**TECNOLÓGICO NACIONAL DE MEXICO**

**INSTITUTO TECNOLÓGICO DE LA LAGUNA**

****

**REPORTE DE PRACTICA**

**UNIDAD 2: Actuadores PRÁCTICA 01-04:**

**DOCENTE: LAMIA HAMDAN M.**

| **NUM DE CONTROL** | **NOMBRE** |
| --- | --- |
| 19130514 | Isaias Gerardo Cordova Palomares |
| 19130545 | Oscar Martinez Ruiz |
| 17130763 | Raúl Martín Ayala Salais |
| 19130541 | Pedro Lopez Ramirez |
| 19130535 | Ivan Herrera Garcia |
| 18131263 | Gerardo Alberto Orozco Villegas |

**FECHA DE ENTREGA:** 18/10/2022

**TABLA DE CONTENIDO**

[**1.INTRODUCCIÓN**](#_555xbbp1n8y5) **3**

[**2. COMPETENCIA A DESARROLLAR**](#_30j0zll) **3**

[**3. CIRCUITO LÓGICO Y/O PROGRAMA**](#_vyaxf5uqhjrg) **3**

[**4. METODOLOGÍA**](#_uikunnlbl3c) **3**

[**5. RESULTADOS**](#_4j5zl4o09ir7) **4**

[**6.CONCLUSIONES Y RECOMENDACIONES**](#_swudbzgg3llj) **11**

[**7. REFERENCIAS**](#_s8u6prgrx9vv) **11**

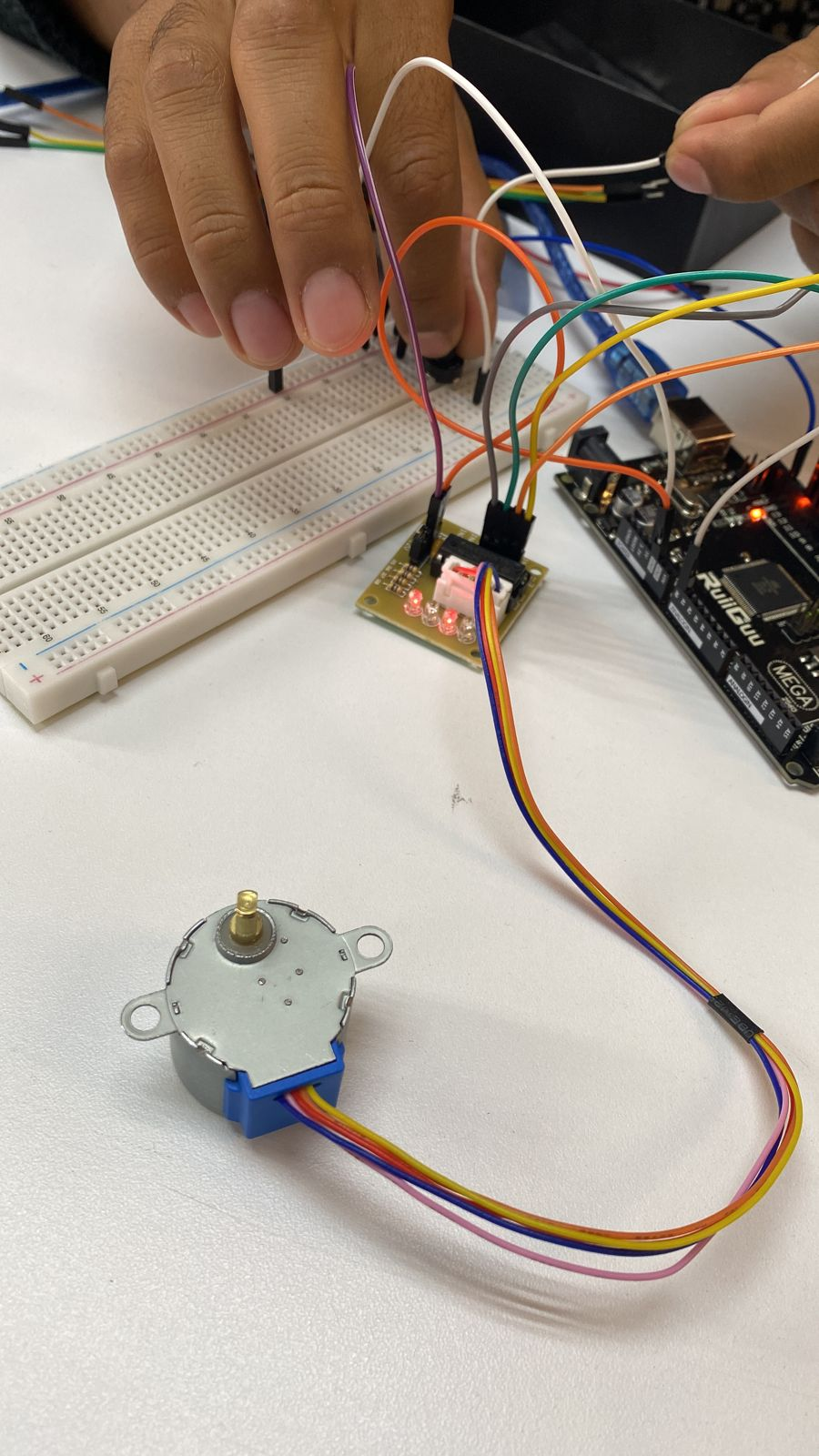
# **1.INTRODUCCIÓN**

Durante las practicas 01 hasta la 04 se buscara ver el funcionamiento y manipulación del motor paso a paso.

# **2. COMPETENCIA A DESARROLLAR**

Aprender el funcionamiento y manipulación de un motor paso a paso (Stepper)

# **3. CIRCUITO LÓGICO Y/O PROGRAMA**



*Circuito fisico de la practica*

# **4. METODOLOGÍA**

1. Conectar los puertos digitales correspondientes en la tarjeta controladora.
2. Conectar el cable de voltaje y tierra a la tarjeta controladora
3. Conectar los cables del motor paso a paso a la tarjeta controladora

**Material necesario:**

* Arduino
* Tarjeta controladora
* Cables
* Motor paso a paso (Stepper)
* Potenciómetro
* Protoboard
* Botón/PushButton

# 

# **5. RESULTADOS**

A lo largo de estas practicas se obtuvieron los siguientes resultados :

**Practica 01 : “Motor Knob”**

En esta primer practica se manipulan los pasos del motor mediante un potenciómetro, el cual por cada vuelta que demos hacia adelante este valor incrementara en 10, en caso de girar al lado contrario, este disminuye su valor en 10. El numero maximo de pasos definidos en un inicio es de 200, estos pueden aumentar o disminuir.

**Codigo**

/\*

\* MotorKnob

\*

\* A stepper motor follows the turns of a potentiometer

\* (or other sensor) on analog input 0.

\*f

\* http://www.arduino.cc/en/Reference/Stepper

\* This example code is in the public domain.

\*/

#include <Stepper.h>

// change this to the number of steps on your motor

#define STEPS 200

// create an instance of the stepper class, specifying

// the number of steps of the motor and the pins it's

// attached to

Stepper stepper(STEPS, 8, 9, 10, 11);

// the previous reading from the analog input

int previous = 0;

int potVal = 0;

int pVal = 0;

void setup() {

// set the speed of the motor to 30 RPMs

stepper.setSpeed(100);

}

void loop() {

// get the sensor value

// int val = analogRead(0);

potVal = map(analogRead(A0),0,1024,0,500);

if(potVal > pVal)

{

stepper.step(10);

}

if(potVal < pVal)

{

stepper.step(-10);

}Serial

pVal = potVal;

Serial.println(pVal);

/\*

// move a number of steps equal to the change in the

// sensor reading

stepper.step(val - previous);

// remember the previous value of the sensor

previous = val;\*/

}

**Practica 02 : oneRevolution**

En esta practica lo que se observo, esta fue una practica en la que no hubo necesidad de que intervinieran los practicantes, ya que mediante el codigo hacemos que cada 500 milisegundos o mejor dicho, cada 0.5 segundos. El motor cambie de sentido automáticamente.

En el IDE Arduino viendo el monitor se nos desplegara si el stepper esta girando en el sentido de las manecillas del reloj o si esta girando en contra de estas.

**Codigo :**

/\*

Stepper Motor Control - one revolution

This program drives a unipolar or bipolar stepper motor.

The motor is attached to digital pins 8 - 11 of the Arduino.

The motor should revolve one revolution in one direction, then

one revolution in the other direction.

Created 11 Mar. 2007

Modified 30 Nov. 2009

by Tom Igoe

\*/

#include <Stepper.h>

const int stepsPerRevolution = 200; // change this to fit the number of steps per revolution

// for your motor

// initialize the stepper library on pins 8 through 11:

Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);

void setup() {

// set the speed at 60 rpm:

myStepper.setSpeed(60);

// initialize the serial port:

Serial.begin(9600);

}

void loop() {

// step one revolution in one direction:

Serial.println("clockwise");

myStepper.step(stepsPerRevolution);

delay(500);

// step one revolution in the other direction:

Serial.println("counterclockwise");

myStepper.step(-stepsPerRevolution);

delay(500);

}

**Practica 03 : One Step At TIme**

El funcionamiento de esta practica, es basicamente. Llevar la cuenta de las veces en las que el stepper ha dado pasos o “steps”, esto se lleva acabo mediante un contador. Este ira incrementando de uno en uno.

La información del contador la podremos ver en el monitor con el que cuenta el IDE de Arduino

**Codigo :**

/\*

Stepper Motor Control - one step at a time

This program drives a unipolar or bipolar stepper motor.

The motor is attached to digital pins 8 - 11 of the Arduino.

The motor will step one step at a time, very slowly. You can use this to

test that you've got the four wires of your stepper wired to the correct

pins. If wired correctly, all steps should be in the same direction.

Use this also to count the number of steps per revolution of your motor,

if you don't know it. Then plug that number into the oneRevolution

example to see if you got it right.

Created 30 Nov. 2009

by Tom Igoe

\*/

#include <Stepper.h>

const int stepsPerRevolution = 200; // change this to fit the number of steps per revolution

// for your motor

// initialize the stepper library on pins 8 through 11:

Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);

int stepCount = 0; // number of steps the motor has taken

void setup() {

// initialize the serial port:

Serial.begin(9600);

}

void loop() {

// step one step:

myStepper.step(1);

Serial.print("steps:");

Serial.println(stepCount);

stepCount++;

delay(100);

}

**Practica 04 : speedControl**

Para esta practica hacemos uso nuevamente del potenciómetro, pero además de eso se añadió un botón/pushbutton.

El funcionamiento de la practica en si es, que con el potenciómetro se va aumentando la velocidad y/o pasos que da el mismo. Mientas que con el botoncito, lo que hacemos es invertir el sentido de giro del botón.

/\*

Stepper Motor Control - speed control

This program drives a unipolar or bipolar stepper motor.

The motor is attached to digital pins 8 - 11 of the Arduino.

A potentiometer is connected to analog input 0.

The motor will rotate in a clockwise direction. The higher the potentiometer value,

the faster the motor speed. Because setSpeed() sets the delay between steps,

you may notice the motor is less responsive to changes in the sensor value at

low speeds.

Created 30 Nov. 2009

Modified 28 Oct 2010

by Tom Igoe

\*/

#include <Stepper.h>

const int stepsPerRevolution = 200; // change this to fit the number of steps per revolution

// for your motor

// initialize the stepper library on pins 8 through 11:

Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);

int stepCount = 0; // number of steps the motor has taken

void setup() {

// nothing to do inside the setup

}

void loop() {

// read the sensor value:

int sensorReading = analogRead(A0);

// map it to a range from 0 to 100:

int motorSpeed = map(sensorReading, 0, 1023, 0, 100);

// set the motor speed:

if (motorSpeed > 0) {

myStepper.setSpeed(motorSpeed);

// step 1/100 of a revolution:

myStepper.step(stepsPerRevolution / 100);

}

}

# **6.CONCLUSIONES Y RECOMENDACIONES**

Realmente estas practicas fueron muy sencillas, ya que realmente no había mucho que conectar, además de que el codigo nos fue proveído por el IDE de Arduino, por lo que solo hubo que implementar. Lo que se aprendieron de estas mini practicas fue ver las diferentes formas en que puede funcionar el motor paso a paso.

# **7. REFERENCIAS**

Arduino.cc. 2022. *Stepper - Arduino Reference*. [online] Available at: <http://www.arduino.cc/en/Reference/Stepper> [Accessed 13 October 2022].

(Cátedra Digital - Agosto Diciembre 2022: Ingresar al sitio, 2022)