

## 例程十一 ADC 及其应用

在计算机过程控制的数据采集等系统中，经常要对一些过程参数进行测量和控制，这些参数往往是连续变化的物理量，如温度，压力，流量和速度等。这里所指的连续变化即数值是随着时间连续可变的，通常称这些物理量为模拟量，然而计算机本身所能识别和处理的都是数字量。这些模拟量在进入计算机之前必须转换成二进制数码表示的数字信号。能够把模拟量变成数字量的器件称之为模数转换器。

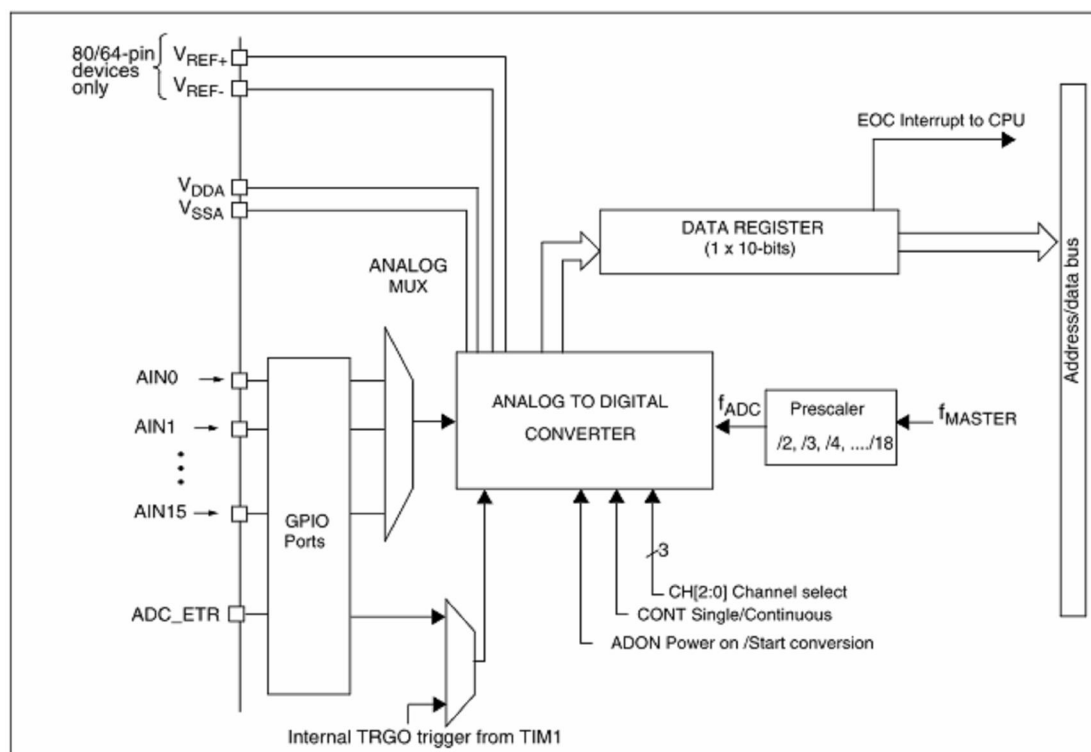
### 15.1 STM8S207ADC 模块概述

STM8S207 系列高性能产品包括了一个 10 位连续渐进式模数转换器 ADC2，（注意 STM8S2XX 都没有 ADC1）提供了多达 16 个多功能的输入通道。主要性能有：

转换时间为 14 个时钟周期，带有参考电压引脚，可设置为单次或联系的转换模式，可设置转换结束产生中断。

### 15.2 ATD 模块接口组成和特点

STM8S207 单片机内置的 ADC 模块如下图所示：



ADC 模块可以通过 ADC\_CR1 寄存器来开启或者关闭。然后 ADC 的时钟是有 f\_master 时钟经过预分频后提供的。STM8S207 高达 16 个输入通道，支持多种转换模式

### 15.3 ADC 模块寄存器设置

STM8S207 的 ADC 模块共有 8 个寄存器。分为 4 个设置寄存器，2 个数据寄存

器和 2 个施密特触发禁止寄存器。

### 15.3.1 ADC 控制/状态寄存器 ADC\_CSR

7	6	5	4	3	2	1	0
EOC	AWD	EOCIE	AWDIE	CH[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw

EOC：转换结束。此位在 AD 转换结束后由硬件置位。由软件写“0”来清零

0：转换未结束

1：转换结束

AWD：因为 STM8S207 没有 ADC1，所以此位无效

EOCIE：转换结束 EOC 的中断使能

0：禁止转换结束中断

1：使能转换结束中断

AWDIE：在 STM8S207 中无效位

CH：选择转换通道，分别选择 0 到 15 共 16 个通道

### 15.3.2 ADC 配置寄存器 1 ADC\_CR1

7	6	5	4	3	2	1	0
保留	SPSEL[2:0]			保留		CONT	ADON
	rw	rw	rw			rw	rw

SPSEL：预分频选择位

000：f\_adc = f\_master/2

001：f\_adc = f\_master/3

010：f\_adc = f\_master/4

011：f\_adc = f\_master/6

100：f\_adc = f\_master/8

101：f\_adc = f\_master/10

110：f\_adc = f\_master/12

111：f\_adc = f\_master/18

CONT：连续转换

0：单次转换模式

1：连续转换模式

ADON：AD 转换开关

0：禁止 ADC 转换，进入低功耗模式

1：使能 ADC 并开始转换

需要注意的是：如果此位是 0 时，并且写 1 到此位，那么将把 ADC 从低功耗模式下唤醒。如果此位是 1，并且写 1 到此位，那么将启动 AD 转换。一旦 ADC 上电，所选通道的 IO 输出功能就被禁用了。

### 15.3.3 ADC 配置寄存器 2 ADC\_CR2

7	6	5	4	3	2	1	0
保留	EXTTRIG	EXSEL[1:0]		ALIGN	保留	SCAN	保留
	rw	rw	rw	rw		rw	

EXTTRIG：外触发使能位

0：禁止外部触发转换

1：使能外部触发转换

主要：为了避免错误的触发事件，使用 BSET 指令来设置 EXTTRIG 位，不用改变其它位

EXTSEL：外部事件选择位

00：内部定时器 1TRG 事件

01：ADC\_ETR 引脚的外部中断

10：保留

11：保留

ALIGN：数据排列

0：数据左对齐。高 8 位在 ADC\_DRH，其余的在低位字节

1：数据右对齐。低 8 位在 ADC\_DRL，其余的在高位字节

SCAN：STM8S207 无效

### 15.3.4 ADC 配置寄存器 3 ADC\_CR3

7	6	5	4	3	2	1	0
DBUF	OVR	保留					
rw	rc_w0						

STM8S207 此寄存器无效

### 15.3.5 ADC 数据寄存器 ADC\_DRH、ADC\_DRL

根据 ADC\_CR2 中的 ALIGN 设置数据对齐方式，ADC\_DRH 和 ADC\_DRL 存放这转换结果。

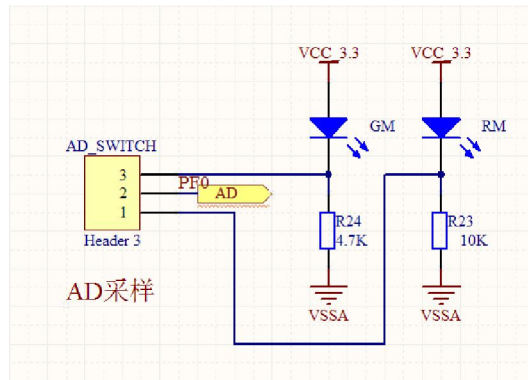
### 15.3.6 ADC 施密特触发器禁止寄存器 ADC\_TDRH、ADC\_TDRL

这些位由软件设置。当 TDx 为 1 时，禁止施密特触发功能，降低 IO 的静态功耗；为 0 时开启施密特触发器。

### 15.4 ADC 模块基础应用实例

本节通过一个简单示例，让读者掌握 ADC 模块的使用和编程方法。

本实例硬件连接入下图所示，在 STM8S207 实验板中 ADC 的外部输入可以使光敏电阻或者热敏电阻，通过实物图的短路帽可以自行选择。本节 ADC 内容只是简单说明 ADC 的编程实例，通过 ADC 的转换，通过串口把相应的 AD 值显示出来，实验中使用的 AD 接口是 PF0，也就是 AIN10



下面看看怎样初始话AD模块，还是从主函数看起

```
int main(void)
{
    /* Infinite loop */
    /*设置内部时钟16M为主时钟*/
    CLK_HSIPrescalerConfig(CLK_PRESCALER_HSIDIV1);
    /*!<Set High speed internal clock */

    Uart_Init();
    UART1_SendString("ADC_CONVERSION AD_Value:", sizeof("ADC_CONVERSION AD_Value:"));
    ADC_Init();
    #if CONVERSIONMODE == CONVERSIONMODE_CONTINUOUS
    Tim1_Init();
    #endif
    Delay(0xffff);
    Delay(0xffff);
    Delay(0xffff);
    __enable_interrupt();
    while (1)
    {
        }
    }
}
```

当大家看到

```
#if CONVERSIONMODE == CONVERSIONMODE_CONTINUOUS
Tim1_Init();
#endif
```

就知道，在这个例程中我设置的2中转换模式，有单次转换和连续转换模式，单次转换模式就是转换一次就不转了，就停在那里了，在这个例程中为了调试方便我才设置为了单次转换模式，在实际应用中一般都是连续转换模式，在连续转换模式要开一个定时器去读取AD值回来，一般的实际应用都是这样。如果大家想用连续转换模式的话，很简单，在我的例程中，只要该下宏定义就行。

```
#define CONVERSIONMODE 1
#define CONVERSIONMODE_SINGLE 0
#define CONVERSIONMODE_CONTINUOUS 1
```

说了那么多了，还是看看AD 的初始化是怎样的。

```
void ADC_Init(void)
{
    ADC2_DeInit();
#ifdef CONVERSIONMODE == CONVERSIONMODE_SINGLE
    /**< Single conversion mode */
    /**< Analog channel 10 */
    /**< Prescaler selection fADC2 = fcpu/18 */
    /**< Conversion from Internal TIM TRGO event */
    /** DISABLE ADC2_ExtTriggerState*/
    /**< Data alignment right */
    /**< Schmitt trigger disable on AIN10 */
    /**DISABLE ADC2_SchmittTriggerState*/
    ADC2_Init(ADC2_CONVERSIONMODE_SINGLE, ADC2_CHANNEL_10, ADC2_PRESSEL_FCPU_D18, \
    ADC2_EXTTRIG_TIM, DISABLE, ADC2_ALIGN_RIGHT, ADC2_SCHMITTTRIG_CHANNEL10, DISABLE);
    ADC2_ITConfig(ENABLE);
#elif CONVERSIONMODE == CONVERSIONMODE_CONTINUOUS
    /**< Continuous conversion mode */
    /**< Analog channel 10 */
    /**< Prescaler selection fADC2 = fcpu/18 */
    /**< Conversion from Internal TIM TRGO event */
    /** DISABLE ADC2_ExtTriggerState*/
    /**< Data alignment right */
    /**< Schmitt trigger disable on AIN10 */
    /**DISABLE ADC2_SchmittTriggerState*/
    ADC2_Init(ADC2_CONVERSIONMODE_CONTINUOUS, ADC2_CHANNEL_10, ADC2_PRESSEL_FCPU_D18, \
    ADC2_EXTTRIG_TIM, DISABLE, ADC2_ALIGN_RIGHT, ADC2_SCHMITTTRIG_CHANNEL10, DISABLE);
    ADC2_ITConfig(DISABLE);
#endif
    ADC2_Cmd(ENABLE);
    ADC2_StartConversion();
}
```

里面已经注释得很清楚了。当大家把该例程下载进去，打开串口助手，调好波特

- BaudRate = 115200 baud
- Word Length = 8 Bits
- One Stop Bit
- No parity

如果你是选择是单次转换模式的话，你需要按一下复位键，在串口就看到相对应的数据打印出来。

**风驰电子祝您学习愉快!!! ~~~**