DISPOSITIVO DE APRENDIZAJE BRAILLE

AUTOR: Celia María Romero Vílchez

TUTOR: Enrique Norro Gañán

GRADO: Grado Superior en Mantenimiento Electrónico

CURSO: 2021/22

ÍNDICE

Introducción **3**
Justificación **3**
Objetivos **6**
Desarrollo **6**
Montaje en protoboard 6
Programación 7
Diseño PCB 8
Creación PCB mediante insolado 9
Soldadura componentes 12
Mecánica 13
Cálculos **14**
Diseño **16**
Esquemático 16
PCB 17
Código **18**
Cuerpo 18
Inicialización DFPlayer 21
Juego letras 22
Juego palabras 23
Introducir palabra 24
Introducir letra 29
Banco de palabras 34
BOM **36**
Anexos **37**
Módulo cargador TP4056 37
Arduino Nano 38
DFPlayer Mini 38
HW-106 DC-DC 39
Bibliografía **40**

1. Introducción

Este proyecto trata el desarrollo para la creación de un dispositivo electrónico que facilite el aprendizaje del abecedario Braille, orientado sobre todo a niños con discapacidad visual que se estén iniciando en la lectura. Se pretende que su manejo y composición sea lo más simple posible.

2. Justificación

La creación de este proyecto comenzó durante el curso académico 2020/21 a raíz de una necesidad real con la que se encuentran los profesores de alumnos con discapacidad visual.

Actualmente vivimos en un mundo tecnológico, en el que cualquier ámbito ha sido reforzado, si no sustituido, por un aparato electrónico, o por las denominadas TICS, herramientas informáticas que procesan, presentan, recuperan y sintetizan información de diferentes formas. Como claro ejemplo de su implementación en el día a día tenemos la enseñanza, donde es casi inimaginable concebirla sin un pc que apoye cada actividad realizada. Su uso en el proceso de enseñanza-aprendizaje, aportan experiencias beneficiosas al alumnado que no se pueden obtener con otros tipos de recursos, como son:

- Ofrecen a los niños/as conocimientos y destrezas básicas sobre la informática, que les servirá para adaptarse a la sociedad actual.
- Uso didáctico para hacer más fácil los procesos de aprendizaje.
- Motivación: los niños/as casi siempre asocian las nuevas tecnologías con actividades divertidas, lo que potencia su interés y el saber más.
- Aprendizaje autónomo: con programas informáticos adecuados, que sean capaz de dar respuestas al alumnado de manera rápida e inmediata, los niños/as pueden conocer sus errores a tiempo real y poder modificarlas.

Sin embargo, dentro de la enseñanza primaria para personas con discapacidad la aplicación de avances tecnológicos está bastante estancada, se siguen recurriendo a los mismos métodos que años atrás. Y esto, aunque siga tratándose de un método válido para el aprendizaje, es comprensible que un alumno se sienta desplazado o falto de interés cuando el resto de sus compañeros tienen la posibilidad de usar TICs sin ningún impedimento. Por eso, esto puede hacer que su autoestima y sensación de competencia suba a la par que aumenta sus posibilidades de integración, pues pueden comunicarse con sus compañeros, hablar de las mismas cosas, aprender de ellos y enseñarles también.

Actualmente existen dispositivos que fomentan estos puntos, los cuales se presentan a continuación, pero que por desgracia no son muy accesibles para los profesores, ya sea por precio o disponibilidad. Con este proyecto no se pretende crear un aparato totalmente nuevo con el objetivo de sacarlo al mercado, sino crear uno basado en estos para dar una solución y ayudar en la medida de lo posible a gente cercana.

<u>Método Braille</u>

En 1825, Luis Braille ideó su sistema de puntos en relieve: el sistema braille, también conocido como cecografía, aporta a las personas ciegas una herramienta válida y eficaz para leer, escribir, acceder a la educación, a la cultura y a la información sin necesidad de ver, guiándose solo por el tacto.

Se trata de un sistema que parte de seis puntos en relieve, distribuidos en 2 columnas de 3 puntos. La combinación de los seis puntos permite obtener 64 combinaciones diferentes, incluyendo la que no tiene

ningún punto, que se utiliza como espacio en blanco para separar palabras, números, etc. La presencia o ausencia de puntos determina de qué letra se trata.

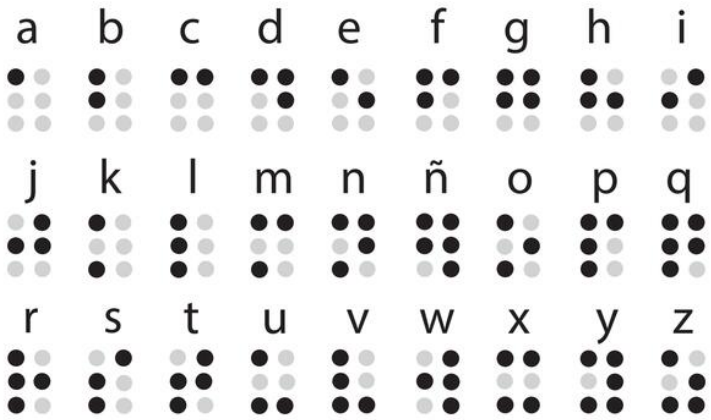


Figura 1

También existen signos especiales en Braille para la taquigrafía, para los caracteres especiales típicos de cada idioma, para los signos matemáticos.

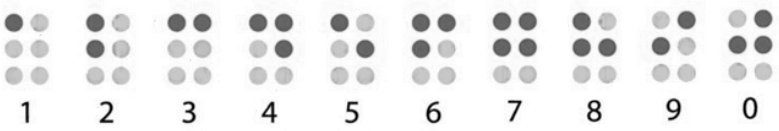


Figura 2

Para empezar a utilizar el método Braille, es necesario que el niño tenga buena orientación espacial y un tacto entrenado. Debe dominar los conceptos arriba, abajo, derecha, u otros, el conocimiento de figuras geométricas elementales y siluetas sencillas y el conocimiento de los signos Braille. La escritura del Braille puede hacerse de dos formas: manual o utilizando un teclado. La lectura se realiza primero colocando el libro en horizontal. Luego se colocan los dedos sobre el renglón, haciendo una pequeña presión. No se puede palpar, y hay que utilizar ambas manos independientemente.

Braillin

A finales del año 2004 la Once lanzó un juguete con formato de un muñeco, para que sea utilizado por cualquier niño o niña, con o sin discapacidad visual. Este juguete constituye la primera iniciativa mundial de este tipo. Familiariza a los más pequeños con el sistema de lectura y escritura, y representa un recurso didáctico de enorme valor para las aulas o para el juego en el hogar. Une el aprendizaje a la diversión.

Figura 3

Braillin fue creado en Argentina por una profesora de niños con discapacidad visual, con el que consiguió un premio en el Concurso de Investigación Educativa sobre Experiencias Escolares. Está especialmente hecho de un material adecuado para la educación infantil, y el primer ciclo de la Educación Primaria, es decir, para niños de 3 a 10 años de edad. Viene acompañado de una Guía Didáctica en la que se explica su utilidad y puede ser adquirido en cualquier centro de la Organización.

Aunque fue diseñado especialmente para el aprendizaje de este sistema, también puede ser utilizado por cualquier niño. Aspectos que favorece:

- Enseña el alfabeto Braille.
- Integra a las personas del entorno de la persona con discapacidad.
- Favorece la empatía y la autoestima de los niños con discapacidad visual.
- Beneficia a la autonomía personal de los niños pudiendo autoevaluarse y corregir sus errores en el momento.
- Desarrolla el tacto, con diferentes texturas, presión manual y refuerzo del tono muscular.
- Instruye acerca de las partes del cuerpo y de la cara.
- Favorece a la expresión, y comunicación.

<u>Braillear</u>

El punto de partida para nuestro prototipo procede del proyecto “Braillear”, que desarrollaron cinco estudiantes para su tesis de Diseño Industrial en la Universidad de Buenos Aires.

Braillear desarrolló un teclado que sirve para el aprendizaje de letras y números, pero también para escribir textos largos, permitiendo, por ejemplo, tomar de apuntes. La impresora posibilita imprimir en braille y traducir textos convencionales.

Figura 4

3. Objetivos

El objetivo de volver a fabricar el mismo dispositivo que el año anterior, es estudiar las necesidades que quedaron por cubrir y tratar de implementarlas en la medida de lo posible.

Tras una breve charla con algunos profesores, estos mostraron los resultados de los proyectos pasados y como habían sido las respuestas de los alumnos invidentes ante cada uno. Así, estudiando los pros y contras de cada uno se realiza una lista con aspectos a mejorar:

- Ampliar el rango de aprendizaje con palabras simples o números.
- Implementación de componentes y cableado en una PCB para reducir tamaño y probabilidad de fallo en conexiones.
- Rediseño de la caja.
    - Diferenciar bien los seis botones que representan los seis puntos Braille de los de verificación del aparato.
    - Diseño abatible para adaptarse a distintos métodos de escritura.
- Control de volumen.
- Salida jack para auriculares.

4. Desarrollo

4.1. Montaje en protoboard

Recopilando los proyectos realizados el año anterior, se montan los componentes en una protoboard y se prueba el código, el cual permite la práctica de letras en Braille.

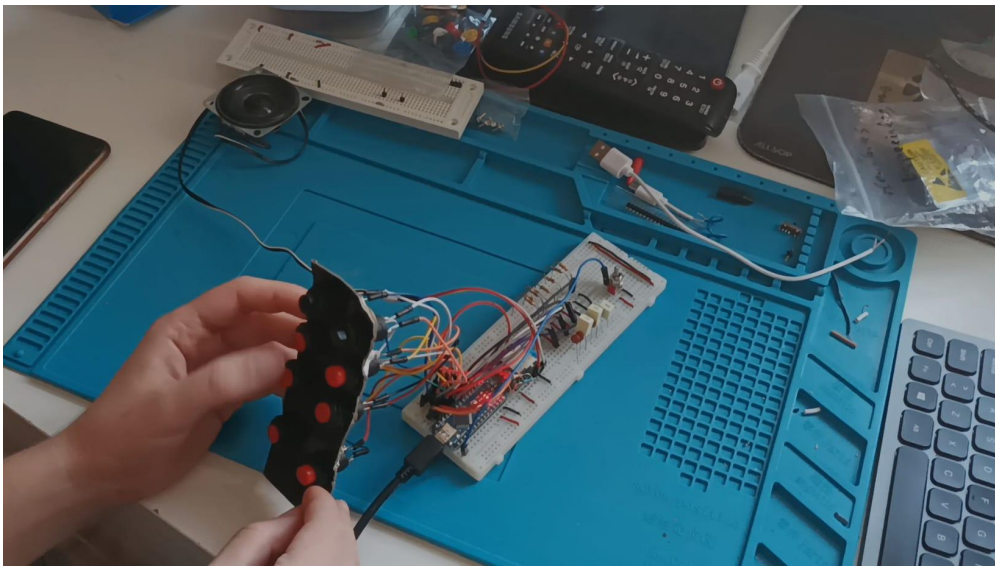


Figura 5

4.2. Programación

Una vez montado y verificando su funcionamiento, se implementan nuevas funcionalidades al código. En este caso se han introducido palabras simples de tres letras, compuestas con las todas las vocales y las consonantes *b*, *l*, *c*, *m*, y *p* ,que corresponden a la signografía más fácil de percibir para aquellos que están comenzando a aprender Braille.

Otra implementación en el código es la opción de control de volumen y el recuento de veces que se introduce una letra o palabra incorrectamente, algo que puede favorecer a que el niño no se sienta bajo presión al fallar y pueda pasar a otra opción.

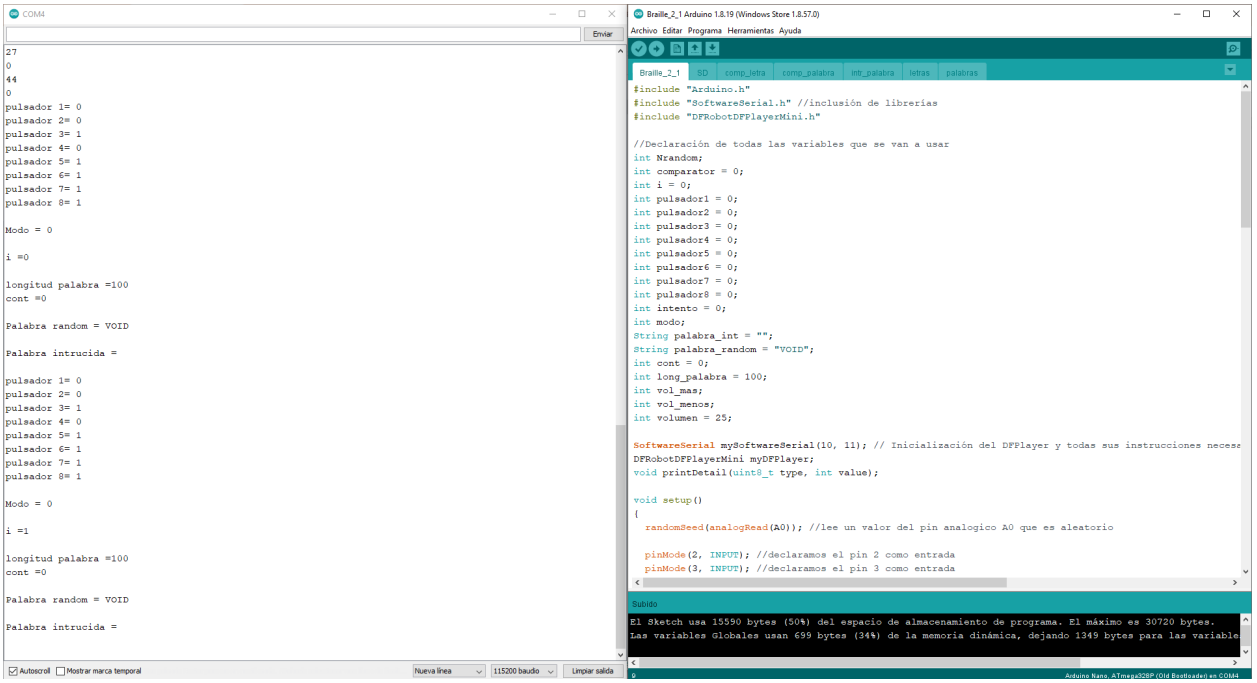


Figura 6

4.3. Diseño PCB

A nivel hardware, se ha introducido un sistema antirrebote a todos los pulsadores, para evitar falsos contactos, favoreciendo así la experiencia de la persona que utilice el dispositivo. Obtener falsa información debido a un error de diseño puede hacer que el juguete no cumpla su objetivo, que es ayudar al aprendizaje de letras y palabras de una manera sencilla y amena.

En el nuevo diseño también se incluyen las conexiones para el cambio de modo de juego y el control de volumen.

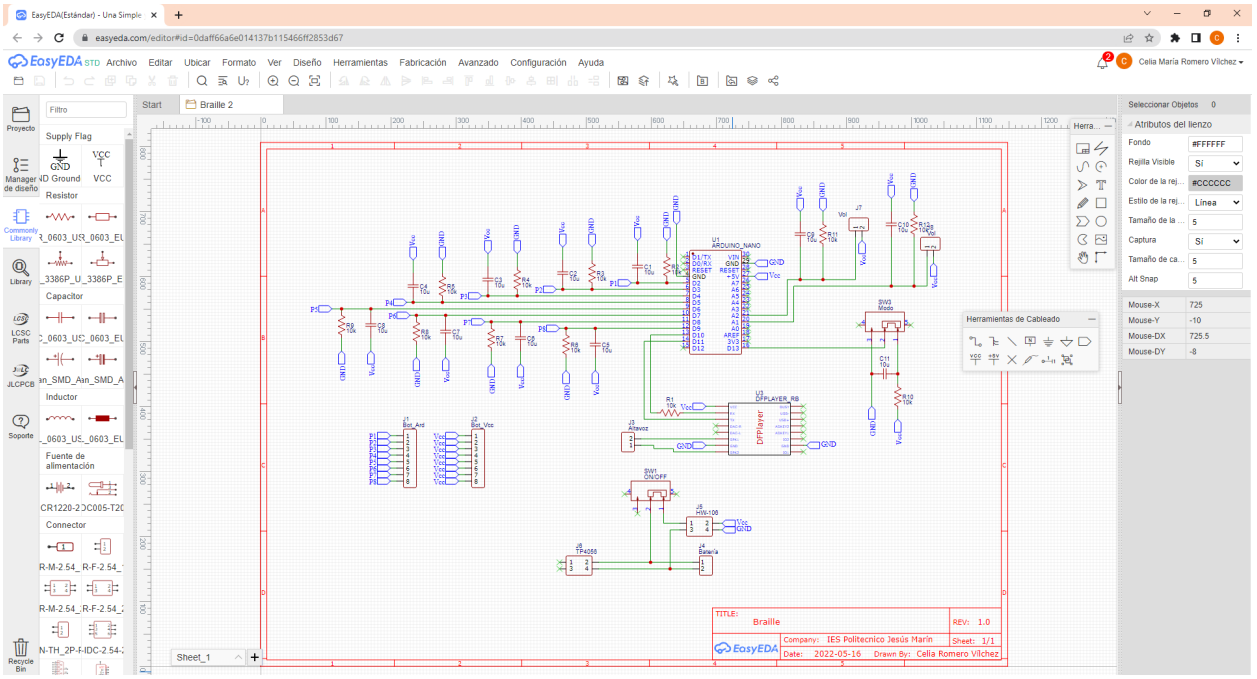


Figura 7

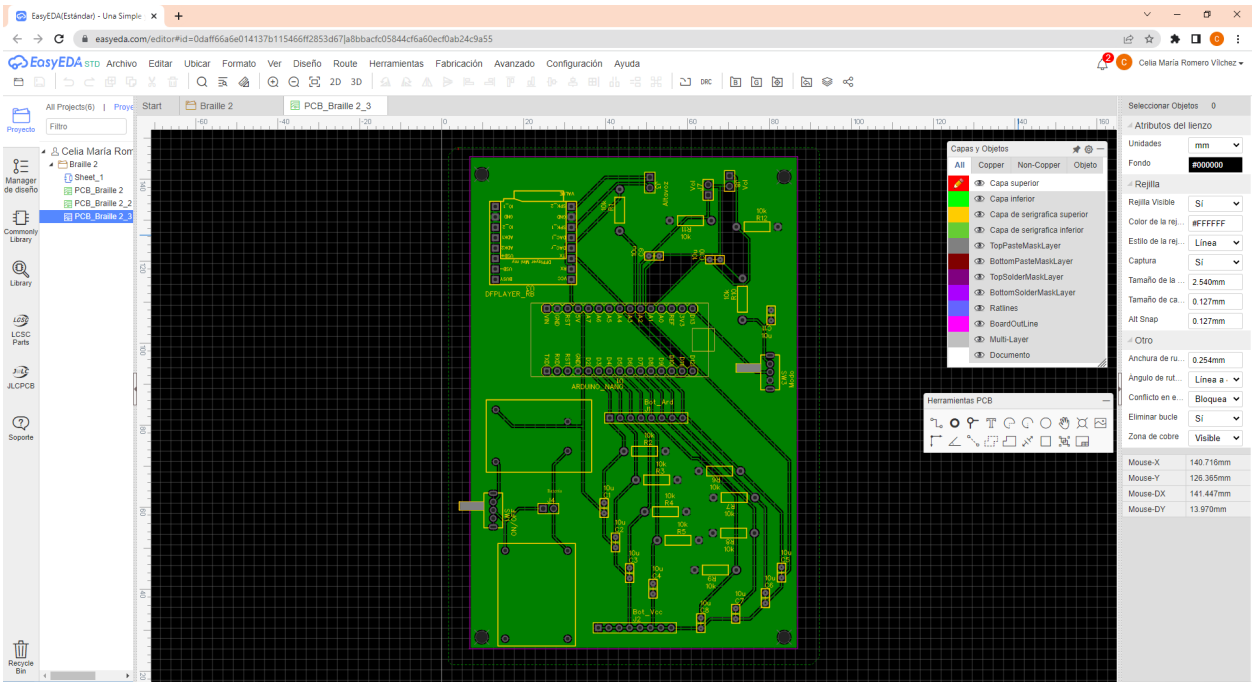


Figura 8

4.4. Creación PCB mediante insolado

- Impresión fotolito

Se imprime, con una impresora láser, el diseño del fotolito en una lámina de acetato transparente. De esta forma se puede colocar sobre la placa para su posterior insolado.

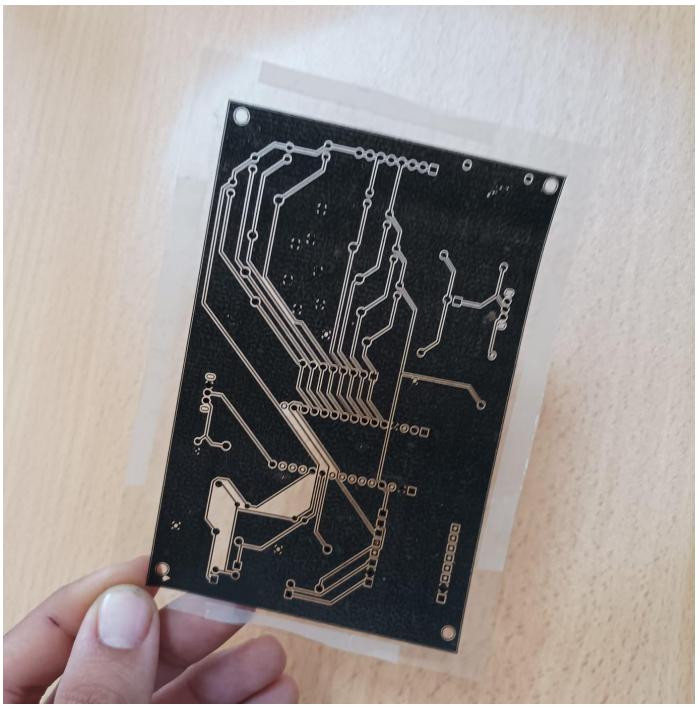


Figura 9

- Colocación fotolito en placa

Se elige una placa de resina fotosensible positiva.

Procedimiento:

1) Se retira la capa protectora, evitando la luz directa sobre la placa
2) Se coloca el fotolito, fijándolo con cinta adhesiva transparente para no perjudicar el proceso de insolado.

- Insolado

Se introduce la placa en la insoladora aproximadamente durante 5 minutos. Al ser una placa positiva, la resina que queda expuesta se reblandece, mientras que la resina sobre las pistas que han quedado cubiertas por la tinta del fotolito permanecerán duras.

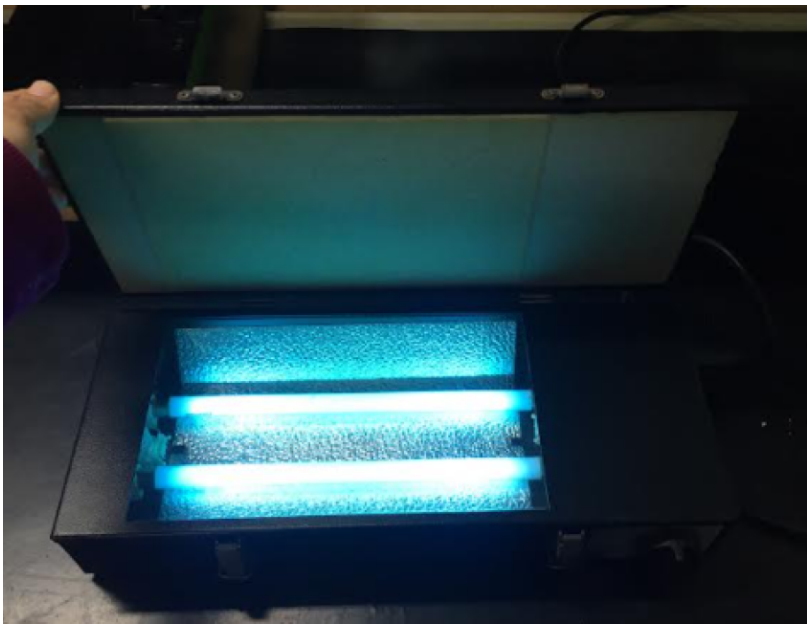


Figura 10

- Revelado

Para eliminar la resina sobrante y dejar solamente cubiertas las pistas para su posterior atacado, se introduce la placa en una solución de sosa caustica disuelta en agua (12gr/l).

El tiempo de revelado dependerá del tiempo que haya permanecido en la insoladora y de la concentración de la solución.

Figura 11

- Atacado

Se sumerge la placa en una solución corrosiva de cloruro férrico activado en agua caliente (2 partes de cloruro por 1 de agua). La solución se encarga de desprender el cobre de la placa, dejando al descubierto la base de fibra de vidrio. Las pistas cubiertas por la resina permanecen intactas.

La velocidad de atacado dependerá de la pureza de la solución y de la temperatura.

Figura 12

Desgraciadamente, se realizaron dos intentos fallidos, teniendo como resultado una primera placa cuyos tiempos de insolado y proporciones en el revelado no fueron los apropiados, por lo que la capa protectora fue dañada.

En el segundo caso, por algún motivo el comportamiento de la resina no fue el esperado, y aún después de ajustar los tiempo y proporciones de componentes para su revelado, la resina no se desprendía de manera uniforme, llegando el cloruro férrico a comerse las partes de cobre que no correspondían.

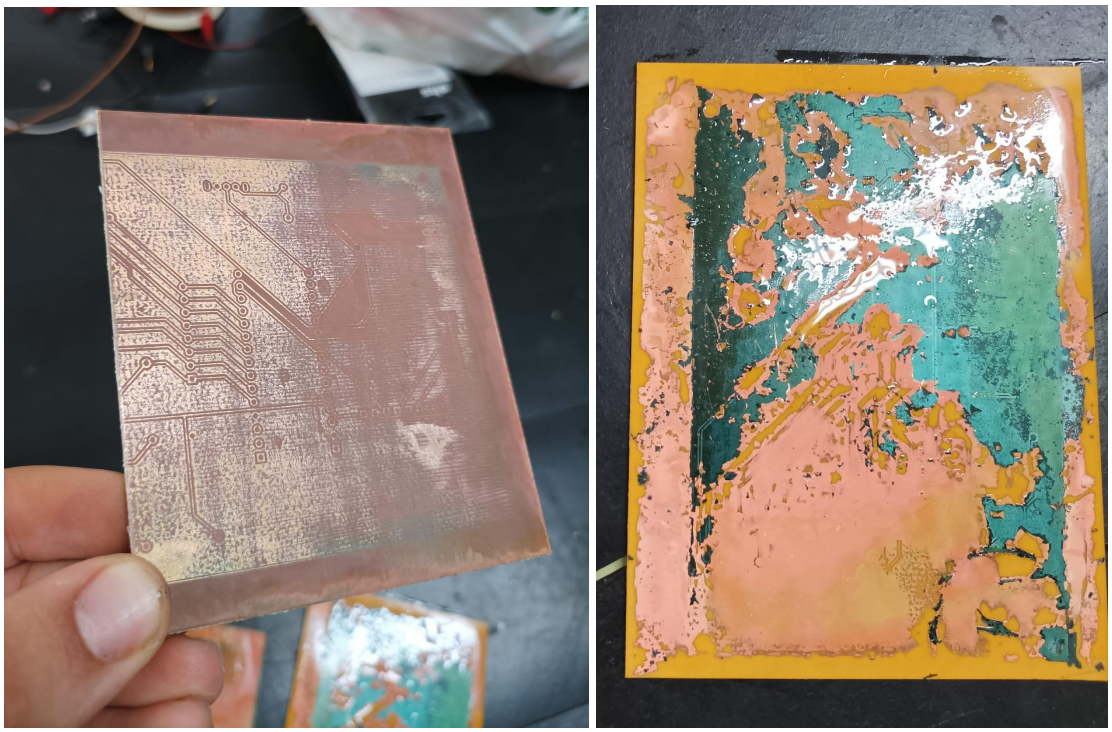


Figura 13

4.5. Soldadura componentes

Debido a la imposibilidad de realizar la PCB se procede a soldar los componentes en una placa perforada, tomando como referencia la distribución diseñada originalmente.

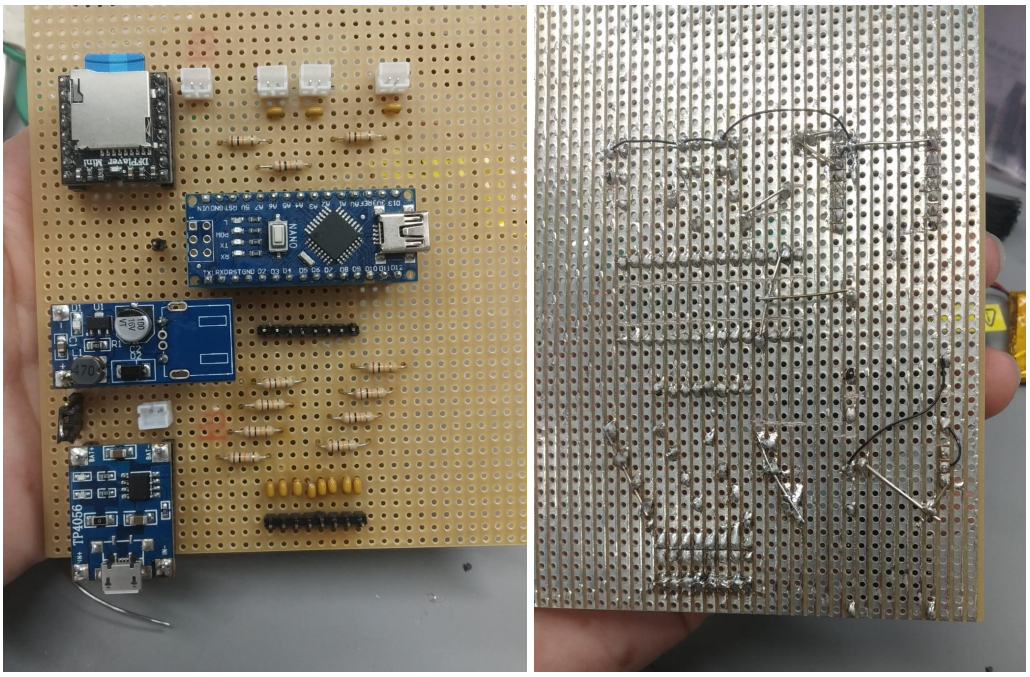


Figura 14

4.6. Mecánica

Se elige una caja de madera, ya que su textura resulta agradable y no contiene mucha información extra para el tacto. Se pretende que no lleve ningún tipo de anclaje o cerradura exterior por este mismo motivo.

Figura 15

La distribución final consta de seis botones situados de forma equidistante, simulando en la medida de lo posible los seis puntos del sistema Braille. Justo encima, pero intentando que las distancias sean diferenciables, se sitúan los dos pulsadores de actuación, con los cuales se generan las letras y palabras aleatorias para el modo juego y se verifica la letra introducida cada vez.

En la parte superior se sitúan el interruptor de encendido y los botones de cambio de modo de juego y de control de volumen. Se pretende que interfieran en la medida de lo posible a la hora de interactuar con el dispositivo y que no suponga una distracción para la persona que lo esté usando.

En la parte interior del dispositivo se encuentra el puerto de carga de la batería.

Figura 16

5. Cálculos

En la primera prueba realizada, se descubren falsos contactos debido a rebotes generados por los pulsadores a la entrada del Arduino. Para solucionar este problema, se utiliza un circuito antirrebote compuesto por una red RC. Mediante la carga y descarga del condensador se estabiliza la señal que entra al pin del Arduino, evitando los rebotes de señal producidos por el pulsador.

El retardo producido en la entrada dependerá de los valores de R y C. De forma general, los valores estándares utilizados en la entrada de un microcontrolador son de 10kΩ y 10μF para resistencia y condensador respectivamente.

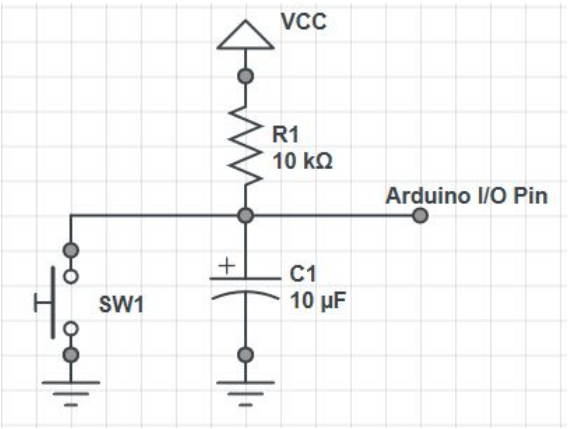
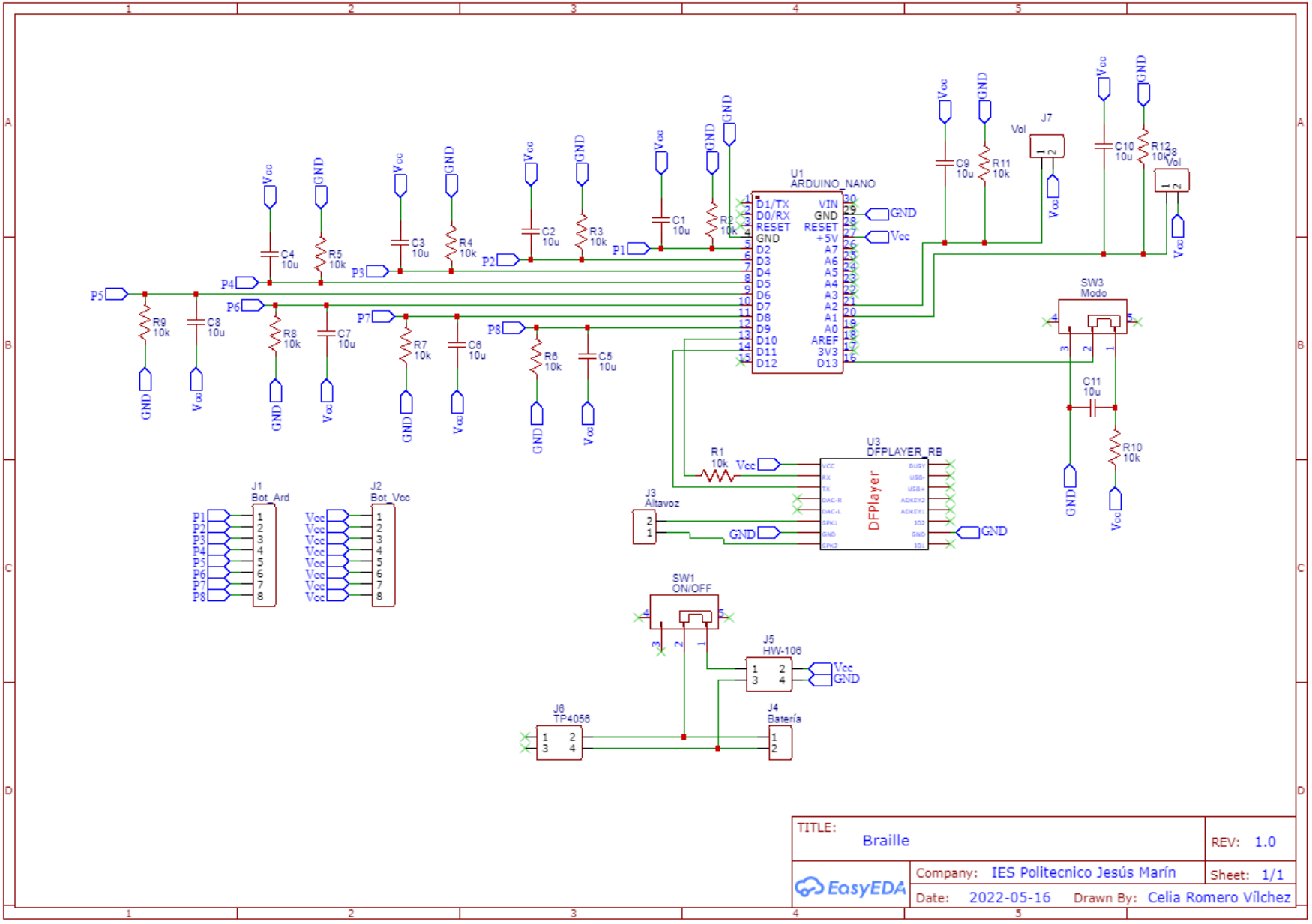


Figura 17

6. Diseño

6.1. Esquemático



6.2. PCB

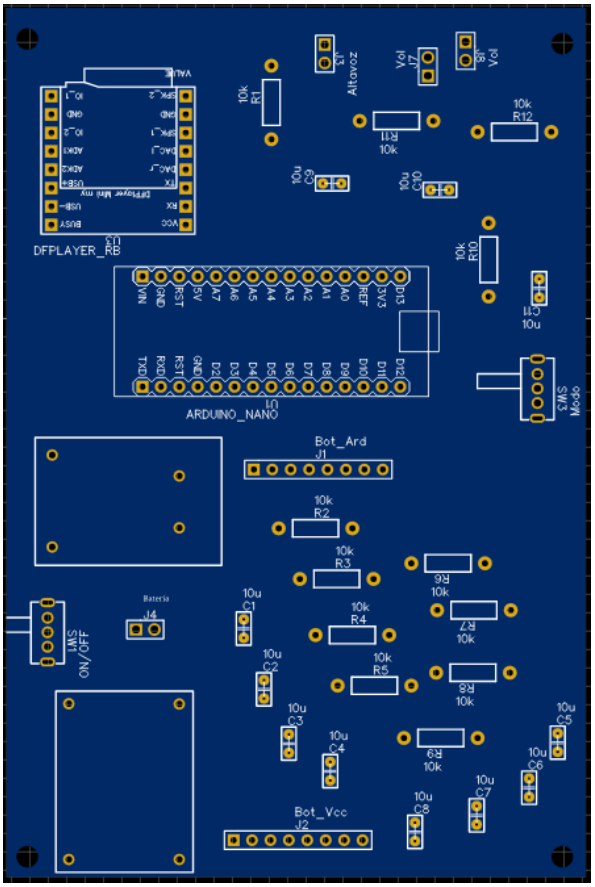


Figura 18

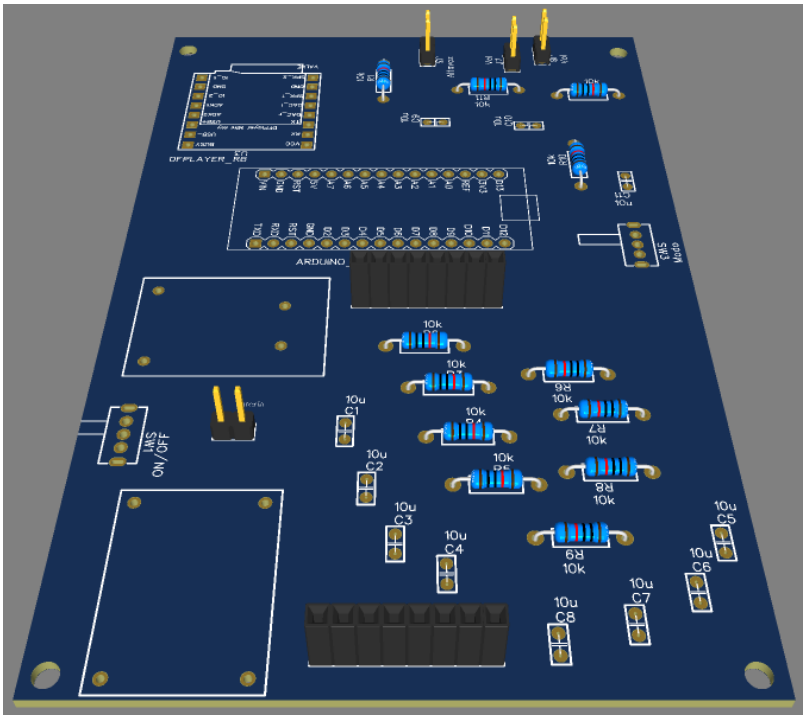


Figura 19

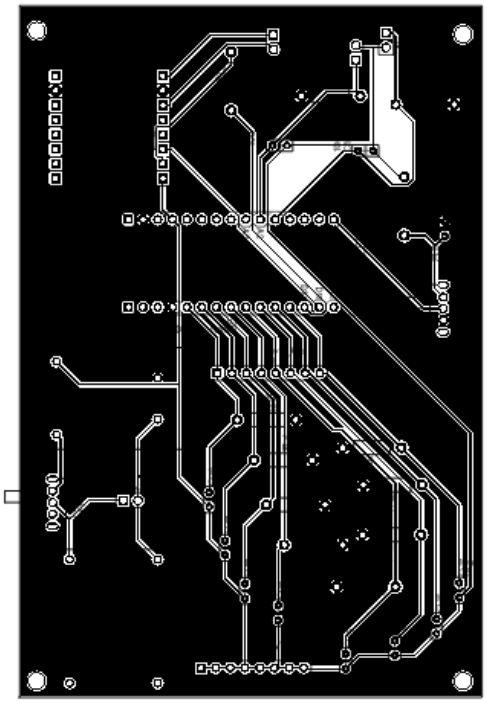


Figura 20

7. Código

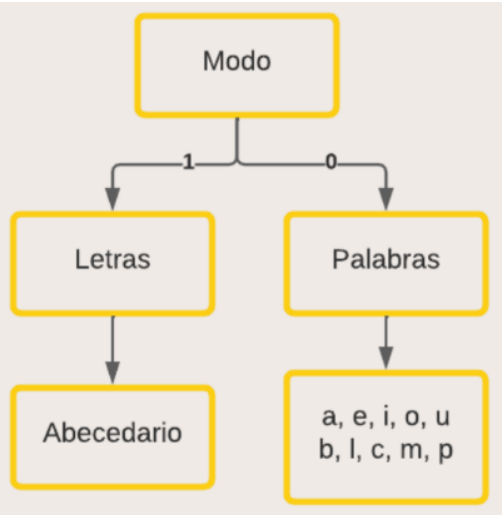


Diagrama 1

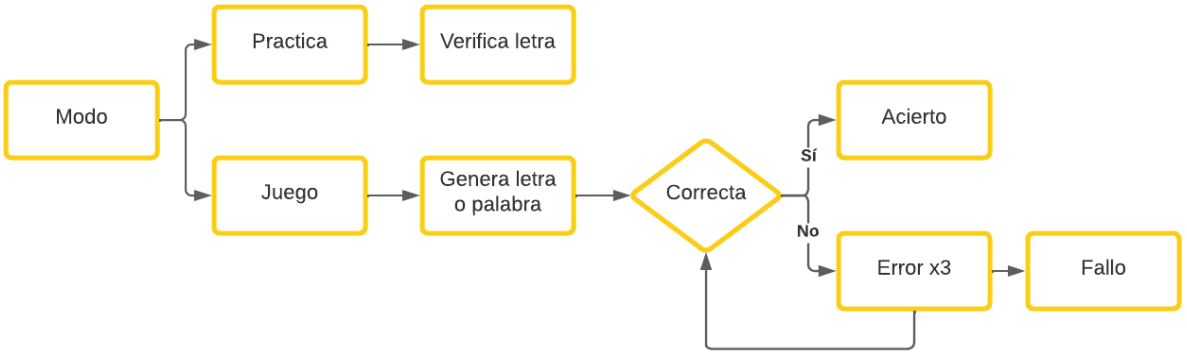


Diagrama 2

7.1. Cuerpo

```
#include "Arduino.h"
#include "SoftwareSerial.h" //inclusión de librerías
#include "DFRobotDFPlayerMini.h"

//Declaración de todas las variables que se van a usar
int Nrandom;
int comparator = 0;
int i = 0;
int pulsador1 = 0;
int pulsador2 = 0;
int pulsador3 = 0;
int pulsador4 = 0;
int pulsador5 = 0;
int pulsador6 = 0;
int pulsador7 = 0;
int pulsador8 = 0;
int intento = 0;
int modo;
String palabra_int = "";
String palabra_random = "VOID";
int cont = 0;
int long_palabra = 100;
int vol_mas;
int vol_menos;
int volumen = 25;

SoftwareSerial mySoftwareSerial(10, 11); // Inicialización del DFPlayer y
todas sus instrucciones necesarias
DFRobotDFPlayerMini myDFPlayer;
```

```
void printDetail(uint8_t type, int value);

void setup()
{
  randomSeed(analogRead(A0)); //lee un valor del pin analogico A0 que es
aleatorio

  pinMode(2, INPUT); //declaramos el pin 2 como entrada
  pinMode(3, INPUT); //declaramos el pin 3 como entrada
  pinMode(4, INPUT); //declaramos el pin 4 como entrada
  pinMode(5, INPUT); //declaramos el pin 5 como entrada
  pinMode(6, INPUT); //declaramos el pin 6 como entrada
  pinMode(7, INPUT); //declaramos el pin 7 como entrada
  pinMode(8, INPUT); //declaramos el pin 8 como entrada
  pinMode(9, INPUT); //declaramos el pin 9 como entrada
  pinMode(13, INPUT); //interruptor letra/palara
  pinMode(A1, INPUT); //Volumen +
  pinMode(A2, INPUT); //Volumen -

  Serial.begin(115200); //inicialización del puerto serie

  init_sd();
}

void loop()
{
  //lee el estado de los botones
  pulsador1 = digitalRead(2);
  pulsador2 = digitalRead(3);
  pulsador3 = digitalRead(4);
  pulsador4 = digitalRead(5);
  pulsador5 = digitalRead(6);
  pulsador6 = digitalRead(7);
  pulsador7 = digitalRead(8); //pulsador comprobación
  pulsador8 = digitalRead(9); //pulsador random

  modo = digitalRead(13);

  vol_mas = digitalRead(A1);
  vol_menos = digitalRead(A2);

  set_volumen();

  Serial.print("pulsador 1= ");
  Serial.println(pulsador1);
  Serial.print("pulsador 2= ");
```

```
  Serial.println(pulsador2);
  Serial.print("pulsador 3= ");
  Serial.println(pulsador3);
  Serial.print("pulsador 4= ");
  Serial.println(pulsador4);
  Serial.print("pulsador 5= ");
  Serial.println(pulsador5);
  Serial.print("pulsador 6= ");
  Serial.println(pulsador6);
  Serial.print("pulsador 7= ");
  Serial.println(pulsador7);
  Serial.print("pulsador 8= ");
  Serial.println(pulsador8);
  Serial.println("");

  Serial.print("Modo = ");
  Serial.println(modo);
  Serial.println("");

  Serial.print("i =");
  Serial.println(i);
  Serial.println("");

  Serial.print("longitud palabra =");
  Serial.println(long_palabra);
  Serial.print("cont =");
  Serial.println(cont);
  Serial.println("");

  Serial.print("Palabra random = ");
  Serial.println(palabra_random);
  Serial.println("");

  Serial.print("Palabra intrucida = ");
  Serial.println(palabra_int);
  Serial.println("");

  //Función para el MODO PRACTICA
  if (pulsador8 == HIGH) //Dependiendo de los botones rojos pulsados al
acionar el pulsador izquierdo busca entre todos los casos el
correspondiente y ejecuta el audio
  {
    letras();
  }

  //Función para el MODO JUEGO generando un valor random de 1,27
```

```
  if (modo == 0) {
    comp_letra();
  }

  if (modo == 1) {
    comp_palabra();
  }
  delay(1000);
}
```

7.2. Inicialización DFPlayer

```
void init_sd() {
  mySoftwareSerial.begin(9600);
  Serial.println();
  Serial.println(F("DFRobot DFPlayer Mini Demo"));
  Serial.println(F("Initializing DFPlayer ... (May take 3~5 seconds)"));
  if (!myDFPlayer.begin(mySoftwareSerial)) //Uso del puerto serie para la
comunicación con mp3
  {
    Serial.println(F("Unable to begin:")); //avisos del DFPlayer si hay
algo mal conectado
    Serial.println(F("1.Please recheck the connection!"));
    Serial.println(F("2.Please insert the SD card!"));
    while (true);
  }

  Serial.println(F("DFPlayer Mini online."));
  myDFPlayer.setTimeOut(500); //Selecciona la comunicacIón con el puerto
serie cada 500ms
  myDFPlayer.volume(25); //Selección del volumen (0~30).
  myDFPlayer.volumeUp(); //Subida de volumen
  myDFPlayer.volumeDown(); //Bajada de volumen
  myDFPlayer.EQ(DFPLAYER_EQ_NORMAL); //equalizacion
  myDFPlayer.outputDevice(DFPLAYER_DEVICE_SD); // uso de tarjeta sd
  Serial.println(myDFPlayer.readState()); //Lee el estado del mp3
  Serial.println(myDFPlayer.readVolume()); //Lee el volumen
  Serial.println(myDFPlayer.readEQ()); //Lee la configuración de
equializacion
  Serial.println(myDFPlayer.readFileCounts()); //Lee los archivos de la SD
  Serial.println(myDFPlayer.readCurrentFileNumber()); //Lee el número de
archivo de reproducción actual
}

void set_volumen()
```

```
{
  if (vol_mas == HIGH)
  {
    myDFPlayer.volumeUp();
    myDFPlayer.play(31); //Notificación volumen
    delay(100);
  }
  if (vol_menos == HIGH)
  {
    myDFPlayer.volumeDown();
    myDFPlayer.play(31); //Notificación volumen
    delay(100);
  }
}
```

7.3. Juego letras

```
void comp_letra()
{
  if (pulsador7 == HIGH)
  {
    i = 1;
    Nrandom = random(1, 27);
    myDFPlayer.play(Nrandom);
    delay(1000);
    comparator = 0;
  }

  if (i == 1 && pulsador8 == HIGH)
  {
    if (comparator == Nrandom) //configuracion de las funciones cuando en
el modo juego el usuario acciona una letra correcta
    {
      myDFPlayer.play(28);
      delay(2000);
      myDFPlayer.pause(); //pause the mp3
      i = 0;
    }
    else//configuracion de las funciones cuando en el modo juego el usuario
acciona una letra incorrecta
    {
      intento++;
      if (intento < 3)
      {
        myDFPlayer.play(29);
```

```
        delay(2000);
        myDFPlayer.pause(); //pause the mp3
      }
      if (intento == 3)
      {
        myDFPlayer.play(30);
        delay(3000);
        myDFPlayer.pause(); //pause the mp3
        intento = 0;
        i = 0;
      }
    }
  }
}
```

7.4. Juego palabras

```
void comp_palabra ()
{
  if (pulsador7 == HIGH)
  {
    i = 1;
    Nrandom = random(31, 49);
    myDFPlayer.play(Nrandom);
    palabras();
    delay(1000);
  }

  intr_palabra();

  if (long_palabra == cont)
  {
    if (palabra_random == palabra_int)
    {
      myDFPlayer.play(Nrandom);
      delay(1000);
      myDFPlayer.pause();
      myDFPlayer.play(28);
      delay(2000);
      myDFPlayer.pause(); //pause the mp3
      i = 0;
      palabra_int = "";
      palabra_random = "VOID";
      cont = 0;
```

```
    }
    else//configuracion de las funciones cuando en el modo juego el usuario
acciona una letra incorrecta
    {
      intento++;
      if (intento < 3)
      {
        myDFPlayer.play(29);
        delay(2000);
        myDFPlayer.pause(); //pause the mp3
        palabra_int = "";
        cont = 0;
      }
      if (intento == 3)
      {
        myDFPlayer.play(30);
        delay(3000);
        myDFPlayer.pause(); //pause the mp3
        intento = 0;
        i = 0;
        palabra_int = "";
        palabra_random = "VOID";
        cont = 0;
      }
    }
  }
}
```

7.5. Introducir palabra

```
void intr_palabra() {
  if (i == 1 && pulsador8 == HIGH)
  {
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == LOW
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "a";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == LOW
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "b";
```

```
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "c";
      cont++;
      delay(50);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "d";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == LOW
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "e";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "f";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "g";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == LOW
        && pulsador5 == HIGH && pulsador6 == LOW)
```

```
    {
      palabra_int = palabra_int + "h";
      cont++;
      delay(1000);
    }
    if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "i";
      cont++;
      delay(1000);
    }
    if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "j";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "k";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "l";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "m";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
```

```
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "n";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == HIGH)
    {
      palabra_int = palabra_int + "ñ";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "o";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "p";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "q";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "r";
      cont++;
      delay(1000);
```

```
    }
    if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "s";
      cont++;
      delay(1000);
    }
    if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == LOW)
    {
      palabra_int = palabra_int + "t";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == LOW && pulsador6 == HIGH)
    {
      palabra_int = palabra_int + "u";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == LOW && pulsador6 == HIGH)
    {
      palabra_int = palabra_int + "v";
      cont++;
      delay(1000);
    }
    if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == HIGH)
    {
      palabra_int = palabra_int + "w";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == LOW && pulsador6 == HIGH)
    {
      palabra_int  = palabra_int  + "x";
```

```
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
        && pulsador5 == HIGH && pulsador6 == HIGH)
    {
      palabra_int  = palabra_int  + "y";
      cont++;
      delay(1000);
    }
    if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
        && pulsador5 == HIGH && pulsador6 == HIGH)
    {
      palabra_int = palabra_int  + "z";
      cont++;
      delay(1000);
    }
  }
}
```

7.6. Introducir letra

```
void letras() {
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == LOW
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 1;
    myDFPlayer.play(1); //A
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == LOW
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 2;
    myDFPlayer.play(2); //B
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == LOW && pulsador6 == LOW)
  {
```

```
    comparator = 3;
    myDFPlayer.play(3); //C
    delay(50);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 4;
    myDFPlayer.play(4); //D
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == LOW &&
pulsador4 == LOW
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 5;
    myDFPlayer.play(5); //E
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 6;
    myDFPlayer.play(6); //F
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 7;
    myDFPlayer.play(7); //G
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == LOW
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 8;
    myDFPlayer.play(8); //H
    delay(1000);
  }
  if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
```

```
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 9;
    myDFPlayer.play(9); //I
    delay(1000);
  }
  if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 10;
    myDFPlayer.play(10); //J
    Serial.println("J");
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 11;
    myDFPlayer.play(11); //K
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 12;
    myDFPlayer.play(12); //L
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 13;
    myDFPlayer.play(13); //M
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 14;
    myDFPlayer.play(14); //N
    delay(1000);
```

```
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == HIGH)
  {
    comparator = 15;
    myDFPlayer.play(15); //Ñ
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 16;
    myDFPlayer.play(16); //O
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 17;
    myDFPlayer.play(17); //P
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 18;
    myDFPlayer.play(18); //Q
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 19;
    myDFPlayer.play(19); //R
    delay(1000);
  }
  if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == LOW && pulsador6 == LOW)
  {
    comparator = 20;
```

```
    myDFPlayer.play(20); //S
    delay(1000);
  }
  if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == LOW)
  {
    comparator = 21;
    myDFPlayer.play(21); //T
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == LOW && pulsador6 == HIGH)
  {
    comparator = 22;
    myDFPlayer.play(22); //U
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == HIGH && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == LOW && pulsador6 == HIGH)
  {
    comparator = 23;
    myDFPlayer.play(23); //V
    delay(1000);
  }
  if (pulsador1 == LOW && pulsador2 == HIGH && pulsador3 == LOW &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == HIGH)
  {
    comparator = 24;
    myDFPlayer.play(24); //W
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == LOW && pulsador6 == HIGH)
  {
    comparator = 25;
    myDFPlayer.play(25); //X
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == HIGH
      && pulsador5 == HIGH && pulsador6 == HIGH)
```

```
  {
    comparator = 26;
    myDFPlayer.play(26); // Y
    delay(1000);
  }
  if (pulsador1 == HIGH && pulsador2 == LOW && pulsador3 == HIGH &&
pulsador4 == LOW
      && pulsador5 == HIGH && pulsador6 == HIGH)
  {
    comparator = 27;
    myDFPlayer.play(27); //Z
    delay(1000);
  }
}
```

7.7. Banco de palabras

```
void palabras() {
  if (Nrandom == 32)
  {
    palabra_random = "ala";
    long_palabra = 3;
  }
  if (Nrandom == 33)
  {
    palabra_random = "ama";
    long_palabra = 3;
  }
  if (Nrandom == 34)
  {
    palabra_random = "amo";
    long_palabra = 3;
  }
  if (Nrandom == 35)
  {
    palabra_random = "boa";
    long_palabra = 3;
  }
  if (Nrandom == 36)
  {
    palabra_random = "col";
    long_palabra = 3;
  }
  if (Nrandom == 37)
  {
```

```
    palabra_random = "eco";
    long_palabra = 3;
  }
  if (Nrandom == 38)
  {
    palabra_random = "leo";
    long_palabra = 3;
  }
  if (Nrandom == 39)
  {
    palabra_random = "mal";
    long_palabra = 3;
  }
  if (Nrandom == 40)
  {
    palabra_random = "mio";
    long_palabra = 3;
  }
  if (Nrandom == 41)
  {
    palabra_random = "oca";
    long_palabra = 3;
  }
  if (Nrandom == 42)
  {
    palabra_random = "ola";
    long_palabra = 3;
  }
  if (Nrandom == 43)
  {
    palabra_random = "pie";
    long_palabra = 3;
  }
  if (Nrandom == 44)
  {
    palabra_random = "pua";
    long_palabra = 3;
  }
}
```

8. BOM

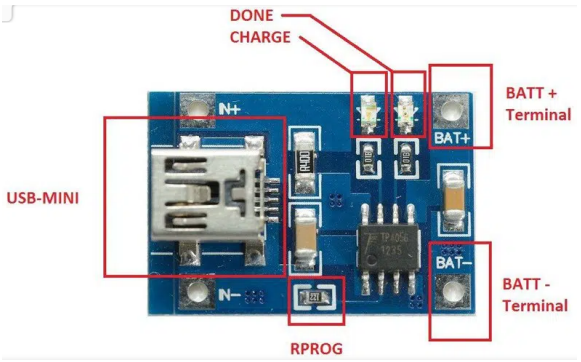
ITEM	COMPONENTE	DESCRIPCIÓN	CANTIDAD
1	Arduino Nano	Controlador	1

2	TP4056	Módulo cargador de baterías	1
3	HW-106 DC-DC	Conversor DC-DC 3.7V a 5V	1
4	DFPlayer Mini	Lector de tarjetas SD para Arduino	1
5	Tarjeta microSD	Tarjeta microSD 16GB	1
6	Conector	Conector mini Molex macho 2 pines	4
7	Conector	Conector mini Molex hembra 2 pines	4
8	Conector	Conector mini Molex hembra 3 pines	1
9	Conector crimpa	Conector amp Amphenol crimpado hembra 8 pines	2
10	Conector	Conector berg macho	19
11	Conector berg	Conector berg hembra	54
12	Resistencia	Resistencia 1kΩ	1
13	Resistencia	Resistencia 10kΩ	11
14	Condensador	Condensador 10µF	11
15	Pulsador	Pulsador enclavamiento 0.472 in Senrise	8
16	Pulsador	Pulsador 0.472 in Senrise	2
17	Pulsador	Mini pulsador 6x6mm 10H	3
18	Interruptor	Interruptor palanca 2 posiciones	1
19	Cinta plana	Cinta plana 8 pines	2
20	Altavoz		1
21	Caja	Caja 19x9x5cm reciclada	1

Tabla 1

9. Anexos

9.1. Módulo cargador TP4056



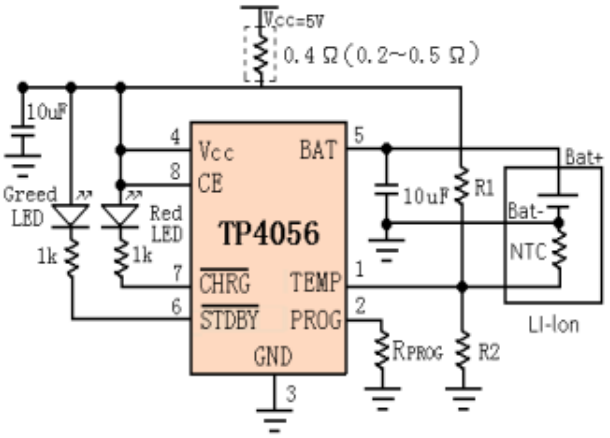


Figura 21

SYMBOL	PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
V_{CC}	Input Supply Voltage		●	4.0	5	8.0	V
I_{CC}	Input Supply Current	Charge Mode, R_{PROG} = 1.2k StandbyMode(Charge Terminated) Shutdown Mode (R_{PROG} Not Connected, $V_{CC} < V_{BAT}$, or $V_{CC} < V_{UV}$)	● ● ●		150 55 55	500 100 100	μA μA μA
V_{FLOAL}	Regulated Output (Float) Voltage	0℃≤T_A≤85℃, I_{BAT}=40mA		4.137	4.2	4.263	V
I_{BAT}	BAT Pin Current Text condition:VBAT=4.0V	R_{PROG} = 2.4k, Current Mode R_{PROG} = 1.2k, Current Mode Standby Mode, V_{BAT} = 4.2V	● ● ●	450 950 0	500 1000 −2.5	550 1050 −6	mA mA μA
I_{TRIKL}	Trickle Charge Current	$V_{BAT}<V_{TRIKL}$, R_{PROG}=1.2K	●	120	130	140	mA
V_{TRIKL}	Trickle Charge Threshold Voltage	R_{PROG}=1.2K, V_{BAT} Rising		2.8	2.9	3.0	V
V_{TRHYS}	Trickle Charge Hysteresis Voltage	R_{PROG}=1.2K		60	80	100	mV
T_{LIM}	Junction Temperature in Constant Temperature Mode				145		℃

Tabla 2

9.2. Arduino Nano

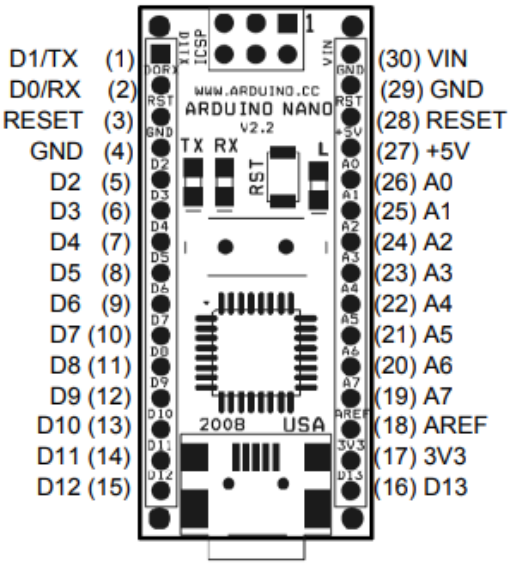


Figura 22

Pin No.	Name	Type	Description
1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
3, 28	RESET	Input	Reset (active low)
4, 29	GND	PWR	Supply ground
17	3V3	Output	+3.3V output (from FTDI)
18	AREF	Input	ADC reference
19-26	A7-A0	Input	Analog input channel 0 to 7
27	+5V	Output or Input	+5V output (from on-board regulator) or +5V (input from external power supply)
30	VIN	PWR	Supply voltage

Tabla 3

9.3. DFPlayer Mini

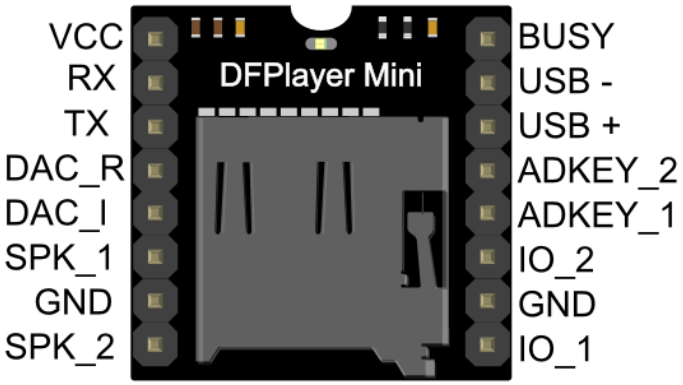


Figura 23

Pin	Description	Note
VCC	Input Voltage	DC3.2~5.0V;Type: DC4.2V
RX	UART serial input	
TX	UART serial output	
DAC_R	Audio output right channel	Drive earphone and amplifier
DAC_L	Audio output left channel	Drive earphone and amplifier
SPK2	Speaker-	Drive speaker less than 3W
GND	Ground	Power GND
SPK1	Speaker+	Drive speaker less than 3W
IO1	Trigger port 1	Short press to play previous（long press to decrease volume）
GND	Ground	Power GND
IO2	Trigger port 2	Short press to play next（long press to increase volume）
ADKEY1	AD Port 1	Trigger play first segment
ADKEY2	AD Port 2	Trigger play fifth segment
USB+	USB+ DP	USB Port
USB-	USB- DM	USB Port
BUSY	Playing Status	Low means playing \High means no

Tabla 4

9.4. HW-106 DC-DC

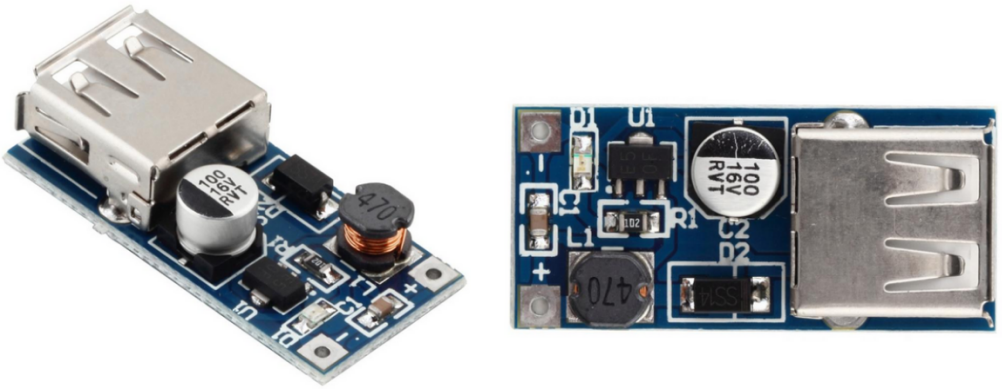


Figura 24

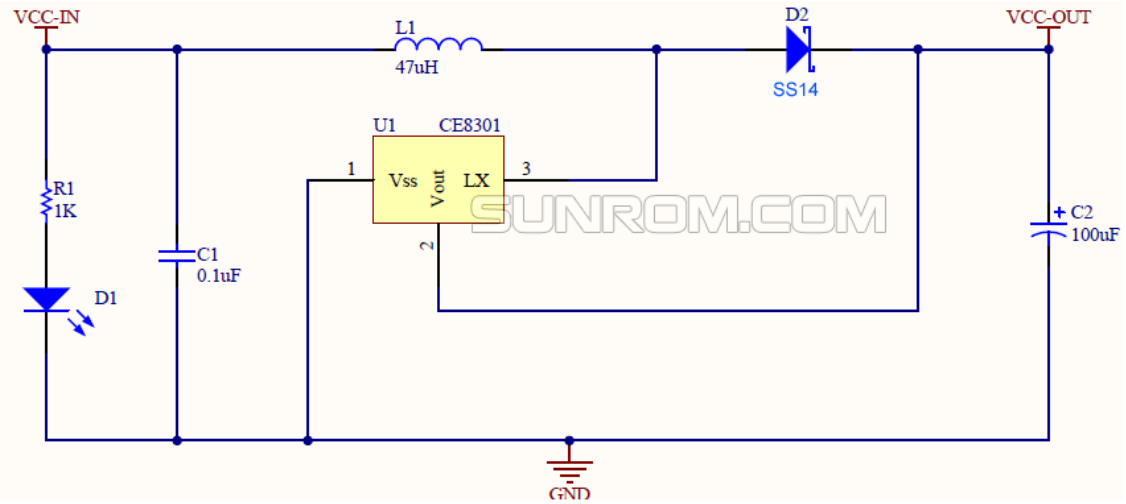
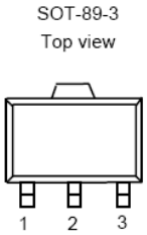


Figura 25



PIN NO.	PIN NAME	FUNCTION
1	V_{SS}	GND pin
2	V_{OUT}	Output voltage pin
3	LX	External inductor connection pin

Tabla 5

10. Bibliografía

(PDF) Educación y Tecnología, de lo excepcional a lo cotidiano

La discapacidad visual y las TIC en la etapa escolar | Observatorio Tecnológico

Braille en español, alfabeto, números, y sistema - Web ONCE

BRAILLÍN, UN MÉTODO PARA NIÑOS PARA APRENDER BRAILLE | DISCAPACIDAD VISUAL DOCE (DISCAPACITADOS OTROS CIEGOS DE ESPAÑA)

datasheet TP4056.pdf

Arduino Nano (V2.3)

DFPlayer_Mini_SKU_DFR0299-DFRobot