

## پروژه هفتم:

### ساخت ربات تعقیب خط با تکنیک پردازش تصویر



این ربات با استفاده از دوربینی که بر روی آن نصب شده است تصاویر را به نرم افزار متلب ارسال می کند و نرم افزار متلب نیز بر اساس برنامه ای که در آن نوشته شده است با پورت سریال فرمان های لازم را به ربات می دهد. البته در این پروژه از یک مبدل سریال به USB استفاده شده است.

### سخت افزار:

همانطور که در عکس دیدید این ربات دارای دو موتور برای دو چرخ و یک چرخ هرزگرد می باشد. در سخت افزار این پروژه از Mosfet برای راه اندازی موتورهای DC استفاده شده است. با استفاده از آی سی MAX232 ارتباط سریال بین کامپیوتر و میکرو برقرار می شود. عکس مدار این ربات در محیط شبیه سازی در زیر آورده شده است:



```
#include <mega32.h>

#include <stdio.h>

void main (void)
{
    char a;

    DDRB.3=1;

    PORTB.3=0;

    DDRD.7=1;

    PORTD.7=0;

    TCNT0=0;

    OCR0=50;

    TCNT2=0;
```

```

{
    a=getchar();
    if (a=='u')
    {
        TCCR0=0X6b;
        TCCR2=0X6c;
    }
    else if (a=='r')
    {
        TCCR0=0X6b;
        TCCR2=0X00;
    }
    else if (a=='l')
    {
        TCCR0=0X00;
        TCCR2=0X6c;
    }
}
}

```

### توضیح برنامه AVR:

در برنامه میکروکنترلر علاوه بر فعال کردن پورت سریال برای دریافت اطلاعات، از دو تایمر در مد PWM برای کنترل سرعت چرخ ها استفاده شده است. این برنامه بسیار ساده نوشته شده است به این صورت که در تابع main و قبل از لوپ اصلی، مقادیر ثابتی را به OCR تایمرها می دهیم. (با کم و زیاد کردن این مقادیر می توانید سرعت چرخ ها را تغییر دهید) سپس پورت سریال را با Baud rate (نرخ انتقال) 9600 فعال می کنیم.

در لوپ اصلی نیز کاراکتری از پورت سریال دریافت می شود و در صورتی که کاراکتر مورد نظر برابر 'u' بود هردو چرخ می چرخد (هر دو تایمر روشن می شود) تا ربات به صورت مستقیم حرکت کند. اگر کاراکتر برابر 'r' بود چرخ سمت راست خاموش می شود تا ربات به سمت راست بچرخد و اگر کاراکتر برابر 'l' بود چرخ سمت چپ خاموش می شود تا ربات به سمت چپ بچرخد.

## متن کامل برنامه Matlab:

```
***** PART1 *****
o close all;clear all;clc;closepreview;
o cam = videoinput('winvideo' , 1, 'YUY2_160x120');
o preview(cam)
o set(cam.source,'Brightness',200);
o set(cam,'ReturnedColorSpace','rgb');
o pause(2);
o se = strel ( 'line' , 10 , 90 ) ;
***** PART2 *****
o SR=serial('COM3');
o set(SR,'baudrate',9600);
o fopen(SR);
***** PART3 *****
o while(1)
o pic = getsnapshot(cam);
o bw = im2bw(pic , graythresh(pic));
o bw = bwareaopen(bw,100);
o bw = imfill(bw, 'holes');
o bw = imclose ( bw , se ) ;
o bw = ~bw;
o bw = imdilate ( bw , se ) ;
o imshow(bw)
***** PART4 *****
o up=bw(40:50,72:88);
o L1=bw(50:55,50:65);
o L2=bw(55:60,35:50);
o L3=bw(60:65,20:35);
o L4=bw(65:70,5:20);
o R1=bw(50:55,95:110);
o R2=bw(55:60,110:125);
o R3=bw(60:65,125:140);
o R4=bw(65:70,140:155);
***** PART5 *****
o up1=mean(mean(up));
o L(1)=mean(mean(L1));
o L(2)=mean(mean(L2));
o L(3)=mean(mean(L3));
o L(4)=mean(mean(L4));
o R(1)=mean(mean(R1));
o R(2)=mean(mean(R2));
o R(3)=mean(mean(R3));
o R(4)=mean(mean(R4));
o left = mean (L);
o right = mean (R);
***** PART6 *****
o if left > 0.15
o left = 0 ;
o else
o left = 1 ;
o end
```



مستطیل بالا را با  $up$ ، مربع های سمت راست را با  $R1$  تا  $R4$  و مربع های سمت چپ را با  $L1$  تا  $L4$  نامگذاری می کنیم.

در بخش پنجم از مربع های بالا، راست و چپ میانگین گرفته و آنها را در متغیرهای  $up1$ ،  $right$  و  $left$  می ریزیم.

در قسمت ششم از برنامه، با چند شرط ساده تصمیم گیری می کنیم که ربات به کدام سمت حرکت کند. اگر میانگین سمت راست از  $0.15$  بیشتر بود موتور سمت راست خاموش و در غیر اینصورت آنرا روشن می کنیم. به همین صورت برای موتور سمت چپ هم عمل می کنیم و در آخر اگر میانگین  $up$  از  $0.4$  بیشتر بود هر دو موتور را روشن می کنیم تا ربات به سمت جلو حرکت کند.

در قسمت هفتم طبق تصمیمی که برای حرکت ربات به سمت جلو، راست یا چپ گرفته ایم، یکی از سه کاراکتر  $'u'$ ،  $'r'$  یا  $'l'$  را در متغیر  $xout$  می نویسیم و در بخش هشتم آنرا با پورت سریال به میکرو ارسال می کنیم.