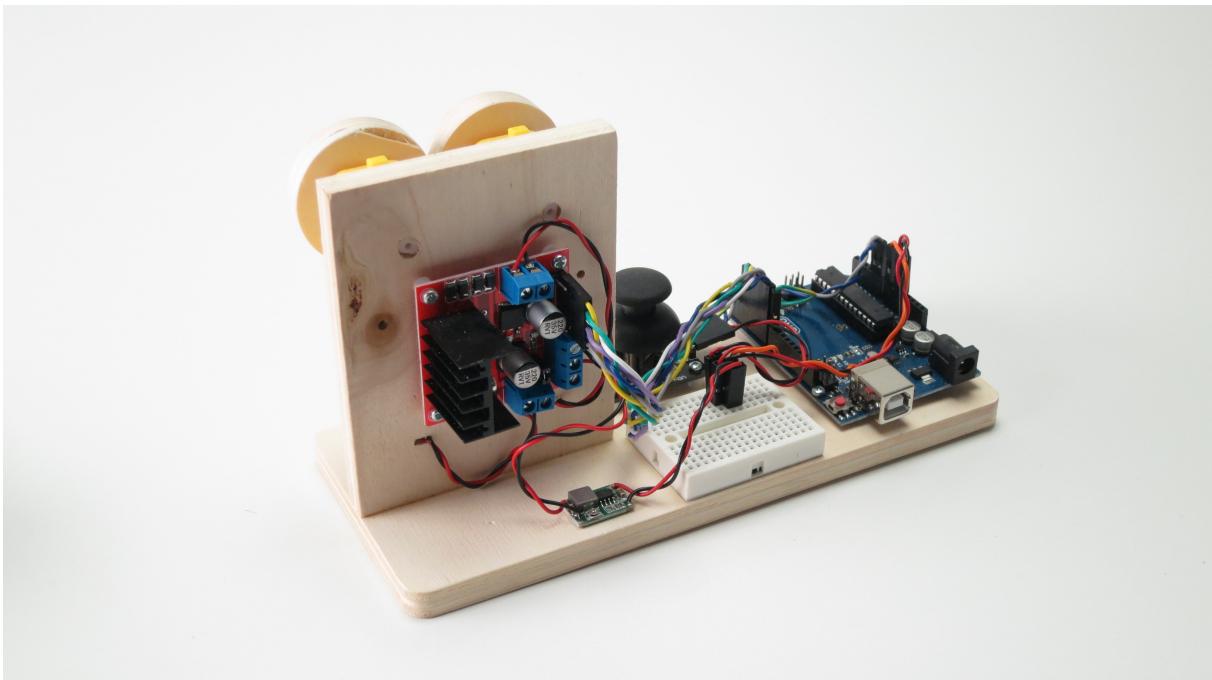
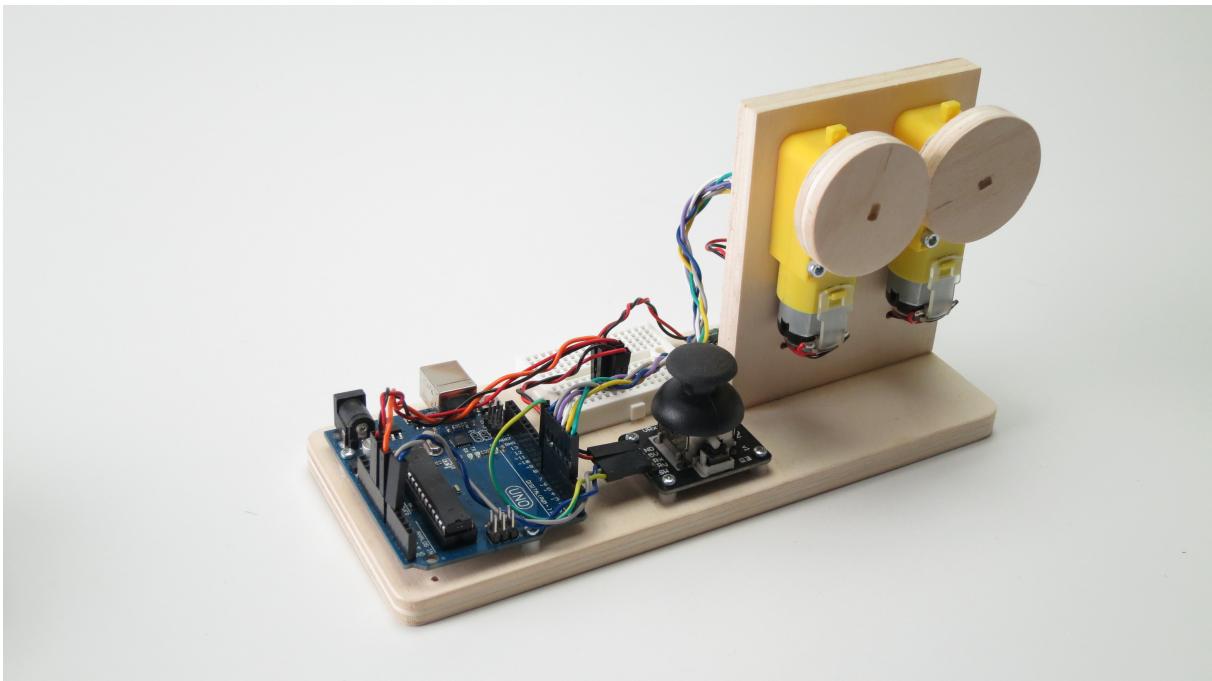


Motorsturing met joystick en H-brug

Foto's:



links:

uitleg code:

<http://dronebotworkshop.com/dc-motors-l298n-h-bridge/>

video:

https://www.youtube.com/watch?v=dyjo_ggEtVU

Sketch:

```
/*
L298N Motor Control Demonstration with Joystick
L298N-Motor-Control-Demo-Joystick.ino
Demonstrates use of Joystick control with Arduino and L298N Motor Controller

DroneBot Workshop 2017
http://dronebotworkshop.com
*/



// Motor A

int enA = 6;
int in1 = 7;
int in2 = 5;

// Motor B

int enB = 3;
int in3 = 4;
int in4 = 2;

// Joystick Input

int joyVert = A0; // Vertical
int joyHorz = A1; // Horizontal

// Motor Speed Values - Start at zero

int MotorSpeed1 = 0;
int MotorSpeed2 = 0;

// Joystick Values - Start at 512 (middle position)

int joyposVert = 512;
int joyposHorz = 512;

void setup()
```

```

{

// Set all the motor control pins to outputs

pinMode(enA, OUTPUT);
pinMode(enB, OUTPUT);
pinMode(in1, OUTPUT);
pinMode(in2, OUTPUT);
pinMode(in3, OUTPUT);
pinMode(in4, OUTPUT);

// Start with motors disabled and direction forward

// Motor A

digitalWrite(enA, LOW);
digitalWrite(in1, HIGH);
digitalWrite(in2, LOW);

// Motor B

digitalWrite(enB, LOW);
digitalWrite(in3, HIGH);
digitalWrite(in4, LOW);

}

void loop() {

// Read the Joystick X and Y positions

joyposVert = analogRead(joyVert);
joyposHorz = analogRead(joyHorz);

// Determine if this is a forward or backward motion
// Do this by reading the Verticle Value
// Apply results to MotorSpeed and to Direction

if (joyposVert < 460)
{
    // This is Backward

    // Set Motor A backward

    digitalWrite(in1, LOW);
    digitalWrite(in2, HIGH);
}
}

```

```

// Set Motor B backward

digitalWrite(in3, LOW);
digitalWrite(in4, HIGH);

//Determine Motor Speeds

// As we are going backwards we need to reverse readings

joyposVert = joyposVert - 460; // This produces a negative number
joyposVert = joyposVert * -1; // Make the number positive

MotorSpeed1 = map(joyposVert, 0, 460, 50, 255);
MotorSpeed2 = map(joyposVert, 0, 460, 50, 255);

}

else if (joyposVert > 564)
{
    // This is Forward

    // Set Motor A forward

    digitalWrite(in1, HIGH);
    digitalWrite(in2, LOW);

    // Set Motor B forward

    digitalWrite(in3, HIGH);
    digitalWrite(in4, LOW);

    //Determine Motor Speeds

    MotorSpeed1 = map(joyposVert, 564, 1023, 50, 255);
    MotorSpeed2 = map(joyposVert, 564, 1023, 50, 255);

}

else
{
    // This is Stopped

    MotorSpeed1 = 0;
    MotorSpeed2 = 0;

}

// Now do the steering
// The Horizontal position will "weigh" the motor speed
// Values for each motor

```

```

if (jo yposHorz < 460)
{
    // Move Left

    // As we are going left we need to reverse readings

    jo yposHorz = jo yposHorz - 460; // This produces a negative number
    jo yposHorz = jo yposHorz * -1; // Make the number positive

    // Map the number to a value of 255 maximum

    jo yposHorz = map(jo yposHorz, 0, 460, 50, 255);

    MotorSpeed1 = MotorSpeed1 - jo yposHorz;
    MotorSpeed2 = MotorSpeed2 + jo yposHorz;

    // Don't exceed range of 0-255 for motor speeds

    if (MotorSpeed1 < 0) MotorSpeed1 = 50;
    if (MotorSpeed2 > 255) MotorSpeed2 = 255;

}

else if (jo yposHorz > 564)
{
    // Move Right

    // Map the number to a value of 255 maximum

    jo yposHorz = map(jo yposHorz, 564, 1023, 50, 255);

    MotorSpeed1 = MotorSpeed1 + jo yposHorz;
    MotorSpeed2 = MotorSpeed2 - jo yposHorz;

    // Don't exceed range of 0-255 for motor speeds

    if (MotorSpeed1 > 255) MotorSpeed1 = 255;
    if (MotorSpeed2 < 0) MotorSpeed2 = 50;

}

// Adjust to prevent "buzzing" at very low speed

if (MotorSpeed1 < 8) MotorSpeed1 = 0;
if (MotorSpeed2 < 8) MotorSpeed2 = 0;

```

```
// Set the motor speeds  
analogWrite(enA, MotorSpeed1);  
analogWrite(enB, MotorSpeed2);  
}
```