

// PARCIAL 5 - DISEÑO CON uP y uC. 2024-1.

// NOMBRE: Diego Andrés García Díaz.

// CÓDIGO: 2195533.

// -----

// DEFINIR VELOCIDADES

#define VEL_BASE 120

#define VEL_giro 90

//#define VEL_BASE 140

//#define VEL_giro 100

// DEFINIR NIVELES DIGIRALES PARA LA LINEA A SEGUIR

#define negro 1

#define blanco 0

// DEFINIR COMO ENTRADA LOS PINES DE LOS SENSORES

#define SI RA2_bit // Sensor de la IZQUIERDA

#define SC RA4_bit // Sensor del centro

#define SD RA5_bit // Sensor de la DERECHA

// CONFIGURACIÓN DE LOS PINES DEL SENSOR PARA ENTRADA/SALIDA

#define SI_DIR TRISA2_bit // Sensor de la IZQUIERDA

#define SC_DIR TRISA4_bit // Sensor del centro

#define SD_DIR TRISA5_bit // Sensor de la DERECHA

// DEFINIR COMO SALIDA LOS PINES DEL MOTOR DE LA DERECHA

#define M_D1 LATC0_bit

#define M_D2 LATC1_bit

// CONFIGURACIÓN DE LOS PINES DEL MOTOR DE LA DERECHA SALIDA

#define M_D1_DIR TRISC0_bit

#define M_D2_DIR TRISC1_bit

// DEFINIR COMO SALIDA LOS PINES DEL MOTOR DE LA IZQUIERDA

#define M_I1 LATC2_bit

#define M_I2 LATC4_bit

```
// CONFIGURACIÓN DE LOS PINES DEL MOTOR DE LA IZQUIERDA SALIDA
```

```
#define M_I1_DIR TRISC2_bit
```

```
#define M_I2_DIR TRISC4_bit
```

```
signed int VELOCIDAD;
```

```
// Motor DERECHA STOP
```

```
void M_DSTOP(){
```

```
    M_D1=0;
```

```
    M_D2=0;
```

```
}
```

```
// Motor DERECHA ADELANTE
```

```
void M_DADELANTE(){
```

```
    M_D1=1;
```

```
    M_D2=0;
```

```
}
```

```
// Motor DERECHA ATRAS
```

```
void M_DATRAS(){
```

```
    M_D1=0;
```

```
    M_D2=1;
```

```
}
```

```
// Motor IZQUIERDA STOP
```

```
void M_ISTOP(){
```

```
    M_I1=0;
```

```
    M_I2=0;
```

```
}
```

```
// Motor IZQUIERDA ADELANTE
```

```
void M_IADELANTE(){
```

```
    M_I1=0;
```

```
    M_I2=1;
```

```
}
```

```
// Motor IZQUIERDA ATRAS
```

```
void M_IATRAS(){  
    M_I1=1;  
    M_I2=0;  
}
```

```
// SE QUEDA QUIETO
```

```
void Robot_QUIETO(){  
    M_DSTOP();  
    M_ISTOP();  
}
```

```
// VA HACIA ADELANTE
```

```
void Robot_ADELANTE(signed int VEL){  
    signed int VEL_D, VEL_I;  
    // SE SUMA O RESTA LA VELOCIDAD BASE CON LA VELOCIDAD QUE SE QUIERE (VEL)  
    VEL_D = VEL_BASE + VEL;  
    VEL_I = VEL_BASE - VEL;
```

```
    // LIMITES PARA LOS MOTORES
```

```
    if (VEL_D >= 255) VEL_D = 255;  
    if (VEL_D <= 0) VEL_D = 0;  
    if (VEL_I >= 255) VEL_I = 255;  
    if (VEL_I <= 0) VEL_I = 0;
```

```
    PWM1_Set_Duty(VEL_D); // VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
```

```
    PWM2_Set_Duty(VEL_I); // VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
```

```
    M_DADELANTE();
```

```
    M_IADELANTE();
```

```
}
```

```
// VA HACIA ATRAS
```

```
void Robot_ATRAS(unsigned char VEL){
```

```
    PWM1_Set_Duty(VEL); // VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
```

```
    PWM2_Set_Duty(VEL); // VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
```

```
    M_DATRAS();
```

```

M_IATRAS();
}

// VA HACIA LA IZQUIERDA
void Robot_IZQUIERDA(unsigned char VEL){
    PWM1_Set_Duty(VEL);// VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
    PWM2_Set_Duty(VEL);// VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
    M_DADELANTE();
    M_IATRAS();
}

// VA HACIA LA DERECHA
void Robot_DERECHA(unsigned char VEL){
    PWM1_Set_Duty(VEL);// VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
    PWM2_Set_Duty(VEL);// VELOCIDAD INICIAL DEL PWM D ELOS MOTORES
    M_IADELANTE();
    M_DATRAS();
}

void LEER_SENSORES(){
    if ((SI==negro) && (SC==negro) &&(SD==negro) )           VELOCIDAD = VEL_giro;
    else if ((SI==negro) && (SC==negro) && (SD==blanco) )      VELOCIDAD = -VEL_giro;
    else if ((SI==negro) && (SC==blanco) && (SD==negro) )      VELOCIDAD = -VEL_giro; // VALOR
IMPOSIBLE
    else if ((SI==negro) && (SC==blanco) && (SD==blanco) )      VELOCIDAD = -VEL_giro;
    else if ((SI==blanco) && (SC==negro) && (SD==negro) )      VELOCIDAD = VEL_giro;
    else if ((SI==blanco) && (SC==negro) && (SD==blanco) )      VELOCIDAD = 0;
    else if ((SI==blanco) && (SC==blanco) && (SD==negro) )      VELOCIDAD = VEL_giro;
    else if ((SI==blanco) && (SC==blanco) && (SD==blanco) )      VELOCIDAD = 0;
}

// FUNCIÓN PRICIPAL
void main(){
    // CONFIGURACIÓN DEL OSCILADOR
    OSCCON=0;
    OSCTUNE=0;
    // OSILADOR EN PALABRA DE CONFIGURACIÓN

```

```
SCS1_bit=0;
```

```
SCS0_bit=0;
```

```
//OSCILADOR INTERNO 8 MHZ
```

```
IRCF3_bit=1;
```

```
IRCF2_bit=1;
```

```
IRCF1_bit=1;
```

```
IRCF0_bit=0;
```

```
// TODO DIGITAL
```

```
ANSELA=0;
```

```
ANSELC=0;
```

```
// CONFIGURACIÓN DE ENTRADAS Y SALIDAS
```

```
// ENTRADAS SENSORES
```

```
SI_DIR=1;
```

```
SC_DIR=1;
```

```
SD_DIR=1;
```

```
// SALIDAS MOTORES
```

```
M_D1_DIR=0;
```

```
M_D2_DIR=0;
```

```
M_I1_DIR=0;
```

```
M_I2_DIR=0;
```

```
// SALIDA PINES PWM
```

```
TRISC3_bit=0;
```

```
TRISC5_bit=0;
```

```
SPLEN_bit=0; // Prende el PLL x 4 = 32MHz, si es 0 da 8 MHz.
```

```
PWM1_Init(250);
```

```
PWM2_Init(250);
```

```
PWM1_Set_Duty(127); // VELOCIDAD INICIAL DEL PWM DE LOS MOTORES
```

```
PWM2_Set_Duty(127); // VELOCIDAD INICIAL DEL PWM DE LOS MOTORES
```

```
PWM1_Start();  
PWM2_Start();  
while (1){  
    LEER_SENSORES();  
    Robot_ADELANTE(VELOCIDAD);  
    //Delay_ms(1);  
}  
}
```