



Universidad Autónoma de Yucatán

Física General I

Dr. Guillermo Cordourier

Equipo:

*Carrillo Romero Mauricio
*Euan Herrera Ana Cecilia

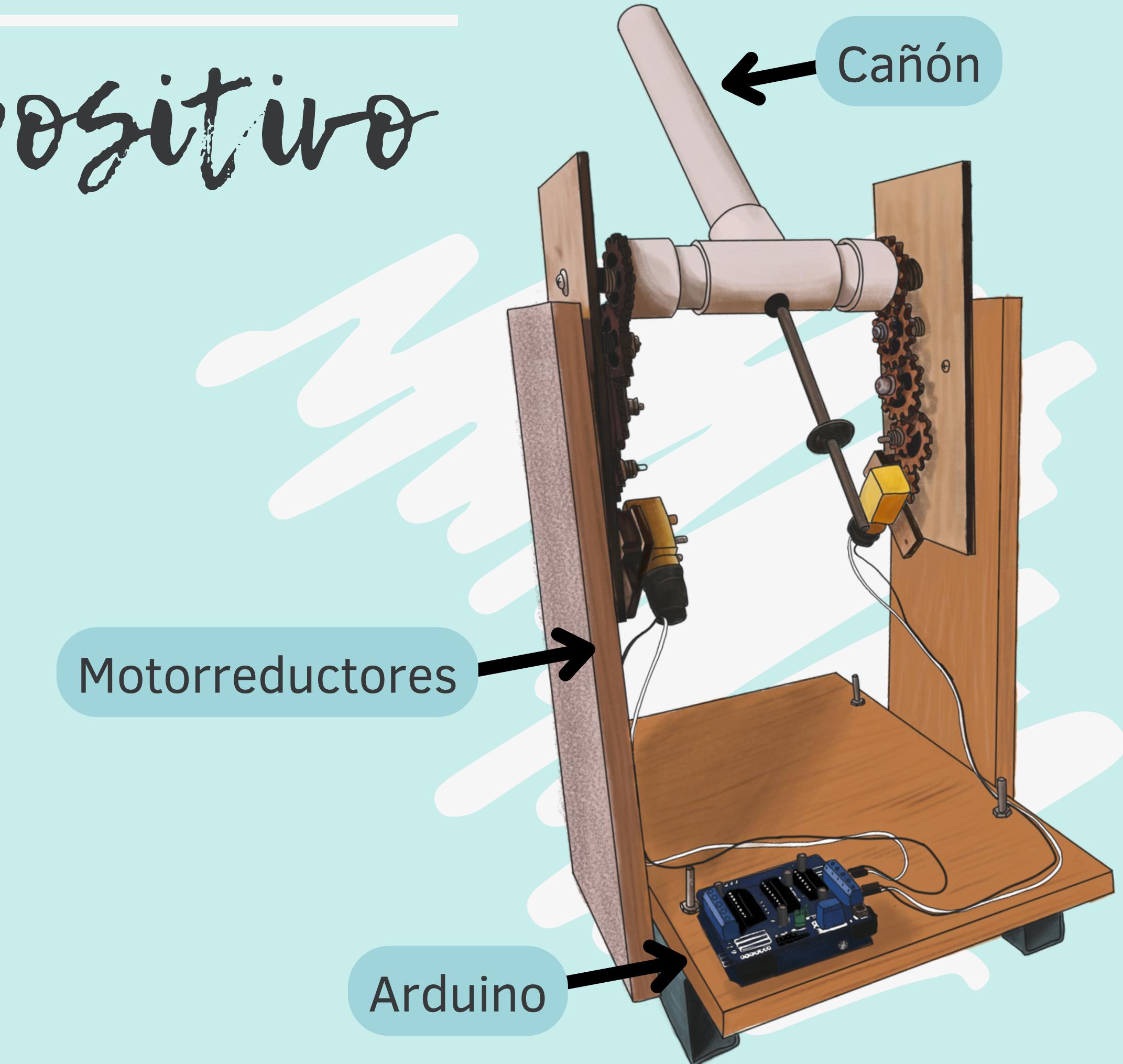
PROYECTO FINAL

CATAPULTA



Sobre el dispositivo

- Manejo sencillo
- Presentación estética
- Obtención de datos
- Estabilidad
- Precisión



Materiales

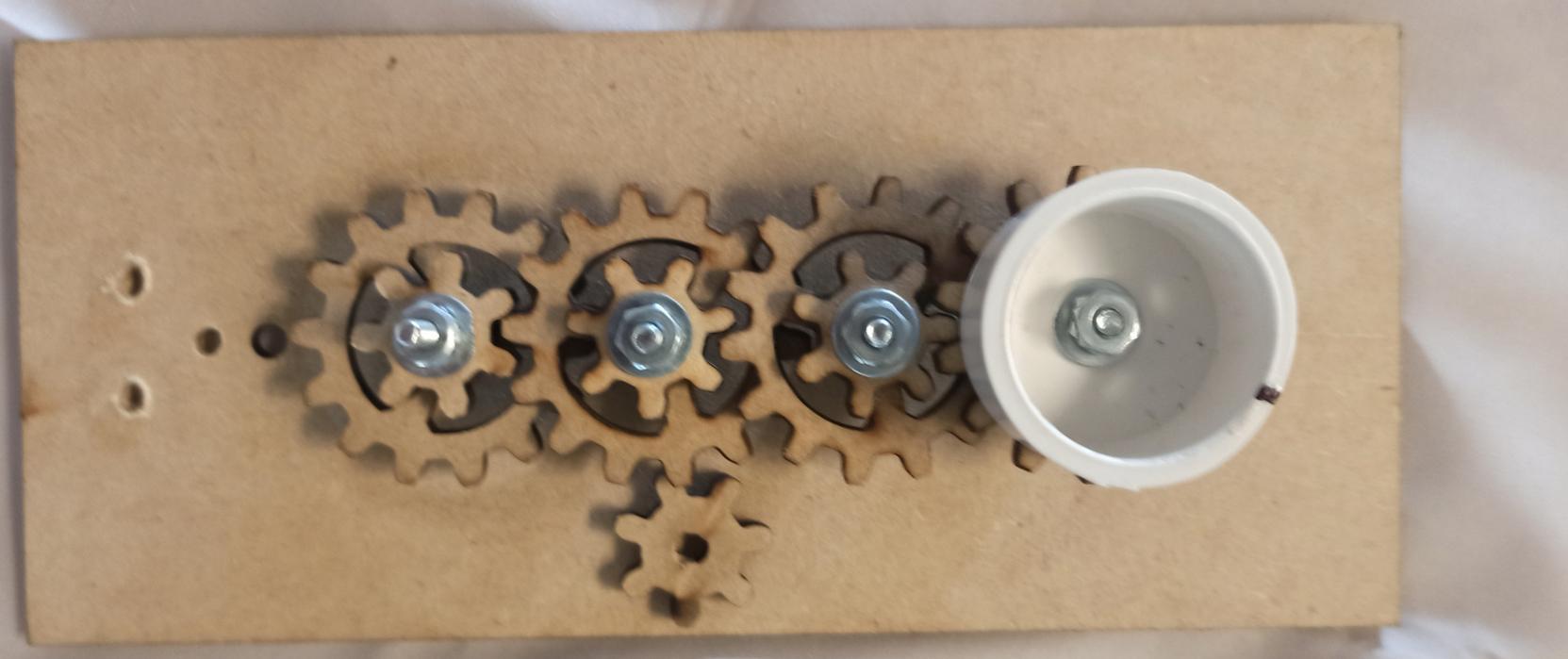
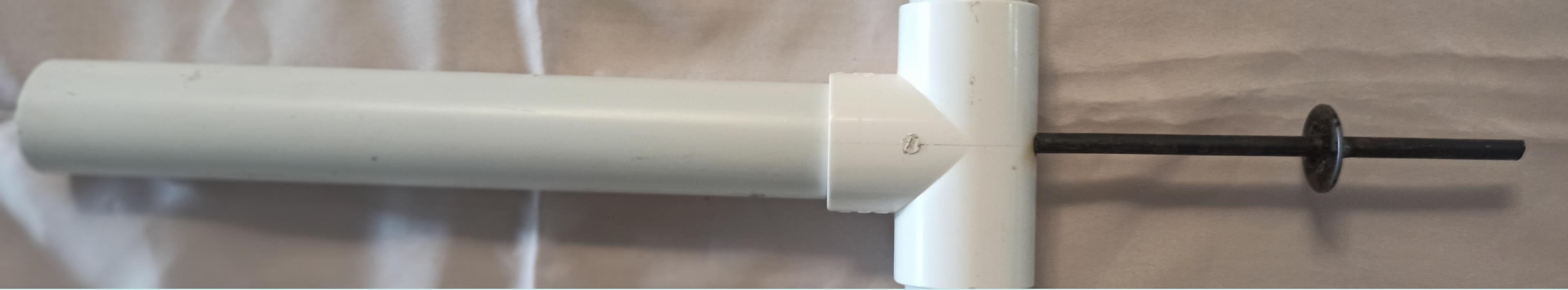


Ensamble

- Tubos de pvc
- Conexiones pvc
- Engranes fabricados con mdf
- Madera
- Tornillos, tuercas, rondanas
- Aluminio
- Piedras
- Cinta aislante

Electrónica

- Placa Arduino
- Tarjeta de control de motores L293D
- Cables
- Motorreductores
- Cable usb
- Alimentación por computadora de 5V



Proceso

Cálculos previos

Hallamos experimentalmente la velocidad de salida de la piedra y también la velocidad angular del cañón. Hallar la velocidad angular del cañón es importante porque ello nos permitirá conocer cuánto tiempo es necesario que el motor gire, para recorrer los grados que necesitemos. Así, la velocidad angular del motor estará dada por:

$$\omega = \frac{\theta}{t}$$

Se colocó el cañón a 0° y se accionaron los motores, cuando llegó a 90° el cronómetro marcó 4 segundos. Por lo tanto:

$$\omega = (90^\circ) / 3.54\text{s} = 25.4237288^\circ/\text{s}$$

Calculamos un error de $\pm 5^\circ$;

Proceso

Cálculos previos

Ahora calculamos la velocidad de salida de la piedra.

Recordando la fórmula que relaciona velocidad, ángulo y distancia máxima:

$$x_{máx} = \frac{V_0^2 \operatorname{Sen}(2\theta)}{g}$$

Por tanto, la fórmula para hallar velocidad es:

$$V_0 = \pm \sqrt{\frac{x_{máx} g}{\operatorname{Sen}(2\theta)}}$$

Tomamos el valor positivo de la velocidad. Además, se realizaron 10 pruebas para medir la distancia máxima a la que llegó la piedra, de modo que tenemos los siguientes datos:

Proceso

Cálculos previos

De este modo, en promedio, la distancia máxima que recorre la piedra tiene un valor de 297.9cm. Así, podemos calcular la magnitud de la velocidad y su error:

Tiro	Dist. Máx. (cm)
1	308.5
2	296
3	295
4	292
5	293
6	285
7	291
8	311
9	297.5
10	310
Promedio dist. Máx.	Error
297.9	0.845813218



Proceso

Cálculos previos

De este modo, en promedio, la distancia máxima que recorre la piedra tiene un valor de 297.9cm. Así, podemos calcular la magnitud de la velocidad y su error:

$$V_0 = \pm \sqrt{\frac{x_{\max} g}{\operatorname{Sen}(2\theta)}} = \pm \sqrt{\frac{(2.979 \text{ m})(9.81 \text{ m/s}^2)}{\operatorname{Sen}(2[45^\circ])}} = \pm \sqrt{(2.979 \text{ m})(9.81 \text{ m/s}^2)}$$

$$V_0 = 5.40592175 \text{ m/s}$$

Y el error es:

$$dV_0 = \frac{\partial V_0}{\partial x} dx + \frac{\partial V_0}{\partial \theta} d\theta$$

$$dV_0 = \sqrt{\frac{\operatorname{Sen}(2\theta)g \csc(\theta)}{x}} \frac{dx}{2} - \sqrt{gx\operatorname{Sen}(2\theta)} \csc(2\theta) \cot(2\theta) d\theta$$

Con $x = 2.979$, $g = 9.8$, $\theta = 45^\circ$;

$$dV_0 = 1.29001$$

Tiro	Dist. Máx. (cm)
1	308.5
2	296
3	295
4	292
5	293
6	285
7	291
8	311
9	297.5
10	310
Promedio dist. Máx.	Error
297.9	0.845813218

Proceso

Programa en arduino

```
programa_arduino

#include <AFMotor.h>
#include <Servo.h>
double variable = 0, variable_2 = 0;
int a = 0;
unsigned long tiempo;
String cadena;
AF_DCMotor motor1(1);
AF_DCMotor motor2(2);

//Condiciones iniciales
void setup() {
  tiempo = millis();
  motor1.setSpeed(255);
  motor2.setSpeed(255);
  // put your setup code here, to run once:
  Serial.begin(9600);
}

void loop() {
  //Si hay una entrada, se guarda el valor del tiempo
  if(Serial.available() > 0){
    cadena = Serial.readStringUntil('\n');
    cadena = cadena.substring(0, cadena.length() - 1);
    tiempo = millis();
  }
  //Si mandamos la letra r, se produce una reversa en el tiempo mandado con anterioridad
  if(cadena == "r"){
    if(millis()-tiempo <= variable_2 * 1000){
      Serial.println("RECARGA");
      motor1.run(BACKWARD);
    }
  }
}
```

```
programa_arduino

//Si mandamos la letra r, se produce una reversa en el tiempo mandado con anterioridad
if(cadena == "r"){
  if(millis()-tiempo <= variable_2 * 1000){
    Serial.println("RECARGA");
    motor1.run(BACKWARD);
    motor2.run(BACKWARD);
  }
  if(millis()-tiempo > variable_2 * 1000 ){
    motor1.run(RELEASE);
    motor2.run(RELEASE);
    Serial.println("Detenido EN RECARGA");
    cadena = "";
  }
}
//Si mandamos cualquier tiempo, el motor avanza ese tiempo
else {
  variable = cadena.toFloat();
  if(millis()-tiempo <= variable * 1000){
    Serial.println("ADELANTE");
    motor1.run(FORWARD);
    motor2.run(FORWARD);
    variable_2 = variable;
  }
  if(millis()-tiempo > variable * 1000 ){
    motor1.run(RELEASE);
    motor2.run(RELEASE);
    Serial.println("Detenido EN ADELANTE");
    cadena = "";
  }
}
```

Proceso

Programa en C#

Al ingresa este valor máximo en la caja de distancia, el ángulo nos da un resultado de 45° . Al obtener el valor del ángulo necesario, el programa realiza la operación:

$$t = \frac{\theta}{\omega}$$

Dado que el programa calcula el ángulo con la velocidad inicial calculada, y el valor de $\omega=25.4237288^\circ/\text{s}$, obtenemos el valor del tiempo de giro de los motores. Así, con ello le damos el ángulo correcto al cañón.

El lanzamiento de la piedra se realiza manualmente a través del muelle realizado.

Proceso

Programa en C#

```
}

1 referencia
private void ButtonConnect_Click(object sender, EventArgs e)
{
    try
    {
        serialPort1.PortName = comboBoxPuerto.Text;
        serialPort1.BaudRate = Convert.ToInt32(comboBoxBaudRate.Text);
        serialPort1.Open();
        progressBarConnection.Value = 100;

        ButtonDisconnect.Enabled = true;
        ButtonRecargar.Enabled = true;
        ButtonApuntar.Enabled = true;
        comboBoxBaudRate.Enabled = false;
        comboBoxPuerto.Enabled = false;
    }
    catch (Exception error)
    {
        MessageBox.Show(error.Message);
    }
}

1 referencia
private void ButtonCalcular_Click(object sender, EventArgs e)
{
```

