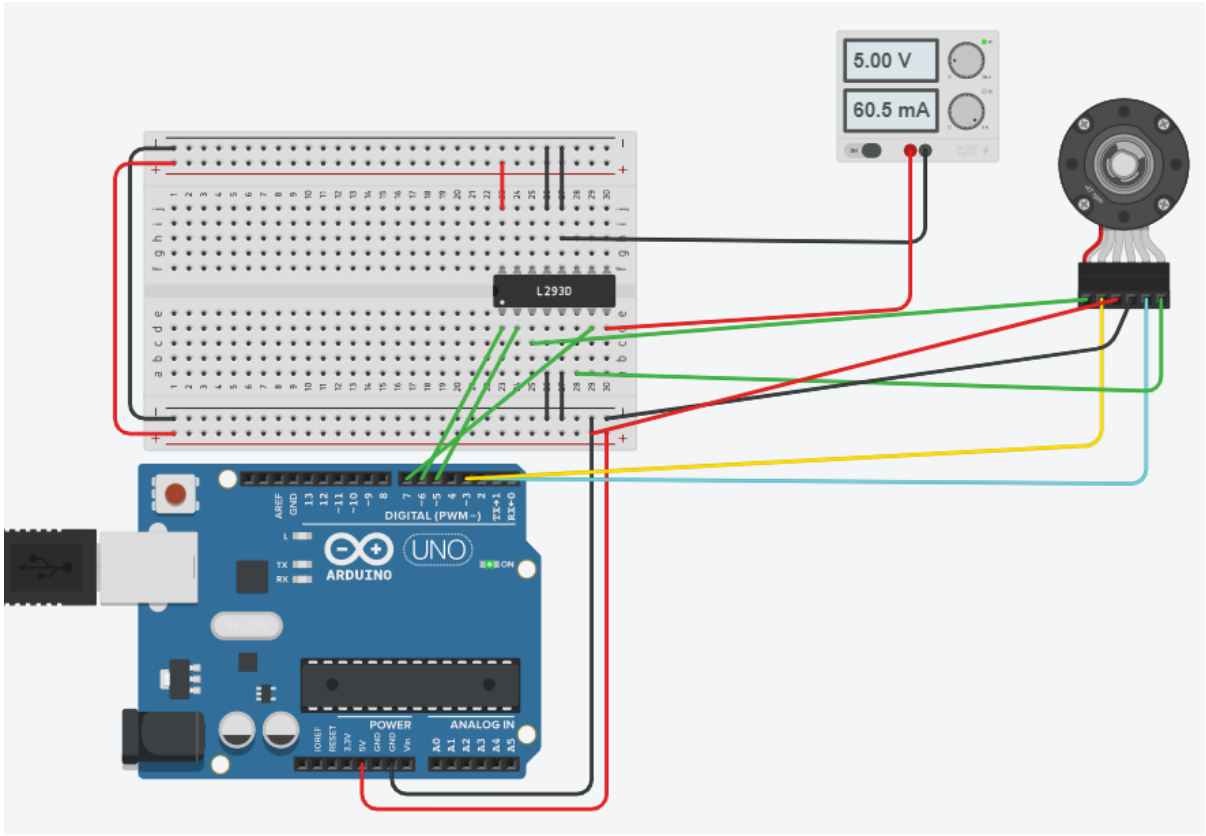
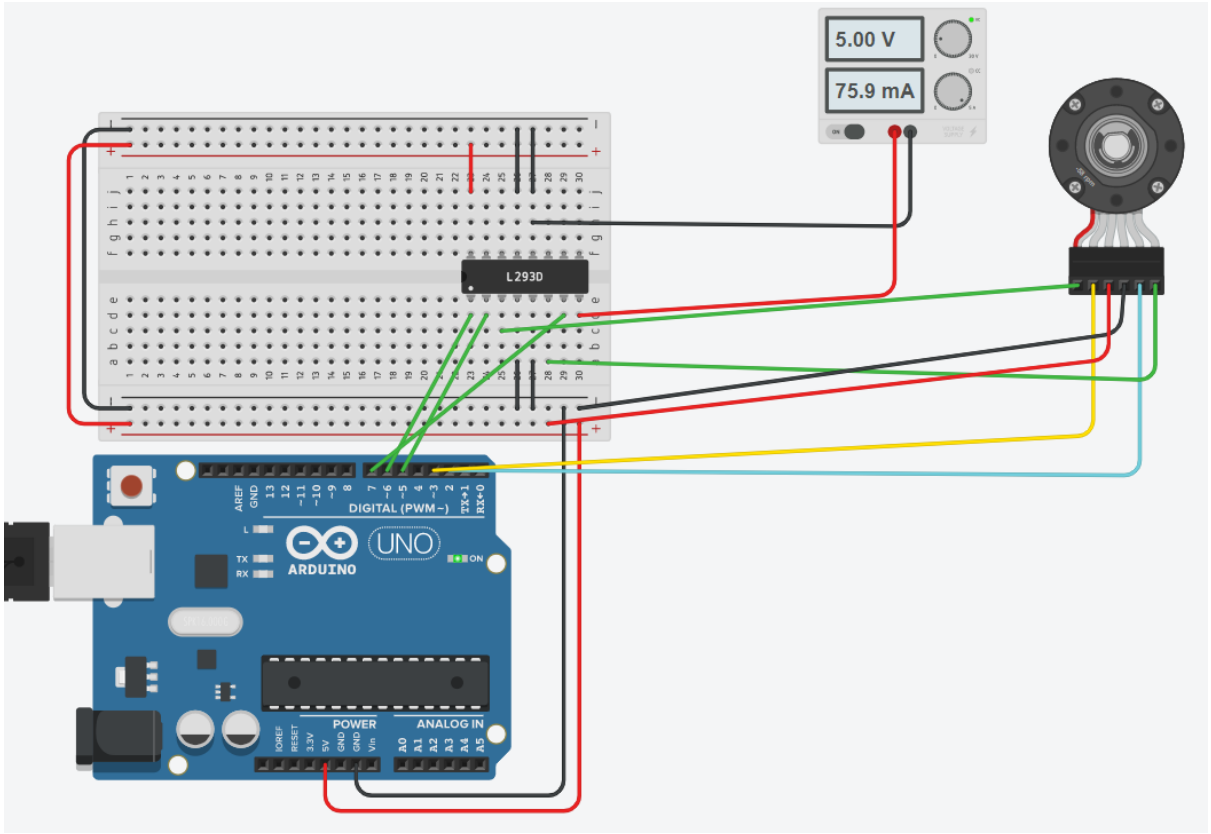



```
#define MOTOR_D1_PIN 7

#define MOTOR_D2_PIN 5

#define MOTOR_PWM_PIN 6

int interruptChannel1APin = 2;

int interruptChannel1BPin = 3;

int setpoint=0;

int speedFB = 0;

int error2 = 0;

float A = 1.0;

float B = 0.3;

volatile int encoderCount = 0;

volatile int encoderDir = 0;


void setup()

{

  pinMode(MOTOR_D1_PIN,OUTPUT);

  pinMode(MOTOR_D2_PIN,OUTPUT);
```

```
  pinMode(MOTOR_PWM_PIN,OUTPUT);

  pinMode(interruptChannel1APin,INPUT_PULLUP);
  pinMode(interruptChannel1BPin,INPUT_PULLUP);

  attachInterrupt(digitalPinToInterrupt(interruptChannel1APin),
            Channel1A_callback, RISING);
  attachInterrupt(digitalPinToInterrupt(interruptChannel1BPin),
            Channel1B_callback, RISING);
  Serial.begin(9600);
  Timer1_initialize(100);
}

void moveForward (int speed)
{
  digitalWrite(MOTOR_D1_PIN,HIGH);
  digitalWrite(MOTOR_D2_PIN,LOW);
  analogWrite(MOTOR_PWM_PIN,speed);
}

void moveBackward (int speed)
{
  digitalWrite(MOTOR_D1_PIN,LOW);
  digitalWrite(MOTOR_D2_PIN,HIGH);
  analogWrite(MOTOR_PWM_PIN,speed);
}

void setSpeed (int speed)
{
  if ((speed>0)&&(speed<100))
  {
```

```
   if(speed>100)
   {speed=100;}
   moveForward(speed);
 }
 else if ((speed<0)&&(speed<-100))
 {
  speed=speed*(-1);
  if (speed>100)
  {speed=100;}
  moveBackward(speed);
 }
 else
 {moveForward(1);}
}

void Timer1_initialize (int period)
{
 noInterrupts();
 TCCR1A = 0;
 TCCR1B = 0;
 TCNT1  = 0;
 OCR1A  = (16000000.0/(256.0*1000.0))*period - 1;
 TCCR1B |= (1 << WGM12);
 TCCR1B |= (1 << CS12) | (0 << CS11) | (0 << CS10);
 TIMSK1 |= (1 << OCIE1A);
 interrupts();
}

void loop()
{
 int error = setpoint - speedFB;
```

```
  int pid = A*(float)(error)+ B*(float)(error2);
  error2 += error;
  setSpeed(pid);

  Serial.print(setpoint);
  Serial.print(",");
  Serial.println(speedFB);
  delay(100);
}

void serialEvent()
{
  setpoint = Serial.parseInt();
}
void Channel1A_callback()
{
  if (digitalRead(interruptChannel1APin)==1 &&
  digitalRead(interruptChannel1BPin)==0)
   {encoderCount++;}
}

void Channel1B_callback()
{
  if (digitalRead(interruptChannel1APin)==0 &&
  digitalRead(interruptChannel1BPin)==1)
   {encoderCount--;}
}
ISR(TIMER1_COMPA_vect)
{
  speedFB= encoderCount;
  encoderCount = 0;}
```



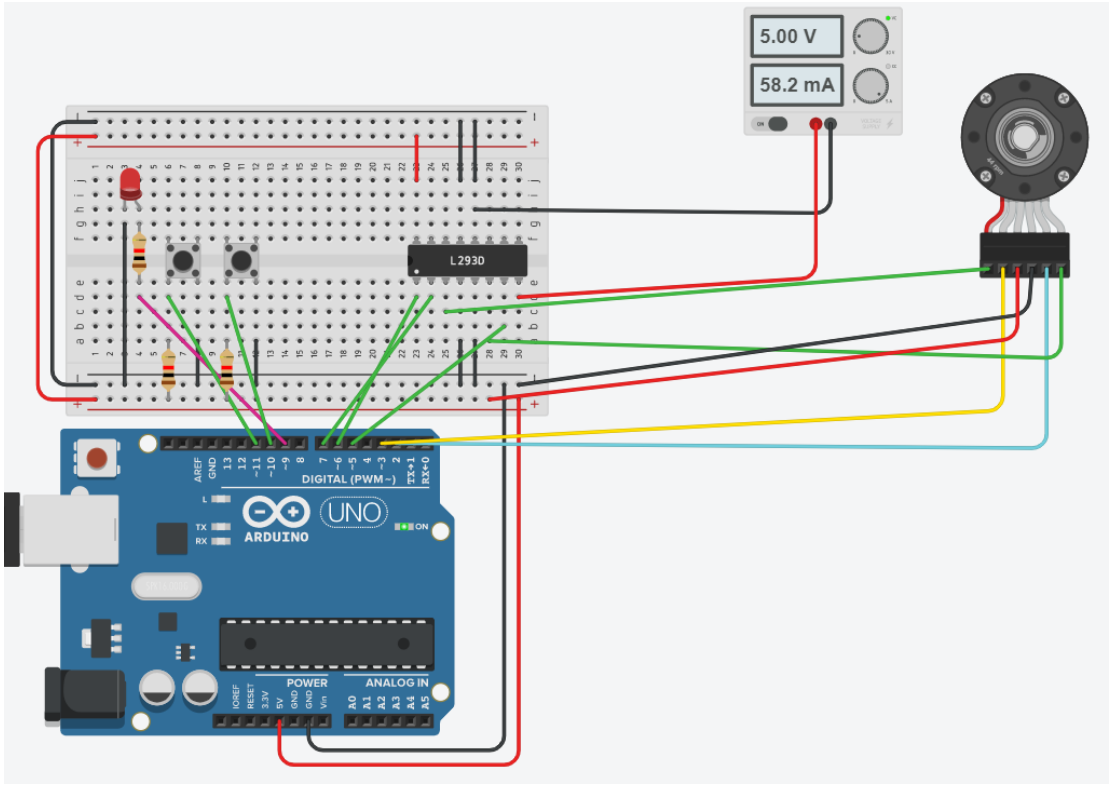

```
#define MOTOR_D1_PIN 7

#define MOTOR_D2_PIN 5

#define MOTOR_PWM_PIN 6

int interruptChannel1APin = 2;

int interruptChannel1BPin = 3;

int setpoint=0;

int speedFB = 0;

int error2 = 0;

float A = 1.0;

float B = 0.3;

volatile int encoderCount = 0;

volatile int encoderDir = 0;


void setup()

{

  pinMode(MOTOR_D1_PIN,OUTPUT);

  pinMode(MOTOR_D2_PIN,OUTPUT);
```

```
  pinMode(MOTOR_PWM_PIN,OUTPUT);

  pinMode(interruptChannel1APin,INPUT_PULLUP);
  pinMode(interruptChannel1BPin,INPUT_PULLUP);

  attachInterrupt(digitalPinToInterrupt(interruptChannel1APin),
            Channel1A_callback, RISING);
  attachInterrupt(digitalPinToInterrupt(interruptChannel1BPin),
            Channel1B_callback, RISING);
  Serial.begin(9600);
  Timer1_initialize(100);
}

void moveForward (int speed)
{
  digitalWrite(MOTOR_D1_PIN,HIGH);
  digitalWrite(MOTOR_D2_PIN,LOW);
  analogWrite(MOTOR_PWM_PIN,speed);
}

void moveBackward (int speed)
{
  digitalWrite(MOTOR_D1_PIN,LOW);
  digitalWrite(MOTOR_D2_PIN,HIGH);
  analogWrite(MOTOR_PWM_PIN,speed);
}

void setSpeed (int speed)
{
  if ((speed>0))
  {
```

```
   if(speed>100)
   {speed=100;}
   moveForward(speed);
 }
 else if ((speed<0))
 {
  speed=speed*(-1);
   if (speed>255)
   {speed=255;}
   moveBackward(speed);
 }
 else
 {moveForward(1);}
}

void Timer1_initialize (int period)
{
  noInterrupts();
  TCCR1A = 0;
  TCCR1B = 0;
  TCNT1  = 0;
  OCR1A  = (16000000.0/(256.0*1000.0))*period - 1;
  TCCR1B |= (1 << WGM12);
  TCCR1B |= (1 << CS12) | (0 << CS11) | (0 << CS10);
  TIMSK1 |= (1 << OCIE1A);
  interrupts();
}

void loop()
{
  int error = setpoint - speedFB;
```

```
  int pid = A*(float)(error)+ B*(float)(error2);
  error2 += error;
  setSpeed(pid);

  Serial.print(setpoint);
  Serial.print(",");
  Serial.println(speedFB);
  delay(100);
}

void serialEvent()
{
  if(Serial.read() != 's') {return;}
  float val = Serial.parseFloat();
  if(Serial.read() != 'n') {return;}
  if(Serial.available()) {
     Serial.read();
  }

  setpoint = constrain(val, -100.0, 100.0);

  Serial.println(setpoint);
}

void Channel1A_callback()
{
  if (digitalRead(interruptChannel1APin)==1 &&
  digitalRead(interruptChannel1BPin)==0)
   {encoderCount++;}
}
```

```
void Channel1B_callback()
{
  if (digitalRead(interruptChannel1APin)==0 &&
  digitalRead(interruptChannel1BPin)==1)
   {encoderCount--;}
}
ISR(TIMER1_COMPA_vect)
{
  speedFB= encoderCount;
  encoderCount = 0;
}
```




```
#define MOTOR_D1_PIN 5

#define MOTOR_D2_PIN 7

#define MOTOR_PWM_PIN 6

int interruptChannel1APin = 2;

int interruptChannel1BPin = 3;

int delay_count = 0;

int setpoint=0;

int fb_speed = 0;

int target;

int led_state= LOW;

unsigned long previous_millis_led = 0;

const long led_interval = 500;

volatile int encoderCount = 0;

volatile int encoderDir = 0;

int errori = 0;

float kp = 1.0;

float ki = 0.3;
```

```
void setup()
{
  pinMode(MOTOR_D1_PIN,OUTPUT);
  pinMode(MOTOR_D2_PIN,OUTPUT);
  pinMode(MOTOR_PWM_PIN,OUTPUT);
  pinMode(10,INPUT_PULLUP);
  pinMode(11,INPUT_PULLUP);
  pinMode(9,OUTPUT);
  pinMode(interruptChannel1APin,INPUT_PULLUP);
  pinMode(interruptChannel1BPin,INPUT_PULLUP);
  attachInterrupt(digitalPinToInterrupt(interruptChannel1APin),
            Channel1A_callback, RISING);
  attachInterrupt(digitalPinToInterrupt(interruptChannel1BPin),
            Channel1B_callback, RISING);
  Serial.begin(9600);
  Timer1_initialize(300);
}

void moveForward (int speed)
{
  digitalWrite(MOTOR_D1_PIN,HIGH);
  digitalWrite(MOTOR_D2_PIN,LOW);
  analogWrite(MOTOR_PWM_PIN,speed);
}

void moveBackward (int speed)
{
  digitalWrite(MOTOR_D1_PIN,LOW);
  digitalWrite(MOTOR_D2_PIN,HIGH);
  analogWrite(MOTOR_PWM_PIN,speed);
}
```

```
void setSpeed (int speed)
{
 if (speed>0)
 {
   if(speed>255)
   {speed=255;}
   moveForward(speed);
 }
 else if (speed<0)
 {
  speed=speed*(-1);
  if (speed>255)
  {speed=255;}
  moveBackward(speed);
 }
 else
 {moveForward(1);}
}

void Timer1_initialize (int period)
{
 noInterrupts();
 TCCR1A = 0;
 TCCR1B = 0;
 TCNT1  = 0;
 OCR1A  = (16000000.0/(256.0*1000.0))*period - 1;
 TCCR1B |= (1 << WGM12);
 TCCR1B |= (1 << CS12) | (0 << CS11) | (0 << CS10);
 TIMSK1 |= (1 << OCIE1A);
 interrupts();
```

```
}

void loop()
{
  unsigned long current_milli=millis();
  if(current_milli - previous_millis_led>=led_interval){
    previous_millis_led = current_milli;
    digitalWrite(9,led_state);
    led_state=!led_state;
  }
  if(digitalRead(10)==0){
    if(target<0){
      target=target*-1;}
    while(digitalRead(10)==0){delay(10);}
  }
  if(digitalRead(11)==0){
    if(target>0){
      target=target*-1;
    }
    while(digitalRead(11)==0){delay(10);}
  }
  setpoint = target;
  int error = setpoint - fb_speed;
  int pid = kp*(float)(error)+ ki*(float)(errori);
  errori += error;
  setSpeed(pid);
  Serial.print(setpoint);
  Serial.print(",");
  Serial.println(fb_speed);
  delay(100);
}
```

```
void serialEvent() {
  if(Serial.read() != 's') {return;}
  float val = Serial.parseInt();
  if(Serial.read() != 'n') {return;}
  if(Serial.available()) {
     Serial.read();
  }
  target = constrain(val, -100.0, 100.0);
  Serial.println(target);
}

void Channel1A_callback()
{
  if (digitalRead(interruptChannel1APin)==1 &&
  digitalRead(interruptChannel1BPin)==0)
   {encoderCount++;}
}

void Channel1B_callback()
{
  if (digitalRead(interruptChannel1APin)==0 &&
  digitalRead(interruptChannel1BPin)==1)
   {encoderCount--;}
}
ISR(TIMER1_COMPA_vect)
{
  fb_speed = encoderCount;
  encoderCount = 0;
}
```