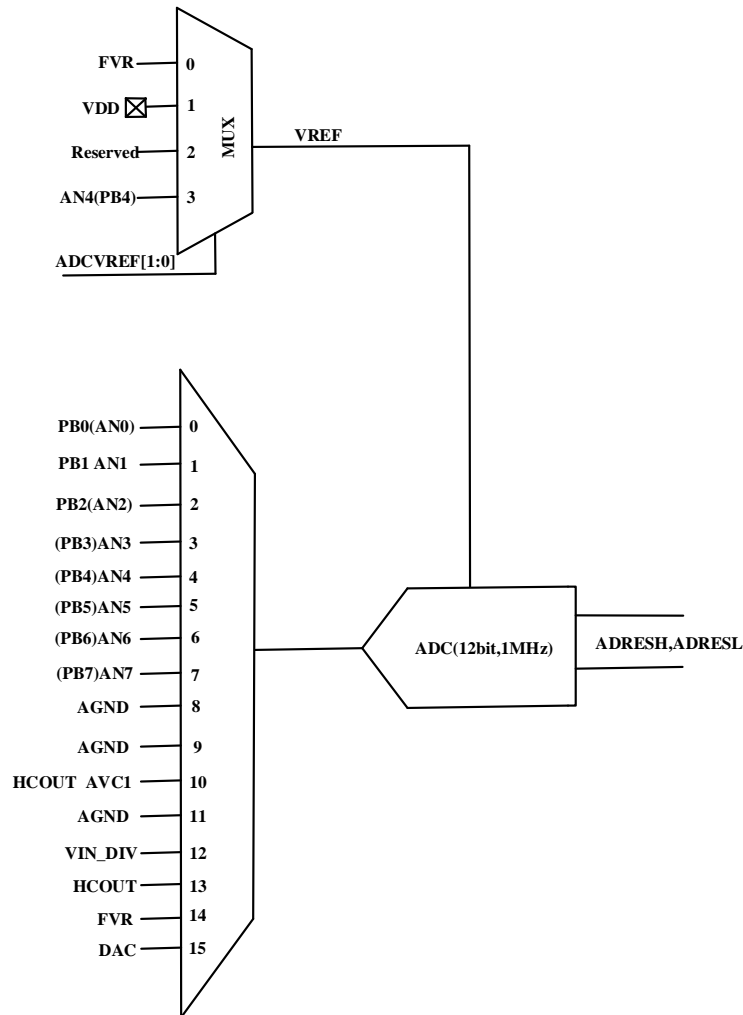


图 14.1 A/D 结构图

执行 A/D 转换步骤：

- 1) 配置 A/D 模块：
 - 选择参考电压（通过 ADCON1[7:6]寄存器）
 - 选择 A/D 输入通道（通过 ADCON0[5:2]寄存器）
 - 选择 A/D 采集时间（通过 ADCON1[5:3]寄存器）
 - 选择 A/D 转换时间（通过 ADCON1[2:0]寄存器）
 - 使能 A/D 模块（通过 ADCON0[0]寄存器）
- 2) 需要时，配置 A/D 中断：
 - 清零 ADIF 位
 - 将 ADIE 位置 1
 - 将 GIE 位置 1
- 3) 如果需要，需等待所需的采集时间：

- 4) 启动转换:
 - 将 $\overline{GO/DONE}$ 位置 1(ADCON0[1])
- 5) 等待 A/D 转换完成, 通过以下两种方式之一判断转换是否完成:
 - 查询 $\overline{GO/DONE}$ 位是否被清零
 - 等待 A/D 中断
- 6) 读取 A/D 结果寄存器(ADRESH:ADRESL), 需要时将 ADIF 位清零
- 7) 如需再次进行 A/D 转换, 返回步骤 1 或者步骤 2

注: 当等待 AD 转换时, 建议增加超时处理, 如下图:

```

/*****
* Description   : ADC采集
* Input        : None
* Output       : None
* Return       : None
* Attention    : 默认采集FVR 1.0V ->反推VCC电压
*****/
void ADC_Exce(void)
{
    unsigned short timeout;

    GO = 1;      //start ADC

    //等待AD转换结束
    timeout = 0;
    while(GO)
    {
        CLRWDT();

        timeout++;
        if(timeout > 2000)      //超时时间需按照ADC采集时间适当更改
        {
            ADON = 0;
            GO = 0;
            ADON = 1;
            GO = 1;
            break;
        }
    }
}

```

ADC 的相关寄存器

ADRESH (AD 转换结果的高四位)

地址: 0x1C

Bit	Name	Description	Attribute	Reset
7:4	Reserved			
3:0	ADRESH	AD 转换结果的高 4 位	R/W	0000

ADRESL (AD 转换结果的低四位)

地址: 0x1D

Bit	Name	Description	Attribute	Reset
7:0	ADRESL	AD 转换结果的低 8 位	R/W	0x00

ADCON0 (ADC 控制寄存器 0)

地址: 0x1E

Bit	Name	Description	Attribute	Reset
7	ADFM	AD 结果对齐标志 1: 左对齐 ADRESH = adc【11:4】左对齐	R/W	0

Bit	Name	Description	Attribute	Reset
		ADRESL = {adc 【3:0】 ,4'b0} 0: 右对齐 ADRESH = {4'b0,adc 【11:8】 }右对齐 ADRESL = adc 【7:0】		
6	ADSP	ADC 速度选择 1: 高速 0: 低速	R/W	0
5:2	CHS [3:0]	CHS3:CHS0 – 模拟通道选择位 0000: 通道 0 AN0 (PB0) 0001: 通道 1 AN1 (PB1) 0010: 通道 2 AN2 (PB2) 0011: 通道 3 AN3 (PB3) 0100: 通道 4 AN4 (PB4) 0101: 通道 5 AN5 (PB5) 0110: 通道 6 AN6 (PB6) 0111: 通道 7 AN7 (PB7) 1000: 通道 8 AGND 1001: 通道 9 AGND 1010: 通道 10 HCOUT_AVG 电流采集 1:1 平均输出) 1011: 通道 11 ANGD 1100: 通道 12 VIN_DIV (VIN 分压 1/16 或 1/32) 1101: 通道 13 HCOUT(电流采集 1:1 瞬态输出) 1110: 通道 14 FVR 1111: 通道 15 DAC VIN 分压 1/16 或 1/32 需要配置 HBC0[5]	R/W	0000
1	GO/DONE	GO/DONG – A/D 转换状态位, 当 ADON=1 时: 1: A/D 转换正在进行 0: A/D 空闲	R/W	0
0	ADON	ADON – A/D 模拟使能位 1: 使能 A/D 转换器模块 0: 禁止 A/D 转换器模块	R/W	0

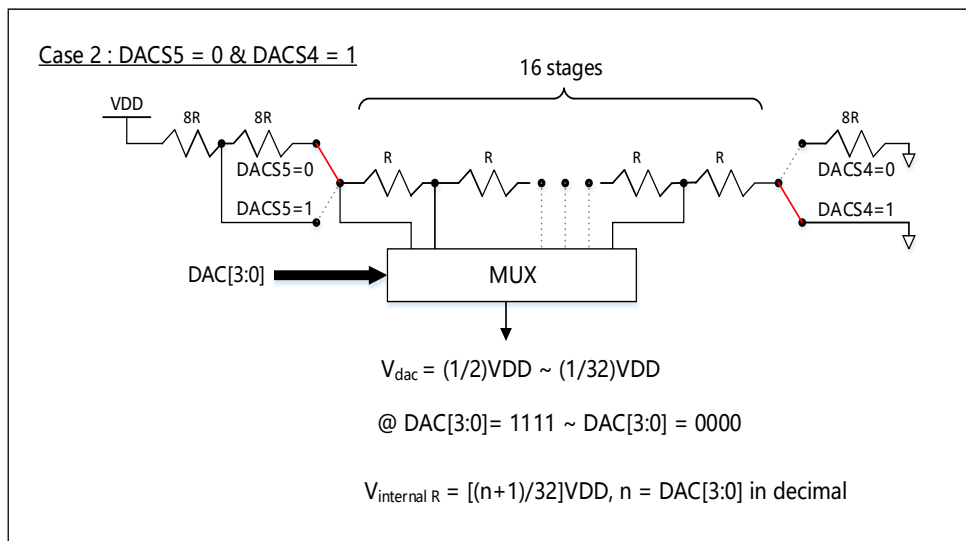
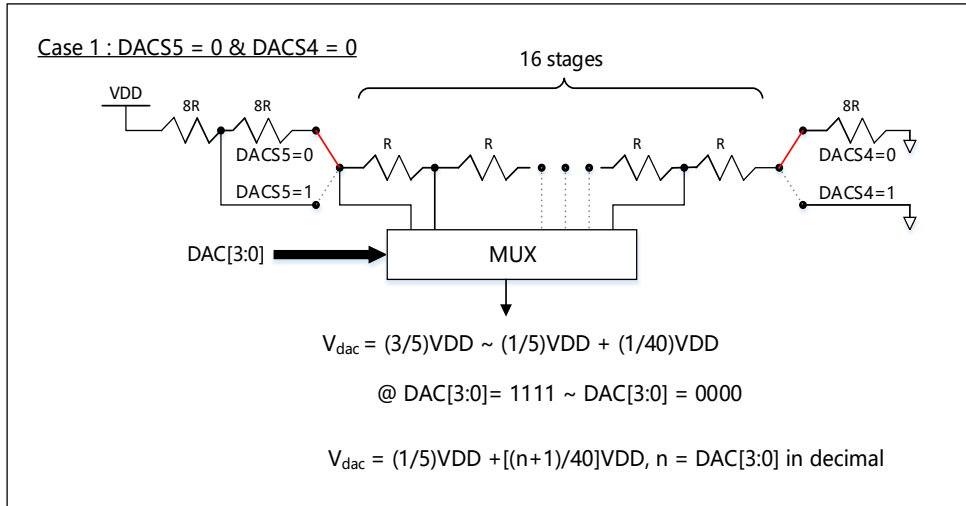
ADCON1 (ADC 控制寄存器 1)

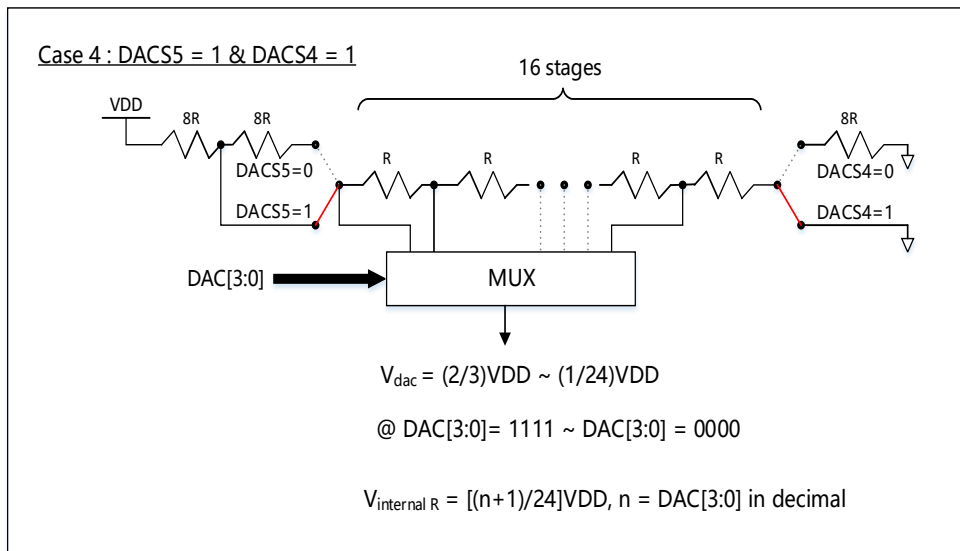
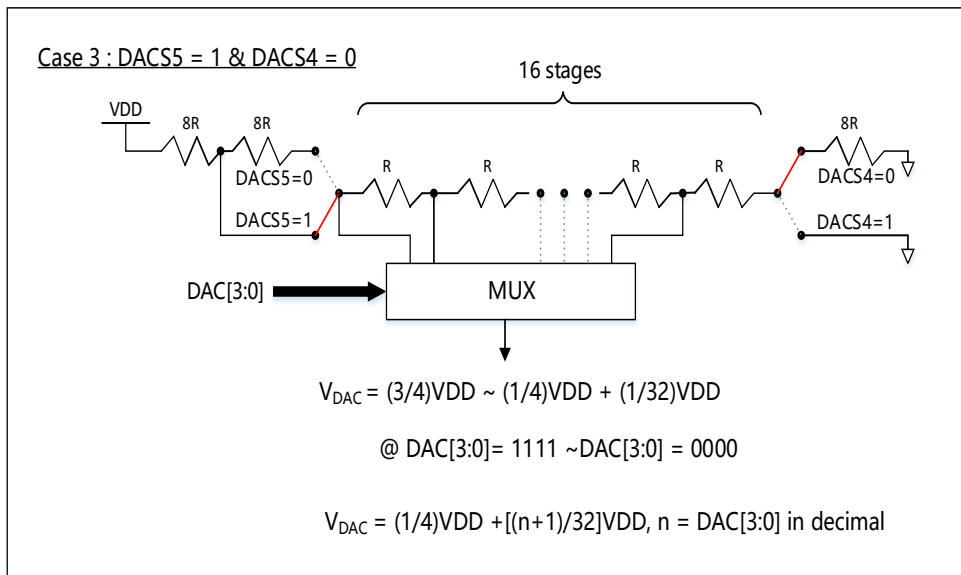
地址: 0x1F

Bit	Name	Description	Attribute	Reset
7:6	ADVREF [1:0]	A/D 参考电压选择位 00: FVR 01: VDD 10: Reserved 11: AN4(PB4)	R/W	00
5:3	ACQT [2:0]	A/D 采集时间选择位 111: 20 TAD 110: 16 TAD 101: 12 TAD 100: 8 TAD 011: 6 TAD 010: 4 TAD 001: 2 TAD 000: 0 TAD	R/W	000
2:0	ADCS [2:0]	A/D 转换时钟选择位 111: Reserved 110: FSYS/512(ADSP=0), FSYS /64(ADSP=1) 101: FSYS /128(ADSP=0), FSYS /16(ADSP=1) 100: FSYS /32(ADSP=0), FSYS /4(ADSP=1) 011: Reserved 010: FSYS /256(ADSP=0), FSYS /32(ADSP=1) 001: FSYS /32 (ADSP=0), FSYS /8(ADSP=1) 000: FSYS /16(ADSP=0), FSYS /2(ADSP=1)	R/W	000

15 D/A 转换

DAC 是由一串电阻所组成，可以产生不同层次的参考电压，DACON 寄存器的 4 和 5 位用来选择电阻串的最高和最低值；DAC[3:0]用于选择所要的电压值，该值由 DACS5，DAC4 来决定。下图显示了四个不同选择时，内部参考电压值的计算。DAC 输出电压范围可以从 $(1/32) \times VDD$ 到 $(3/4) \times VDD$ 。





DAC 的相关寄存器

DACON (DAC 控制寄存器)

地址：0x13

Bit	Name	Description	Attribute	Reset
7				
6	DACEN	使能 DAC 1: 使能 DAC 0: 禁止 DAC	R/W	0
5	DACS5	DAC 正端电阻抽头选择	R/W	0
4	DACS4	DAC 负端电阻抽头选择	R/W	0
3:0	DAC [3:0]	DAC 输出选择 (0x0-0xF)	R/W	0

16 LDO 输出功能

AD16H02 内部集成了线性稳压 LDO，可提供 2.5V/3.0V/3.3V/5V 稳压输出，可以为外部 MCU 提供稳定电源。输出精度达到 1%，最大可提供 20mA 输出电流。

ALG3 (MAPEN=1, 寄存器读写)

地址: 0x9B

Bit	Name	Description	Attribute	Reset
7:6	OVP_SEL [1:0]	过压保护电压的选择 00: 8V 01: 10V 10: 12V 11: 16V	R/W	11
5:4	LDOV [1:0]	LDO 输出电压的选择 00: 5V 01: 3.0V 10: 3.3V 11: 2.5V	R/W	00
3:0	Reserved			

17 十六进制转化为十进制 (Hexadecimal Convert to Decimal HCD)

AD16H02 具有十进制格式化功能. 当一个寄存器里面的内容需要十进制转化的时候, 在执行操作 ALU 以后必须把结果进行相应的进制转化. 一个数据在处理过程中进行了转化成了十进制, 那么所有对这个数进行的操作 (包含存放该数据的 RAM 单元, WREG (W), 立即数, 以及所要查表信息) 都的进行十进制转化, 这样的运算结果才正确.

DAA 指令能在加法运算完成以后将 W 里的数据从十六进制转化为十进制重存给 W 转换操作在例子 2.2 中被说明.

例 2.2: DAA 转化

```
举例如下: 十六进制转十进制
Test content: 0x0a , daa(0x0a) = 0x10, c = 0
test2
bcf      STATUS,C
movlw   0x0a
DAA
movwf   result      ; w -> result
movf    STATUS, w    ; status -> w
movwf   status_save ; w -> status
movf    result, w    ; result -> w
xorlw   0x10; w=09
;xorlw  0x0a
btfss   STATUS,2
;btfsc  STATUS,2
goto    err03
movf    status_save,w; 0x1c
xorlw   0x1c
btfss   STATUS,2
goto    err04
goto    test3
err03
movlw   0x03
movwf   PORTB
goto$
err04
movlw   0x04
movwf   PORTB
goto$
```

DAS 指令能在减法运算完成以后将 W 里的数据从十六进制转化为十进制重存给 W 转换操作在例子 2.3 中被说明

例 2.3: DAS 转化

举例如下:

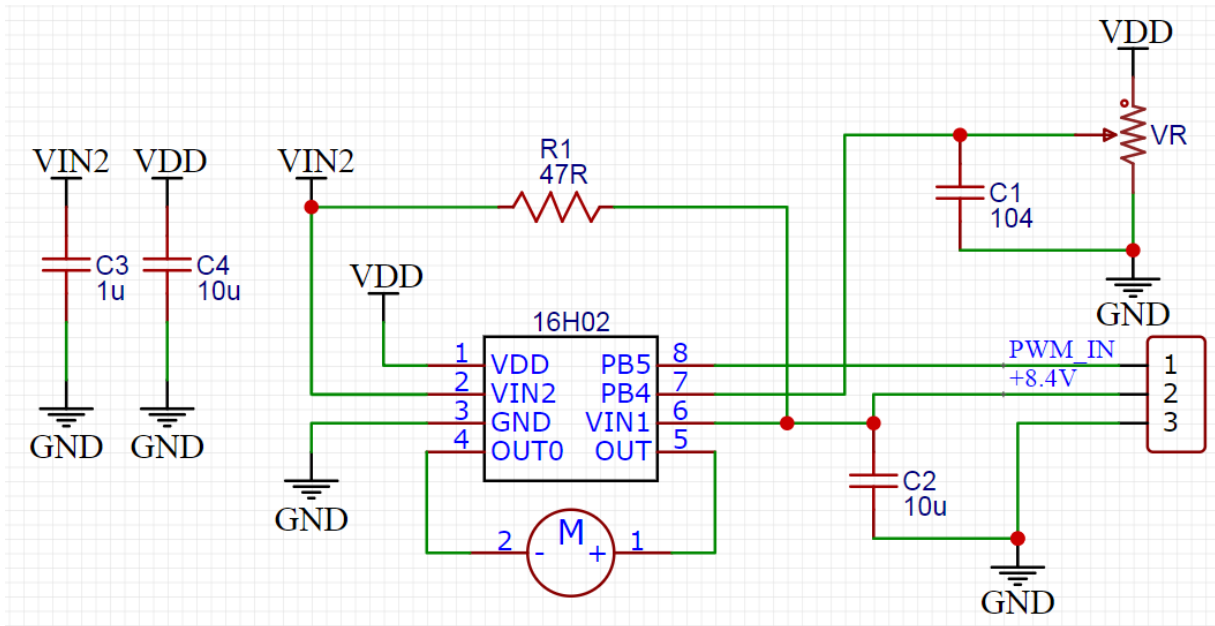
```
; Test content: 0x01-0x02=0xff,C=0 , das(0xFF) = 0x99, c = 0
test2
bcf    STATUS,C
movlw  0x01
movwf  buf
movlw  0x02
subwf  buf,w
DAS
movwf  rs1
movf   STATUS,w
movwf  status_save
movf   rs1,w;0x99
xorlw  0x99
btfss  STATUS,2
goto   err03
movf   status_save,w; 0x18
xorlw  0x18
btfss  STATUS,2
goto   err04
```

18 绝对最大额定值

符号	描述	最小值	最大值	单位
VIN-VSS	电源电压	-0.33	+12.0	V
I_{VIN}	VIN 的最大电流	-	+100	mA
I_{VSS}	VSS 的最大电流	-	-100	mA
T_J	最大结温	-	+150	°C
T_{STG}	存储温度范围	-55	+150	°C
T_A	工作温度	-40	+85	°C

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

19 典型的电路应用



AD16H02 典型电路应用

20 操作条件

DC 供电电压:

VDD 工作电压 2.5V~5.5V

VIN 工作电压 3.3V~12V

端口输入电压 $V_{SS}-0.3V\sim V_{DD}+0.3V$

工作温度 $-40\sim 85^{\circ}\text{C}$

H 桥驱动电流 1A

21 电气特性

直流特性

符号	参数	测试条件		最小	典型	最大	单位
		VDD	条件				
VDD	工作电压	--	$f_{\text{SYS}}=16\text{MHz}$	2.50	3.30	5.50	V
I _{DD}	工作电流	--	$f_{\text{SYS}}=16\text{MHz}$	--	1.5	--	mA

注：数据由设计保证，并未在生产时测试。

IO(直流特性)

符号	参数	测试条件		最小	典型	最大	单位
		VDD	条件				
V _{IH}	I/O 口高电平输入电压 (SMT)	5.0V	SMTV=0	0.4*VDD		VDD	V
V _{IL}	I/O 口低电平输入电压 (SMT)	5.0V	SMTV=0	0		0.2*VDD	V
V _{IH}	I/O 口高电平输入电压 (SMT)	5.0V	SMTV=1	0.7*VDD		VDD	V
V _{IL}	I/O 口低电平输入电压 (SMT)	5.0V	SMTV=1	0		0.3*VDD	V
I _{OL}	I/O 口输入灌电流	5.0V	V _{IO} =0.1*VDD	---	25	---	mA
		3.3V		---	15	---	mA
I _{OH}	I/O 口输出拉电流	5.0V	V _{IO} =0.9*VDD	---	10	---	mA
		3.3V		---	4.5	---	mA
R _{PU}	I/O 口上拉电阻	5.0V	RSEL=0	---	30	---	K Ω
R _{PD}	I/O 口下拉电阻	5.0V	RSEL=0	---	30	---	K Ω
R _{PU}	I/O 口上拉电阻	5.0V	RSEL=1	---	190	---	K Ω
R _{PD}	I/O 口下拉电阻	5.0V	RSEL=1	---	300	---	K Ω

注：数据由设计保证，并未在生产时测试。

FVR

符号	参数	测试条件		最小	典型	最大	单位
		FVR	条件				
FVR		1.0	25°C	-1%	1.0	+1%	V

注：数据由设计保证，并未在生产时测试。

LVT

符号	参数	测试条件	最小值	典型	最大值	单位
I _Q	LVD Quiescent Current			1.8		uA
V _{LVR}	Low Voltage Reset Voltage	LVT 使能	-5%	2.3V	+5%	V
		LVT 使能		2.6V		
		LVT 使能		2.8V		
		LVT 使能		3.2V		
		LVT 使能		3.4V		
		LVT 使能		3.6V		
		LVT 使能		4.1V		

注：数据由设计保证，并未在生产时测试。

WDT|OSC(交流特性)

符号	参数	测试条件		最小	典型	最大	单位
		VDD	条件				
f _{sys}	系统时钟(OSCF)	3.3V	25℃	-1%	16	+1%	MHz
	系统时钟(OSCF)		25℃	-3%	256	+3%	KHz
tr _{STD}	WDT 复位时间		TWDT_111	-3.25%	32.8	+3.25%	ms
			TWDT_110		4		
		TWDT_101	262.2				
		TWDT_100	65.6				

注：数据由设计保证，并未在生产时测试。

ADC

符号	参数	测试条件		最小	典型	最大	单位
		VDD	条件				
VDD	工作电压	V _{OUT_LDO} =5.0V	---	2.7	---	5.5	V
vin	输入电压		---	0	---	VREF	V
Resolution	分辨率		---	---	12	---	Bit

注：数据由设计保证，并未在生产时测试。

LDO

符号	参数	测试条件		最小	典型	最大	单位
		LDO	条件				
I _Q	LDO 静态电流			---	1.3	---	uA
V _{OUT_LDO}	LDO 输出电压			---	3.3	---	V
Δ V _{LOAD}	LDO 负载调整率			---	0.3	---	%/mA
Δ V _{LINE_LDO}	LDO 线性调整率			---	0.28	---	%/V
V _{DROP_LDO}	LDO 压降电压	V _{OUT_LDO} =3.3V	空载	---	60	---	mV

注：数据由设计保证，并未在生产时测试。

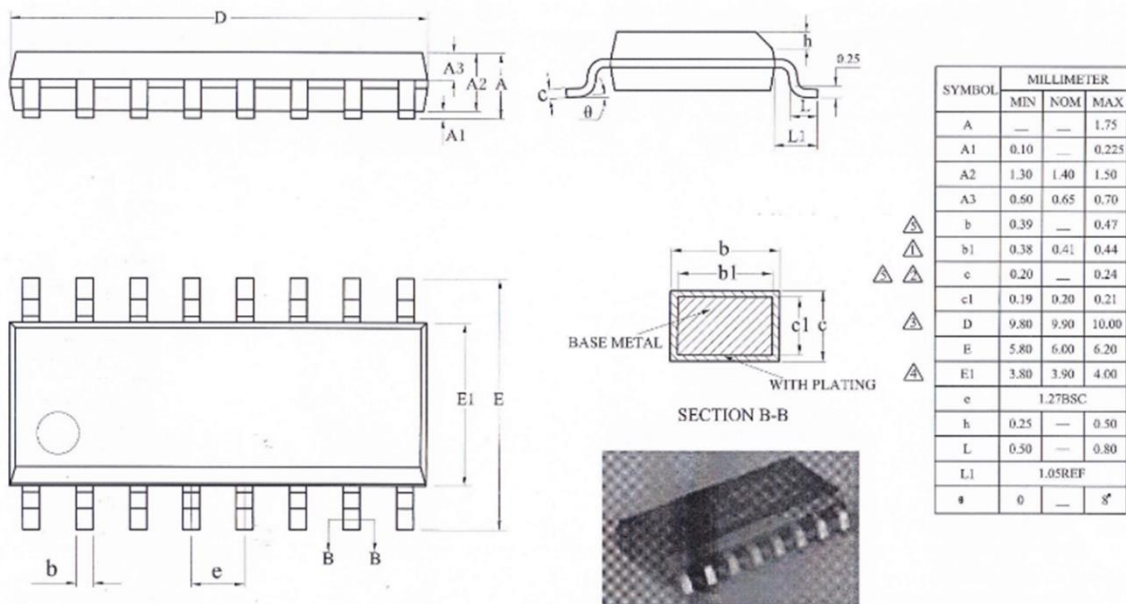
SLEEP

符号	参数	测试条件		最小	典型	最大	单位
		VDD	测试条件				
SLEEP	PWIDEL	V _{OUT_LDO} =5.0	25°C	---	1000	---	uA
	PWSAVE			---	12	---	uA
	PWOFF			---	4	---	uA
	PWIDEL	V _{OUT_LDO} =3.3	25°C	---	750	---	uA
	PWSAVE			---	9	---	uA
	PWOFF			---	3.5	---	uA
	PWIDEL	V _{OUT_LDO} =3.0	25°C	---	700	---	uA
	PWSAVE			---	8	---	uA
	PWOFF			---	3	---	uA
	PWIDEL	V _{OUT_LDO} =2.5	25°C	---	600	---	uA
	PWSAVE			---	8	---	uA
	PWOFF			---	3	---	uA

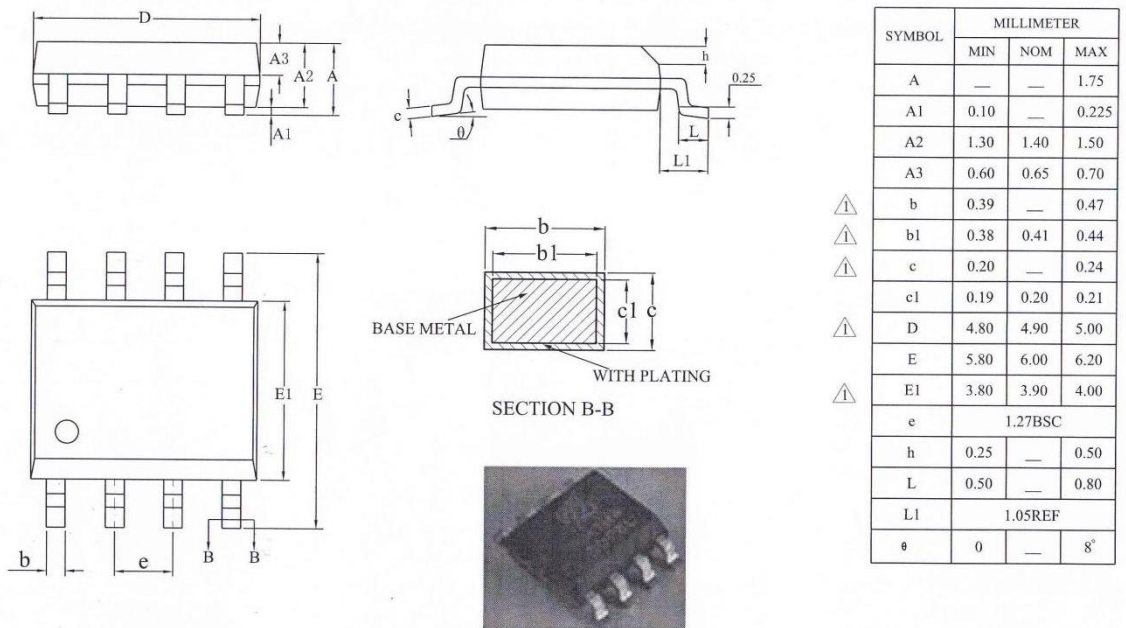
注：数据由设计保证，并未在生产时测试。

22 封装信息

SOP16



SOP8



23 订购信息

丝印信息

现行ADUC的单片机表面印有一栏信息：产品代码和日期码。

Marking

H02	1	20	1
Device code	Year	Week	SeriesNo
Year: 0: 2020; 1: 2021			
Week: 01:第1周; 23:第23周			
SeriesNo: 序列号0-Z, 1:当前周第2个工单			

标签信息

货品内外包装上粘贴的标签上包含：产品名称，封装信息，芯片批号，丝印信息，出货日期及包装数量。

产品名称

封装信息

芯片批号

丝印信息

出货日期

包装数量

Part No: AD16H02S16-XT

Package: SOP16

Lot No: R2H401500

Marking: H021371

Date: 2021-09-16

QTY: 10000pcs

空片

采购信息

AD16H02			
产品名称	封装信息	工作温度	包装方式及数量
AD16H02S16-XT	SOP16, 绿色封装	-40~85℃	Tube 50/tube