

DISPENSADOR DE ALIMENTO PARA MASCOTAS



¿Para quién va dirigido?

Dirigido para jóvenes adultos que tienen mascotas en casa.



Problemática

La importancia de nutrir adecuadamente a las mascotas.

Objetivo General

Desarrollar un dispensador de comida para mascotas que sea automático y programable.

Objetivo Específico

Realizar las especificaciones del sistema y diseñar la estructura mecánica y el control electrónico del dispensador automático.

La mala nutrición en las mascotas puede generar enfermedades que afectan su metabolismo y el desarrollo de sus huesos y órganos a una temprana edad, además de enfermedades como la: obesidad, problemas cardíacos, respiratorios,

La alternativa de solución que se propone en este proyecto es la de un prototipo de dispensador de comida, electrónico y automático que se pueda tener en el hogar.

IDEA

Materiales



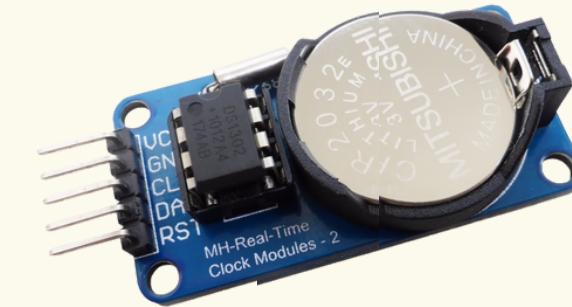
TRANSFORMADOR

DE 5V **S/13**



PROTOBOARD

S/2.50



MÓDULO DE RELOJ

S/10



SERVOMOTOR

S/13

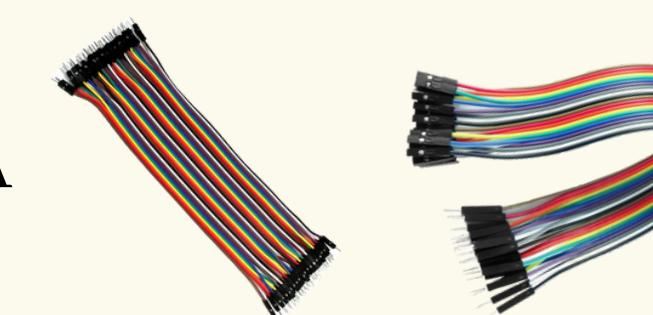


CABLE USB AB



ARDUINO UNO

S/49



PALITOS DE
MAQUETA



ENVASE DE
PLÁSTICO

JUMPER **S/5**

MACHO-HEMBRA

MACHO-MACHO

PASOS

CONECTAR SCL CON A5

CONECTAR SDA CON A4

CONECTAR CNO GND

CABLE CAFÉ a GND TIERRA

CABLE NARANJA a PIN 9

CONECTAR PROTOBOARD

CONECTAR VCC a PROTOB

CABLE ROJO a PROTOBOARD

CONECTAR USB-AB

ARDUINO UNO

+

MÓDULO DE RELOJ

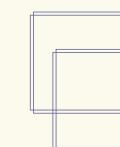
SERVOMOTOR

+

ARDUINO UNO

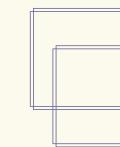


SISTEMA OPERATIVO

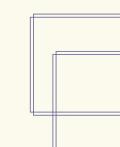
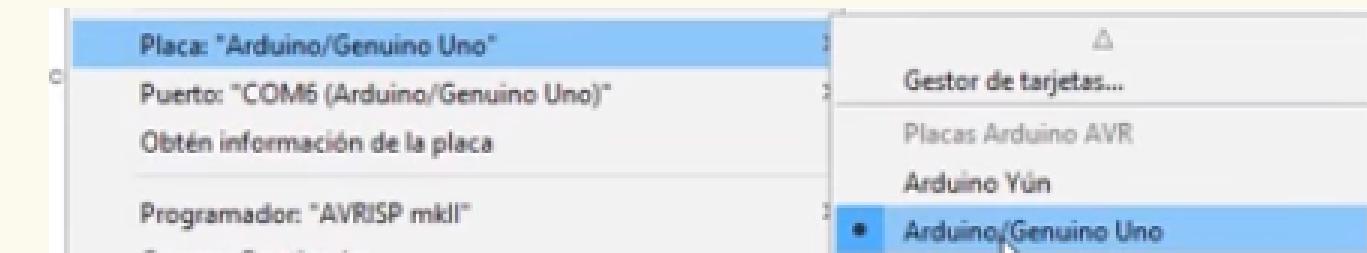


DESCARGAR ARDUINO EDITOR

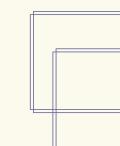
<https://www.arduino.cc/en/software>



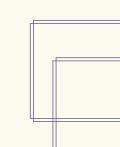
AÑADIR LIBRERÍA DS3121



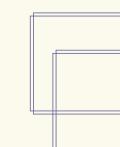
COPIAR SINCRONIZADOR



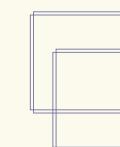
PEGAR EN EDITOR ARDUINO



SINCRONIZAR HORA

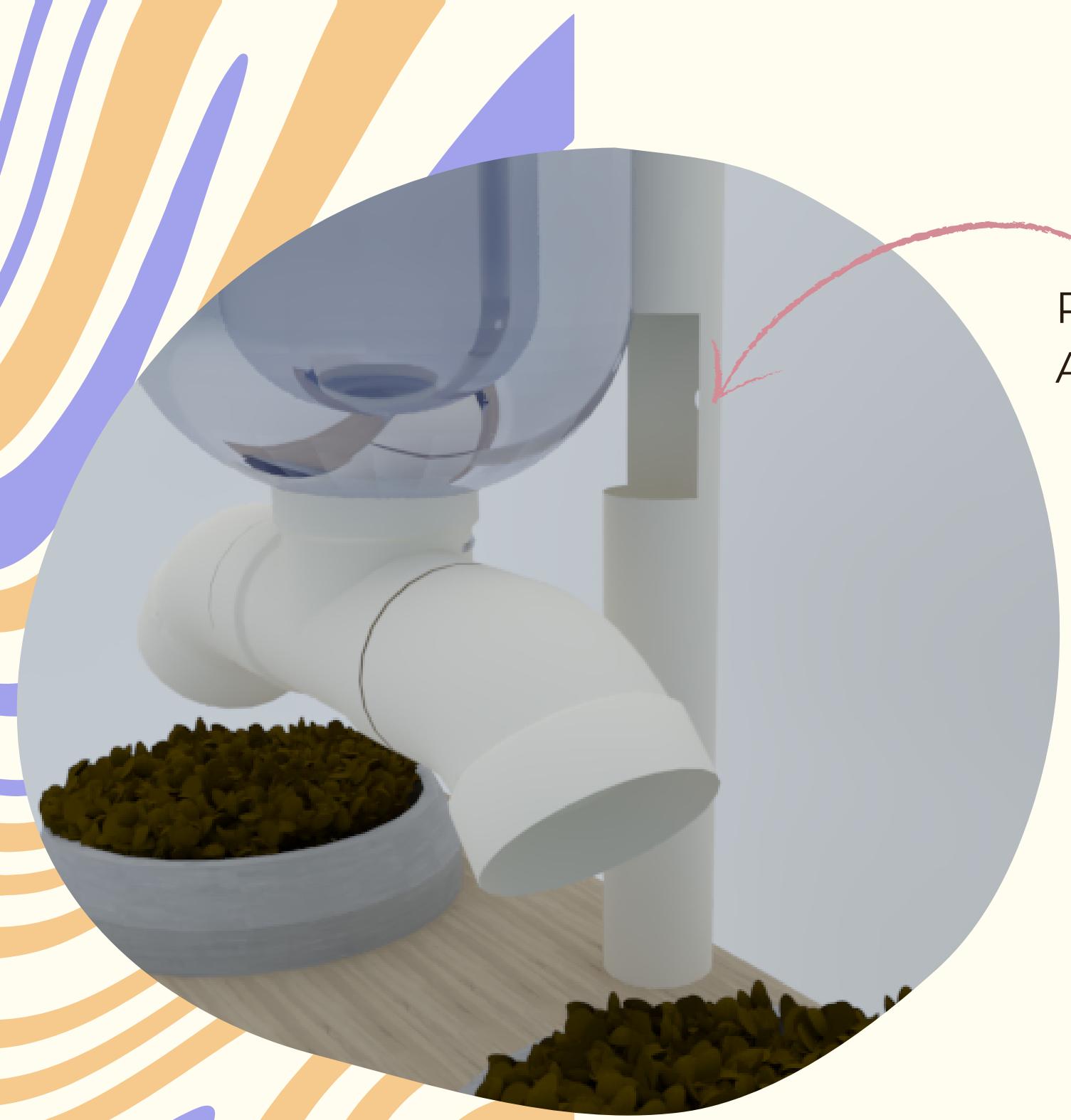


COPIAR Y PEGAR ROTADOR

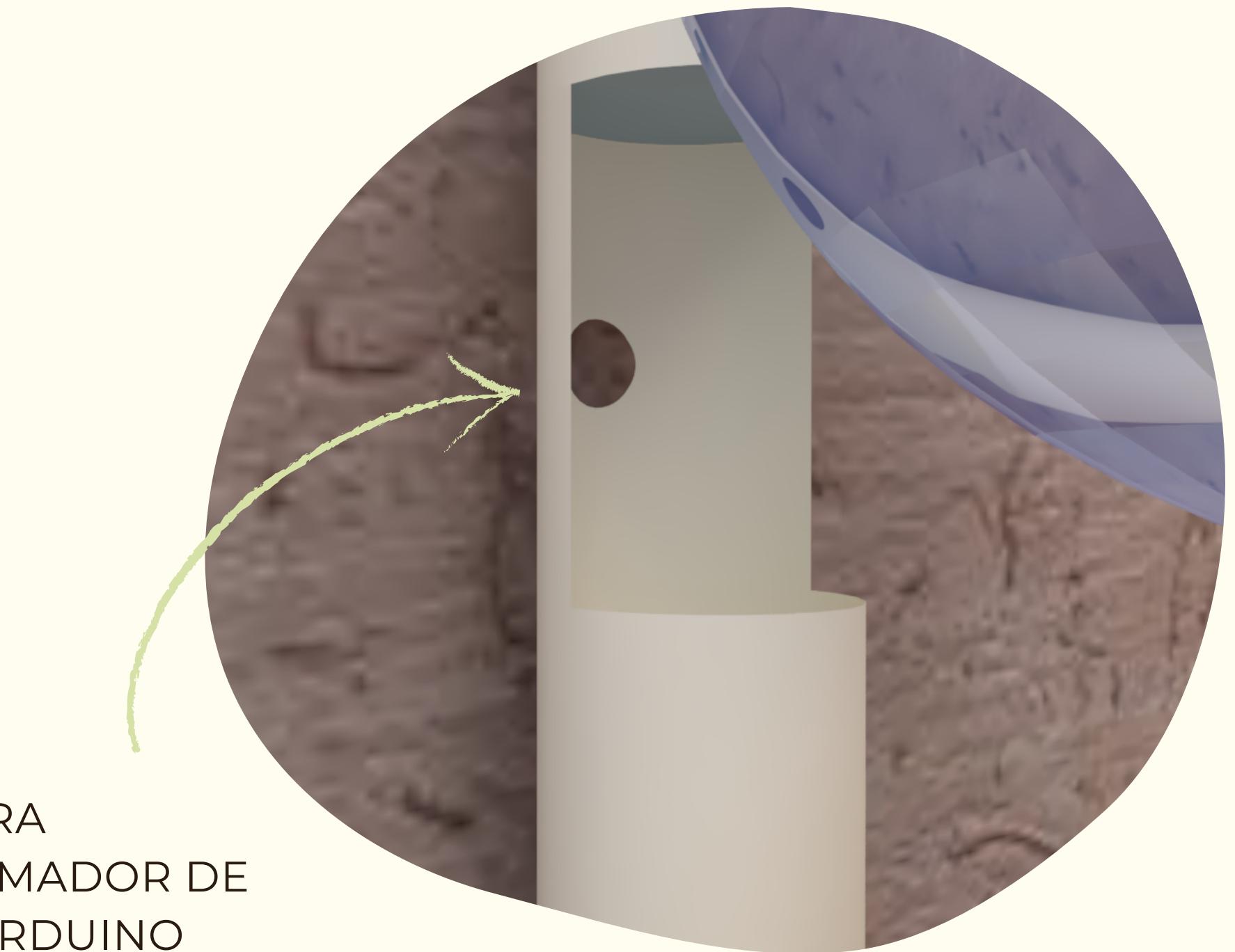


CONFIGURAR HORAS



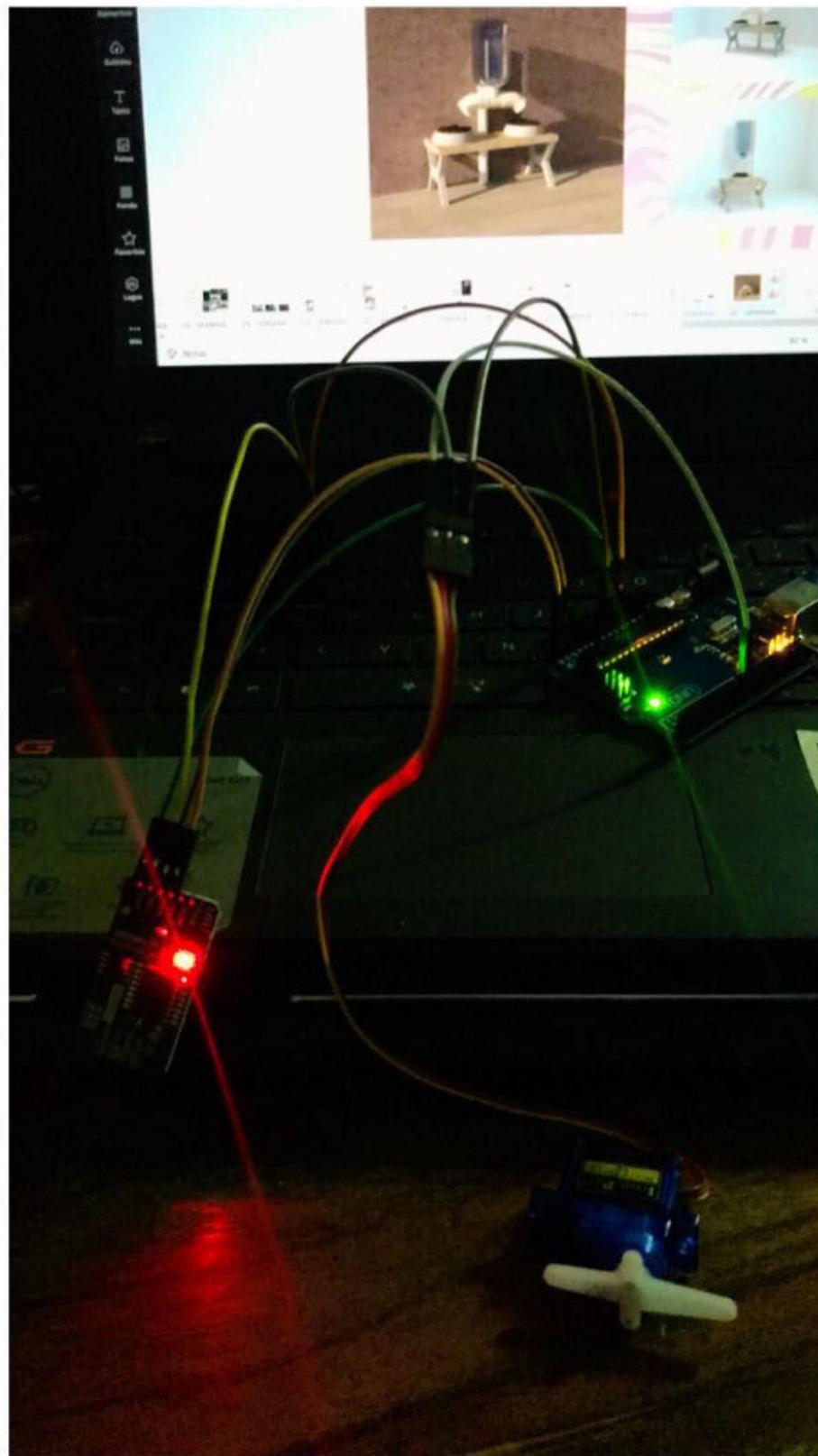


PERFORACIÓN EN TUBO PARA
ALMACENAR COMPONENTES

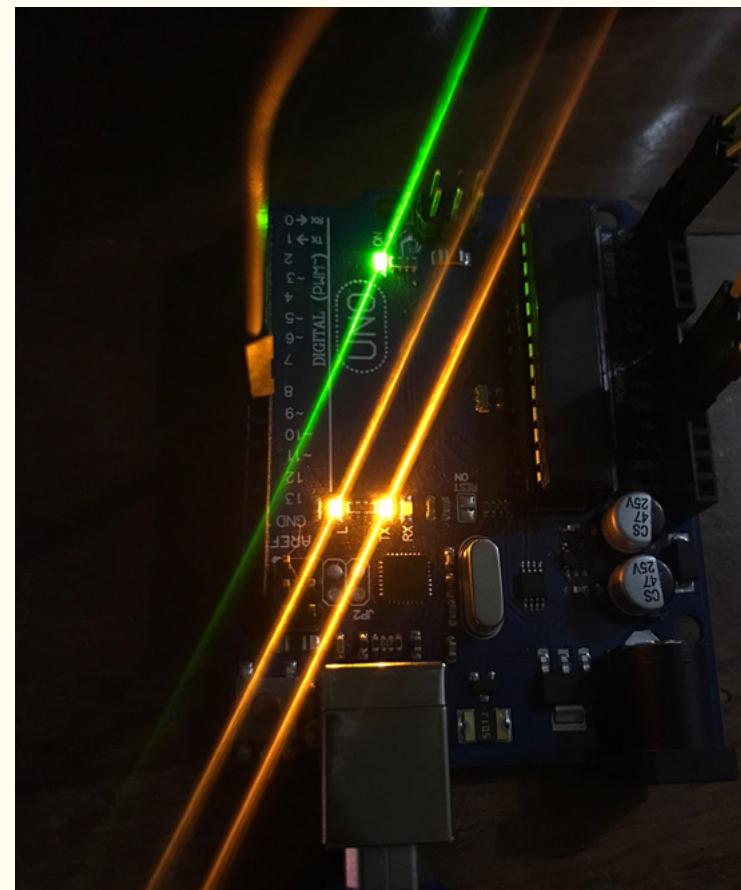


CORTE CIRCULAR PARA
INSERTAR TRANSFORMADOR DE
5V Y ALIMENTAR AL ARDUINO

ARDUINO CONECTADO



- Intento n°1: no gira el servo.
- Intento n°2: anadí la biblioteca DS3231 pero sigue sin funcionar.
- Intento n°3: desconecté todos los cables y los volví a colocar uno a uno, las luces se encendieron como se esperaba, el programa se compiló y se subió correctamente, pero aún el servo motor no funciona.



- Incluí la librería "RTClib.h" para luego cambiar el horario para alimentar a mis mascotas.
- Compilé la programación para verificar que todo ese correctamente.
- Subí la programación en el Arduino.
- Se encendieron las luces "L", "TX" y "RX" mientras iba subiendo luego se mantuvo las dos primeras.

A screenshot of the Arduino IDE interface. The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". The title bar says "sketch_may03a". The code editor contains the following C++ code:

```
Archivo Editar Programa Herramientas Ayuda
sketch_may03a
#include <DS3231.h>
#include <Servo.h>
#include <Wire.h>
#include "RTClib.h"

char *horas[]={
    "07:20",
    "11:30",
    "14:20",
    "18:45",
    "22:20",
};

long nhoras = (sizeof(horas)/sizeof(char *));
bool h12;
bool PM;
RTC_DS1307 Clock;
Servo servoMotor;
byte year, month, date, DoW, hour, minute, second;
```

The status bar at the bottom right shows "Subido".

PROCESO

PROGRAMACIÓN #2

sketch_aug03a Arduino 1.8.3

```
sketch_aug03a$  
Archivo Editar Programa Herramientas Ayuda  
  
#include <Servo.h>  
#include <DS3231.h>  
#include <Wire.h>  
  
char *horas[] = {  
    "16:45",  
    "16:46",  
    "16:47",  
    "16:47",  
    "16:49"  
};  
  
long nhoras = (sizeof(horas)/sizeof(char *));  
bool h12;  
bool PM;  
DS3231 Clock;  
Servo servoMotor;  
byte year, month, date, DoW, hour, minute, second;  
  
void setup() {  
    Wire.begin();  
    Serial.begin(9600);  
}  
void loop() {  
    delay(1000);  
    Clock.getTime(year, month, date, DoW, hour, minute, second);  
    if(deboGirar()==1)  
        ...  
}
```

Subido

El Sketch usa 4818 bytes (14%) del espacio de almacenamiento de programación

sketch_aug03a Arduino 1.8.3

```
sketch_aug03a$  
Archivo Editar Programa Herramientas Ayuda  
  
void loop() {  
    delay(1000);  
    Clock.getTime(year, month, date, DoW, hour, minute, second);  
    if(deboGirar()==1)  
        girar();  
  
    void girar(){  
        servoMotor.attach(9);  
        servoMotor.write(30);  
        delay(575);  
        servoMotor.write(180);  
        delay(1500);  
        servoMotor.detach();  
    }  
  
    int deboGirar(){  
        String h = "";  
        String m = "";  
        for(int i=0;i<nhoras;i++){  
            h= getValue(horas[i], ':', 0);  
            m= getValue(horas[i], ':', 1);  
            hour = Clock.getHour(h12, PM);  
            if(int(hour)==atoi(h.c_str()) && int(minute)==atoi(m.c_str()) && int(second)==0)  
                return 1;  
        }  
        return 0;  
    }  
}
```

Subido

sketch_aug03a Arduino 1.8.3

```
sketch_aug03a$  
Archivo Editar Programa Herramientas Ayuda  
  
int deboGirar(){  
    String h = "";  
    String m = "";  
    for(int i=0;i<nhoras;i++){  
        h= getValue(horas[i], ':', 0);  
        m= getValue(horas[i], ':', 1);  
        hour = Clock.getHour(h12, PM);  
        if(int(hour)==atoi(h.c_str()) && int(minute)==atoi(m.c_str()) && int(second)==0)  
            return 1;  
    }  
    return 0;  
}  
  
String getValue(String data, char separator, int index)  
{  
    int found = 0;  
    int strIndex[] = { 0, -1 };  
    int maxIndex = data.length() - 1;  
  
    for (int i = 0; i <= maxIndex && found <= index; i++) {  
        if (data.charAt(i) == separator || i == maxIndex) {  
            found++;  
            strIndex[0] = strIndex[1] + 1;  
            strIndex[1] = (i == maxIndex) ? i+1 : i;  
        }  
    }  
    return found > index ? data.substring(strIndex[0], strIndex[1]) : "";  
}
```

Subido