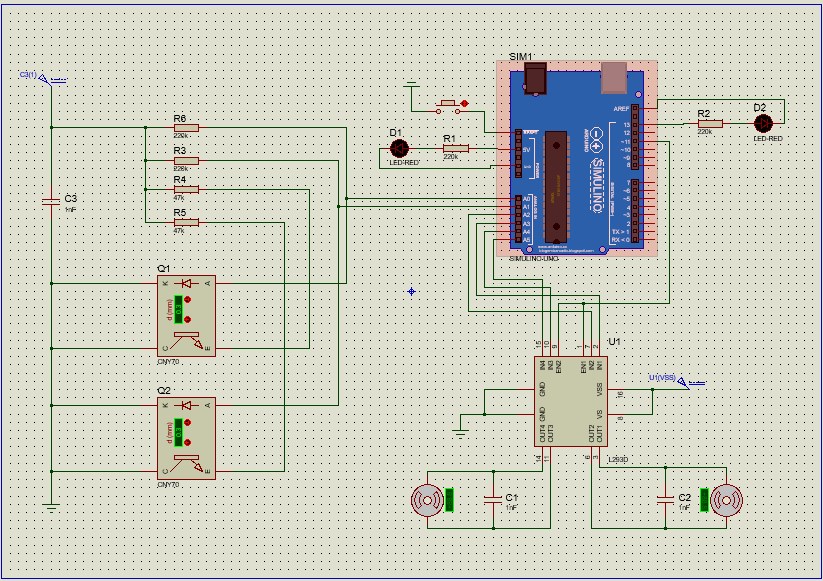
Documentación N°2

Diana Marcela Álzate B, Andrés Felipe Veloz Peralta, Juan Sebastián Sánchez Vega. Programa de Ingeniería Electrónica, Facultad de Ingeniería Universidad del Quindío - Colombia

Luego de realizar el montaje físico (mecánico) del seguidor de línea y de socializar tanto el diagrama del circuito con su respectiva programación, se realizaron los siguientes cambios:

* Se realizó el manejo de las señales del controlador de motores en los pines digitales del Arduino Uno y no en los pines análogos como estaba planteado anteriormente.
* Se realiza un control PID para manipular la velocidad de cada uno de los motores por medio del cambio en su voltaje de entrada, esta manipulación se realiza por medio de potenciómetros.
* Se controlan los motores por medio de dos pines PWM (en el esquema anterior se realizaba dicho control con solo un pin PWM) para darle más independencia a cada uno de los motores.
* En controlador de motores solo se van a tener control, valga la redundancia de solo una entrada de cada motor, esta elección se hizo para buscar una facilidad en el manejo del seguidor a nivel de programación.
* Finamente se instaló la librería correspondiente al Arduino Uno para poder hacer la simulación de todo el circuito.



* Finalmente se realizó la programación adecuada para el montaje realizado como se muestra a continuación, este programa se realizó por medio de la plataforma Arduino.
* En el código se utilizó el modulo diferenciador estático dado a que las velocidades del seguidor de línea se manipularan mediante del triger para manipular los voltajes del giro de los motores

#define M1A 19 //Motor 1A

#define M1B 18 //Motor 1B

#define M2A 17 //Motor 2A

#define M2B 16 //Motor 2B

#define PIN\_PWM 11 //Modulacion porancho de pulso - Emular salida analogicacon salida digital

#define VEL 100 //Velocidad

#define S1 0 //IZQ

#define S2 1 //DER

#define \_UMBRAL\_ 200 //Umbral de lossensores

void STOP(uint16\_t time);

void ADE(uint16\_t time);

void IZQ(uint16\_t time);

void DER(uint16\_t time);

void ATR(uint16\_t time);

unsigned long timeserial;

void setup (){

Serial.begin(9600);

timeserial = millis();

pinMode(M1A, OUTPUT);

pinMode(M1B, OUTPUT);

pinMode(M2A, OUTPUT);

pinMode(M2B, OUTPUT);

STOP(10000);

analogWrite(PIN\_PWM, VEL);

}

void loop(){

byte SDER = (analogRead(S1)> \_UMBRAL\_)?0:1;

byte SIZQ = (analogRead(S2)> \_UMBRAL\_)?0:1;

if(SDER && SIZQ)

STOP(0);

else if (!SDER && SIZQ)

DER(0);

else if (SDER && !SIZQ)IZQ(0);

else ADE(0);

if(millis() - timeserial > 500){

timeserial = millis();

}

Serial.print("Sensor1: ");//

Serial.print(SDER);

Serial.print(analogRead(S1));

Serial.print(" Sensor2: ");//

Serial.println(SIZQ);

Serial.println(analogRead(S2));

}

void ADE(uint16\_t time){

digitalWrite(M1A, LOW);

digitalWrite(M1B, HIGH);

digitalWrite(M2A, LOW);

digitalWrite(M2B, HIGH);

delay(time);

}

void DER(uint16\_t time){//Llanta Izquierda

digitalWrite(M1A, HIGH);

digitalWrite(M1B, LOW);//llanta Derecha

digitalWrite(M2A, LOW);

digitalWrite(M2B, HIGH);

delay(time);

}

void IZQ(uint16\_t time){//Llanta Izquierda

digitalWrite(M1A, LOW);

digitalWrite(M1B, HIGH);//Llanta Derecha

digitalWrite(M2A, HIGH);

digitalWrite(M2B, LOW);

delay(time);

}

void STOP(uint16\_t time){

digitalWrite(M1A, LOW);

digitalWrite(M1B, LOW);

digitalWrite(M2A, LOW);

digitalWrite(M2B, LOW);

delay(time);

}