

Microcontrol

MSP430外围模块功能简介[4]

作者:利尔达

模数转换器ADC12的工作原理及使用

MSP430F14X和44X系列单片机内嵌入一个高精度12位ADC转换模块。该转换模块具有采样速率高，（最大采样速率达二十万次每秒，这无疑提高了测量精度）。另外MSP430系列中的FE427也具有ADC模块功能，该模块中的16位ADC是采用 Σ - Δ 转换技术来将模拟信号数字化的。因此这两个模块转换的原理是有差别的，具体使用参见芯片参考手册。在此讲座中只介绍MSP430F44X、14X系列的ADC转换模块。

从该模块电路看出，其模式转换是采用逐次逼近的方法来实现测量的。该电路分5大功能模块组成，在配置上这5个模块都可以独立配置。

五大模块分析：

1、带有采样/保持功能的ADC内核

该采样器是以电荷为转换辅助量的及采用电荷重分布技术的逼近型ADC，其特点是高效经济。其核心为1权电容网络。其采样过程是一个电容充电过程。其保持就是根据电容中的电荷不变实现在比较寄存器中的总电荷量在逐次逼近，随着逐次逼近过程的进行，权电容网络中的各电容两端的电压在不断变化，因而导致总电荷量在每个电容之间不断重新分布。该原理要比传统逼近型ADC中控制精密电阻的相对精度要容易，因此实现较为经济，同时消除了电阻网络中因温度变化引起的阻值失配。

[EXIT](#)

由于A/D转换的原理是基于电荷再分配，当内部开关切换输入信号进行采样时，会产生流入或流出电流，但这种电流由于外部的等效时间常数很小而不会影响转换精度。但如果外部的阻抗很大，在确定的采样时间内，这些瞬变的电流就会影响采样的精度。

关于采样时序可见书中**223**页采样时序分析和**218**页的采样一节。

2、可控制的转换存储：

MSP430F449的A/D转换存储是由**16**个转换存储寄存器完成的，在使用中可以进行多次转换而不需要用软件干预。**16**个转换存储寄存器都配有一个控制器，通过控制器的说明可灵活的对每一采样通道的选择、参考电平的选择以及转换模式等其他工作条件进行独立定义。一旦进行采样，A/D就会按照定义的方式进行工作，直到用软件去停止它。该转换存储模式有**4**种：**a**、单通道单次。

b、单通道重复转换。**c**、序列通道单次转换。**d**、序列通道重复转换。

3、可控制的参考电平发生器

V_{R+} 的选择：A/D12内部有两种参考电平，可以选择**1.5V**或**2.5V**。参考电平可以加在A/D内核 V_{R+} 上，也可以引到芯片外部的引脚中。如果选用外部的

[EXIT](#)

参考电平，则可通过 $V_{e\text{ REF+}}$ 引脚加到 V_{R+} 上。

V_{R-} 的选择： V_{R-} 的参考电平可以选择 AV_{SS} ，也可以经过 $V_{\text{REF-}}$ 或由外部 $V_{\text{EREF-}}$ 引脚加入。如果内部 $V_{\text{REF-}}$ 和外部 $V_{\text{EREF-}}$ 引脚未用，则 V_{R-} 连接到 AV_{SS} 。

参考电平的配置通过寄存器ADC12MCTLx中的SREF位的设置完成。

如果只用外部的参考电平，那么内部的参考电平发生器可以关闭，以降低功耗。

参考电平的作用是建立的A/D转换的时模拟信号输入的上下限。其分别对应转换结果的量程的上下限。因此在进行A/D模拟采样时不要使模拟信号的输入范围超出了该转换的范围。

注：当第一次打开 V_{REFON} 一定注意建立转换所需要的建立时间。

4、可控制和选择的时钟源

A/D C12的转换时钟可选择多种时钟源，并可作1~8分频，其时钟源的不同将影响转换速度和稳定性。

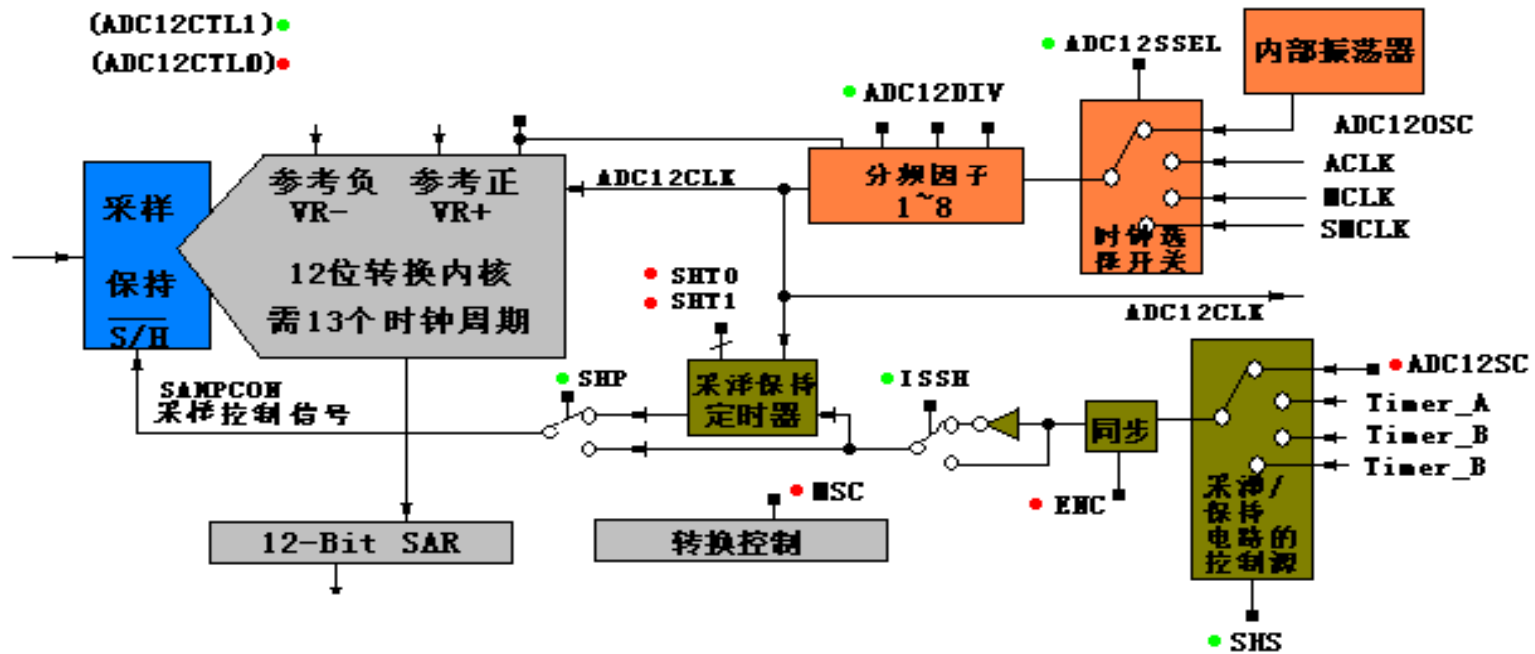
下列式子是转换时钟与转换速度的关系式

$$\text{转换时间} = 13 \times (\text{ADC12DIV} / f_{\text{adc12clk}}) \quad \text{EXIT}$$

式中说明了当分频因子选择1时，其转换时间为13个转换时钟周期。

5、可控制的采样及转换时序电路

ADC12的采样/保持保持电路可由用户通过ADC12CTL0和ADC12CTL1中的控制位配置来灵活地实现控制。



[EXIT](#)

说明1: 模拟信号输入端

MSP430F14x与MSP430F44x的AD完全相同。外接的模拟输入信号占用P6口，及P6.0~P6.7 另外还有4个模拟输入信号； AV_{CC} 、 V_{eREF} 、 $V_{REF} - V_{eREF}$ 、 Ref_x 对应的是电源电压、外部正参考电平、外部负参考电平和温度。对于内部参考电平一般可通过7脚与外部去藕电容相连，使之达到退藕的目的，同时也可作为校准用。外部引进的参考电平由10脚和11脚引入，注意进入的电平不能超过 AV_{CC} 和 AV_{SS} 关于内部参考有两种基准，一个是2.5v，一个是1.5v，这样AD的参考电平可以组成7种不同的组合来决定转换的范围的上下限，而且所有的输入模拟信号的转换码都与该参考电平有关，而每一个通道的参考电平可选择不同的组合来指出参考电平的上下限，这样AD的转换就根据此来转换响应的数字位并存储在指定的单元中。注第11~15通道是测量 $Av_{cc}/2$ 电平，实际上12和15是没有的。

关于码制转换公式如下： $NADC=1095 * (V_{in}-V_{r-}) / (V_{r+}-V_{r-})$

式中 V_{in} 输入的模拟信号，范围大于 AV_{SS} 小于 AV_{CC}

式中 V_{r-} 参考电平下限，最小为 AV_{SS}

式中 V_{r+} 参考电平上限，最大为 Av_{cc}

[EXIT](#)

关于各个通道的模拟输入的参考电平的设置

可通过各个通道的存储控制寄存器（ADC12MCTLx）中Sref 三个位来定义。同用INCH中的四个位来指出是选用哪个通道。

1、 单通道单次转换

第一步： 将ADC12内核打开，及由ADC12ON=0 修改成ADC12ON=1 。

第二步： 将指定通道地址，也就是开始存放的地址。由控制寄存器1中的第12~15位确定。 同时定义选定的通道和定义该通道的参考电平和保存结果存储器。

第三步： 启动转换，用ENC=1启动，等待转换，转换需要13个时钟周期。其中12个时钟周期用于产生转换结果，1个时钟周期用于存储转换结果。

第四步： 获取转换结果可通过查询方式或中断方式，采用查询方式时必须获取数据后将 ENC=0 以及中断标志位复位。如将结果写入选定的存储寄存器ADC12MEMx时，中断标志自动的复位。

第五步： 从复执行第三步进行下一个转换。

注意的是： 当选用ADC12SC（软件转换）控制转换时，每次转换还要启动一次ADC12SC，如果用定时器启动就可在定时中断中启动ENC一次即可。

如

[EXIT](#)

2、 序列通道单次转换

通道结束

第一步~第二步同上，只是用一个EOS来定义转换的结束指针在哪一个即可。

其余的步骤也同上。

3、 单通道多次转换

多次与单次转换类似，主要要设定它为多次转换，不同的是如何让转换停止，对单通道序列转换的停止方法有三种，

此时中

a 改变CONSEQ=0，此时多次转换结束，可将结果存入ADC12MEMx中，中断标志置位。

b 用ENC来停止，并将结果存入ADC12MEMx中，此时中断标志置位。

c 上述两个过程合在一起用。但建议不采用此方法。

4、 序列通道多次转换

序列通道多次转换与序列通道单次转换类似

[EXIT](#)

上述模式切换即由2转换成3或由4转换成2可不必停止ENC转换允许，就可直接进行。但对转换到1模式是不可以的，要先停止转换后切换。

可以接受的转换模式有：

00——>01 00——>10 01——>00 01——>11 10——>10

10——>11 11——>01 11——>10

00 单通单次 01序列单次 10 单通多次 11 序列多次

如果转换发生冲突，可分成两步切换即可。

MSC位的使用：

多次采样/转换控制位（MSC）在不用采样定时器产生SAMPCON信号时不起作用。如果用采样定时器产生SAMPCON且不处于单通道单次转换模式时（CONSEQ>0），则MSC位可以控制实现尽可能快的连续转换。

如果MSC=0，那么不管在哪种模式下，每次转换都需要由一个SHI的上升沿来触发。如果MSC=1，并且CONSEQ>0，那么首次转换由SHI的第一个上升沿触发，但是后续的转换会在每次转换完成后立即自动触发启动。

根据不同的工作模式，在序列转换完成或ENC位翻转前，SHI的后续上升沿不起作用，在使用MSC位时，ENC位的作用不变。

[EXIT](#)

```

// 这个程序，用到了中断程序被初始化成单通道、重复转换， //
// 转换结果在中断服务程序中，被放在变量Result中。 //
// 新的转换结果覆盖旧的结果，也可定义缓冲区来存放转换结果 //
//*****//
#include "MSP430x44x.h"
#define ADCMEM ((int*) 0x0140) // ADC12MEMx 定义
void Init(void); // 初始化系统及ADC寄存器
unsigned int Result;
//-----
void main(void){
    Init();
    ADC12CTL0 |= 0x01; // 开始转换
    LPM0; //进入低功耗状态，等待中断
    _NOP();
}
void Init(void) {
    WDTCTL = WDTPW + WDTHOLD; // 停止Wochdog.
    P6SEL = 0xFF; // 所有P6口线均为ADC模块使用
    ADC12CTL0 &=~ 0x02; // 在进行设置时首先复位ADC的
    //转换使能

```

```

ADC12CTL0 = SHT0_8+MSC+ADC12ON; //内部振荡器，
    //置位MSC位，因此转换能自动进行
ADC12CTL1 = 0x0204; // ADC12SC 位触发采样和保持
    // 采样脉冲由采样定时器产生
    // 时钟源：内部振荡器
    // 时钟分频: 1
    // 转换模式: 单通道、重复转换
//选则参考电压和输入管脚
ADC12MCTL0 = 0x0a; // Ref = AVss, AVcc; Input =
    // source = Temperature diode
ADC12IE = 0x001; //使能通道10转换完成后中断
_EINT();
ADC12CTL0 |= 0x02; // 使能ADC转换
}
#pragma vector=ADC_VECTOR
__interrupt void ADC12(void){
    Result = ADCMEM[0];
}

```

[EXIT](#)

```

//*****//
// 这个程序，没有用到中断程序被初始化成多通道、 //
// 单次转换，转换结果放在变量Result[0]--Result[1]中 //
//*****//
#include "MSP430x14x.h"
#define ADCMEM ((int*) 0x0140) // ADC12MEMx 定义
void Init(void); // 初始化系统及ADC寄存器
void main(void){
    unsigned int i;
    unsigned int j;
    unsigned int Result[2]; //用以保存转换结果
    Init(); //对ADC进行初始化
    while (1)
    {
        for (i=0;i<2;i++)
        {
            ADC12CTL0 |= 0x01; // 开始转换
            ADC12CTL0 &=~0x01;

```

```

for (j=0;j<=1000;j++); // 延时，准备触发序列中
//的下一转换
        }
    while ((ADC12CTL1&0x01)==1); // 等待整个序列的转换完成
        for(j = 0 ; j <2; j++)
        {
            Result[j] = ADCMEM[j];
        }

        for (i=0;i<=40000;i++); // 两个序列之间的一个延时
    }
}

```

[EXIT](#)

```

/*****//
// 这个程序，用到了中断程序被初始化成多通道、 //
//重复转换，转换结果在中断中被放在 //
//变量Result[0]和Result[1]中或是放在其它的缓冲区中 //
/*****//

#include <mmsp430x14x.h>
#define ADCMEM ((int*) 0x0140) // ADC12MEMx 定义
void Init(void);
unsigned int Result[2];
void main(void)
{
    Init();
    ADC12CTL0 |= 0x01; // 开始转换
    LPM0; //进入低功耗状态，等待中断
    _NOP(); }
void Init(void){
    WDTCTL = WDTPW + WDTHOLD; // Stop watchdog timer
    P6SEL = 0xFF; // 所有P6口线均为ADC模块使用
    ADC12CTL0 &=~ 0x02; //在进行设置时首先复位ADC的转换使能
    ADC12MCTL0 = 0x00; // Ref = AVss, AVcc; 输入通道 = A0
    ADC12MCTL1 = 0x8a; // Ref = AVss, AVcc; 输入通道 = A10

```

```

ADC12CTL1 = 0x0206; // 第一个转换结果被放在ADC12MEM0
// 第二个转换结果被放在ADC12MEM1
// 采样脉冲由采样定时器产生
// 时钟源：内部振荡器
// 时钟分频: 1
// 转换模式: 多通道、重复转换

ADC12CTL0 = SHT0_8+ MSC+ADC12ON ;
ADC12IE = 0x002; //使能转换中断
ADC12CTL0 |= 0x02;
_EINT(); //使能全局中断
}
#if __VER__ < 200
interrupt [ void (void) // ADC 中断处理程序
#else
#pragma vector=ADC_VECTOR
__interrupt void ADC12(void){
#endif
    unsigned int i;
    for(i = 0 ; i < 2 ; i++ )
    {
        Result[i] = ADCMEM[i] ;
    }
}

```

EXIT

控制寄存器ADC12CTL0

位	15 14 13 12	11 10 9 8	7	6	5	4	3	2	1	0
符号	SHT1	SHT0	MSC	2.5V	REFON	ADC12ON	QD12OVIE	ADC12TOVIE	ENC	ADC12SC
含义	用于采样保持 定时时间设定 8 - 15	用于采样保持 定时时间设定 0 - 7	多次采样 转换位	参考电平 1.5/2.5	启动内部 参考电压	打开AD内 核实现转 换	EMIE溢出 中断允许	转换时间 溢出中断 允许	转换允许	软件控制 采样转换

注意：此寄存器中的高12位的设置只有在ENC位复位的的情况下才能对其修改。

控制寄存器ADC12CTL1

位	15 14 13 12	11 10	9	8	7 6 5	4 3	2 1	0
符号	CSStartAddr	SHS	SHP	ISSH	ADC12DIV	ADC12SSEL	CONSEQ	ADC12BUSY
含义	定义转换存储 首地址	选择控制 采样转换 输入信号源	采样信 号控制 源	采样信 号极 性	选择AD时钟 分频因子	选择时钟原	选择转换 模式	指示转换 活动状态

[EXIT](#)

采样转换定时时间：

在控制寄存器ADC12CTL0中的高字节定义了转换存储器中的采样时序，其含义是设置了采样转换定时时间。其计算公式如下：

$$T_{\text{sample}} = 4 \times \text{ADC12CLK} \times n$$

SHT0	0	1	2	3	4	5	6	7	8	9	10	11	12 15
n	1	2	4	8	16	24	32	48	64	96	128	192	256

SHT1	0	1	2	3	4	5	6	7	8	9	10	11	12 15
n	1	2	4	8	16	24	32	48	64	96	128	192	256

[EXIT](#)

中断标志寄存器ADC12IFG和中断允许寄存器ADC12IE

AD12具有16个中断标志位ADC12IFG，16个中断允许位ADC12IE，1个中断向量字ADIFG。

中断标志位和中断允许位对应于响应的16个寄存器ADC12MEMx。

中断标志ADC12IFG在转换结果装入转换存储寄存器时置位。在AD12MEMx被访问时复位。

如果在ADC12IFG在置位期间有新的数据写入转换存储器时则将会发生溢出。

中断允许标志位是禁止当ADC12IFG置位后禁止产生中断服务请求。

中断向量寄存器ADC12IV

ADC12中断事件共有18个，他们共用同一个中断向量，18个中断标志按中断优先级组织。该中断可通过ADC12IV向量寄存器中的值获得。其中0表示无中断挂起，在程序进入中断服务程序后将此值加到PC 指针上后来实现进入响应中断的处理程序中。

ADC12OVIFG和ADC12IV寄存器后自动复位。而ADC12IFG必须用软件或通过访问ADC12MEM来自动复位。

[EXIT](#)

中断向量ADC12IV真值表

IFG15	IFG14	IFG13	IFG12	IFG11	IFG10	IFG9	IFG8	IFG7	IFG6	IFG5	IFG4	IFG3	IFG2	IFG1	IFG0	OVIE	TOVIE	12IV
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	4
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	8
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	10
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	12
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	14
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	16
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	18
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	20
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	22
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	24
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	26
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	28
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	36



优先级

[EXIT](#)

12位ADC的主要特性:

最大采样速率: 200Ksps

转换分辨率: $1 / 4096$ (用最低有效位LSB的步长表示)

控制采样周期: 可用软件选择或编程的方式

可配8个外部通道

可选则4个内部通道用语特定的数据采样

每个通道可用软件设定 2种之一的内部参考电平

内部具有采样/保持电路

可进入省电模式

具有多种转换模式

可用软件设置转换时钟源

[EXIT](#)

逐次逼近：这种ADC是用一个电压比较器将模拟输入电压与一个N位DAC的输出电压进行比较，而这个比较的DAC是有模拟转换器是由该内部的寄存器所提供。在工作时输入的模拟信号逐次与这个位从高到低进行比较使之将模拟电压转换成响应的数字电平信号。

Σ - Δ : 这是以很低的分辨率用过采样速率实现及过采样技术、噪声整形和数字滤波技术来增加分辨率的一种可达到很高分辨率的技术，这种技术在很早旧有人从理论上分析可以达到很高的分辨率，但由于制造技术上的限制直到近来的微米技术的成熟才使得采用这种技术的进行高分辨率越来越应用很广。目前使用该技术的制造的芯片主要用于高分辨率的中、低频（可达到直流）的测量和数字音频电路中。向MSP430FE427和MSP430F169芯片。

[EXIT](#)

微控设计网提供MSP430单片机发开工具

