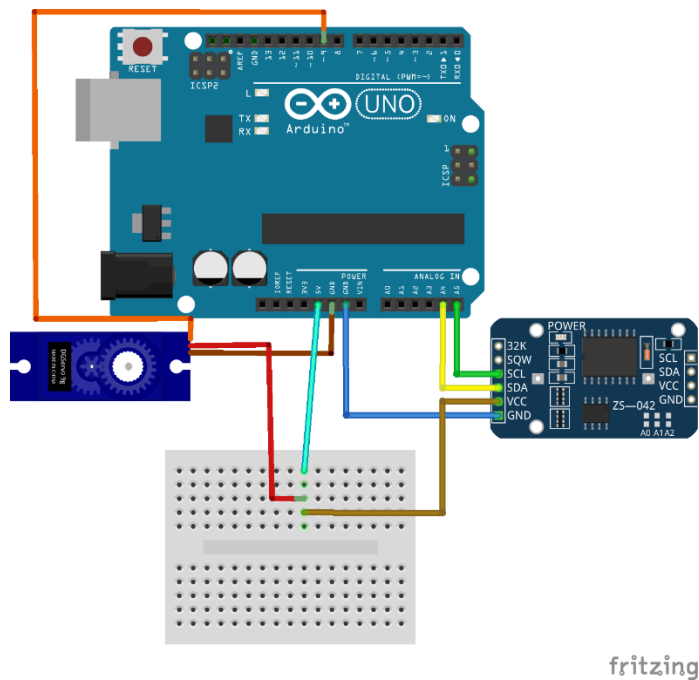


En cuanto a la identificación de los subprogramas (funciones) podemos destacar dos funciones clave.

La primera será leer_hora la cual ira leyendo nuestro sensor de reloj arduino DS3231 de segundo en segundo, conectado al tiempo real del ordenador. Ella leerá el tiempo del ordenador, ese será su argumento, mientras que cuando llegue a la hora programada mandará como salida una instrucción a nuestra segunda función de girar_servo de tal manera que esta recibirá como argumento la orden de la función leer_tiempo y la salida de girar_servo será mover el motor de tal manera que caiga la comida.

Foto del montaje



En cuanto a la programación de nuestro reloj DS3231 en Arduino sería así.

Código para leer la hora.

```

#include <DS3231.h>
#include <Wire.h>

DS3231 Clock;
void setup() {
  // Se inicial la interfaz I2c
  Wire.begin();
  // Se inicia la Comunicación Serial
  Serial.begin(9600);
  //Se establece el modo horario en 12 horas (false = 24h)
  Clock.setClockMode(true);
  //Se establece el año
  Clock.setYear((byte)año);
  //Mes
  Clock.setMonth((byte)mes);
  //Dia de la semana (no lo estoy considerando)
  Clock.setDoW((byte)dia);
  //Día
  Clock.setDate((byte)dia);
  //Hora
  Clock.setHour((byte)hora);
  //Minutos
  Clock.setMinute((byte)minuto);
  //Segundos
  Clock.setSecond((byte)segundo);
}

void loop() {
}

```

Código para rescatar la hora de nuestro ordenador.

```

#include <DS3231.h>
#include <Wire.h>

DS3231 Clock;

byte year, month, date, DoW, hour, minute, second;

void setup() {
  // Se inicial la interfaz I2c
  Wire.begin();

```

```

// Se inicia la Comunicación Serial
Serial.begin(9600);
}

void loop() {

  delay(1000);
  //Se rescata la información
  Clock.getTime(year, month, date, DoW, hour, minute, second);
  //Se imprime
  Serial.print(date, DEC);
  Serial.print("/");
  Serial.print(month, DEC);
  Serial.print("/");
  Serial.print(year, DEC);
  Serial.print(" ");
  Serial.print(hour, DEC);
  Serial.print(":");
  Serial.print(minute, DEC);
  Serial.print(":");
  Serial.println(second, DEC);
}

```

En cuanto a la programación de nuestro servomotor.

```

// Incluimos la librería para poder controlar el servo

#include <Servo.h>

// Declaramos la variable para controlar el servo

Servo servoMotor;

void setup() {

  // Iniciamos el monitor serie para mostrar el resultado

  Serial.begin(9600);

  // Iniciamos el servo para que empiece a trabajar con el pin 9

```

```
servoMotor.attach(9);  
  
}  
  
void loop() {  
  
    // Desplazamos a la posición 0º  
  
    servoMotor.write(0);  
  
    // Esperamos 1 segundo  
  
    delay(1000);  
  
    // Desplazamos a la posición 90º  
  
    servoMotor.write(90);  
  
    // Esperamos 1 segundo  
  
    delay(1000);  
  
    // Desplazamos a la posición 180º  
  
    servoMotor.write(180);  
  
    // Esperamos 1 segundo  
  
    delay(1000);  
  
}
```