

# 电化学开发板 模拟量输出 设计指南

Ver1.0

威海精讯畅通电子科技有限公司

Weihai JXCT Electronics Co., Ltd.

## 目录

一、产品简介.....	3
二、具体参数.....	3
三、工作原理图.....	3
四、编程和标定思路.....	6
五、开发板外观图.....	9

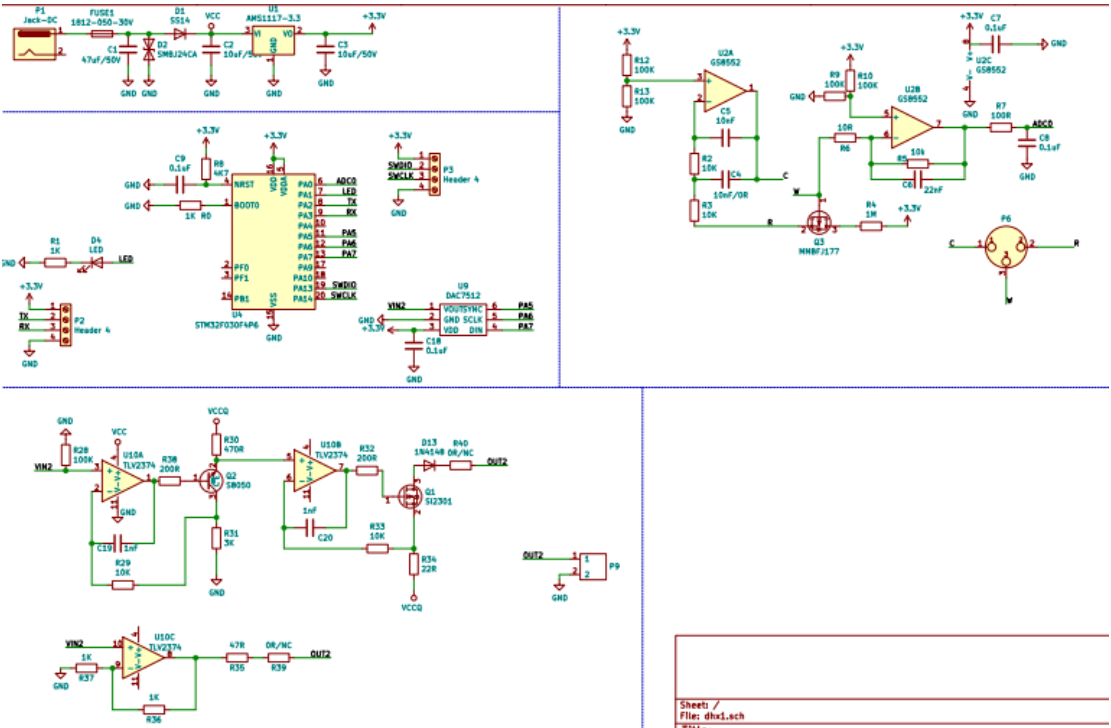
一、产品简介

此开发板是我司针对于 2 电极和 3 电极电化学探头检测不同的气体含量而开发。此开发板支持 4-20mA 输出、0-5V 输出、0-10V 输出，供电方式支持 5/12V。此开发板是基于 STM32F030F4P6 芯片，采用精密运放芯片 GS8552 采集电化学探头的输出信号，经过算法处理，可以用不同的方式输出。我司提供开发例程，模块，客户可以通过更改部分电路以及例程实现相应的功能。

二、具体参数

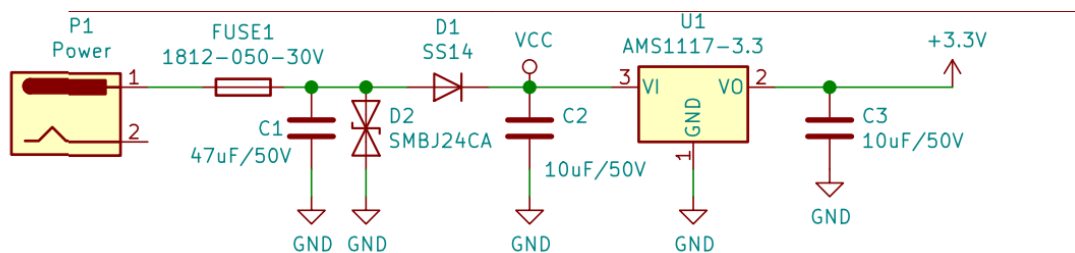
参数	技术指标
输入电源	5V/12V DC
输出方式	4-20mA/0-5V/0-10V 输出
尺寸	82mm*35mm*16mm(长*宽*高)
储存温度	-25 <sup>0</sup> C - +85 <sup>0</sup> C
工作温度	-10 <sup>0</sup> C - +70 <sup>0</sup> C
重量	0.05kg

三、工作原理图



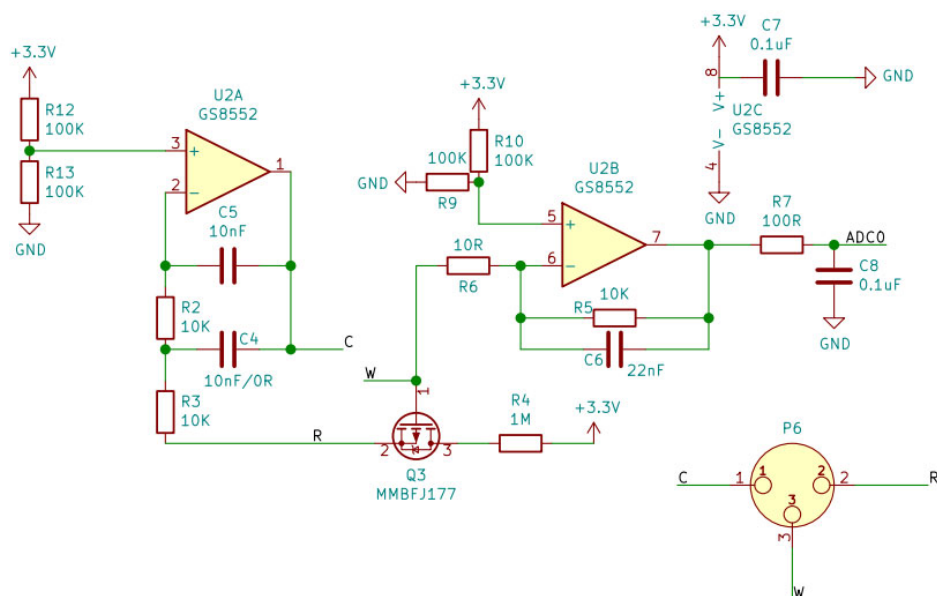
图一 总原理图

如图一所示，这是总原理图，包含降压模块、ADC 采集模块、STM32F030F4P6 单片机最小系统模块、485 模块、232 模块。



图二 电源模块

如图二所示，这是基于 AMS1117-3.3V 的降压电路，可以把输入为 5V 或者输入为 12V 的电压转换成 3.3V 输出。

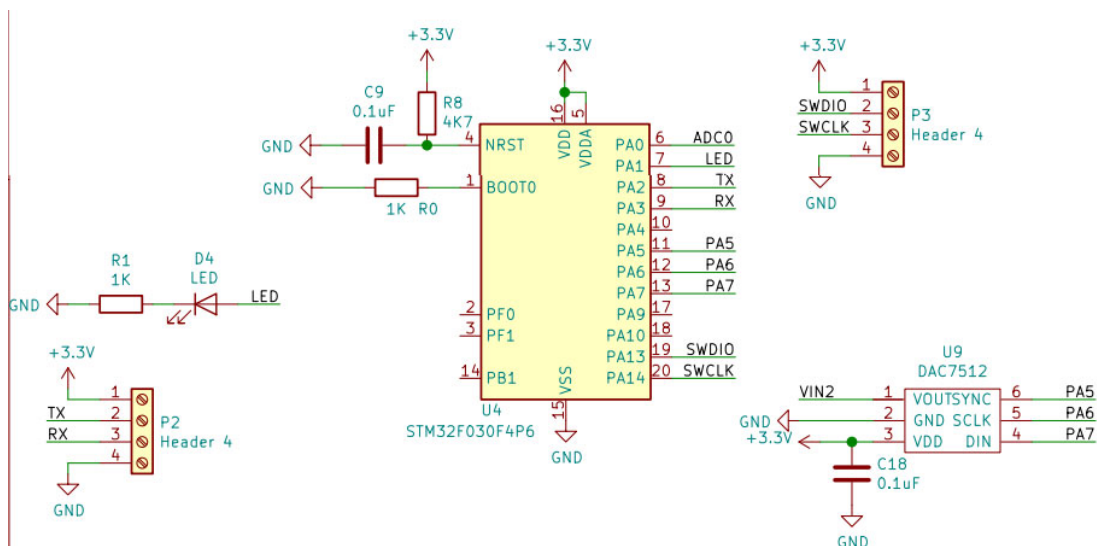


图三 ADC 采集模块

如图三所示是 ADC 采集模块，使用的是 GS8552 运放芯片，该 GS8552 放大器是单电源，微功率，零漂移 CMOS 运算放大器，放大器提供带宽为 1.5MHz，轨对轨输入和输出，单电源操作从 2.1V 到 5.5V。本采集模块支持 2 电极和 3 电极电化学气体探头。C 是对电极，W 是工作电极，R 是参比电极。

关于 2 电极和 3 电极的修改说明

- 1) 若使用 2 电极电化学探头，C4 应焊接 0R 电阻，R3 应焊接 100R 电阻；
- 2) 若使用 3 电极电化学探头，C4 应焊接 10nF 电容，R3 应焊接 10K 电阻。



2) 使用 0-5V 电压输出时, R40 不需要焊接, R39 焊接 0R 电阻, 此时供电电源 5V/12V 均可, 可通过调换 R36 的阻值改变输出电压;

3) 使用 0-10V 电压输出时, R40 不需要焊接, R39 焊接 0R 电阻, 此时供电电源应为 12V, 可通过调换 R36 的阻值改变输出电压。

## 四、编程和标定思路

### 1. 编程思路

#### 1) adc 采集

一路 adc 采集, 单片机处理采集到的数据, 对采集到的数据进行软件滤波处理, 得到一个较为稳定的数值。

#### 2) 输出方式

输出方式为模拟量输出, 如 4-20ma, 0-5V, 0-10V, 模拟量的值可以通过修改 DAC7512(u16 DADData)的 DADData 值来改变。

### 2. 标定思路

模拟量和检测气体的浓度有一定的线性关系, 根据线性关系, 进行零点和满点标定。

### 3. 参考程序

#### 1) adc 采集处理程序

```
#define A 101
u32 caiji[A];
uint32_t ADC_Check(void)//采集处理程序
{
    uint16_t i,j,k,h;
    u32 ResultI2;
    for(i=0;i<A;i++)
    {
        while(!ADC_GetFlagStatus(ADC1, ADC_FLAG_EOC)); //等待转换结束
        caiji[i]=(uint32_t)ADC_ConvertedValue[i];
        for(j=0;j<500;j++)
        {}
    }
    for(j=0;j<A-1;j++) //排序取中值
    {
        for(k=0;k<A-j-1;k++)
        {
            if(caiji[k]>caiji[k+1])
            {
                ResultI2=caiji[k];
                caiji[k]=caiji[k+1];
                caiji[k+1]=ResultI2;
            }
        }
    }
}
```

```

    }
    ResultI=caiji[(A-1)/2];
    ResultI=ResultI*3300;
    ResultI=ResultI/4096;
    ADC_ClearITPendingBit(ADC1,ADC_IT_EOC);
    ResultI1 = ResultI;
    return ResultI1;
}

```

## 2) DAC7512 驱动程序

```

#define IO_SYNC_H GPIO_SetBits(GPIOA,GPIO_Pin_5)
#define IO_SCLK_H GPIO_SetBits(GPIOA,GPIO_Pin_6)
#define IO_DIN_H   GPIO_SetBits(GPIOA,GPIO_Pin_7)
#define IO_SYNC_L GPIO_ResetBits(GPIOA,GPIO_Pin_5)
#define IO_SCLK_L GPIO_ResetBits(GPIOA,GPIO_Pin_6)
#define IO_DIN_L   GPIO_ResetBits(GPIOA,GPIO_Pin_7)

void SetSDAPP()
{
    GPIO_InitTypeDef GPIO_InitStructure;
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_5|GPIO_Pin_6|GPIO_Pin_7;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT;
    GPIO_InitStructure.GPIO_OType = GPIO_OType_PP;
    GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz; //高速 GPIO
    GPIO_Init(GPIOA,&GPIO_InitStructure);
    GPIO_SetBits(GPIOA,GPIO_Pin_7);
}

void SetSDAOD()
{
    GPIO_InitTypeDef GPIO_InitStructure;
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_7;
    GPIO_InitStructure.GPIO_Mode = GPIO_Mode_OUT;
    GPIO_InitStructure.GPIO_OType = GPIO_OType_OD;
    GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz; //高速 GPIO
    GPIO_Init(GPIOA,&GPIO_InitStructure);
}

void DAC7512(u16 DADData)          //满电压是 5000mV  4096 分度值
{
    u8 i;
    u32 temp=DADData;
    SetSDAPP();
    IO_SCLK_L;
    IO_SYNC_L;//启动转换
    IO_SCLK_H;
    temp=temp*5000/4096;//    /5000*4096 输出电压值 0-3.3V
}

```

```

if(temp>4095)
    temp=4095 ;
DADData=temp;
for(i=0;i<16;i++)//AD7888 写入命令与获取数值一次。
{
    if((DADData&(0x8000>>i))>>(15-i))
    {
        IO_DIN_H;//启动转换
    }
    else
    {
        IO_DIN_L;//启动转换
    }
    IO_SCLK_L;    //下降写入
    IO_DIN_L;//启动转换;每次数据移入后，DIN 还需要置 0 真变态
    IO_SCLK_H;//
}
IO_SCLK_L;    // 这里还需要 SCLK 置 0，更变态,datasheet 上却没提
IO_SYNC_H;//下次转换必须先置 1

```

```

SetSDAOD();
}

```

### 3) 标定程序

```

void Tiaozheng()
{
    dianliu=Result11;//**ADCcheng/ADCchu+ADCjia;

    if(dianliu_0<=dianliu<=dianliu_1)
    {
        dianliutruel=1*(dianliu-dianliu_0)/(dianliu_1-dianliu_0);
    }

    else if(dianliu_1<dianliu)
    {
        dianliutruel=1*(dianliu-dianliu_1)/(dianliu_1-dianliu_0)+1;
    }

}

```

### 4) 主程序

```

void main()//主程序
{

    ReadInf();
    Systick_Init();
}

```



