

文档版本	V1.0
发布日期	20210524

APT32F102 ADC 应用指南

APTCHIP



目录

1 概述	1
2. 适用的硬件.....	1
3. 应用方案代码说明	1
3.1 ADC 配置.....	1
3.2 ADC 单次转换模式.....	2
3.3 中断接收数据.....	4
3.4 ADC 连续转换模式	5
4. 程序下载和运行	7

1 概述

本文介绍了在APT32F102中使用ADC的应用范例。

2. 适用的硬件

该例程使用于 APT32F102x 系列学习板

3. 应用方案代码说明

3.1 ADC 配置

基于 APT32F102 完整的库文件系统，可以很方便的对 ADC 进行配置。

硬件配置：

ADC 的参考电压源支持选择内部(VDD)或者外部(VREF+)，同时负向参考电压源也可以由外部提供。

使用 FVR 做参考时，需要在 GPIO 的配置中使能对应的 AF 功能（VREF+），在 VREF 管脚上增加一个 0.1uF 的电容到地。

参考电源选择配置：

1. 正向为内部 VDD，负向为 VSS
2. 正向为外部 VREF+管脚，负向为 VSS
3. 正向为 FVR 2.048V 输出，负向为 VSS
4. 正向为 FVR 4.096V 输出，负向为 VSS
5. 正向为内部 INTVREF 输出，负向为 VSS
6. 正向为内部 VDD，负向为 VREF-
7. 正向为外部 VREF+管脚，负向为 VREF-
8. 正向为 FVR 2.048V 输出，负向为 VREF-
9. 正向为 FVR 4.096V 输出，负向为 VREF-
10. 正向为内部 INTVREF 输出，负向为 VREF-

FVR 固定电压： 2.048V /4.096V

INTVREF 电压： 1.0V

● ADC 模块图

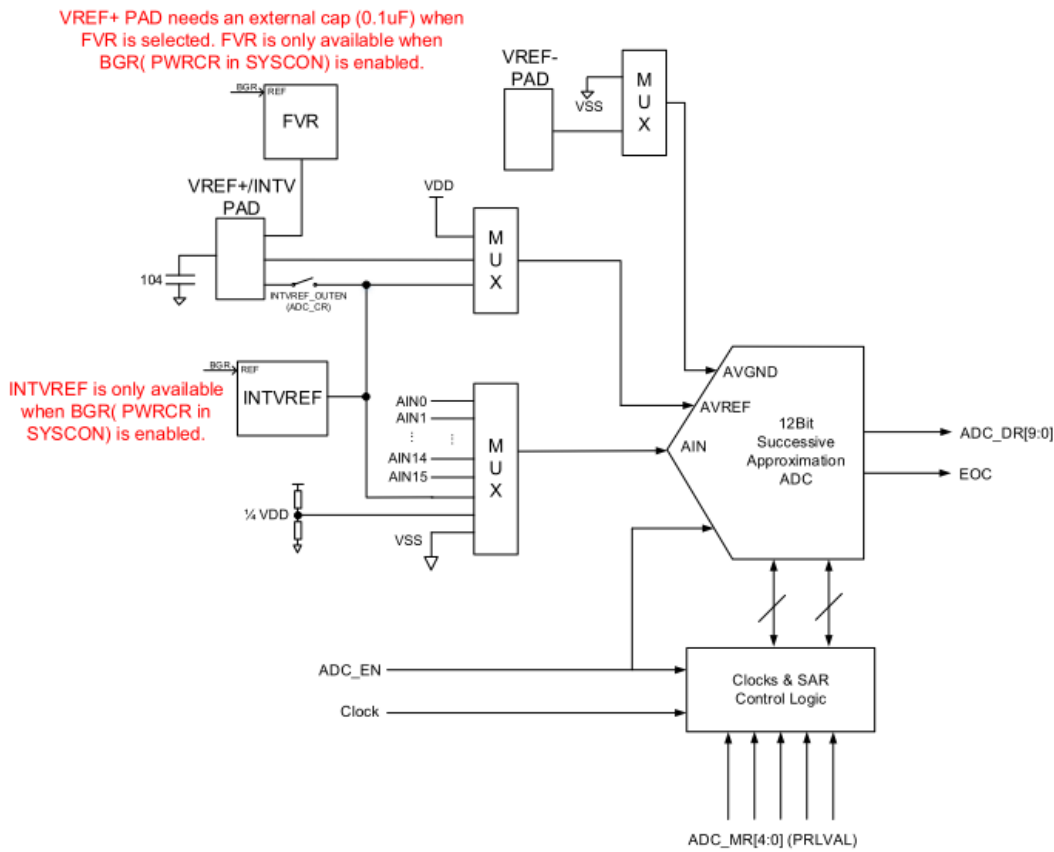


图 3.1.1 ADC 模框图

需要注意设置转换序列个数为 1，那么 ADC 只会转换 ADC_SEQ0 设置的通道，并且将结果存入 ADC_DR0。

● 软件配置:

可在 apt32f102_initial.c 文件中 ADC12_CONFIG 进行初始化的配置

3.2 ADC 单次转换模式

系统主频 48MHZ，参考电压选择内部 4.096V， PIN23-PA0.2 (Vref+) 需要接 104 电容到 GND。

```

/*****/
//adc config
//EntryParameter:NONE
//ReturnValue:NONE

```

```

/*****
void ADC12_CONFIG(void)
{
    ADC12_RESET_VALUE();           //ADC 所有寄存器复位赋值
    ADC12_CLK_CMD(ADC_CLK_CR, ENABLE); //使能 ADC CLK
    ADC12_Software_Reset();       //ADC 软件复位
    //选择 12BIT ADC; 连续模式; 转换优先序列寄存器为 0; 采样保持时间=6 ;ADC_CLK=PCLK/2*2=0.2us; 转换序列个数为 1
    ADC12_Configure_Mode(ADC12_12BIT, One_shot_mode ,0, 6, 2, 1);
    //选择内部 INTVREF=1v 做参考电压, VREF-选择 VSS,选择 INTVREF 做参考, ADC_CLK 不能超过 2M
    ADC12_Configure_VREF_Selecte(ADC12_VREFP_FVR4096_VREFN_VSS);

    //选择内部 INTVREF=1v 做参考电压, VREF-选择 VSS,选择 INTVREF 做参考, ADC_CLK 不能超过 2M
    ADC12_ConversionChannel_Config(ADC12_ADCIN10,ADC12_CV_RepeatNum1,ADC12_AVGDIS,0);
    ADC12_CMD(ENABLE);           //使能 ADC 模块
    ADC12_ready_wait();          //等待 ADC 模块配置完成
    ADC12_Control(ADC12_START);  //开始转换
}

```

● 代码说明:

ADC12_RESET_VALUE() ----用于将 ADC 的寄存器复位到上电复位值。

ADC12_CLK_CMD() ----用于开启和关闭 ADC 主时钟

ADC12_Software_Reset () ----用于软复位 ADC 模块

ADC12_Configure_Mode () ----用于软复位 ADC 模块

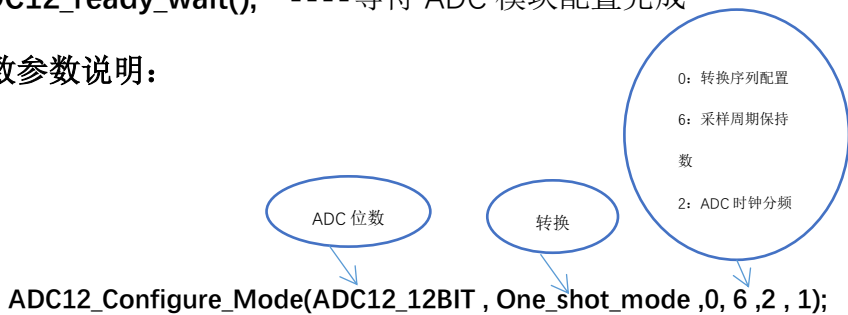
ADC12_Configure_VREF_Selecte()----用于选择参考电压

ADC12_ConversionChannel_Config () ----用于配置 GPIO 口以及映射模拟输入与转换序列

ADC12_CMD() ----用于开启和关闭 ADC 模块

ADC12_ready_wait(); ----等待 ADC 模块配置完成

● 函数参数说明:





ADC12_ConversionChannel_Config(ADC12_ADCIN10,ADC12_CV_RepeatNum1,ADC12_AVGDIS,0);

● **ADC 单次转换模式运行:**

```
volatile U32_T R_ADC_Buf1, R_ADC_Buf2;

/*****/
//main
/*****/

int main(void)
{
    APT32F102_init();
    while(1)
    {
        SYSCON_IWDGNT_Reload();           //清除看门狗
        //
        ADC12_SEQEND_wait(0);
        R_ADC_Buf1 = ADC0->DR[0];
        ADC12_Control(ADC12_START);
    }
}
```

3.3 中断接收数据

```
void ADC12_CONFIG(void)
{
    ADC12_RESET_VALUE();                 //ADC 所有寄存器复位赋值
    ADC12_CLK_CMD(ADC_CLK_CR, ENABLE);  //使能 ADC CLK
    ADC12_Software_Reset();              //ADC 软件复位
    //选择 12BIT ADC; 连续模式; 转换优先序列寄存器为 0; 采样保持时间=6;ADC_CLK=PCLK/2*2=0.2us; 转换序列个数为 2
    ADC12_Configure_Mode(ADC12_12BIT, One_shot_mode, 0, 6, 2, 2);
    //选择内部 INTVREF=1v 做参考电压, VREF-选择 VSS,选择 INTVREF 做参考, ADC_CLK 不能超过 2M
    ADC12_Configure_VREF_Selecte(ADC12_VREFP_FVR4096_VREFN_VSS);
    //转换序列 0,选择 ADCIN0 通道, 6 个转换周期, 连续重复采样次数为 1 平均值计算禁止

    ADC12_ConversionChannel_Config(ADC12_ADCIN10,ADC12_CV_RepeatNum1,ADC12_AVGDIS,0);
    ADC12_CMD(ENABLE);                   //使能 ADC 模块
    ADC12_ready_wait();                   //等待 ADC 模块配置完成
}
```

```

//
ADC12_ConfigInterrupt_CMD(ADC12_EOC,ENABLE);
ADC_Int_Enable();
ADC12_Control(ADC12_START);           //开始转换
}
    
```

● 代码说明:

ADC12_ConfigInterrupt_CMD(); ----用于使能 ADC 转换所有通道结束中断

ADC_Int_Enable(); ----用于开始 ADC 中断

● 中断接收:

```

/*****/
//ADC Interrupt
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void ADCIntHandler(void)
{
    // ISR content ...
    if(ADC0->SR&0x1)           //EOC
    {
        R_ADC_Buf1=ADC12_DATA_OUPUT(0);
        ADC12_Control(ADC12_START);
    }
}
    
```

● 代码说明:

U16_T ADC12_DATA_OUPUT(U16_T Data_index)----直接读出 adc 转换后的值.

ADC12_Control(ADC12_START);----开启下一次转换

3.4 ADC 连续转换模式

系统时钟选择内部 48Mhz，使能 ADCIN10、ADCIN11 通道,12BIT ADC,参考电压选择内部 4.096V。采集外部电压 3.3V

```

/*****/
//adc config
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void ADC12_CONFIG(void)
{
    
```

```

ADC12_RESET_VALUE(); //ADC 所有寄存器复位赋值
ADC12_CLK_CMD(ADC_CLK_CR , ENABLE); //使能 ADC CLK
ADC12_Software_Reset(); //ADC 软件复位
//选择 12BIT ADC; 连续模式; 转换优先序列寄存器为 0; 采样保持时间=6;ADC_CLK=PCLK/2=0.2us; 转换序列个数为 2
ADC12_Configure_Mode(ADC12_12BIT , One_shot_mode ,0, 6 ,2 , 2);
//选择内部 INTVREF=1v 做参考电压, VREF-选择 VSS,选择 INTVREF 做参考, ADC_CLK 不能超过 2M
ADC12_Configure_VREF_Selecte(ADC12_VREFP_FVR4096_VREFN_VSS);
//转换序列 0,选择 ADCIN0 通道, 6 个转换周期, 连续重复采样次数为 1 平均值计算禁止
ADC12_ConversionChannel_Config(ADC12_ADCIN0,ADC12_CV_RepeatNum1,ADC12_AVGDIS,0);
//转换序列 1,选择 ADCIN1 通道, 6 个转换周期, 连续重复采样次数为 1,平均值计算禁止
ADC12_ConversionChannel_Config(ADC12_ADCIN1,ADC12_CV_RepeatNum1,ADC12_AVGDIS,1);
ADC12_CMD(ENABLE); //使能 ADC 模块
ADC12_ready_wait(); //等待 ADC 模块配置完成
ADC12_Control(ADC12_START); //开始
}
volatile U32_T R_ADC_Buf1, R_ADC_Buf2;
/*****/
//main
/*****/
int main(void)
{
    APT32F102_init();
    while(1)
    {
        SYSCON_IWDCNT_Reload(); //看门狗
        ADC12_SEQEND_wait(0); //等待转换序列 0 转换完成
        R_ADC_Buf1 = ADC0->DR[0]; //保存转换结果
        //
        ADC12_SEQEND_wait(1); //等待转换序列 1 数据完成
        R_ADC_Buf2 = ADC0->DR[1]; //保存数据
    }
}

```

● **采集数据:**



图 3.4.1 采样值

计算公式：待测电压 = AD 值 / (满量程) * 基准电压

$$\text{待测电压} = 3274 / 4096 * 4.096 = 3.27V$$

4. 程序下载和运行

1. 将目标板与仿真器连接，分别为 VDD SCLK SWIO GND
2. 将需要配置好的 ADC 口连接待测试的外部电路。
3. 程序编译后仿真运行
4. 图 3.4.1 所示 R_ADC_Buf1 得到的就是需要采集的 ADC 数据。