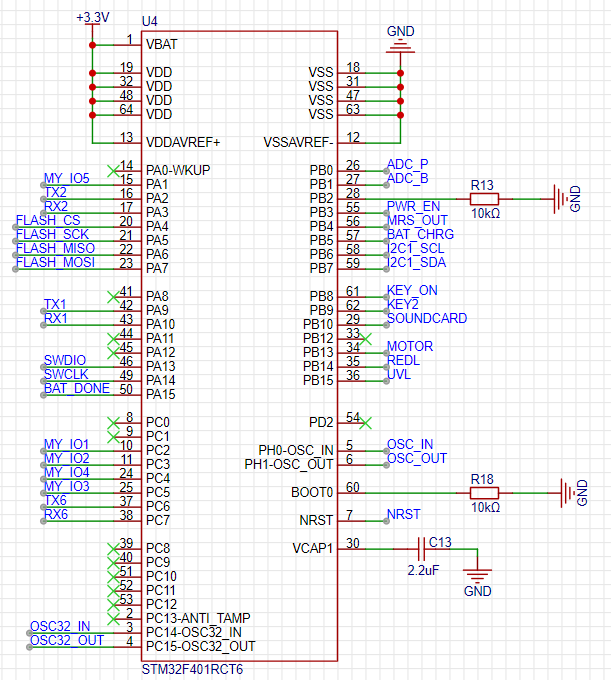
# 激光枪主板说明

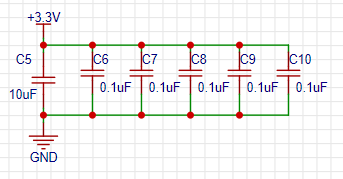
## 硬件

### 电路

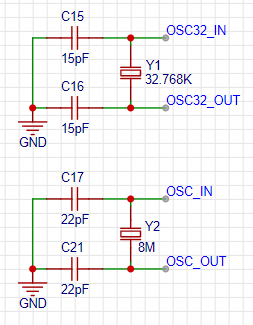
#### 单片机相关电路图



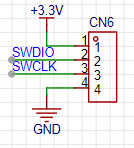
其中TXRX为usart串口接入口、FLASH是连接存储器芯片的spi引脚、SWDIO和SWCLK用于烧录程序、MY\_IOx是MP3的控制引脚、OSC是晶振引脚、ADC引脚是压力传感器板的值读取和电压值读取、PWR\_EN是电源开关接口、BAT是充电相关的引脚、KEY是按键、SOUNDCARD是MP3、MOTOR是电机、REDL是红点可见光、UVL是不可见光、NRST是复位、I2C是oled屏幕的引脚、MRS\_OUT是磁感应的读取口。



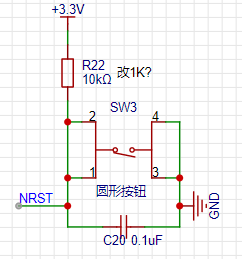
滤波电容放在单片机芯片的电源口，每个口放一个。



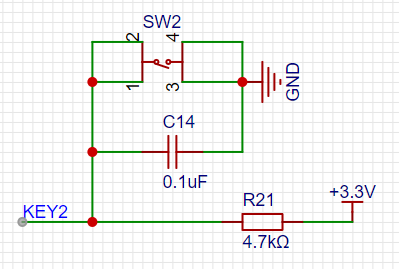
晶振电路。



烧录电路。

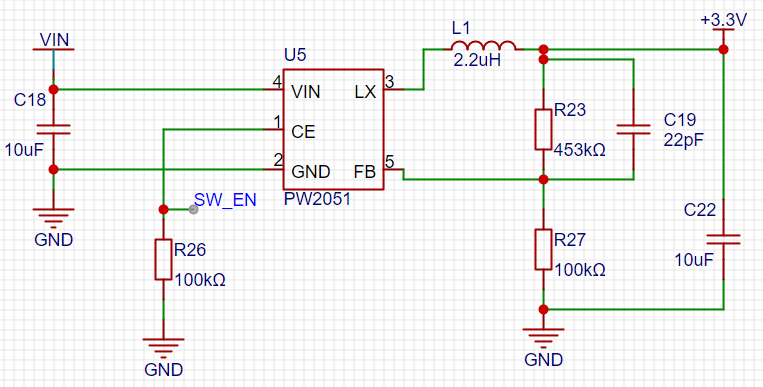


复位按钮电路。在上一代中出现过复位问题，有些线路在硬件中通了（具体见“PCB制作”的文档），此时把10K电阻换成1K就应付了过去，不过下一代修改过电路应该已不存在问题。

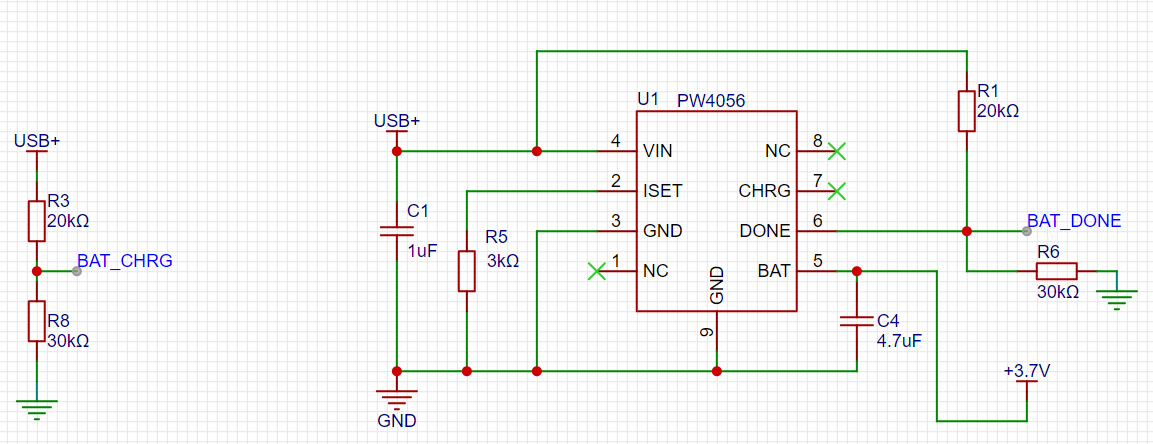


按键电路。

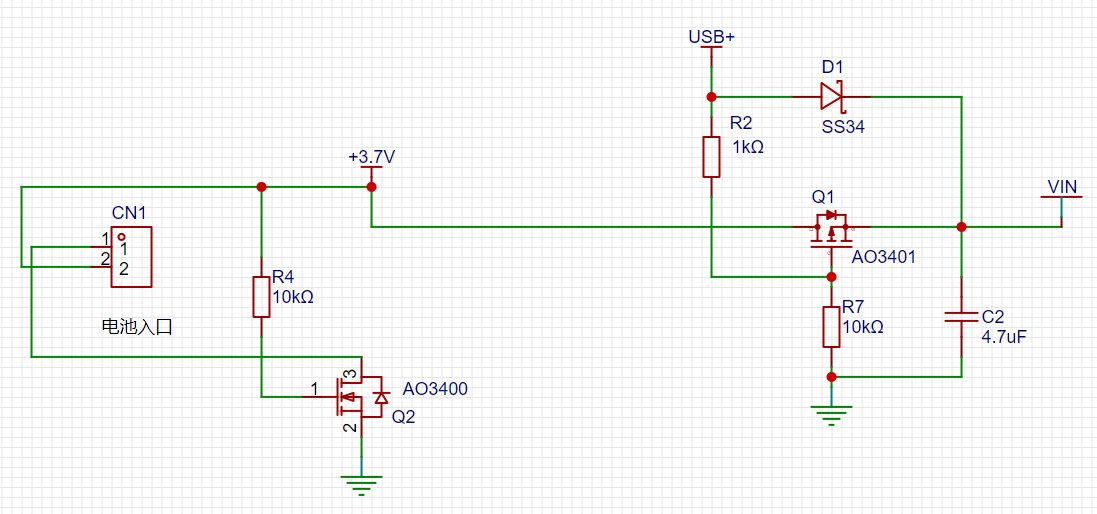
#### 供电电路



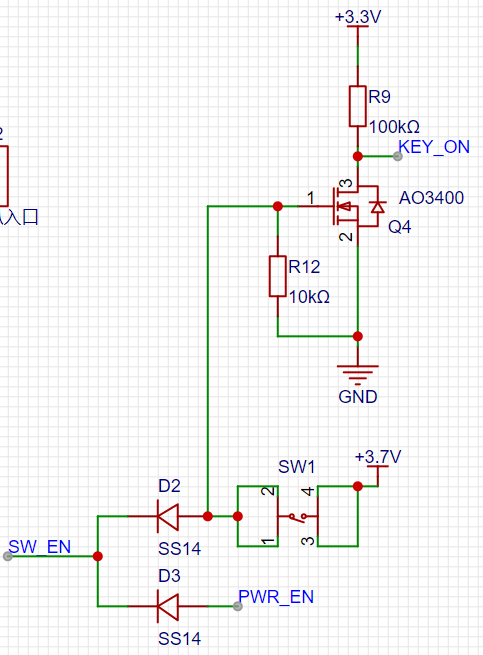
3.7V转3.3V主电源电路。



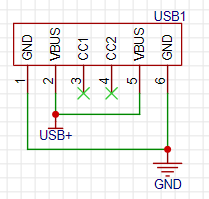
充电电路。



防电池反接电路。

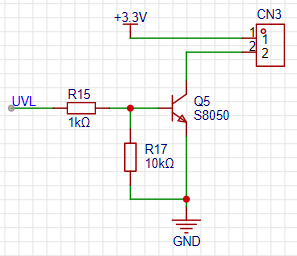


开关机电路，实现了长按开机。

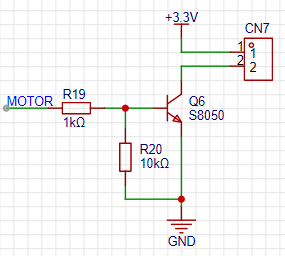


TYPE-C充电插口。

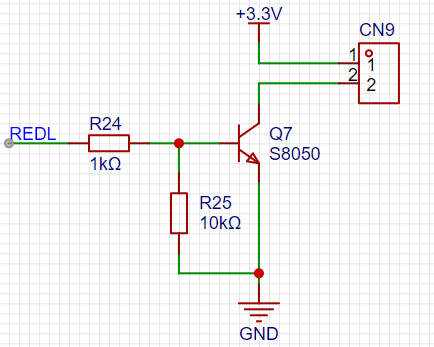
#### 3）开关控制电路



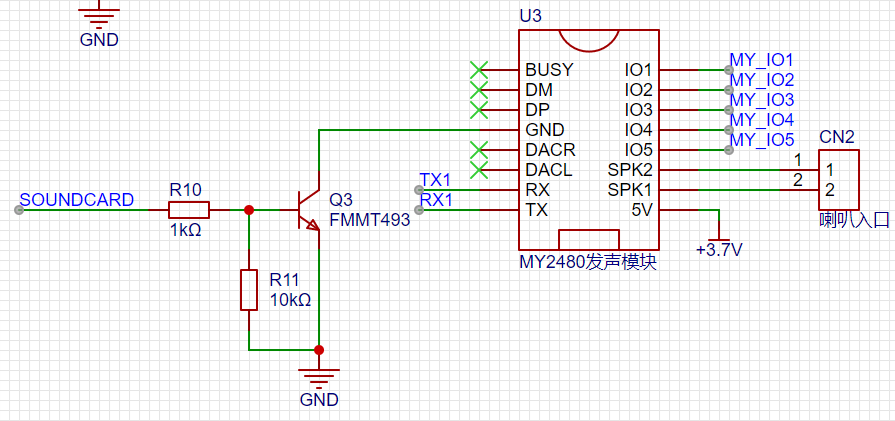
不可见光电路。



电机电路。

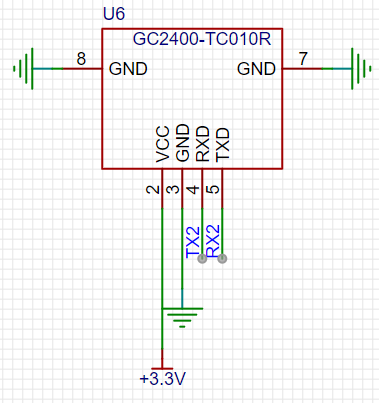


红点激光电路。

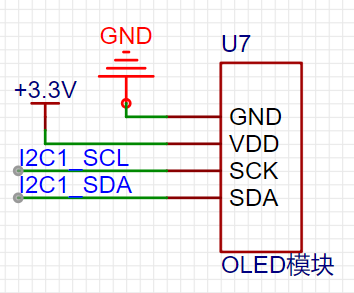


MP3电路。

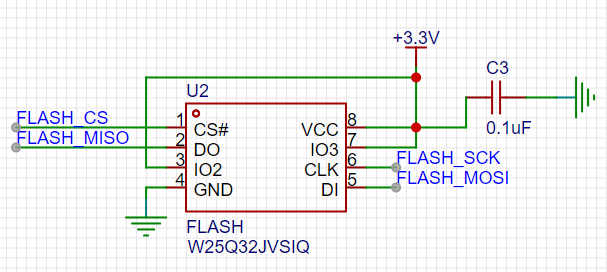
#### 4）模块电路



无线模块电路。

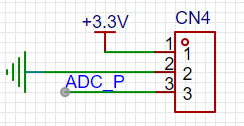


OLED显示屏电路。

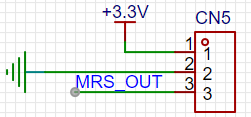


FLASH存储电路。

#### 5）传感器插口电路



压力传感器插口。



磁力传感器插口。

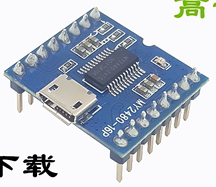
### 1.2 选型

单片机选择STM32F401RCT6。

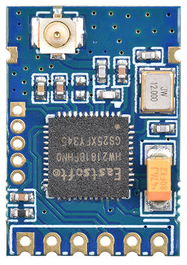
显示屏采用0.91寸OLED屏，I2C协议，非裸屏。



MP3采用MY2480-16P型号，内存大小采用2M。



无线模块采用硅传科技的GC2400-010R。



FLASH存储器W25Q都可以。

TYPE-C选择6P直插。

电源电池插口选择XH2.54-2P弯针座、电机红灯不可见灯开关控制三个选择PH2.0-2P弯针座、压力传感器和磁感应选择ZH1.5-3P弯针座。

烧录口选择XH2.54-4P弯针，不要带座的只要针。

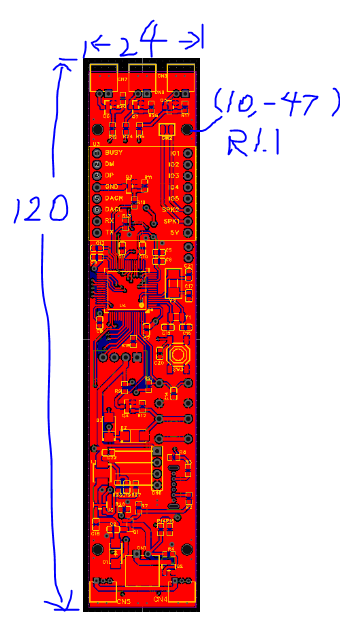
电池选择3.7V-1400mAH锂电池，长50宽25厚10。线头用XH2.54-2P。



喇叭用8欧2W。喇叭的线用28AWG，长8mm。



### 1.3 尺寸图



## 2.功能描述

激光枪板主要的功能是按键控制菜单进行设置、OLED屏幕显示各项信息、接收来自压力传感器板和磁感应板的信息、FLASH内存存储信息、无线模块传送信息给靶面板、MP3发出枪声、电机振动模拟后坐力、控制两个激光的运作。

## 3.软件

### 3.1 开发环境

使用RT-Thread进行系统开发，单片机使用STM32F401RCT6型号进行开发。项目名为AcGunShooterV2，解压后导入RT-Thread Studio即可编程开发。Git的话在网址https://github.com/hafuli123/AcGun-Gun/tree/master。

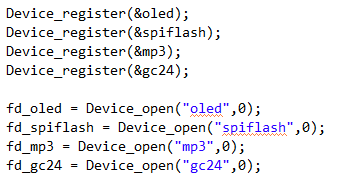
音频素材可在网上搜索，注意需要MP3文件的音效才行。

图片素材在网址https://ps.gaoding.com/，稿定设计网，进行在线PS的画图设计。PS画好图片后导出为BMP，然后在PCtoLCD软件中导出码。

### 3.2 main函数、启用设备和线程

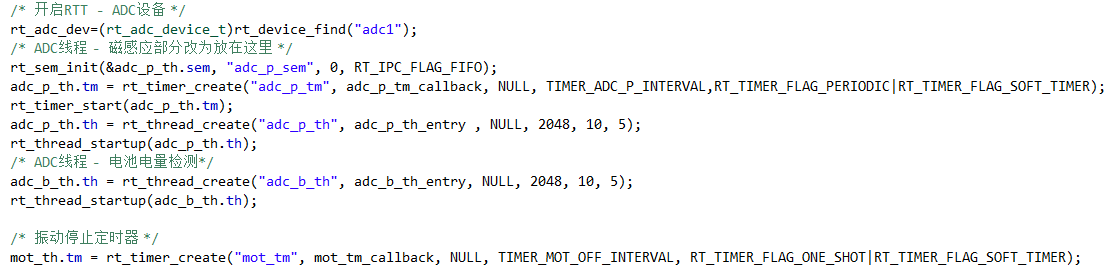
打开RT-Thread项目，进入applications文件夹下的main.c，这里便是主函数所在的c文件。main函数在这里面。

接着注册设备和打开设备。事先在每个设备的驱动文件里我已经写好了底层驱动，因此只要直接使用应用层的代码即可。使用的设备为OLED屏幕，FLASH存储芯片，MP3模块、GC24无线模块。



接着需要读FLASH里的信息，FLASH里面是枪的设置和各种信息。

接着开启线程，如下：

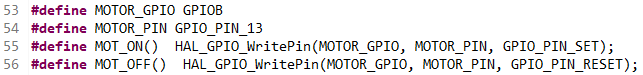


使用到3个子线程，adc\_p是检测压力，adc\_b是检测电池电量，mot\_th是电机停止振动的线程。这里还开启了RT-Thread官方库的ADC设备，这里不使用HAL库来控制ADC设备，HAL库的使用会有些BUG。当开枪后，电机振动产生后坐力，此时会激发mot\_th.tm定时器，定时时间到达后激发mot\_th子线程，在线程中停止电机。

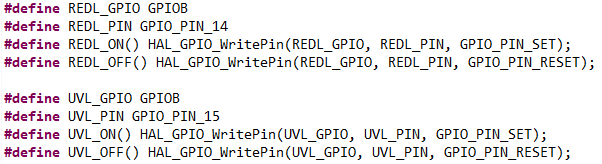
### 3.3 开关量控制

外围设备的运作主要是开关量控制。电机、激光、MP3模块、系统供电。

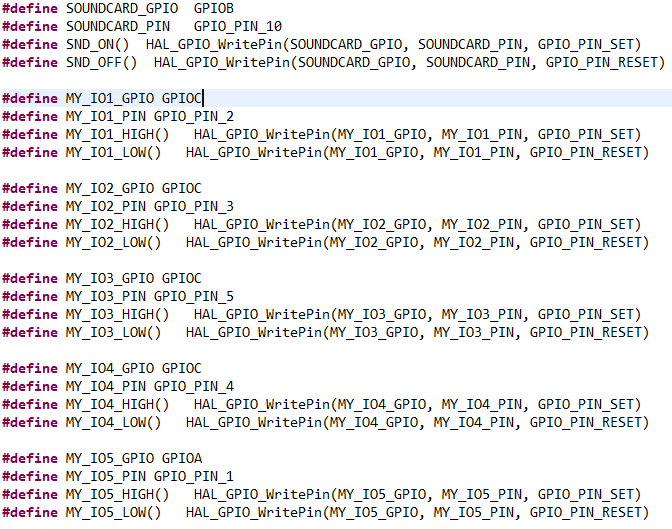
电机的控制引脚是PB13，写1可使电机振动。



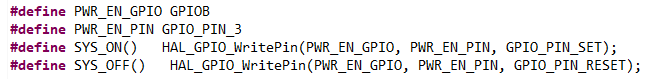
不可见光是PB15，可见红光是PB14。写1为开启。



MP3模块的开启由PB10控制，而IO口控制播放的内容。



系统供电是PB3，写1可开启系统，关机的话写0便会断电。



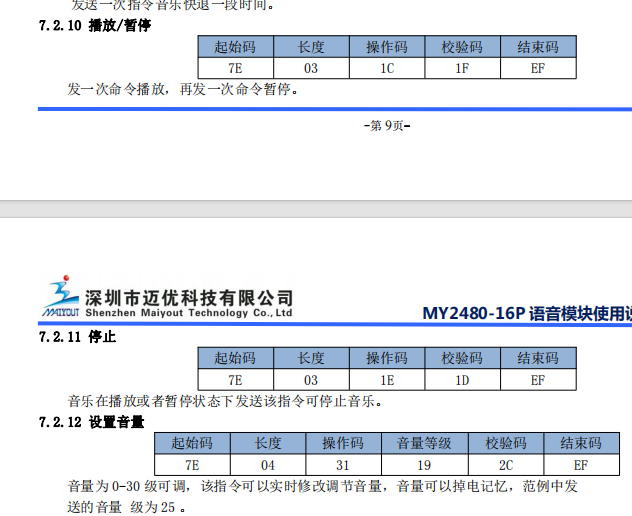
### 3.4 MP3模块

MP3模块名称是MY2480，有两种控制方式，通过IO口控制和通过串口控制。两种方式都硬件实现了可以随意选择使用，在该项目软件选择使用串口控制。

RT-Thread注册和开启了MP3设备后，控制的代码如下：



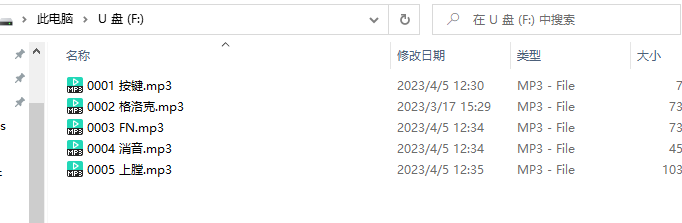
中间but\_snd数组是串口的内容，7是数组字节数。串口数组的内容看MY2480手册，想播放什么内容，或者调节多少音量，只要按照MY2480手册中的介绍来填写这个数组就可以实现了。



MP3的喇叭使用8Ω2W小喇叭，喇叭线采用红黑、外径1.3、长度80mm。

使用线连接MP3模块和电脑，把素材文件夹里“音效”里的枪声文件复制到模块中。





### 3.5 无线模块和信息传递

无线模块采用硅传科技的国产无线透传模块GC2400。控制采用串口控制。

#### 3.5.1 传输原理

无线传输采用了2.4G频率的传输。发送方根据信息内容，往空中发射2.4G频率的电磁波，接收方收到后进行模数转换为数字信号得到信息。

#### 3.5.2 无线模块原理

每个GC2400模块有自己的PID地址。另外通信有信道。

PID类似于IP地址。

信道是通信的具体频率，是由模块自己规定的。在通信中2.4G通信只是个大概的总称，实际通信中2.40G到2.48G都叫2.4G通信，所以具体频率是多少就有很多选择。信道就用作选择，比如该无线模块中信道0是2.402G，信道1是2.403G，等等。该模块共有82信道。一个信道不要联通太多模块，最好是一个信道就两个无线模块，不然太多了会拥堵。

要实现两个模块互相能通信，需要PID地址相同，信道也需要相同。

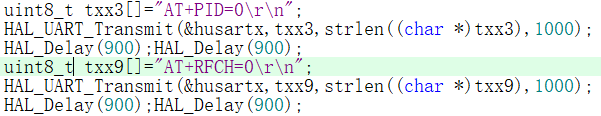
无线模块空中波特率影响信号在空中传播的速度，直接开最大。而发射功率也直接开最大。

无线模块自身实现了对射频芯片的控制，实现了无线透传，即是写入什么信息就发送什么信息，接收方也随时接收，两者互相直接通信。

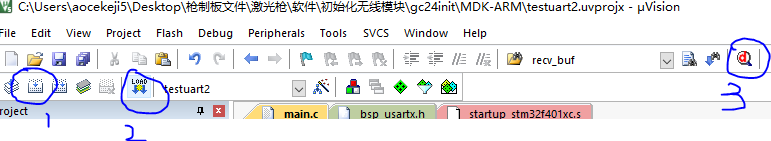
#### 3.5.3 无线模块初始化

在激光枪主板中无线模块采用串口控制。串口波特率为115200，因此需要在“初始化无线模块”文件中，使用keil5裸机的初始化文件先烧录一遍。

具体操作是接好烧录器。修改想要配置的PID和信道。



先1编译，然后2下载烧录。记得2**烧录完成后**，要**立刻马上**按3监控，在Debug中运行。



在监控Debug中，按1运行。看下面Watch1窗口（如果无这个窗口，按上面View-》Watch Window-》Watch 1，然后把代码中aRxBuffer和apple拖入窗口），看aRxBuffer和apple。看到结果是aRxBuffer中是几个OK，apple最后变成1就成功了。

初始化完成后的无线模块，功率为最大，波特率为115200，配置好PID和信道。

#### 3.5.4 无线模块控制

GC2400无线模块通过AT指令进行控制，通过AT指令控制其进入透传模式就可以通信了。

在RT-Thread该枪板项目中，只需要发送不需要接收。以下代码实现发送枪的各项信息：

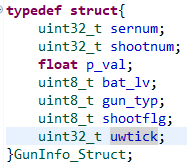


guninfo是枪信息结构体，而第三个参则是枪当前的状态。

### 3.6 各类信息

#### 3.6.1 枪信息

枪信息文件是cubemx/src里的gun\_info.c文件。枪信息结构体如下：



从上而下是：枪编号、射击总次数、压力值、电池电量、枪的类型、枪当前状态和时间戳。

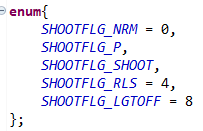
无线模块发送的信息数组是32字节，具体格式看文件“枪与靶子通信”。

枪编号是该枪的编号0xTTYYYYYY，命名规则是0xTT批次，第0xYYYYYY批。

枪的类型为1步枪，2手枪。



枪当前状态有如下内容：



普通状态、按压扳机有压力的状态、开枪击发状态、松开扳机状态。关灯状态该项目已取消不用，但保留。

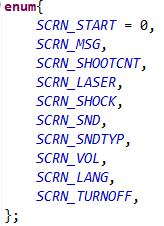
#### 3.6.2 枪配置信息

枪总共可以设置几项，分别为红点可见激光状态、后坐力有无、枪声有无、枪声类型、菜单语言。在\_\_Config结构体中配置：



#### 3.6.3 屏幕信息

屏幕总共有几种状态：

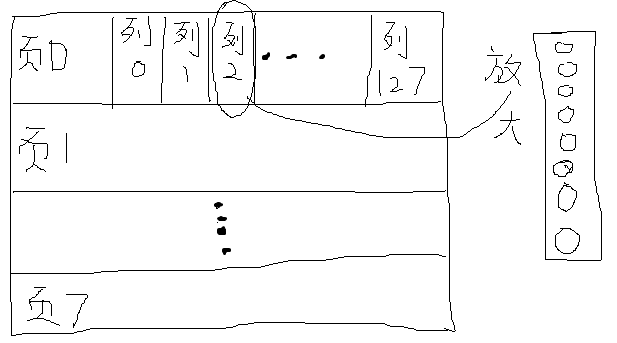


分别为开机时显示公司名的页面、显示编号信息页面、射击总数页面、激光模式选择页面、后坐力选择页面、声音选择页面、声音效果选择页面、音量调节页面、语言选择页面、关机页面。

### 3.7 OLED屏幕

#### 3.7.1 显像原理

枪板采用0.91寸OLED屏幕，STM32的I2C与屏幕的驱动芯片进行通信和控制。I2C发送数据给驱动芯片，驱动芯片根据数据进行各个模式配置。显示图像时，驱动芯片会提供显寸给oled屏幕，STM32通过I2C把地址数据写去驱动芯片，实现图像显示。



整个OLED屏幕分成8页8长条，然后每长条有128小列。这两就为页地址和列地址。先通过I2C写入页地址和列地址。把一小列放大看是8个像素点，想8个点哪个亮就把哪个写为1，不亮的为0，这样可以得出一个uint8\_t的字节，就是往该地址写入的数据内容。

#### 3.7.2 使用代码

使用OLED设备显示图像代码如下：

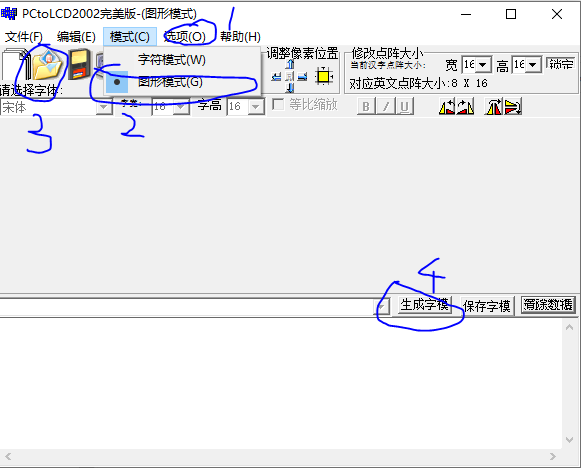


其中pos是显示在屏幕的x,y起点，然后x,y终点。这里是(0,0)到(120,8)。第三个参数为图像码数组。

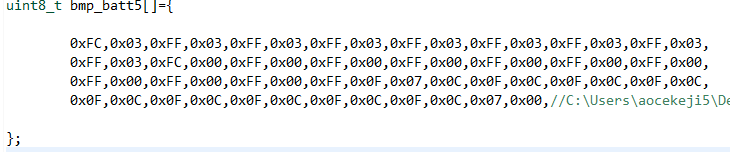
#### 3.7.3 图像码和PS

使用稿定设计网在线PS进行画图。背景宽120高70。注意需要2值图，即是黑白色的。

画好图后，导出为BMP，然后打开PCtoLCD，进行好取字设置后，模式选择图形模式，然后生成字模，即可得到该图片的图像码数组。



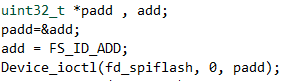
该枪主板RT-Thread项目中，图像码数组都保存在cubemx/src的oled\_code.c文件中。创建一个数组，然后把图像码直接复制进去即可。



### 3.8 存储FLASH芯片

STM32使用SPI控制FLASH芯片。

写FLASH时，需要先清空需要写的地址所在的整块：



第三个参为需要清除的地址的指针。第二个参没用。

写FLASH的代码如下：



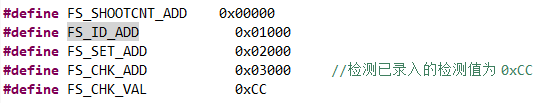
第二个参为写的内容数组，第三个参为地址。

读FLASH代码如下：

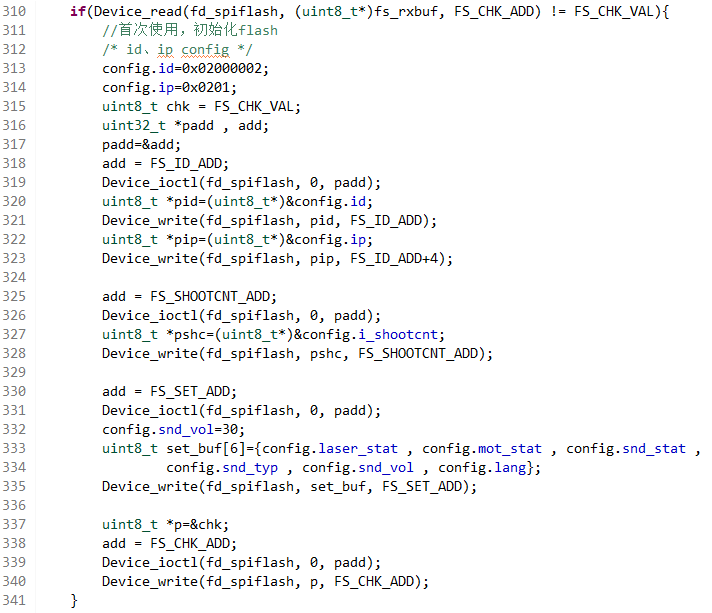


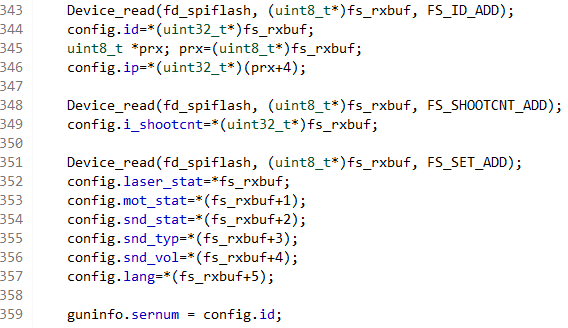
第二个参为读入的数组，第三个参为地址。注意底层驱动设定了最多只能读入10个字节。该函数返回值为该地址读入的第一个字节。

该项目中存储用到的地址如下：

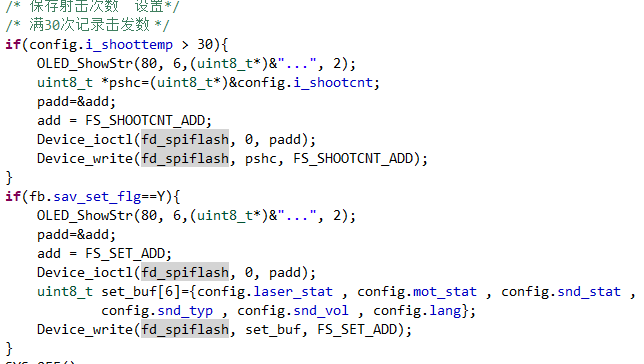


使用到存储读写FLASH的部分有两个：第一是开机时，先检测FS\_CHK\_ADD地址的第一个字节，如果不是0xCC证明这个枪板是第一次使用的，然后往对应的地址写入初始的内容，写入枪编号和IP地址等出厂内容，最后把FS\_CHK\_ADD地址写入0xCC。如果检测FS\_CHK\_ADD地址能读到0xCC，证明不是第一次使用，便只要读各地址的内容。





第二是关机时，若本次使用修改过设置，或者射击次数超过30次，就会写存储芯片进行修改和记录。



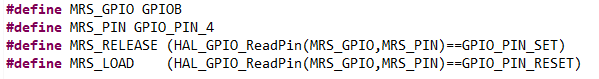
### 3.9 外接传感器

枪板外接两个传感器板，压力传感器板和磁感应板。

#### 3.9.1 磁感应板

磁感应板上有MRS-201或国产TMR1302S磁开关芯片。用于作为开枪的判定。

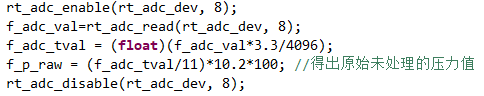
当感应到磁力时，输出低电平；没有感应到时，输出高电平。



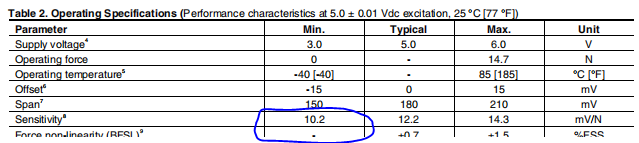
#### 3.9.2 压力传感器板

压力传感器上主要使用FSS1500传感器作为压力检测。FSS1500传感器根据压力输出电压值，电压再经过INA333芯片进行放大以方便获取。使用ADC进行测量得到电压值，再由电压值推算出压力值。

在线程adc\_p中进行压力测量。压力测量需要的ADC设备由RT-Thread库提供。首先需要启用ADC设备，连接传感器的是STM32的ADC通道8。接着获取ADC读取值，获得的是粗值。接着把粗值乘以分辨率（参考电压3.3V除以ADC位数12位即2^12为4096）便可得到电压值。



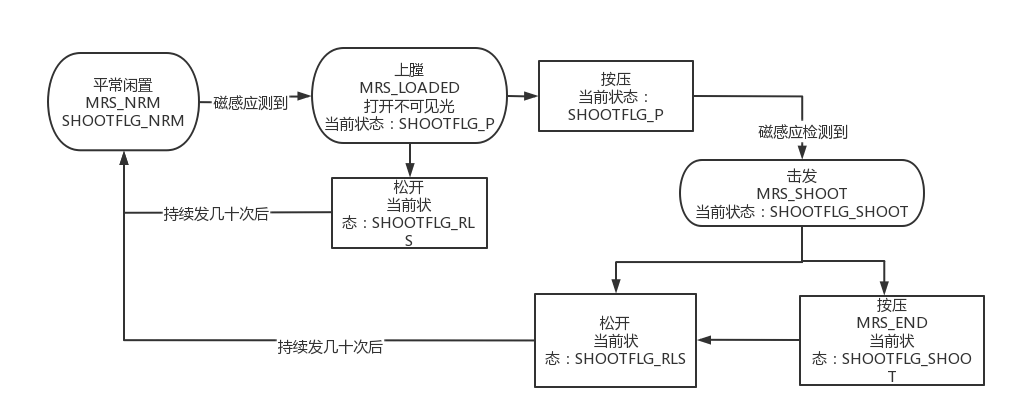
电压值转换为压力值为：首先除以INA333放大器的放大倍数11，得到放大前原本压力传感器输出的电压。然后该电压乘以10.2，这个10.2为FSS1500手册的：



可知每牛为10.2mV，最后乘以100则可计出压力多少克。

### 3.10 射击流程

射击流程由adc\_p线程来实现。由定时器定好时间每隔一段时间执行一次压力读取。



### 3.11 配置参数

#### 3.11.1 配置一些杂项

在main.h头文件可以完成以下配置：



#### 3.11.2 测量数据和压力表测量值

使用ADC读取压力传感器的值并经过计算得到压力值，但实际中拿压力表测扳机得到的力之间有相当的差异，需要匹配一下。

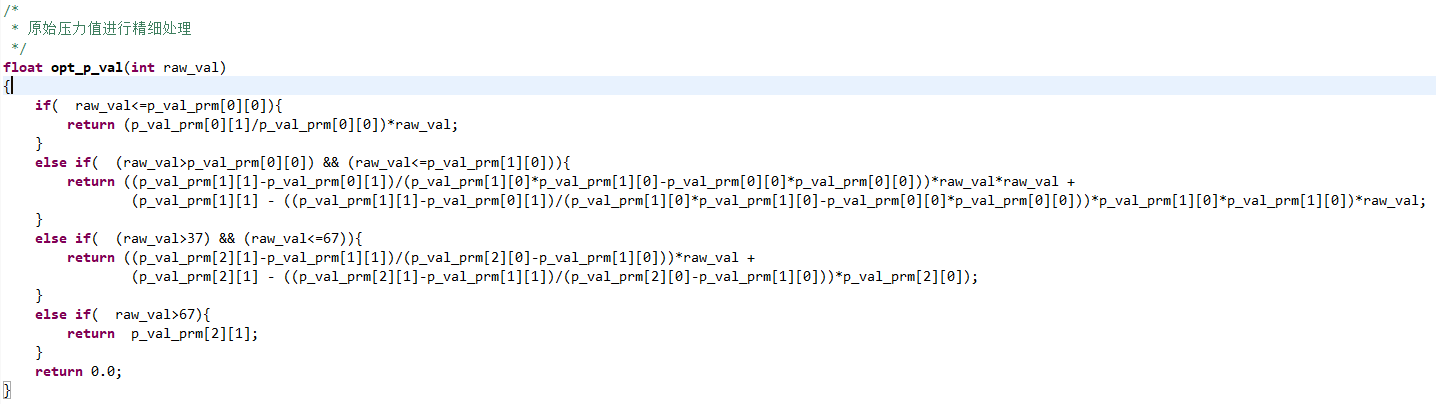
X是软件得到的压力值，Y是实际压力表得到的压力值。整体分成四段，比如实际用压力表测到的300、400、700，这时的软件压力值X记下来：

第一段用Y=AX，(0,0)，(x1,300)来求出A；

第二段用Y=AX²+B，(x1,300)，(x2,400)来求出A、B；

第三段用Y=AX+B，(x2,400)，(x3,700)来求出A、B；

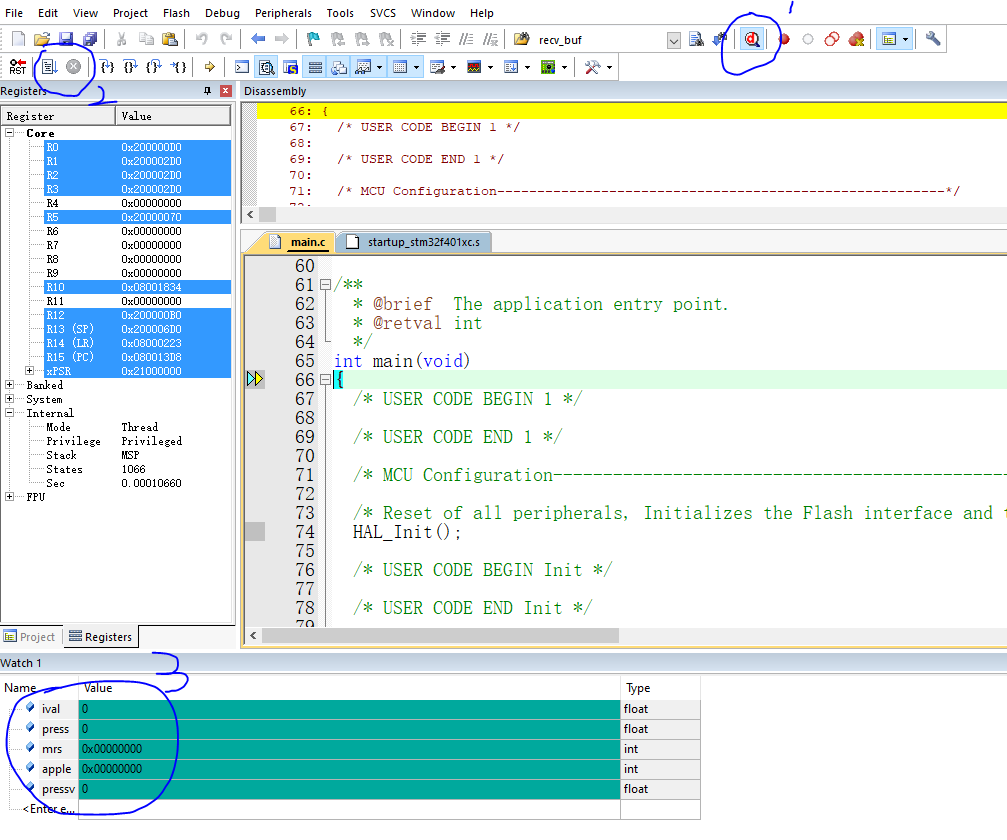
第四段是x3以上的值干脆全都用700表示，因为已经超出范围不需要了。



在main.c文件的p\_val\_prm数组，把这些测量到的xy记录进去。



软件测量X的数值，在“软件测试压力值”文件夹中。打开该keil项目，使用Debug模式。

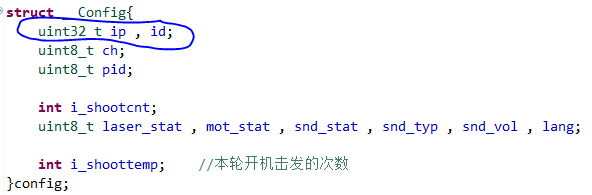


看3，这里的press即为软件压力值，使用压力表在扳机处扣到，比如扣到表显示300g，看这时的press值记录下来便是x，300便是y。另外mrs可以用作磁力传感器的检测，如果磁力检测到会显示0，没有检测到则显示1，测量磁力时可以用到。

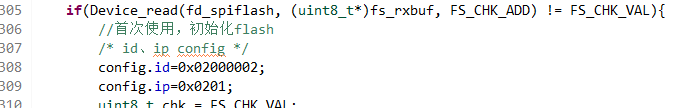
#### 3.11.3 编号和地址

枪的编号是ID。

枪的地址是IP，0xRTYU含义是无线模块信道为0xRT，PID为0xYU。



新板需要在main.c里，以下代码修改ID、IP。



枪的编号和地址是固定的，地址和枪卡一一对应。枪卡采用25mm大小，写入和枪对应的IP地址，详情见读卡器那边的内容。

## 4. 使用流程

拿到一个新做好的电路板。

先把线和各个分离的部件接好。



左边白色头，从上到下是不可见光、可见红光、电机，右边，四根针是烧录口，右边白色头从上到下是压力板、电池、磁感应板。

第一步是进行3.5.3的无线模块初始化。

第二步是进行3.11.2的压力测量匹配。

第三步是进行3.4的MP3模块枪声音效文件复制入。

第三步打开RT-Thread Studio的项目，修改好3.11.2的p\_val\_prm数组，然后修改3.11.3的编号和地址。

最后接好烧录器和枪板点击编译，下载烧录到枪板上。

