MAX31865程序说明文档

Rev 1.0

目录

[一、程序分层 3](#_Toc5955382)

[二、程序功能 3](#_Toc5955383)

[1.hal层功能 3](#_Toc5955384)

[2.max\_i2cdrv层功能 3](#_Toc5955385)

[3.max5825层功能 4](#_Toc5955386)

[4.main层功能 8](#_Toc5955387)

[三、程序运行结果图 10](#_Toc5955388)

[附录：max5825层部分程序 11](#_Toc5955389)

# 一、程序分层



# 二、程序功能

## 1.hal功能

hal层为MCU硬件抽象层，完成GPIO、SPI等功能。

## 2.max\_spidrv功能

max\_spidrv层提供SPI初始化、发送数据、读取数据等功能。函数实现如下：

1）void SPIM\_TranslateInit(void)

提供SPI传输数据初始化功能，初始化MCU中SPI寄存器。具体有发送写数据，写寄存器地址，读单字节数据，读多字节数据的初始化功能。

传入参数：void。

2）void SPI\_Inital(void)

提供SPI的初始化功能，具体有SPI主机端口配置，系统时钟配置，SPI主机初始化配置，片选信号引脚配置，片选引脚高电平使能，SPI传输数据初始化。

传入参数：void。

3）void Configure\_DRDY\_Pin(void)

配置DRDY引脚及功能。

传入参数：void。

4）uint32\_t Get\_DRDYState(void)

使用GPIO\_InGet()函数获得DRDY引脚状态，正常情况下为0。

传入参数：void。

5）void Configure\_SS\_Pin(void)

配置片选信号SS(CS)引脚功能。

传入参数：void。

6）void SPI\_WriteByte(uint8\_t Write\_address, uint8\_t Write\_value)

SPI通信中写单字节

传入参数：Write\_address，写入地址

Write\_value，写入数据

7）uint8\_t SPI\_ReadByte(uint8\_t Read\_address)

SPI通信中读单字节

传入参数：Read\_address，需要读的数据所在的地址

8）void SPI\_Read(uint8\_t Read\_address, uint8\_t \*Read\_buff, uint8\_t Read\_nBytes)

SPI通信读多字节

传入参数：Read\_address，需要读的数据所在的地址

Read\_buff，指向数据所在地址的指针

Read\_nBytes，读取多字节的个数

9）void SPIM1\_CS\_LOW()

使能SS(CS)片选引脚输出低电平

9）void SPIM1\_CS\_HIGH()

使能SS(CS)片选引脚输出高电平

## 3.max31865层功能

Max31865层提供控制max31865芯片8路实现SPI通信功能。函数实现如下：

1）void max31865\_init(max31865\_configuration\* configuration)

初始化MAX31865。

传入参数：configuration，配置MAX31865初始化的结构体指针。

2）void max31865\_startconverntion(uint8\_t uch\_conversion\_mode, uint8\_t fault\_decet\_mode)

使能MAX31865开始进行温度转换。

传入参数：conversion\_mode，转换模式，有三种模式可选，Stop\_Conversion\_Bit，One\_Shot\_Conversion，Automatic\_Conversion。

fault\_decet\_mode，错误检测模式，有四种模式可选，Fault\_No\_Action，Fault\_Enable\_Auto，Fault\_Enable\_Manual\_cl1，Fault\_Enable\_Manual\_cl2。

3）void max31865\_stopconverntion(void)

MAX31865停止进行温度转换。

传入参数：void

4）void max31865\_clearfaultstatus(void)

清除MAX31865错误状态，在一次循环末尾处使用。

传入参数：void

5）void fault\_detectioncyclecontrol(uint8\_t uch\_fault\_detection\_mode)

周期性进行错误状态检测。

传入参数：uch\_fault\_detection\_mode，错误检测模式；

6）void maxim\_setvbias(void)

使能Vbias。使能Vbias后可以开始进行SPI通信

传入参数：void

7）void maxim\_clearvbias(void)

不使能Vbias，关闭Vbias以降低功耗

8）void max31865\_writeregister(uint8\_t uch\_register\_address, uint8\_t uch\_register\_value)

MAX31865向一个寄存器写入单字节。

传入参数：uch\_register\_address，写入的寄存器地址

uch\_register\_value，向寄存器相应地址中写入的数据

9）uint8\_t max31865\_readregister(uint8\_t uch\_register\_address)

MAX31865向一个寄存器读取单字节。

传入参数：uch\_register\_address，读取的单字节所在的寄存器地址。

10）void max31865\_read\_nregisters(uint8\_t uch\_register\_address, uint8\_t \*uch\_buff,uint8\_t uch\_nBytes)

MAX31865读多字节。

传入参数：uch\_register\_address，读取字节所在寄存器地址。

uch\_buff，指向数据所在地址的指针

uch\_nBytes，指定一次读取几个字节

11）void maxim\_set\_fault\_threshold(float high\_threshold, float low\_threshold)

设置温度阈值。

传入参数：high\_threshold，温度阈值上限

low\_threshold，温度阈值下限

## 4.main功能

main层完成SPI、max31865初始化功能，通过SPI通信协议，完成温度数据读取。流程图如下：



# 三、程序运行结果图

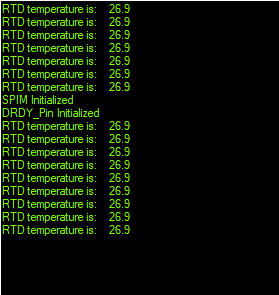


图 1串口调试输出

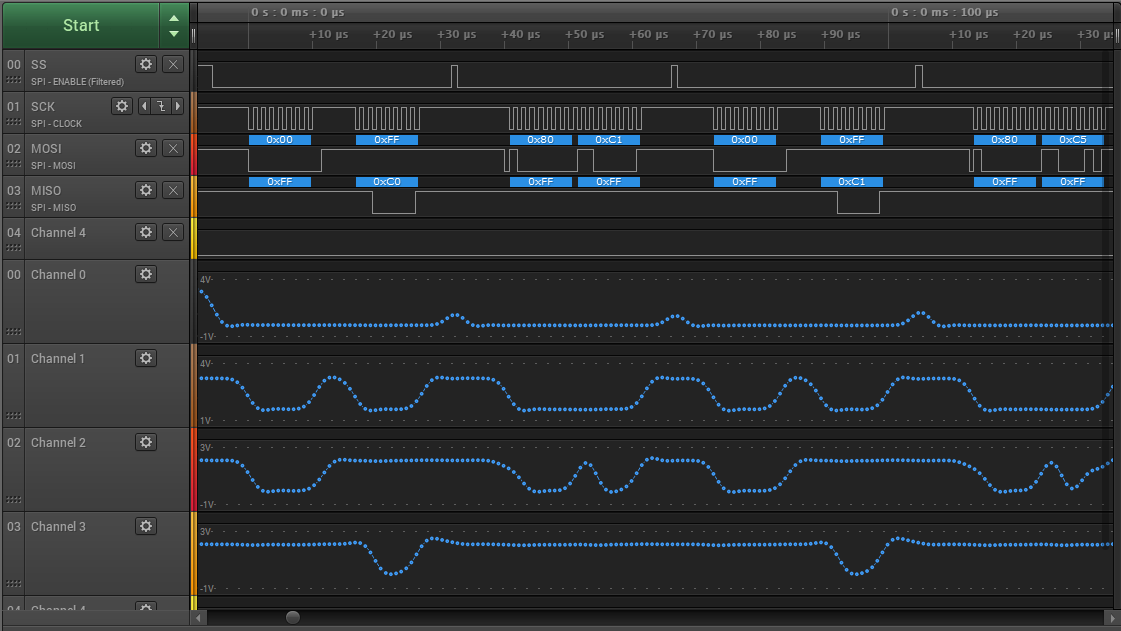


图 2 SPI传输