

# 工程机器人（创意）设计及制作（四）

## 关键元器件与程序编写

### 1、关键元器件

#### （1）红外遥控模块

红外遥控模块由红外发射模块和红外接收模块组成；按下遥控器的某一个键，遥控器会发出一连串经过调制后的信号，这个信号经过红外一体化模块接收后，输出解调后的数字脉冲，每个按键对应不同的脉冲，故识别出不同的脉冲就能识别出不同的按键。

优点：具有抗干扰能力强，信息传输可靠，功耗低，成本低，易实现等显著优点。



图 1 红外接收模块

| 名称    | 编码      | 名称    | 编码      | 名称   | 编码      |
|-------|---------|-------|---------|------|---------|
| A     | b'E'    | B     | b'F'    | C    | b'G'    |
| D     | b'D'    | E     | b'C'    | F    | b'r'    |
| UP    | b'@'    | DOWN  | b'\x19' | Left | b'\x07' |
| Right | b't'    | Enter | b'\x15' | 0    | b'\x16' |
| 1     | b'\x0c' | 2     | b'\x18' | 3    | b'^'    |
| 4     | b'\x08' | 5     | b'\x1c' | 6    | b'Z'    |
| 7     | b'B'    | 8     | b'R'    | 9    | b'J'    |

图 2 按钮编码

(2) 直流电机 只能控制正转反转，模糊的控制转速。



图 3 直流电机

(3) 舵机

舵机是指在自动驾驶仪中操纵飞机舵面（操纵面）转动的一种执行部件。分有：①电动舵机，由电动机、传动部件和离合器组成。接受自动驾驶仪的指令信号而工作，当人工驾驶飞机时，由于离合器保持脱开而传动部件不发生作用。②液压舵机，由液压作动器和旁通活门组成。当人工驾驶飞机时，旁通活门打开，由于作动器活塞两边的液压互相连通而不妨碍人工操纵。此外，还有电动液压舵机，简称“电

液舵机”。



图 4 舵机

## 2、程序目标分析

- (1) 先用红外遥控器控制工程机器人的移动
- (2) 在控制机械臂的运动
- (3) 在控制机械爪的抓起和放下

## 3、程序流程图

分析程序如图所示。

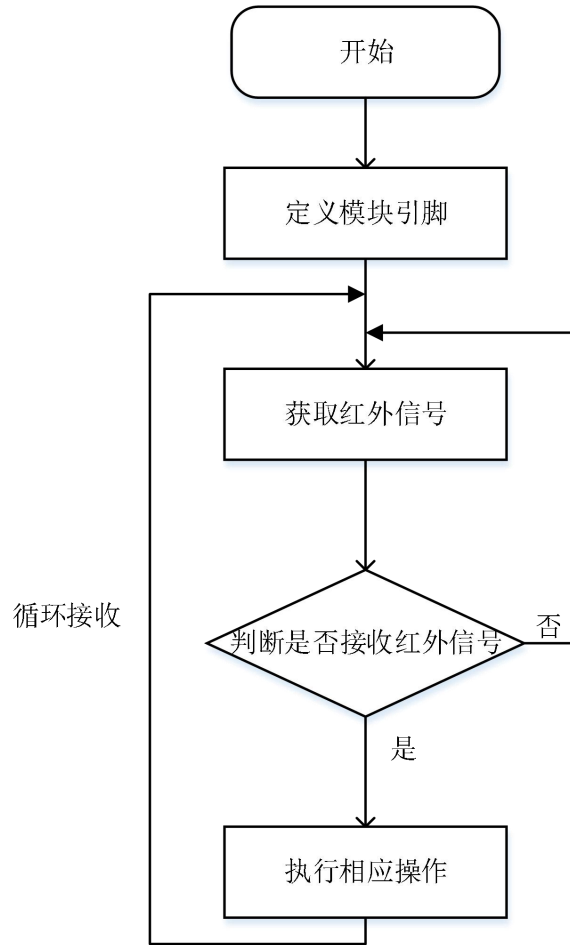


图 5 程序流程图

#### 4、代码分析

```

from ir_remote import IRDecode #红外解码库

remote = IRDecode(Pin.P0) #定义红外引脚

#回调函数
def remote_callback(_address, _command):
    print(_command) #打印红外信号编码
remote.set_callback(remote_callback)
  
```

图 6 红外模块程序

```
from servo import Servo #舵机库

#定义舵机引脚
servo_0 = Servo(0, min_us=500, max_us=2500, actuation_range=180)

#舵机转动的角度
servo_0.write_angle(60)
```

图 7 舵机模块程序

```
#定义电机引脚
p0 = MPythonPin(0, PinMode.PWM)
p1 = MPythonPin(1, PinMode.PWM)
#设置电机转速
speed = 100
#将速度映射为模拟信号0~1023
speed = int(numberMap(speed, 0, 100, 0, 1023))
#将模拟信号给引脚P0
p0.write_analog(speed)
sleep_ms(2)
p1.write_analog(0)
sleep_ms(2)
```

图 8 电机模块程序

```

#机械爪抓取功能
if x == "b'\x07'":#按键左
    if angle_15 > 11:
        angle_15 -= 10
        servo_15.write_angle(angle_15)
    if angle_16 < 169:
        angle_16 += 10
        servo_16.write_angle(angle_16)
#机械爪放开功能
if x == "b'\t'":#按键右
    if angle_15 < 169:
        angle_15 += 10
        servo_15.write_angle(angle_15)
    if angle_16 > 11:
        angle_16 -= 10
        servo_16.write_angle(angle_16)

```

图 9 控制机械爪程序

```

#机械臂抬起功能
if x == "b'@'":#按键上
    p13.write_analog(speed)
    sleep_ms(2)
    p14.write_analog(0)
    sleep_ms(2)
    time.sleep(1)
    p14.write_analog(0)
    sleep_ms(2)
    p13.write_analog(0)
    sleep_ms(2)
#机械臂下降功能
if x == "b'\x19'":#按键下
    p14.write_analog(speed)
    sleep_ms(2)
    p13.write_analog(0)
    sleep_ms(2)
    time.sleep(1)
    p14.write_analog(0)
    sleep_ms(2)
    p13.write_analog(0)
    sleep_ms(2)

```

图 10 控制机械臂程序

```

#停止移动
if x == "b'\x15':#Enter按键
    p1.write_analog(0)
    sleep_ms(2)
    p0.write_analog(0)
    sleep_ms(2)
    p9.write_analog(0)
    sleep_ms(2)
    p8.write_analog(0)
    sleep_ms(2)
#前后移动
if x == "b'E':#按键A 功能前进
    p0.write_analog(0)
    sleep_ms(2)
    p1.write_analog(speed)
    sleep_ms(2)
    p9.write_analog(speed)
    sleep_ms(2)
    p8.write_analog(speed)
    sleep_ms(2)
if x == "b'F':#按键B 功能后退
    p1.write_analog(speed)
    sleep_ms(2)
    p0.write_analog(speed)
    sleep_ms(2)
    p8.write_analog(0)
    sleep_ms(2)
    p9.write_analog(speed)
    sleep_ms(2)

if x == "b'G':#按键C 功能左转
    p0.write_analog(speed)
    sleep_ms(2)
    p1.write_analog(speed)
    sleep_ms(2)
    p8.write_analog(speed)
    sleep_ms(2)
    p9.write_analog(speed)
    sleep_ms(2)
if x == "b'D':#按键D 功能右转
    p1.write_analog(speed)
    sleep_ms(2)
    p0.write_analog(0)
    sleep_ms(2)
    p9.write_analog(speed)
    sleep_ms(2)
    p8.write_analog(0)
    sleep_ms(2)

```

图 11 控制机器人程序