

Memoria proyecto Malakabot 2019.

Spanish Bull

Jesús Delgado Ciendones.
Oliver John Garrard Carmona.

Contenido

| | |
|---|---|
| 1. Introducción. | 2 |
| 2. Funcionamiento del proyecto. | 2 |
| 3. Materiales y herramientas. | 2 |
| 4. Programación. | 3 |
| 4.1. Diagrama de flujo. | 3 |
| 4.2. Código para Arduino UNO. | 4 |
| 5. Montaje e instalación del proyecto. | 5 |
| 5.1. Esquema electrónico. | 5 |
| 5.2. Esquema de montaje. | 6 |
| 5.3. Montaje final. | 6 |

1. Introducción.

Esta memoria tratará sobre el proyecto realizado por los alumnos del Grado Superior Mantenimiento Electrónico S21ME Oliver J. Garrard y Jesús Delgado Ciendones, en conjunto con varios alumnos de 2º de Bachillerato de Artes Plásticas.

El proyecto consiste en la cabeza de un toro, con los ojos iluminados en verde, que detecta la presencia humana, y actúa de distintas maneras dependiendo de la distancia de dicha presencia, como cambiar el color de los ojos a rojo o llorar, entre otras acciones.

La realización de este proyecto pretende criticar y representar el daño y malestar que genera la ser humano sobre los animales, y específicamente los toros.

2. Funcionamiento del proyecto.

Para explicar el funcionamiento del proyecto, dividiremos las funciones de nuestro toro en tres apartados distintos, diferenciados según a la distancia que se encuentre la presencia humana de este:

- ***Presencia a mayor distancia de 50cm, o sin presencia:*** En este caso, el toro emitirá luz de color verde en sus ojos, sin realizar ninguna otra acción.
- ***Presencia a menos de 50cm:*** Esta distancia provocará que el toro cambie la luz de verde a rojo, la apertura de su boca, así como la emisión de bramidos constantes.
- ***Presencia a menos de 25cm:*** El toro realizará las acciones del caso anterior, a las que se añadirá un lloro constante.

3. Materiales y herramientas.

Este proyecto ha requerido los siguientes materiales para su constitución:

- Arduino UNO R3.
- Módulo ultrasonido HC-SR04
- Dos servomotores MG996R.
- Dos diodos LED RGB.
- Módulo lector de tarjetas SD DFPlayerMini.
- Altavoz de 3W.
- Batería *Turnigy LiPo Battery*.
- Dos placas de pruebas (tamaño mediano y pequeño).
- Cables “*jumpers*” varios.
- Estaño para soldar.
- Botella de agua y cánulas de distintos tamaños.
- Gomillas varias.
- Planchas de chapón de diversas medidas.
- Hilo de pescar.
- Una pelota de ping-pong.

Y como herramientas se han requerido:

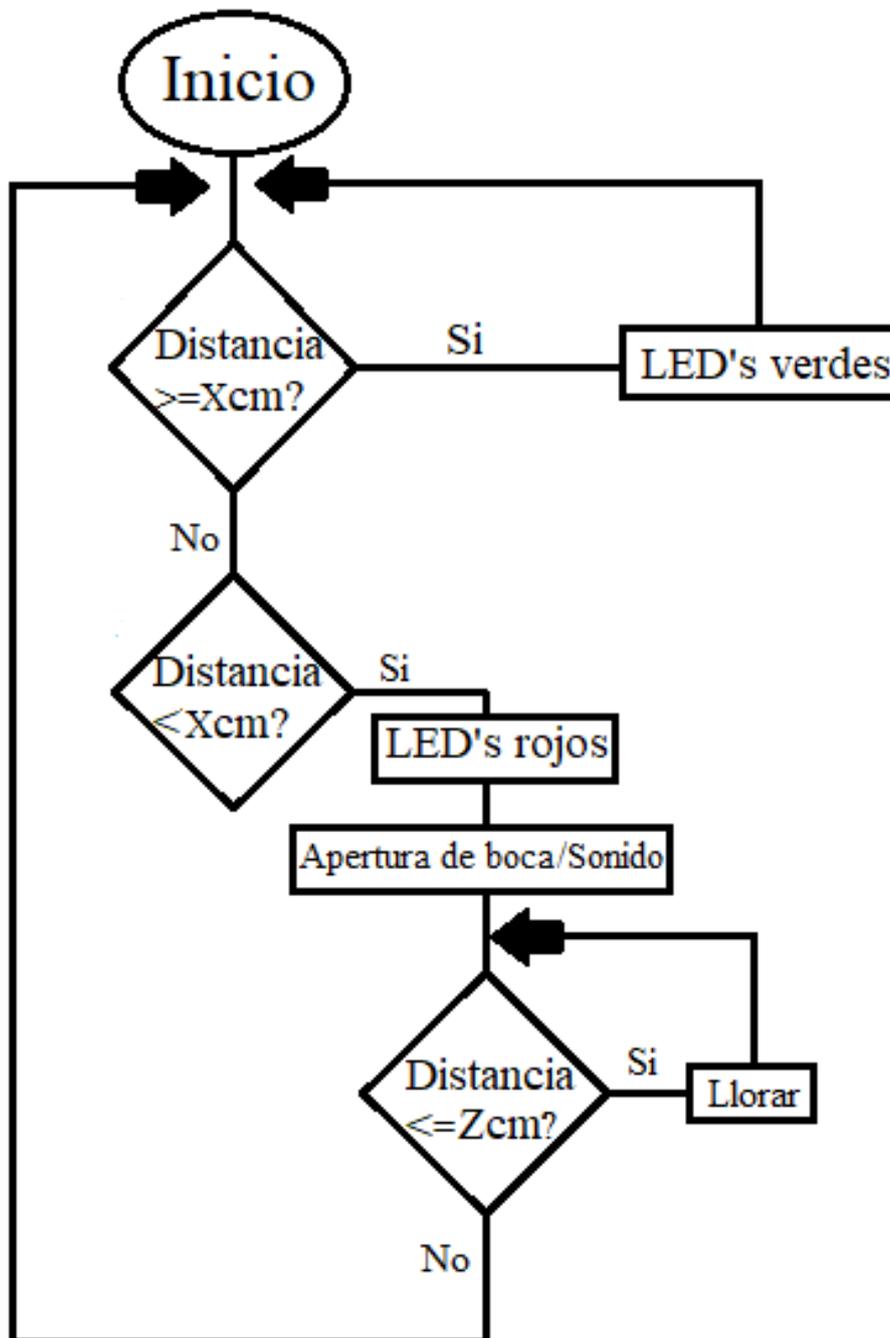
- Ordenador (Arduino IDE).
- Soldador.

- Tijeras.
- Sierra.
- Pistola de silicona.
- Pinzas.

4. Programación.

4.1. Diagrama de flujo.

Para la correcta realización del código, primero resulta necesario tener claro el orden lógico a seguir de nuestro programa, por lo que se ha realiza un diagrama de flujo para mostrar la lógica a seguir:



4.2. Código para Arduino UNO.

```
#include <DFRobotDFPlayerMini.h> //Librería MP3
#include <SoftwareSerial.h> //Librería Puerto Serie
#include <Servo.h> //Librería Servomotores

SoftwareSerial mySoftwareSerial(7, 8); // Asignación de RX y TX en la placa.
DFRobotDFPlayerMini mp3; // Creación de objeto mp3 tipo DFRobotDFPlayerMini
Servo motor_0; Servo motor_1; // Creación de objetos motor_0 y motor_1 tipo Servo
int dist; int duration; int time_out=6000; // Creación de variables para el ultrasonido
int trig=3; int echoe=4; // Asignación de pines 3 y 4 de la placa para variables trig y echoe
int led_vl=5; int led_rl=6; // Asignación de pines 5 y 6 de la placa para variables led_vl y led_rl

void color (int v_l, int r_l) { // Función para el uso de led's
  analogWrite (led_vl, v_l);
  analogWrite (led_rl, r_l);
}

void sonido (int full) { // Función para la reproducción o pausa del DFPlayer mini
  if (full==1) mp3.play(1); //nº del audio
  if (full==0) mp3.stop();
}

int ultrasonido () { //Lectura de distancia por el ultrasonido; la devuelve en cm
  digitalWrite(trig, LOW);
  delayMicroseconds (2);
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  duration = pulseIn (echoe,HIGH,time_out);
  if ( duration == 0 ) {
    duration = time_out; }
  dist = duration /29 / 2;
  return dist;
}

void setup() {
  Serial.begin(115200); //Establece puerto serie
  mySoftwareSerial.begin(9600); //pines placa: Ultrasonido(3,4); LED's (5,6); Sonido(7,8); servo(10,11);
  mp3.begin (mySoftwareSerial);
  mp3.volume(30); //Establece el volumen de DFPlayerMini
  mp3.play(0); //Establece el estado inicial de DFPlayerMini

  motor_0.attach(10);motor_1.attach(11); //pines 10 y 11 de la placa para los servomotores
  motor_0.write(0);motor_1.write(90); //puesta a 0 del primer motor; 90º para el segundo

  pinMode (trig, OUTPUT); digitalWrite (trig, LOW); //pin de trig como salida, inicializado a nivel bajo
  pinMode (echoe, INPUT); //pin de echoe como entrada
  pinMode (led_vl, OUTPUT); //pin de led_vl como salida
  pinMode (led_rl, OUTPUT); //pin de led_rl como salida
}

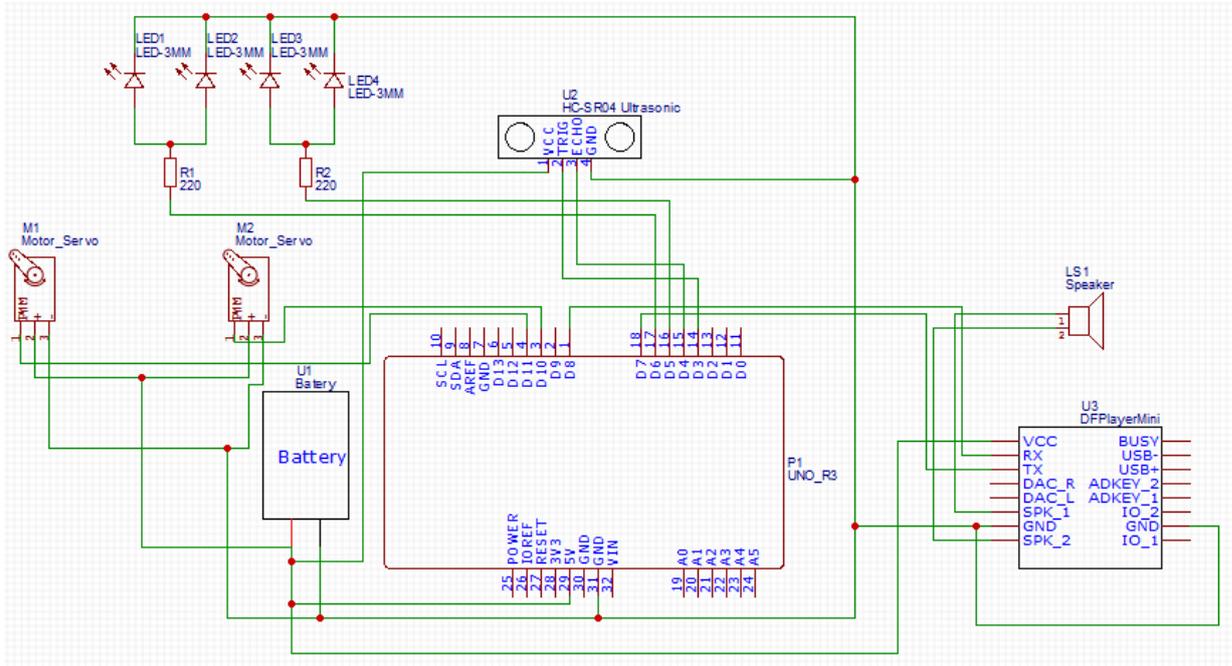
void loop() {
  if (ultrasonido())>=50) { //comprueba si la distancia es mayor o igual a 50cm
    color (255, 0); //ajusta ambos led a color verde
    motor_0.write (0); //ajusta el primer servo a 0º; boca cerrada
    sonido (0); //ajusta el DFPlayer en stop
    delay (2000); //espera de dos segundos
  }
  if (ultrasonido())<50) { //comprueba si la distancia es menor a 50cm
    color (0, 255); //ajusta ambos led a color rojo
    motor_0.write(35); //ajusta el primer servo a 35º; apertura de boca
    sonido (1); //ajusta el DFPlayer en reproducción
    delay(2000);
    while (ultrasonido())<=25){ //Comprobación si está a menos de 25cm o igual; acción inferior continua mientras esté dentro del rango
      motor_1.write(90);delay(300); //El motor dos se mueve 30º y vuelve a su posición en 0.6 segundos; espera un segundo
      motor_1.write(120);delay(300);
      delay(1000);
    }
  }
}
```

Como se puede apreciar, el código en sí no posee gran complejidad, pues la mayoría de líneas consisten en la configuración de los distintos módulos, y dado que viene con gran extensión de comentarios, no realizaremos ninguna aclaración más en este documento.

5. Montaje e instalación del proyecto.

5.1. Esquema electrónico.

Continuaremos la memoria mediante la aportación del esquema electrónico referente al proyecto, y adaptado a la programación del Arduino UNO:

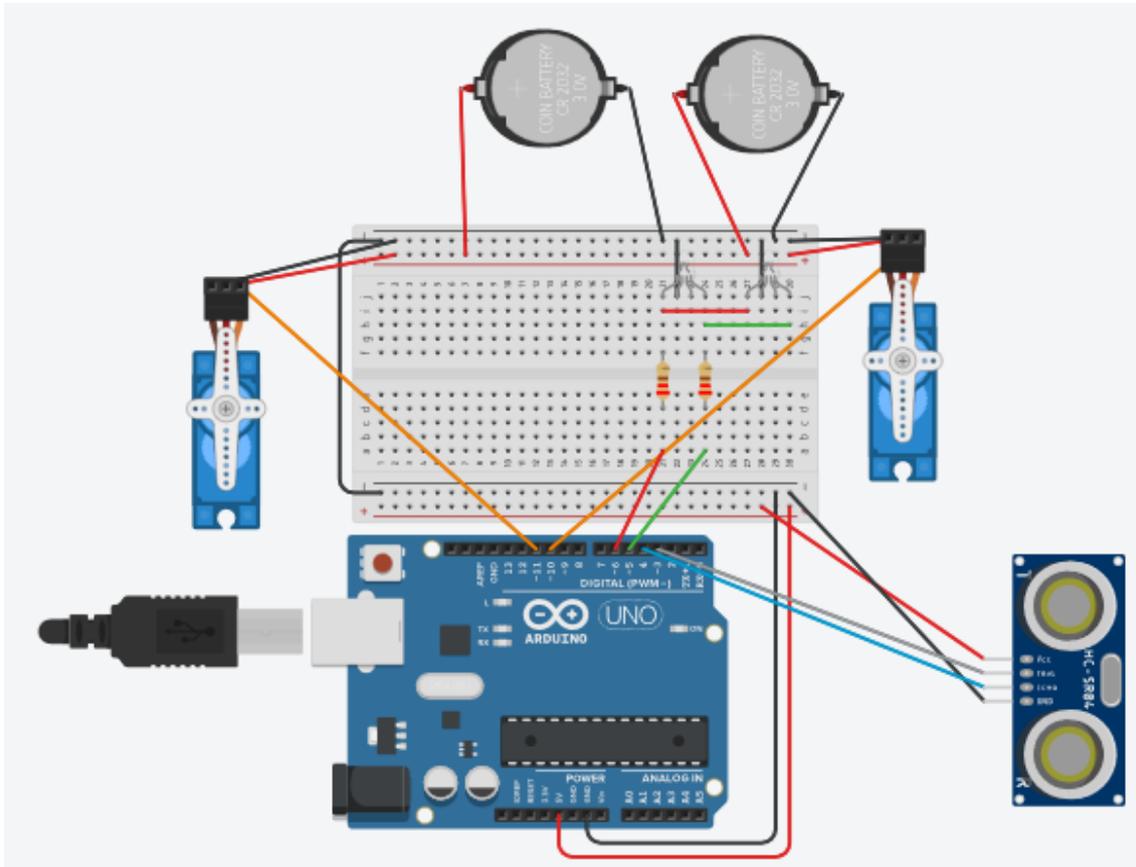


Aquí resaltaremos varios datos importantes:

- Nos podremos fijar en la aparición de una batería externa, que utilizaremos en la alimentación de los distintos módulos.
- También observamos como en cada salida digital del Arduino dedicada a un LED de color, ya sea el rojo o verde, colocamos dos en paralelo, con el fin de ahorrar pines y simplificar la programación.
- En el módulo DFPlayerMini existen varias salidas para altavoces, pero utilizamos SPK_1 y SPK_2 debido a que nos bastará con sonido mono y en un altavoz con menos de 3Watts.

5.2. Esquema de montaje.

A continuación, se incluye un esquema práctico para poder visualizar el montaje con una placa de pruebas, y aunque no se ha podido añadir en este el módulo DFPlayerMini debido a que no estaba incluido en la librería de *TinkerCAD*, esperamos que pueda resultar al menos como una mera orientación teórica:



Para el montaje final, utilizaremos otra placa de pruebas de menor tamaño para poder conectar el módulo lector de tarjeta SD, al que conectaremos el altavoz, los pines 7 y 8 de la paca Arduino, así como tierra y voltaje.

El uso de ambos motores, correspondería al movimiento de la boca y el goteo para las lágrimas respectivamente.

5.3. Montaje final.

Una vez simulado, montado y probado el circuito en distintas ocasiones, procederemos a realizar su montaje final dentro de la cabeza del toro, realizada por los alumnos de 2º de Bachillerato de Artes Plásticas.

Para esto, primero decidimos soldar cableado en cada patilla de los LED's RGB, el sensor ultrasonido y el altavoz, debido a que su colocación en los ojos, nariz y boca (respectivamente) necesitaría cableado para poder llegar hasta donde sería colocada la placa de pruebas, justo en al final de la cabeza del toro. Los LED's irán encajados en perforaciones realizadas sobre medias pelotas de ping-pong.

Tras tener el cableado listo, continuamos cortando el chapón en las cuatro planchas con las distintas medidas del hueco en la cabeza del toro, y realizando en la plancha inferior

Memoria Malakabot Spanish Bull

un hueco para encajar el servomotor de la boca, y otro en la plancha superior que será utilizado para introducir la cánula perteneciente al goteo. Pegaremos las planchas realizando un cubo, y pondremos refuerzos en las esquinas, en el interior, en caso de que los servomotores ejerzan demasiada fuerza sobre la estructura.

Introduciremos el sensor a través del morro, los ojos a través de sus perforaciones metiendo el cableado del LED por dentro, y el altavoz justo al final de la boca; encajamos las piezas de chapón ya pegadas en el hueco.

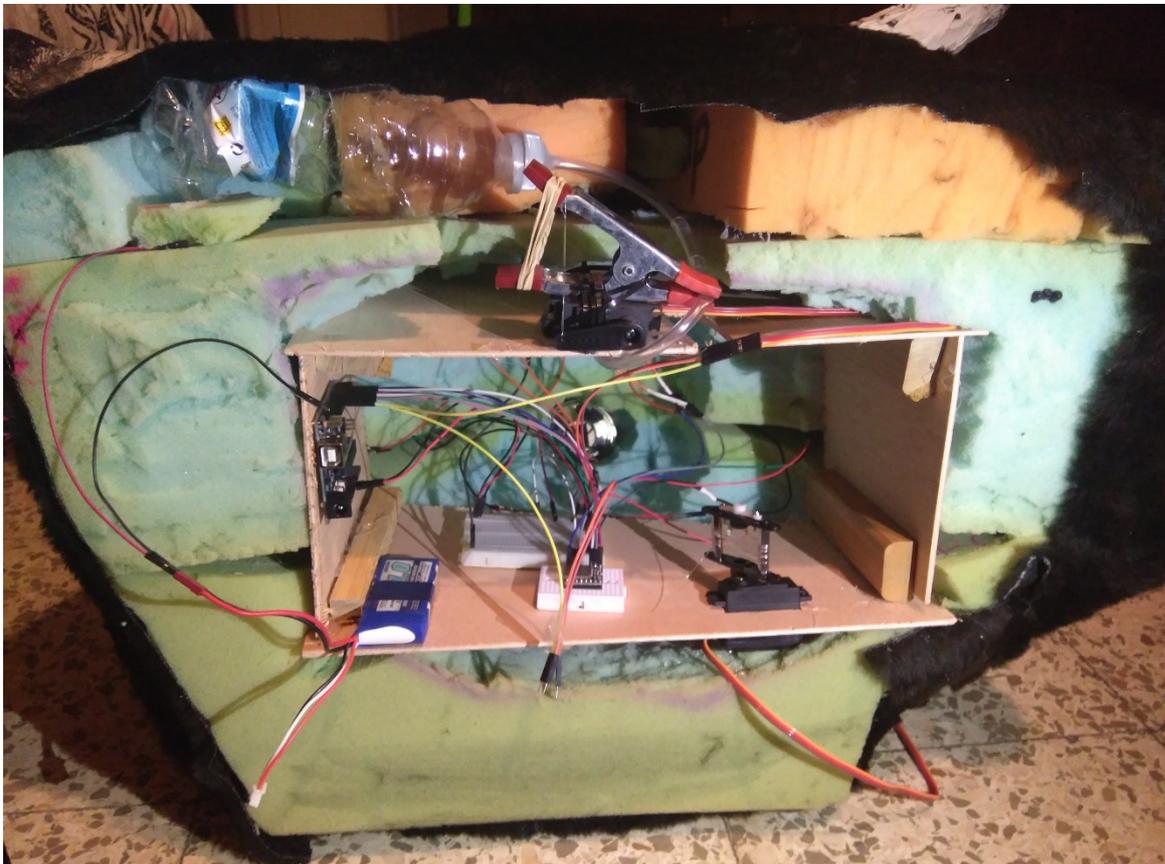
En la parte inferior de la boca engancharemos el hilo de pescar, que enrollaremos en el servomotor encajado en el chapón inferior y la batería, y pegaremos la placa de Arduino en el lateral izquierdo; las placas de pruebas ambas al lado del servomotor.

Al segundo servomotor, lo engancharemos con bridas de plástico a un lado de las pinzas, junto con varias gomillas para que pueda tirar del otro lado sin problemas, y así abrir y cerrar. Lo colocaremos justo encima del chapón superior, en un hueco sin gomaespuma preparado para la botella y las pinzas.

Colocaremos la botella, con el tapón perforado para introducir la cánula en “Y”, en su hueco preparado, y pasaremos su cánula por la pinza, el hueco serrado en la plancha superior y finalmente ambos ojos. Pegaremos la cánula a la plancha superior.

Con todo mecánicamente preparado, sólo queda realizar el conexionado de la placa Arduino con las distintas partes de nuestro circuito en las placas de pruebas (no se ha podido realizar en placas preperforadas por falta de tiempo).

A continuación, se incluyen dos fotos del proyecto concluido:



Memoria Malakabot Spanish Bull

