

实验 10 11 12 13 - ADC 实验

1. 实验目的

掌握 NRF24LE1 的 ADC 的配置和使用。

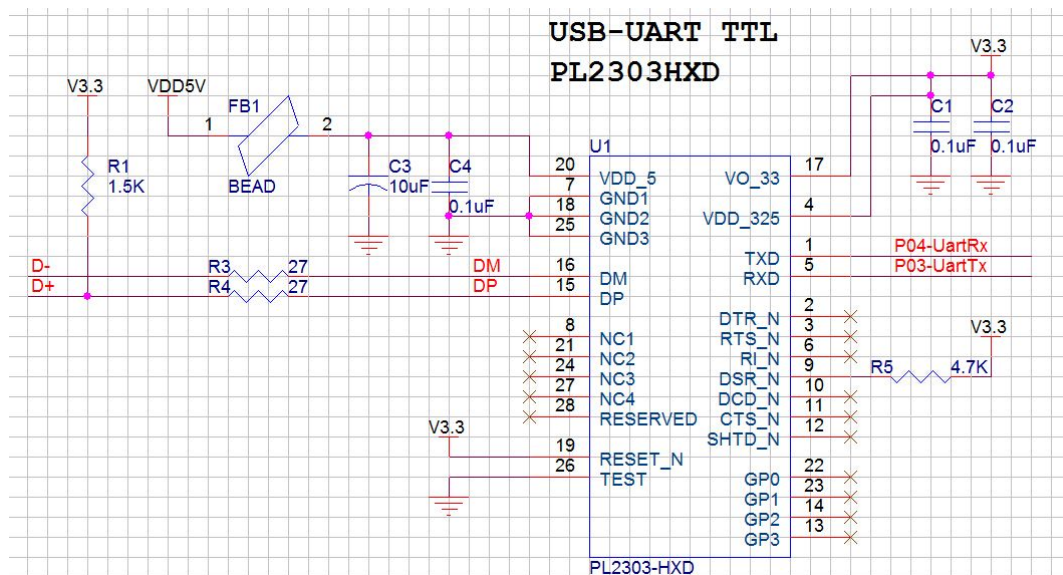
- 采样 VDD 电压，单次采样模式
- 采样 VDD 电压，连续采样模式
- 采样外部电位器电压，单次采样模式
- 采样外部电位器电压，连续采样模式

2. 实验内容

配置 NRF24LE1 的 ADC，分别使用单次和连续采样模式对 VDD 和外部电位器电压进行采样，采样结果通过串口输出。

3. 实验原理

3.1. 电路原理

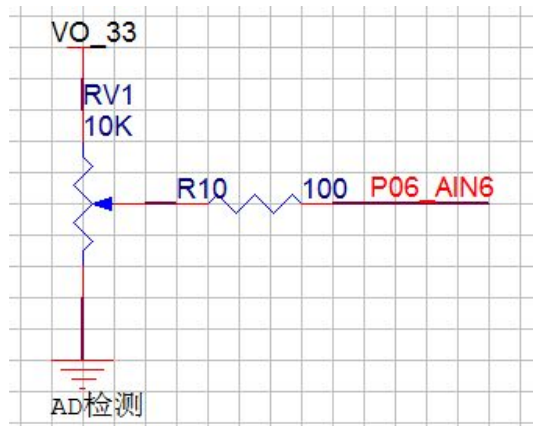


NRF24LE1 集成了一个串口，管脚分配如下：

P0.3: UART TXD

P0.4: UART RXD

开发板上使用了 USB 转 UART TTL 芯片 PL2303-HXD（注意：该芯片是最新的 HXD 版本，不需要外接晶振，早期 HX 版本的需要外接晶振）将串口信号转换为 USB 信号，方便和计算机连接。



开发板上 10K 电位器的抽头连接到 NRF24LE1 的 P0.6 管脚，P0.6 可配置为外部 ADC 检测通道 6。串接的 100 欧电阻是为了保护 IO，防止电位器调到 0 欧时损坏 IO(若此时 IO 不小心配置成了输出，即有可能损坏 IO)。

NRF24LE1 的 ADC 转换器特性如下：

- 6、8、10 或 12 位分辨率；
- 最多 14 路输入通道；
- 单端或差分输入；
- 全量程测量，基准电压可设置为内部基准，外部基准或 VDD；
- 单次转换模式转换时间为 3us；
- 2、4、8 或 16kbps 采样频率的连续采样模式；
- 极低功耗，2kbps 采样时的电流仅为 0.1mA；
- 具备测量供电电源的模式。

3.2. 寄存器配置

NRF24LE1 通过 5 个寄存器对 ADC 单元进行控制和访问：

- ADCCON1、ADCCON2、ADCCON3：包含配置位和控制位；
- ADCDATH、ADCDATL：存储转换结果。

表 1：ADCCON1 寄存器

地址	位	名称	类型	说明	复位值 0x00
0xD3	7	pwrap	RW	加电控制 0: ADC 掉电； 1: ADC 加电并配置所选的引脚为模拟输入。	
	6	busy	R	ADC 忙标志 0: 当前没有转换； 1: 转换正在进行中，当转换完成并且数据存储在寄存器 ADCDATH、ADCDATL 时，busy 标志清除。	
	5~2	chsel	RW	输入通道选择 0000: AIN0； 0001: AIN1； . .	

				<p>1101: AIN13; 1110: 1/3 · VDD; 1111: 2/3 · VDD。</p>
	1~0	refsel	RW	<p>基准源选择 00: 内部 1.2V 基准; 01: VDD; 10: AIN3 上的外部基准; 11: AIN9 上的外部基准。</p>

表 2: ADCCON2 寄存器

地址	位	名称	类型	说明	复位值 0x00
0xD2	7:6	diffm	RW	<p>选择单端或差分模式 00: 单端模式; 01: 差分模式且 AIN2 为反向输入。 10: 差分模式且 AIN6 为反向输入。 11: 未使用。</p>	
	5	diffm	RW	<p>选择单次模式或连续模式 0: 单次采样; 1: 连续采样, 采样率由 rate 位设定。</p>	
	4~2	rate	RW	<p>选择连续采样下的采样率 000: 2 ksps 001: 4 ksps 010: 8 ksps 011: 16 ksps 1XX: 保留 选择单次转换下的掉电时延: 000: 0μs 001: 6μs 010: 24μs 011: 无限 (清除 pwrup 到掉电模式) 1XX: 保留。</p>	
	1~0	tacq	RW	<p>采样窗口持续时间 (TACQ): 00: 0.75μs 01: 3μs 10: 12μs 11: 36μs。</p>	

表 3: ADCCON3 寄存器

地址	位	名称	类型	说明	复位值 0x00
0xD2	7:6	resol	RW	ADC 分辨率	

				00:6 位; 01:8 位; 10:10 位; 11:12 位;
	5	rljust	RW	选择 ADCDATH、ADCDATL 内的数据左对齐还是右对齐 0: 左对齐数据; 1: 右对齐数据。
	4	uflow	R	置位时 ADC 下溢 (转换结果全为 0)
	3	oflow	R	置位时 ADC 上溢 (转换结果全为 1)
	2	range	R	置位时 ADC 下溢或上溢 (oflow 或 uflow)
	1:0	—	—	未使用

表 4: ADCDATH 寄存器

地址	位	名称	类型	说明	复位值
0xD4	7:0	—	R	ADCDATA 左对齐或右对齐后的高有效字节	0x00

表 5: ADCDATL 寄存器

地址	位	名称	类型	说明	复位值
0xD5	7:0	—	R	ADCDATA 左对齐或右对齐后的低有效字节	0x00

通过对上述内容的掌握,对 ADC 进行配置,配置 ADC 用来检测供电电源 VDD 的电压 (使用 NRF24LE1 作为有源电子标签时,可以通过此方式实时检测电池电压)。调用库函数,对 ADC 的配置如下:

```
hal_adc_set_input_channel(HAL_INP_VDD1_3);    //设置通道 检测 1/3 VDD 电压
hal_adc_set_reference(HAL_ADC_REF_INT);        //设置参考电压,内部 1.2V
hal_adc_set_input_mode(HAL_ADC_SINGLE);        //单端输入
hal_adc_set_conversion_mode(HAL_ADC_SINGLE_STEP); //单次采样模式
hal_adc_set_sampling_rate(HAL_ADC_2KSPS);      //采样速率为 2ksps
hal_adc_set_resolution(HAL_ADC_RES_12BIT);     //12 位 ADC
hal_adc_set_data_just(HAL_ADC_JUST_RIGHT);     //数据右对齐
```

配置完成后,在需要进行 ADC 检测时,调用库函数:

```
hal_adc_start();          //启动 ADC
while( hal_adc_busy());   //等待 ADC 转换结束
```

ADC 转换完成后,即可调用库函数:

hal_adc_read_MSB()和 hal_adc_read_LSB()读取转换结果。

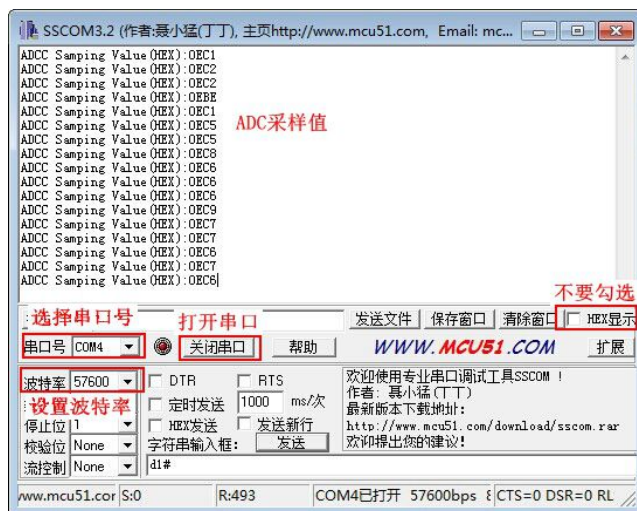
4. ADC 实验：VDD 检测-单次采样

4.1. 实验步骤

- 在 Keil uVision4 中打开工程 “adc_vdd_single.uvproj” 工程；
- 编译工程，注意查看编译输出栏，观察编译的结果，如果有错误，修改程序，直到编译成功为止；



- 将编译生成的 HEX 文件 “adc_vdd_single.hex” (该文件位于工程目录下的 “Object” 文件夹中) 通过编程器下载到开发板中运行。
- 打开串口调试助手，选择串口号，设置波特率为 57600，打开串口，注意不要勾选 “HEX 显示”。



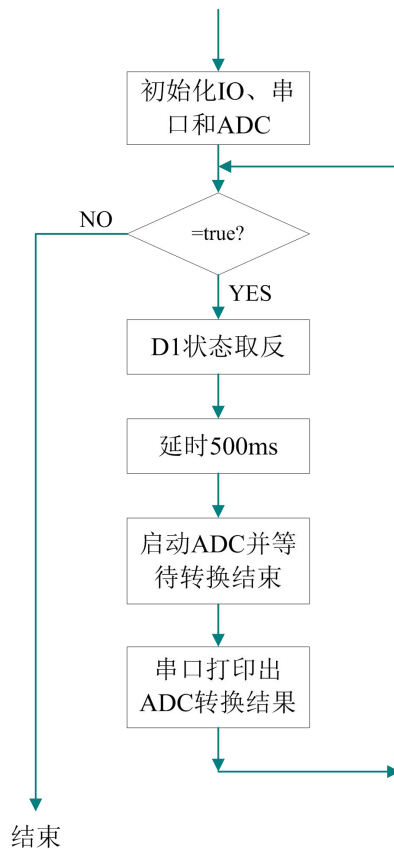
- 观察接收窗口中上报的 ADC 采样值，由此采样值可计算出 VDD 电压。计算公式如下：

$$\text{电压计算公式: } VDD = (1.2 * 3) * \text{采样值} / 4096$$

按照此公式对上图中的采样值 0x0EC7 进行计算，结果是：3.3249V。

4.2. 实验程序

4.2.1. 程序流程



4.2.2. 程序清单

```
#define D1 P00 //开发板上的指示灯 D1
```

```

/*****
*描 述：配置 IO P0.0 为输出，驱动 LED。P03 输出: UART TXD, P04:输入 UART RXD
*入 参：无
*返回值：无
*****/

void IO_Init(void)
{
    P0DIR &= ~0x01; //配置 P0.0 和 P0.1 为输出
    P0DIR &= ~0x08; //P03:输出 UART TXD
    P0DIR |= 0x10; //P04:输入 UART RXD
    D1 = 1; //设置 D1 初始状态为熄灭
}

/*****

```

```

*描 述 : ADC 初始化, 用 ADC 检测 VDD 电压
*入 参 : 无
*返回值 : 无
*****/

void AdcInit(void)
{
    hal_adc_set_input_channel(HAL_INP_VDD1_3); //设置通道 检测 1/3 VDD 电压
    hal_adc_set_reference(HAL_ADC_REF_INT); //设置参考电压, 内部 1.2V
    hal_adc_set_input_mode(HAL_ADC_SINGLE); //单端输入
    hal_adc_set_conversion_mode(HAL_ADC_SINGLE_STEP); //单次采样模式
    hal_adc_set_sampling_rate(HAL_ADC_2KSPS); //采样速率为 2ksps
    hal_adc_set_resolution(HAL_ADC_RES_12BIT); //12 位 ADC
    hal_adc_set_data_just(HAL_ADC_JUST_RIGHT); //数据右对齐

    ADCDATH&=0xF0;
    ADCDATL&=0x00;
}
/*****
*描 述 : 串口打印 16 进制字符
*入 参 : 输出的字符
*返回值 : 无
*****/

void PutHexString (uint8_t x)
{
    uint8_t c;

    c = (x>>4)& 0xf;
    if (c > 9)
        hal_uart_putchar('A'+c-10);
    else
        hal_uart_putchar('0'+c);

    c = x & 0xf;
    if (c > 9)
        hal_uart_putchar('A'+c-10);
    else
        hal_uart_putchar('0'+c);
}
/*****
*描 述 : 串口打印字符串
*入 参 : 无
*返回值 : 无
*****/

```

```

void PutString(char *s)
{
    while(*s != 0)
        hal_uart_putchar(*s++);
}

/*****
*描 述：主函数
*入 参：无
*返回值：无
*****/

void main()
{
    IO_Init(); //初始化 IO
    hal_uart_init(UART_BAUD_57K6); // 初始化 UART，波特率 57600

    while(hal_clk_get_16m_source() != HAL_CLK_XOSC16M) // 等待时钟稳定
        ;
    AdcInit(); //初始化 ADC
    EA = 1;    //开启全局中断

    while(1)
    {
        D1 = ~D1;    //D1 闪烁指示 ADC 转换
        delay_ms(500);

        hal_adc_start();    // 启动 ADC
        while( hal_adc_busy())    // 等待 ADC 转换结束
            ;
        PutString("ADCC Sampling Value(HEX):");
        PutHexString(hal_adc_read_MSB());    //串口输出 ADC 转换结果
        PutHexString(hal_adc_read_LSB());
        PutString("\n");    //换行，方便在串口调试助手中观察数据
    }
}

```

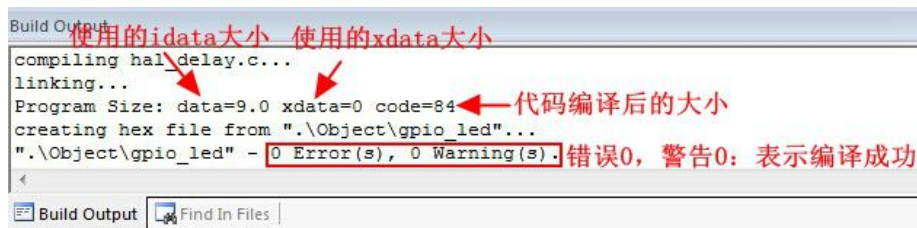
5. ADC 实验：VDD 检测-连续采样

连续采样实验和单次采样实验类似，所不同的是 ADC 配置为“连续采样模式”，采样完成后触发中断，在中断中读出采样结果。主循环中将读出的采样结果通过串口输出，为了防止串口输出过快，在主循环中增加了 500ms 的延时，以方便观察采样结果。

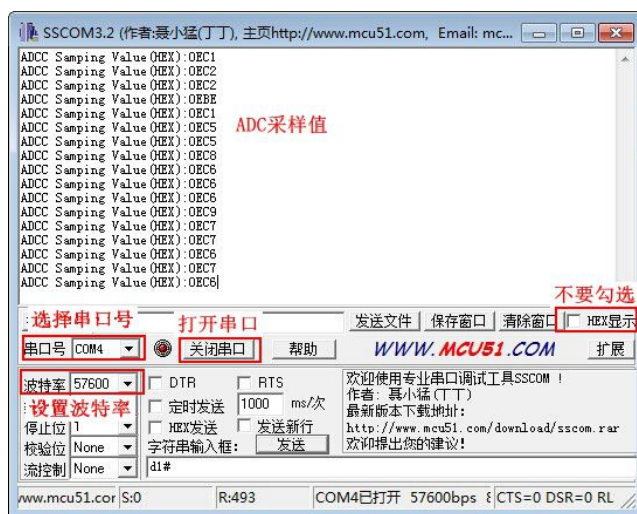
5.1. 实验步骤

- 在 Keil uVision4 中打开工程“adc_vdd_continuous.uvproj”工程；

- 编译工程，注意查看编译输出栏，观察编译的结果，如果有错误，修改程序，直到编译成功为止；



- 将编译生成的 HEX 文件“adc_vdd_continuous.hex” (该文件位于工程目录下的“Object”文件夹中)通过编程器下载到开发板中运行。
- 打开串口调试助手，选择串口号，设置波特率为 57600，打开串口，注意不要勾选“HEX 显示”。



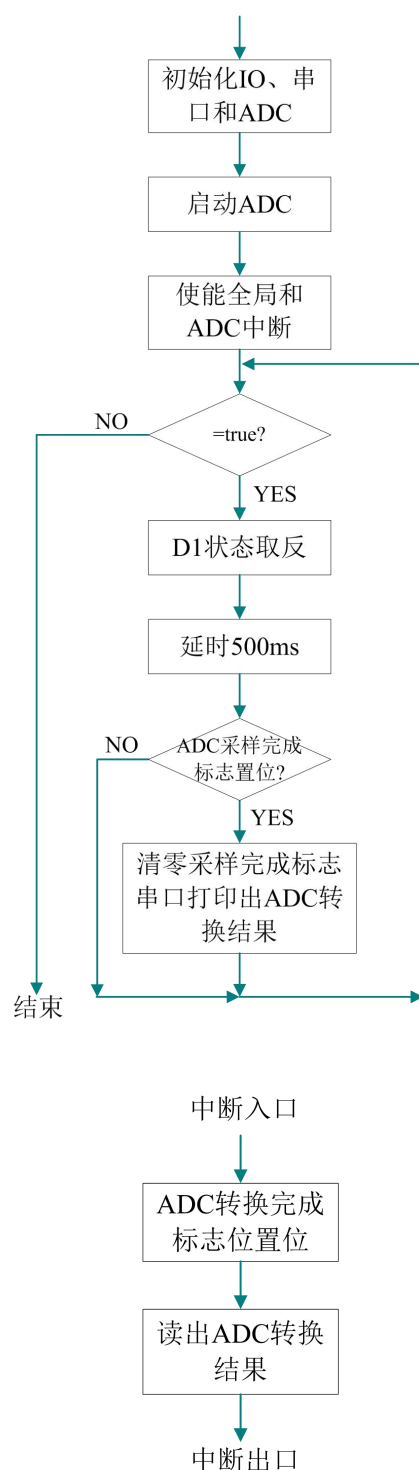
- 观察接收窗口中上报的 ADC 采样值，由此采样值可计算出 VDD 电压。计算公式如下：

$$\text{电压计算公式: } VDD = (1.2 \times 3) \times \text{采样值} / 4096$$

按照此公式对上图中的采样值 0x0EC7 进行计算，结果是：3.3249V。

5.2. 实验程序

5.2.1. 程序流程



5.2.2. 程序清单

连续采样的程序和单次采样类似，所不同的是将 ADC 配置为连续采样模式，如下所以：

`hal_adc_set_conversion_mode(HAL_ADC_CONTINUOUS);` //配置 ADC 为连续采样模式

同时，开启 ADC 中断，在中断中读取采样结果。

MISC = 1; //使能 ADC 中断

```

/*****
*描述：ADC 中断服务函数
*入参：无
*返回值：无
*****/

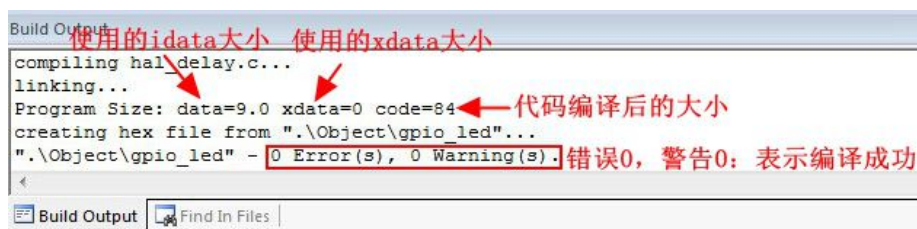
void adc_irq() interrupt INTERRUPT_MISCIQ
{
    SamplingFined = 1;    //采样完成标志置位
    SampValueMsb = hal_adc_read_MSB(); //读取 ADC 采样结果
    SampValueLsb = hal_adc_read_LSB();
}

```

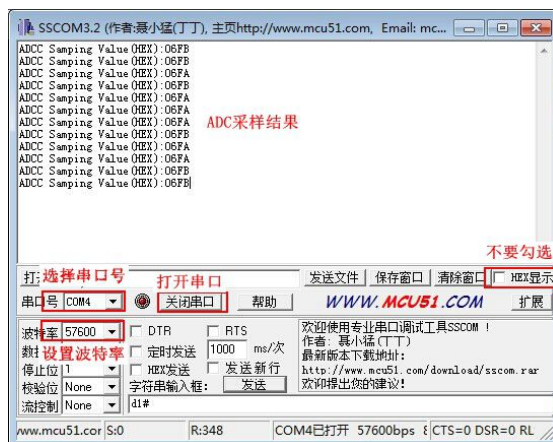
6. ADC 实验：外部采样-单次采样

6.1. 实验步骤

- 在 Keil uVision4 中打开工程 “adc_external_single.uvproj” 工程；
- 编译工程，注意查看编译输出栏，观察编译的结果，如果有错误，修改程序，直到编译成功为止；



- 将编译生成的 HEX 文件 “adc_external_single.hex” (该文件位于工程目录下的 “Object” 文件夹中)通过编程器下载到开发板中运行。
- 打开串口调试助手，选择串口号，设置波特率为 57600，打开串口，注意不要勾选 “HEX 显示”。



- 观察接收窗口中上报的 ADC 采样值，由此采样值可计算出电位器中心抽头上的电

压。计算公式如下：

电压计算公式： $V = (3.3) * \text{采样值} / 4096$

注意：外部ADC采样电压的计算方式和内部是不一样的。

按照此公式对上图中的采样值 0x06FA 进行计算，结果是：1.438V。

- 旋转电位器上的旋钮，观察接收窗口上的采样值的变换。

6.2. 实验程序

6.2.1. 程序流程

参考“VDD 检测-单次采样”程序的流程。

6.2.2. 程序清单

参考“VDD 检测-单次采样”的程序清单。和“VDD 检测-单次采样”不同的是，在 ADC 初始化中，将 ADC 配置为对通道 AIN6 进行采样。

7. ADC 实验：外部采样-连续采样

- 参考“VDD 检测-连续采样”和“外部采样-单次采样”的实验步骤。
- 参考“VDD 检测-连续采样”程序的流程。