

# 基于机器视觉的麦稻播种漏播检测及智能补种装置

郭宋 张妮 仝曦熳

## 第一部分 设计概述

### 1.1 设计目的

由于工厂化播种所使用的普通机械式播种机生产工艺简单、价格低廉，漏播和重播现象较为严重，且在播种完成后，往往需要大量的人力进行间苗和调整柱距。其他的一些精密的播种机，生产工艺复杂、价格昂贵且无反馈部分。本项目将在已有的文献和研究的基础上，并结合实时国情研发一套基于机器视觉及 STM32 的种子检测装置及自动补种装置。

### 1.2 应用领域

该项目能广泛应用于智能农业播种、补种方面，如工厂化育秧作业的水稻、蔬菜、花卉等。播种程序后添加漏播检测系统，在减少空穴和少粒穴情况、提高秧苗成秧率的同时，通过自动化的补种系统将工作人员从繁琐的手动操作中解放出来，还能避免大量空穴带来的环境资源浪费。既保证了秧盘育秧成秧率，又提高了精密播种合格率。

### 1.3 主要技术特点

装置将采用海康威视 CCD 摄像头结合照明装置进行实时通信上传和处理，实现智能化检测和可视化，补种装置采用与上位机相连接的舵机臂，并使用模块化的编程思想对单片机程序进行编写。设计上位机和下位机兼容的数据采集系统，选择合适的深度学习算法去噪、识别、分割并进行图像目标检测，将其相关坐标参数经过串口通信实时反馈至 STM32 控制单元来控制补种装置在流水线上进行快速补种，此外还能通过激光测距装置测量土壤厚度，以便后期补土工作。

### 1.4 关键性能指标

该装置的主要性能指标有三个，一是补种成功率，二是图像识别的准确率，三是对土壤厚度测量的精确度。在不断调整传送带的速度后，测量出补种装置开始作用时秧盘离补种装置的距离，补种成功率在 95%以上。而图像识别准确率会受到外在光线的影响，结果有一定误差，但在可接受范围内。激光测距测量土壤厚度结果较准确，误差在数毫米以内。

### 1.5 主要创新点

(1) 基于机器学习、深度学习的图像处理算法。

(2) 配备漏播检测算法和服务器的低成本、小体积、低功耗的漏播检测装置，并具备实时自动补种功能。

(3) 拟使用掩膜定位技术提取秧盘中每个穴位的种子图像并进行连通域检测，来获得单个连通域的参数，并计算出形状因子，再结合单个连通域的面积和深度学习算法（拟采用 FSSD、BP 神经网络, YOLO 等方法）来计算出穴粒的数量。

(4) 拟使用区块化设计，做到可以在随时将硬件上的设备、软件上的算法进

行实时的更替，以保证使用时间长久的同时，检测效果依旧显著。

## 第二部分 系统组成及功能说明

### 2.1 整体介绍

基于机器视觉的麦稻播种漏播检测及智能补种装置主要包括实现图像采集和补种的硬件部分和实现图像数据处理的软件部分。整个系统主要由摄像机、计算机、补种装置、传送带、暗箱等部分组成。传送带先经过摄像头的下方，育秧盘置于传送带和补种装置之间，摄像头的输出与 PC 机连接，在 PC 机中进行图像处理，判断每个空穴内的播种状况，将需要补种的空穴坐标通过串口通信传送到核心控制板，STM32 核心板驱动舵机转动实现补种功能。

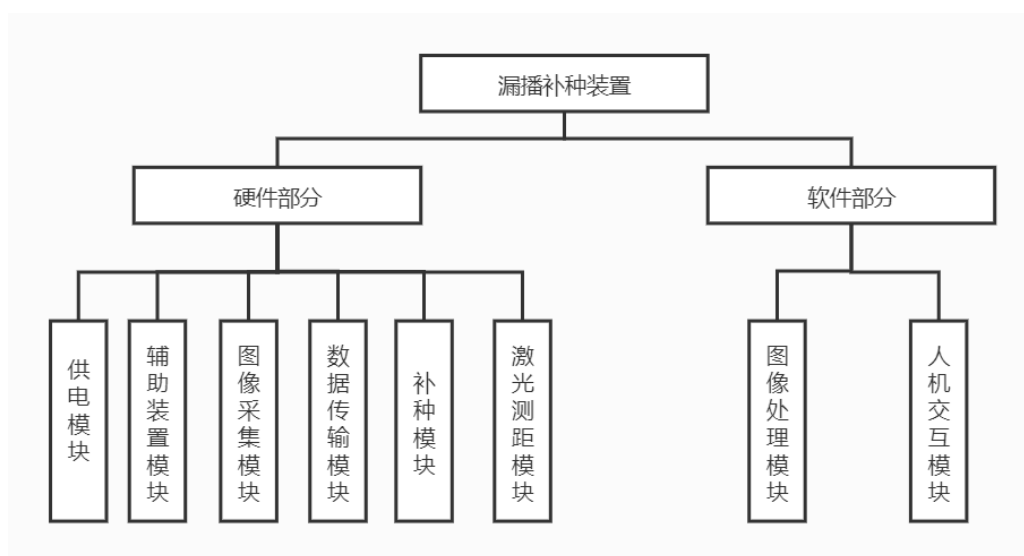


图 1 系统整体框图

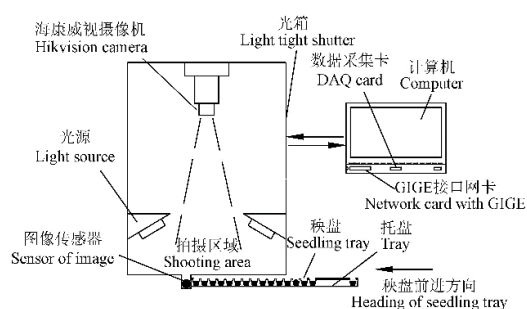


图 2 漏播检测及自动补种的概念图

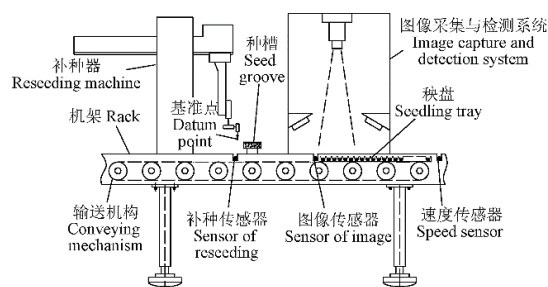


图 3 暗箱的设计概念图

### 2.2 各模块介绍

#### 2.2.1 硬件部分

##### (1) 供电模块

电源采用的是 2200mAh 的航模电池，为整个硬件系统供电，电池可充电，方便更换，提高了产品的可循环利用率。

##### (2) 辅助装置模块

由暗箱、照明灯和传送带组成。在一个图像采集系统中，照明光源的好坏直接影响着图像的成像质量、边缘提取及拟合的效果。其中最关键的要素就是光照的强度。曝光量过大或过小都会使画面产生失真而最终产生测量误差。为了减少外来光线的影响，我们使用四面内壁为黑色的暗箱，暗箱上方有一个小孔用于放入海康威视摄像头，通过调整照明灯的亮暗和摄像头的曝光频率以获得合适的亮度，使采集到的图像更利于后续的去噪、分割等处理。

### （3）图像采集模块

采用海康威视 CCD 摄像头，该摄像头有 600 万像素，帧率达 42.7fps，可以采集到清晰度高、噪点低、适于后续处理分析的种子图像。

### （4）数据传输模块

采用 HC05 蓝牙模块接收测出的土壤厚度，并且将数据传送到人机交互模块的用户界面实时显示，帮助实现数据可视化。

### （5）实时补种模块

采用 STM32H743 为核心控制板，其内核基于 32 位 Cortex-M7 内核，具有极强的处理性能。存储最高 2MB 的 FLASH 存储空间，最高 1MB 的 SRAM 存储空间。外设及 GPIO 多达 168 个 GPIO 口，其中的快速 GPIO 接口可以运行于 133MHz 频率之下。其内部的 22 个定时器，除了传统的定时器之外，高分辨率定时器及低功耗定时器使得其能够广泛应用于各种电子系统中。利用 STM32H743 驱动舵机转动完成实时补种。

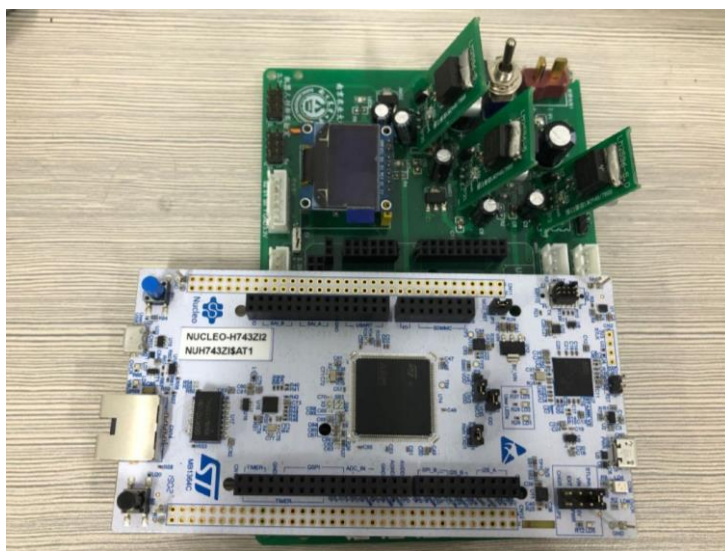


图 4 核心板 STM32H743

### （6）激光测距模块

通过 KLH-01T-20hz 激光测距模块，测量声波在放射后遇到障碍物反射回来的时间，并根据发射和接收的时间差进行计算土壤的厚度，精确度为毫米级别，且光束在空气中的传播速度最快，测得的值较为精准。

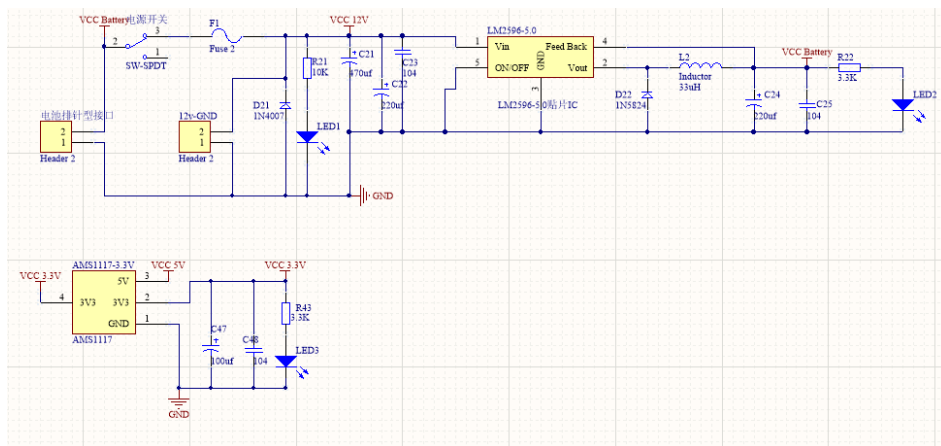


图 5 硬件原理图

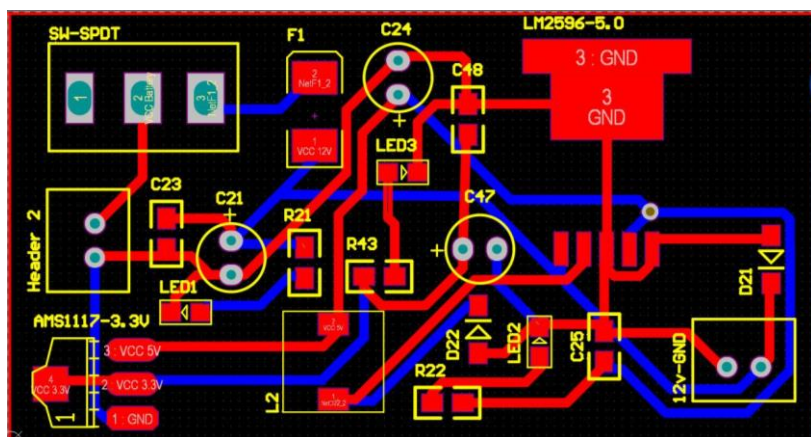


图 6 PCB 设计图

## 2.2.2 软件部分

### (1) 图像处理模块

首先在 PC 端进行图像的读取，并对其进行预处理。主要涉及到提取目标区域、提取种子目标、图像滤波、锐化增强、二值化等过程。目的是为了当处理结束后，系统可以更好的进行图像秧穴的检测，并减少一些外界因素所造成的影响（如：曝光等因素）。然后检测图像各个秧穴的种子图像，并提取其中的连通域形状参数，对各个穴位的穴粒数进行分析，以达到较高的准确率。一直到判断出是最后一个秧穴，即可进行下一张图像的导入。

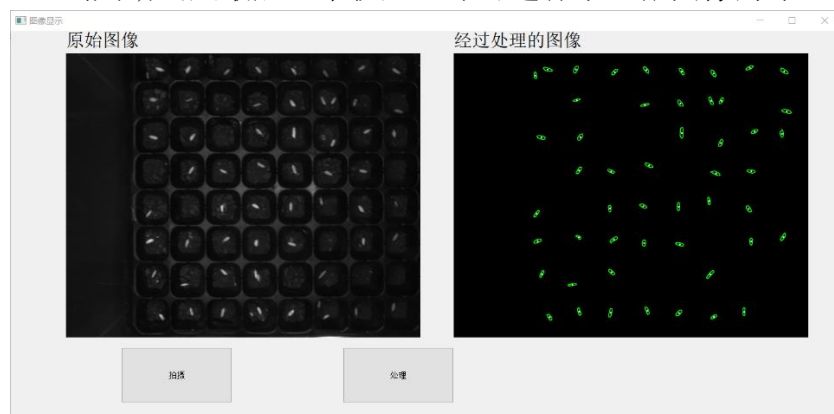


图 7 在 PC 端进行图像处理

## （2）人机交互模块

用 PyQt 设计用户端界面，激光测距模块测量到土壤厚度之后将数据传送给手机端，将检测结果实时显示。在软件部分将这些数据处理可视化，提供给用户一个良好的使用环境。

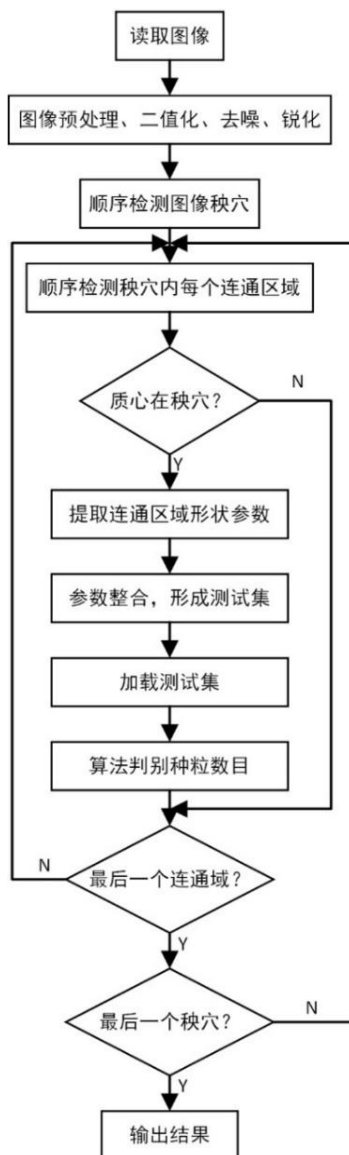


图 8 图像处理流程图

## 第三部分 完成情况及性能参数

传送带先经过摄像头的下方，摄像头采集秧盘内空穴的播种图像，并将图像上传到 PC 端进行图像处理，判断每个空穴内的播种状况，将需要补种的空穴坐标通过串口通信传送到核心控制板，STM32 核心板驱动舵机转动实现补种功能。经过反复试验，补种准确率能保证在 95%以上。





图 9 补种装置实物图

激光测距测量土壤厚度，通过 HC05 蓝牙模块接收测量结果，并且将数据传送到手机端实时显示，帮助实现数据可视化。且测量结果精确度为毫米级别，测得值较为精准。



图 10 在手机端显示激光测距结果

## 第四部分 总结

### 4.1 可扩展之处

(1) 该装置目前只能对秧盘进行一次性补种，一次补种完成后需要手动为补种装置补充种子。为了更好的实用性以投入农业生产中使用，可在补种装置上加一个自动排种装置，实现连续动态补种。

(2) 可单片机上通过串口连接 ESP8266WiFi 模块创建连接服务, 打开手机 APP 通过 WiFi 与其连接, 就可以通过手机 APP 实现对当前状态的查看, 并可以向其发送控制指令, 实现实时监控漏播补播完成情况并进行远程操控。

## 4.2 心得体会

在整个比赛中, 我们小组三人从选题、功能设计、软件和硬件设计到最终搭建实物实现预期功能, 整个过程中遇到了许多困难和意外, 但通过仔细的分析并进行多方面的调整后, 最后还是解决了问题。通过这次比赛, 加深了我们对本专业知识的理解和巩固, 进一步提高了实践能力和独立思考的能力。我们也从中体会了共同协作和团队精神的重要性, 我们组三个人分工协作, 每个人都对自己的任务一丝不苟的去完成, 遇到问题时大家一起解决, 这也是装置成功的重要原因之一。这次比赛让我们学会了更多知识, 拓宽了视野, 令我获益匪浅, 帮助我有效地提高了自身的综合能力, 对日后学习专业核心课程以及日后的读研和就业都有一定的启蒙作用。

## 第五部分 参考文献

- [1]安爱琴,王玉顺,王宏强,聂永芳.基于机器视觉的精播排种器性能检测方法[J].农机化研究,2007(07):48-50.
- [2]王桂莲,刘伟超,王安,柏凯凯,周海波.基于机器视觉的水稻秧盘育秧智能补种装置设计与试验[J].农业工程学报,2018,34(13):35-42.
- [3]谭穗妍,马旭,董文浩,鹿芳媛,李倍旭.基于嵌入式机器视觉的水稻秧盘育秧图像无线传输系统[J].农业机械学报,2017,48(04):22-28.
- [4]齐龙,马旭,周海波.基于机器视觉的超级稻秧盘育秧播种空穴检测技术[J].农业工程学报,2009,25(02):121-125.
- [5]马旭,王剑平,胡少兴,张君.用图像处理技术检测精密排种器性能[J].农业机械学报,2001(04):34-37.
- [6]王欢,姜有忠,曾海峰,王有伟.气吸式播种机播种监测系统设计与试验——基于光敏原件[J].农机化研究,2021,43(02):52-57.
- [7]王臻,李建军,康辉,李秉坤,潘宠,史昊杰,郑思严.基于 Arduino 的电驱精密播种控制系统[J].中国科技信息,2019(18):82-83.
- [8]杨鹏,王志强,王瑞强.一种智能化多功能播种机器人的设计[J].电子世界,2018(24):171-172.
- [9]聂秀展,于艳,江景涛,马云鹏,范玉琪.花生精量播种漏播检测及补偿系统设计与试验[J].青岛农业大学学报(自然科学版),2020,37(02):140-145.
- [10]张冰,袁忠文,谯睿,张天顺,张铭,徐伟城,杨文彩.无盲区三七精密播种机漏播检测装置设计与试验[J].农机化研究,2018,40(12):111-115+120.

## 第六部分 附录

### 6.1 部分重要公式:

#### (1) 双边滤波算法

双边滤波器 (Bilateral filter) 是一种可以保边去噪的滤波器。可以滤除图像数据中的噪声, 且还会保留住图像的边缘、纹理等 (因噪声是高频信号, 边缘、纹理也是高频信息, 高斯滤波会在滤除噪声的同时使得边缘模糊)。

双边滤波器中，输出像素的值依赖于邻域像素的值的加权组合：

$$g(i, j) = \frac{\sum_{k,l} f(k, l) w(i, j, k, l)}{\sum_{k,l} w(i, j, k, l)}$$

权重系数  $w(i, j, k, l)$  取决于定义域核和值域核的乘积：

$$d(i, j, k, l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2}\right)$$

$$r(i, j, k, l) = \exp\left(-\frac{\|f(i, j) - f(k, l)\|}{2\sigma_r^2}\right)$$

$$w(i, j, k, l) = \exp\left(-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2} - \frac{\|f(i, j) - f(k, l)\|}{2\sigma_r^2}\right)$$

同时考虑了空间域与值域的差别。

原理：在平坦区域，像素差值较小，对应值域权重  $r$  接近于 1，此时空域权重  $d$  起主要作用，相当于直接对此区域进行高斯模糊，在边缘区域，像素差值较大，值域系数下降，导致此处核函数下降（因  $w=r*d$ ），当前像素受到的影响就越小，从而保持了边缘的细节信息。

思想：抑制与中心像素值差异较大的像素（即使你们空域相距较近）。

计算方法：对每一个邻域像素点，计算出其对应的空域系数和值域系数，相乘得到总的系数，然后进行加权求和。

## （2）形态学检测算法

数学形态学(Mathematical Morphology)，是基于积分几何和几何概率理论建立的关于图像形状和尺寸的研究方法，其实质是一种非线性滤波方法。数学形态学的方法应用于视觉图象的处理就是用具有一定形状、大小的结构元素去探测、度量和提取图像中的对应形状，以达到对图像进行处理、分析、识别的目的。数学形态学处理能简化图像数据，在保持图像基本形状特性的同时去除不相干的结构。

所使用的公式为：

$$G_1 = [(f \ominus b_1) \oplus b_2] \oplus b_3 - [(f \ominus b_1) \oplus b_2] \times b_3$$

$$G_2 = [(f \oplus b_1) \ominus b_2] \circ b_3 - [(f \oplus b_1) \ominus b_2] \ominus b_3$$

$$G_3 = [(f \ominus b_1) \oplus b_2] \oplus b_3 - [(f \oplus b_1) \ominus b_2] \ominus b_3$$

$$G = G_3 + \theta \bullet \Delta G$$

## 6.2 部分重要代码：

### （1）串口连接控制舵机



```

void My_USART1_Init(void)
{
    GPIO_InitTypeDef GPIO_InitStructure;
    USART_InitTypeDef USART_InitStructure;
    NVIC_InitTypeDef NVIC_InitStructure;

    RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA,ENABLE);//①
    RCC_APB2PeriphClockCmd(RCC_APB2Periph_USART1,ENABLE);

    GPIO_InitStructure.GPIO_Mode=GPIO_Mode_AF_PP;
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_9;
    GPIO_InitStructure.GPIO_Speed=GPIO_Speed_10MHz;
    GPIO_Init(GPIOA,&GPIO_InitStructure);//②

    GPIO_InitStructure.GPIO_Mode=GPIO_Mode_IN_FLOATING;
    GPIO_InitStructure.GPIO_Pin=GPIO_Pin_10;
    GPIO_InitStructure.GPIO_Speed=GPIO_Speed_10MHz;
    GPIO_Init(GPIOA,&GPIO_InitStructure);//②

    USART_InitStructure.USART_BaudRate=115200;
    USART_InitStructure.USART_HardwareFlowControl=USART_HardwareFlowControl_None;
    USART_InitStructure.USART_Mode=USART_Mode_Tx|USART_Mode_Rx;
    USART_InitStructure.USART_Parity=USART_Parity_No;
    USART_InitStructure.USART_StopBits=USART_StopBits_1;
    USART_InitStructure.USART_WordLength=USART_WordLength_8b;

    USART_Init(USART1,&USART_InitStructure);//③

    USART_Cmd(USART1,ENABLE);//使能串口 1

    USART_ITConfig(USART1,USART_IT_RXNE,ENABLE);//开启接收中断

    NVIC_InitStructure.NVIC_IRQChannel=USART1_IRQn;
    NVIC_InitStructure.NVIC_IRQChannelCmd=ENABLE;
    NVIC_InitStructure.NVIC_IRQChannelPreemptionPriority=1;
    NVIC_InitStructure.NVIC_IRQChannelSubPriority=1;
    NVIC_Init(&NVIC_InitStructure);

    LED_Init();//LED 初始化
    OLED_Init();//OLED 初始化
    TIM2_PWM_Init();//右车轮电机初始化
    TIM3_PWM_Init();//左车轮电机初始化
    Xunxian_Init();//循迹模块初始化
    Read();//信号读取

```

```
// EXTI_Init();//外部中断初始化

// uart_init(9600); //串口初始化为 9600

// NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);// 设置中断优先级分组 2

}
u8 *p ;
u16 i=0,j=0,t=0;
u16 x=0,y=0;
u8 headpwmval=25;
u8 led0pwmval=14;

void USART1_IRQHandler(void)
{
u8 res;
if(USART_GetITStatus(USART1,USART_IT_RXNE))
{
res= USART_ReceiveData(USART1);
USART_SendData(USART1,res);
//////////字符串转化为数值

for(i=0;i<8;i++)
{
if(p[i]=='.';) //判断小数点存在的位置
{
t=1; //小数点存在标志
j=i; //记下小数点存在的位置
}
else
{
if(t) //如果在小数点后面
{
// temp=(double)(p[i]-'0')/pow(10,(i-j))+temp; //改为小数
y=(double)(p[i]-'0')+y*10;
}
else //在小数点前面
{
x=(double)(p[i]-'0')+x*10;
}
}
}
}
// return temp;
```

```
TIM1_PWM_Init(199,7199);
//////////舵机转动

TIM_SetCompare1(TIM1,headpwmval);
TIM_SetCompare2(TIM1,led0pwmval);

TIM_SetCompare1(TIM1,headpwmval);
TIM_SetCompare2(TIM1,led0pwmval);

delay_ms(1000);

headpwmval=x;
led0pwmval=y;
TIM_SetCompare1(TIM1,headpwmval);
TIM_SetCompare2(TIM1,led0pwmval);

delay_ms(1000);
headpwmval=25;
led0pwmval=15;
TIM_SetCompare1(TIM1,headpwmval);
TIM_SetCompare2(TIM1,led0pwmval);

delay_ms(1000);
headpwmval=15;
led0pwmval=25;
TIM_SetCompare1(TIM1,headpwmval);
TIM_SetCompare2(TIM1,led0pwmval);

delay_ms(1000);
headpwmval=25;
led0pwmval=14;
TIM_SetCompare1(TIM1,headpwmval);
TIM_SetCompare2(TIM1,led0pwmval);

}

}

int main(void)
{
    NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);
    My_USART1_Init();
```

```
while(1);
```

```
}
```

(2) 激光测距及蓝牙显示模块

```
USART3_RX_STA=0;
```

```
while(1)
```

```
{
```

```
key=KEY_Scan(0);
```

```
/*if(key==KEY1_PRES) //切换模块主从设置
```

```
{
```

```
key=HC05_Get_Role();
```

```
if(key!=0XFF)
```

```
{
```

```
key=!key; //状态取反
```

```
if(key==0)HC05_Set_Cmd("AT+ROLE=0");
```

```
else HC05_Set_Cmd("AT+ROLE=1");
```

```
HC05_Role_Show();
```

```
HC05_Set_Cmd("AT+RESET"); //复位 ATK-HC05 模块
```

```
delay_ms(200);}
```

```
if(key==KEY0_PRES)
```

```
{
```

```
sendmask=!sendmask; //发送/停止发送
```

```
if(sendmask==0)LCD_Fill(30+40,160,240,160+16,WHITE); //清除显示
```

```
}
```

```
else delay_ms(10);
```

```
if(t==50)
```

```
{
```

```
if(sendmask) //定时发送
```

```
{
```

```
u3_printf("jl=%dmm \r\n",value);
```

```
sprintf((char*)sendbuf," HC05 正在发送 %d\r\n",sendcnt);
```

```
}
```

```
// HC05_Sta_Show();
```

```
t=0;
```

```
}
```

```
if(USART3_RX_STA&0X8000) //接收到一次数据了
```

```
{
```

```
LCD_Fill(30,200,240,320,WHITE); //清除显示
```

```
reclen=USART3_RX_STA&0X7FFF; //得到数据长度
```

```
USART3_RX_BUF[reclen]=0; //加入结束符
```

```
LCD_ShowString(30,200,209,119,16,USART3_RX_BUF); //显示接收到的数据
```

```
    USART3_RX_STA=0;  
}  
t++;  
}while(1)  
}
```