

智能仓储系统的开发研究

先进计算与机器人研究所

2023 年 7 月 4 日

目录

1	第六章: 超高频 UHF_R505	2
1.1	RC505 类的声明	2
1.2	RC505.ino	4

1 第六章: 超高频 UHF_R505

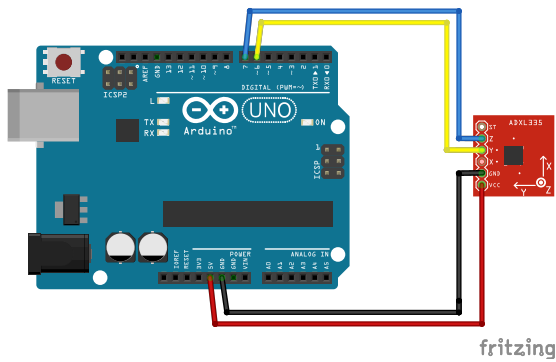


图 1: R505 与 arduino 的实物连接图

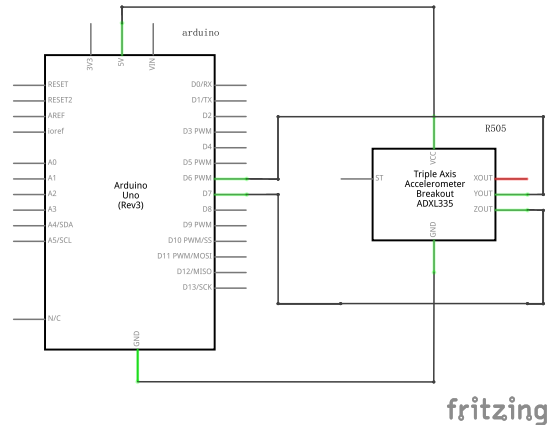


图 2: R505 与 arduino 的电路连接原理图

在程序中, 本文使用了 <SoftwareSerial.h> 库, 创建了软串口对象 WIFISerial(6, 7), 其中定义引脚 6 为 RX, 引脚 7 为 TX(除了引脚 0 和引脚 1 以外的任意空闲数字引脚皆可, 其中引脚 0 和引脚 1 为 Arduino 默认的串口通讯接口, 如果占用了可能会造成通讯堵塞)

- 引脚 6 为 RX 与 R505 的 TX 连接;
- 引脚 7 为 TX 与 R505 的 RX 连接;
- 引脚 5V 与 R505 的 VCC 连接;
- 引脚 GND 与 R505 的 GND 连接;
- R505 的使能引脚 EN 默认为高电平, 不需要连接。

UFID 标签卡的存储空间分为四个区: RESERVED 区, EPC 区, TID 区, USER 区。其中 EPC 区作为识别标签对象的电子产品码, 用户可以手动输入修改。由于我们使用的是“读多卡”的指令进行读, 所以返回的消息中第二位是‘U’, 与“读多卡”指令相对应。除了消息开头结尾固定的字符‘0x0A’和‘0x0D’, 还有刚刚提到的‘U’, 以外卡内还有 32 个字节, 其中前 4 个字节为商家编号例如“3000”, 后四个字节为卡的 ID 编号, 为每张卡所特有, 因此卡片的前四个字节和后四个字节最好不要对其进行修改。中间 24 个字节可以随意存储, 本文这里将这 24 个字节中前 4 个字节用于存储货物类型, 接着后面 4 个字节存储箱内单个物品的质量, 接着后面 4 个字节存储箱子外壳的质量。

当 Arduino 对 R505 进行控制, 需要通过创建的软串口“WIFISerial”使用“write”函数给 R505 发送不同指令, 待 R505 完成操作后可以通过软串口的“read”函数一个字节一个字节的读取返回的信息。

1.1 RC505 类的声明

```
1 class RC505 {
2 private:
3     SoftwareSerial WIFISerial = SoftwareSerial(6, 7);
4 public:
5     char Type_Name[3];
6     long Single_Weight;
7     long Shell_Weight;
8
9     RC505();
10    bool read(char *sub);
11    void write();
12 };
```

默认构造函数: 其中软串口波特率 38400 是为了和 RC505 保持一致。

```

1 RC505::RC505(){
2     WiFiSerial.begin(38400);
3 };

```

在介绍读写函数之前, 首先介绍一下读写指令。不同的指令对应着不同地址的读写。

```

1 const unsigned char MultiEPC[] = {0x0A, 0x55, 0x2C, 0x52, 0x31, 0x2C, 0x31, 0x2C, 0
    x37, 0x0D}; //同时读取多张卡的指令
2 const unsigned char WriteEPC[] = {0x0A, 0x57, 0x31, 0x2C, //第一个0x31表示EPC区
3     0x32, 0x2C, //从第2个位置开始
4     0x33, 0x2C, //这里是写的三个字节, 最多八个地址, 每个地址4字
        节
5     0x30, 0x30, 0x3A, 0x3A, //物品种类
6     0x30, 0x30, 0x31, 0x30, //单个物品质量
7     0x30, 0x30, 0x30, 0x30, 0x0D}; //外壳质量

```

getMessage 返回的是完整的消息, 将读取到的消息更新在传入的字符数组 sub 中。

- 第一步: 向 RC505 写入读的指令后, 判断当前读到的字符是不是 LF, 来决定是否开始接收接下来的字符;
- 第二步: 接收字符, 直到接收长度达 33 为止;
- 第三步: 更新 RC505 类中的 Type_Name 和 Single_Weight。

```

1 bool RC505::read(char *sub){
2     WiFiSerial.write(MultiEPC, sizeof(MultiEPC));
3
4     int start1 = 0;
5     unsigned char buffer = 0;
6     int i=0;
7
8     while(WiFiSerial.available() > 0) {
9         buffer = (char)WiFiSerial.read(); //获取串口接收到的数据
10
11         if(start1 == 0 && buffer == LF){ //当读取到第一个字节为LF
12             start1 = 1;
13             continue;
14         }
15
16         if(start1 == 1 && buffer != CR){ //结尾标志CR和LF
17             // Serial.print((char)buffer);
18             sub[i]=(char)buffer;
19             i++;
20             if(i==arrayMax) {
21                 // Serial.println(' ');
22                 break;
23             }
24             continue;
25         }
26     }
27
28
29     int w=0;
30     for (int i=7;i<9;i++) Type_Name[i-7]=sub[i];
31     for (int i=11;i<13;i++) {
32         w += (sub[i]-48)*pow(10,12-i);
33         // Serial.println(w);
34     }
35     Single_Weight=w;
36     return 1;
37 };

```

write 通过向 RC505 写入指令, 来间接写卡:

```
1 void RC505::write(){
2     WIFISerial.write(WriteEPC, sizeof(WriteEPC));
3 };
```

1.2 RC505.ino

与上一章 RC522.ino 相比, 就是把 RC522 的内容换成了 RC505 的内容, 然后更新 RC505 中 Type_Name 和 Single_Weight。由于本身 RC505 中读写的函数几乎没有判断读写状态的条件语句, 所以反而这里主程序能实现重量发生变化时, 直接发送消息, 不必反复进出刷卡。

```
1 #include "master.h"
2 #include "transform.h"
3 #include "Surface.h"
4 #include "Calibrate.h"
5 #include "Oled.h"
6 #include "RC522.h"
7 #include "RC505.h"
8
9 master m1;
10 Surface YL_Surface;
11 Calibrate YL_Calibrated;
12 transform tf;
13 Oled oled;
14 RC522 rc522;
15 RC505 rc505;
16
17 long numbefore=0, numnow=1;
18 char te[3];
19 unsigned long Sweight;
20 char MessageNow[arrayMax];
21 bool flag=0; //1--->write; 0--->read
22
23 void setup() {
24     //设置串口波特率38400
25     Serial.begin(38400);
26     m1.initialize(8); //8needschanged
27     YL_Calibrated.setpin_SCKDT(4, 5);
28     YL_Calibrated.set_range(20);
29     YL_Calibrated.kb_Initialize();
30     oled.initialize();
31     rc522.initialize(9,10);
32 }
33
34
35 void loop() {
36     switch(flag){
37         case 0:
38             // while(1) {
39             //     bool state = rc505.write();
40             //     if(state==1)
41             //         break;
42             // }
43             rflag=1;
44             // break;
45         case 1:
46             while(1) {
47                 bool state = rc505.read(MessageNow);
48                 if(state==1)
49                     break;
```

```

50     }
51     break;
52 }
53 for (int i=0; i<sizeof(rc505.Type_Name);i++) te[i]=rc505.Type_Name[i];
54 Sweight=rc505.Single_Weight;
55
56 numbefore = numnow;
57 unsigned long CalibratedWeight = YL_Calibrated.Output_CalibratedWeight(YL_Surface.
    Get_Surface());
58 numnow = ceil(CalibratedWeight/Sweight);
59
60 bool flag= (numbefore==numnow?0:1);
61 if(flag){
62     tf.initialize(te, m1.address, numnow, CalibratedWeight);
63     tf.pack();
64     digitalWrite(3,HIGH);
65     m1.send(9, tf.Transmission_Information);
66     Serial.println(tf.Transmission_Information);
67     oled.showIIC(te, numnow);
68     digitalWrite(3,LOW);
69 }
70
71 delay(3000);
72 }

```