

Trabajo final de asignatura: Pianola

Artefactos Digitales e Interactivos - 11808
Universidad Nacional de Luján 2021



Juran, Martín Tomás
Legajo: 143191
tomasjuran96@gmail.com



1. Introducción

En este proyecto se definió como objetivo crear un instrumento musical automático, reproducible, de tamaño y complejidad pequeña utilizando una plataforma de hardware/programación abierta (Arduino). Para esto se decidió construir uno de los elementos más clásicos de la reproducción automática de música: la pianola¹, cuyos primeros prototipos datan del Siglo XIX.

1.1. Trabajos relacionados

Programar una pianola utilizando Arduino no es una idea nueva. Proyectos más clásicos, controlando un piano real² pueden encontrarse en sitios populares. Sin embargo, una dificultad encontrada en estos proyectos es la accesibilidad para obtener los componentes de hardware.

De manera más fiel a la pianola original, es posible leer papel perforado³, pero la dificultad aquí yace en la manera de preparar las "partituras"^{4 5}.

Quizás siendo causa probable las dificultades mencionadas, es más común encontrar proyectos donde la programación de las partituras en el papel se simplifica, utilizando lectores infrarrojo. En esta categoría, se puede priorizar la fidelidad de la música original⁶ o la simpleza del mecanismo⁷.

¹<https://es.wikipedia.org/wiki/Pianola>

²<https://create.arduino.cc/projecthub/MoKo9/arduino-controlled-piano-robot-pibot-641a06>

³<https://www.youtube.com/watch?v=6au08djKRrM>

⁴<https://www.youtube.com/watch?v=HjBh09iqEc0>

⁵<https://www.hackster.io/news/make-a-music-box-punch-machine-with-arduino-d5fc6703e882>

⁶<https://www.instructables.com/Arduino-Barrel-Piano/>

⁷<https://www.instructables.com/Electronic-Music-Box-Powered-by-Arduino-sort-of/>

2. Sobre el prototipo

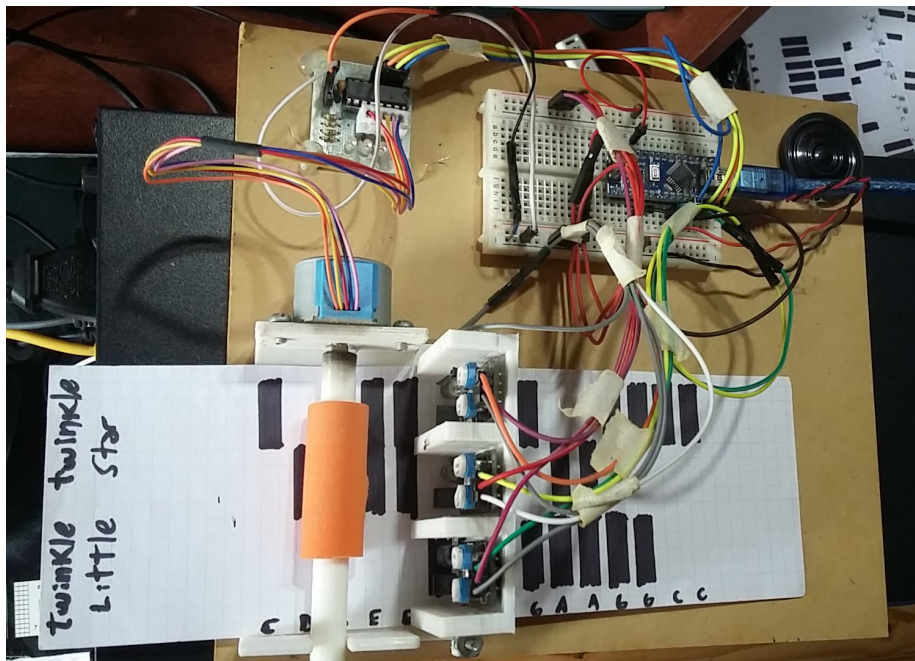


Figura 1: Ejemplo del artefacto pianola leyendo una partitura

2.1. Descripción

Para realizar el artefacto digital "pianola", se decidió que la forma de crear las partituras (elemento que la pianola "lee" para tocar las notas) debería ser lo más simple posible. Así, el artefacto lee y toca partituras realizadas con papel y fibrón negro, utilizando un mecanismo simple de alimentación con un rodillo rotado por un motor paso a paso. Las partituras simplemente deben seguir un formato de codificación binario de tres dígitos⁸, de 0 a 7, donde los blancos representan ceros y los negros uno. El valor 0 en binario corresponde a un silencio (intervalo completamente en blanco), 1 corresponde a la nota Do (todos blancos menos el más a la derecha), y así hasta Si (todos negros).

⁸https://es.wikipedia.org/wiki/Sistema_binario

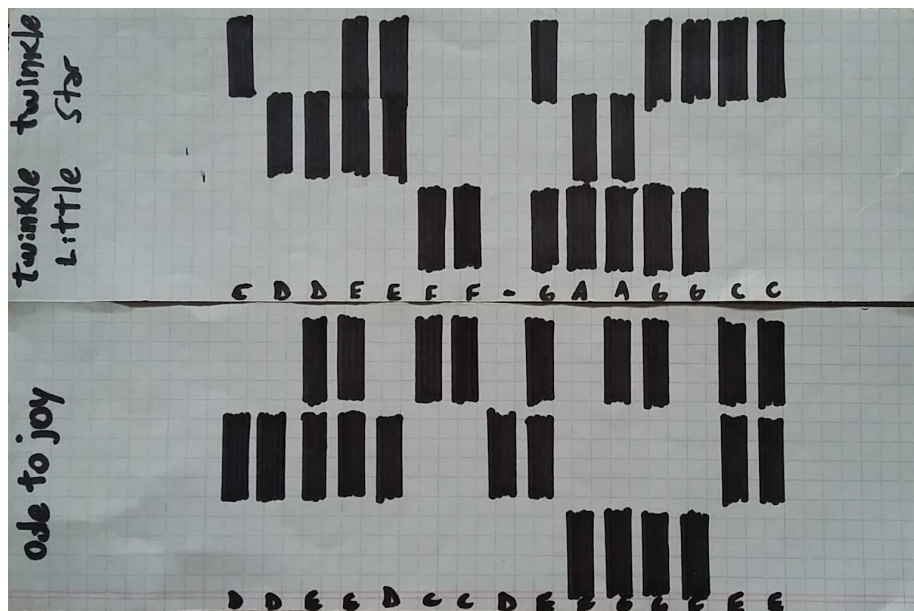


Figura 2: Ejemplo de partitura para el artefacto

2.2. Decisiones de diseño

Para este diseño se decidió utilizar el motor paso a paso de tipo unipolar 28BYJ-48, así como también los lectores infrarrojo KY-032 detectores de objetos; ambos elegidos por su disponibilidad a la hora de realizar el prototipo. La plataforma elegida para el artefacto es el Arduino Nano, por su tamaño. Bajo la misma línea, en el diseño de las piezas 3D también se priorizó la simpleza y el tamaño reducido, con el fin de crear una "pianola de bolsillo".

2.3. Tecnología utilizada

Se utilizó un Arduino Nano, pero el esquema es compatible también con Uno y Mega.

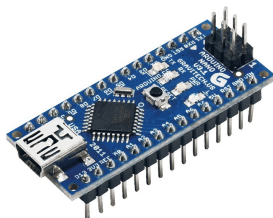


Figura 3: Arduino Nano

La estructura de soporte para el lector de partituras y el rodillo se hicieron con una impresora 3D; su diseño se realizó en Tinkercad⁹.

Para la alimentación de las partituras se utilizó un motor paso a paso de tipo unipolar 28BYJ-48 junto con su módulo driver ULN2003A para utilizar con Arduino, configurado en modo *wave drive*¹⁰, activando una bobina a la vez y ajustando el intervalo o *delay* entre activaciones; para lograr que rote al doble de su velocidad máxima.

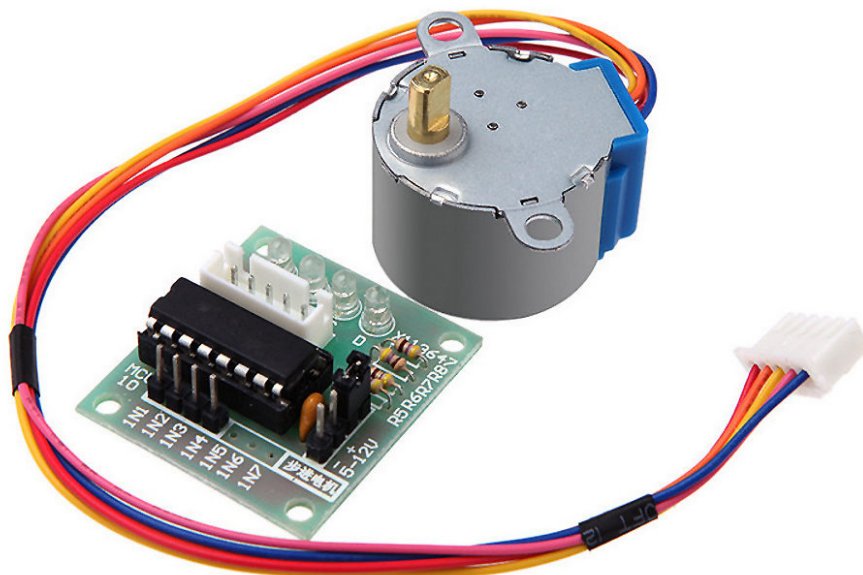


Figura 4: Motor paso a paso 28BYJ-48 y módulo ULN2003A

Para la lectura de las partituras se utilizaron tres lectores infrarrojos KY-032, cuyo orden determina la posición del dígito binario en el momento de la decodificación de la partitura. Es importante la calibración de la frecuencia de dichos módulos para reducir la interferencia entre ellos. El esquema de funcionamiento es utilizar una partitura de fondo blanco tal que en todo momento los lectores detectan un objeto, y pintar de negro la codificación correspondiente a las notas. De esta manera, cuando el papel pintado de negro pasa por el lector, el primero absorbe el rayo infrarrojo y el lector deja de detectar un objeto: intervalo en el cual se procesa la nota.

Se utilizó un buzzer genérico para la reproducción de sonidos; al mismo se le define la duración y la frecuencia del sonido a reproducir en cada iteración.

⁹<https://www.tinkercad.com/>

¹⁰[https://en.wikipedia.org/wiki/Stepper_motor#Wave_drive_\(one_phase_on\)](https://en.wikipedia.org/wiki/Stepper_motor#Wave_drive_(one_phase_on))

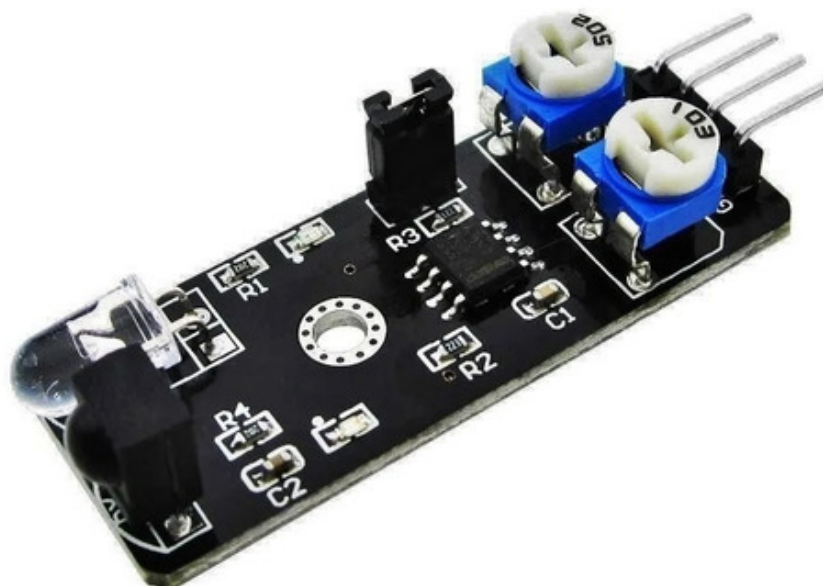


Figura 5: Módulo detector de objetos infrarrojo KY-032

2.4. Experimentos realizados

Una primera versión del prototipo leía y tocaba las notas de forma constante, pero esto dio lugar a muchos errores debido a la interferencia entre los lectores que se encuentran muy próximos. Por esto se pasó al método de codificación.

Para los lectores infrarrojo se intentó simular la aislación del LED presente en dispositivos de mayor complejidad, recubriendo el mismo para intentar focalizar la luz emitida y así reducir interferencias.

En una primera instancia, se utilizó el modo de funcionamiento recomendado del motor paso a paso, haciendo *full-step*, activando dos bobinas por paso. Sin embargo, este no resultó lo suficientemente rápido como para compensar la precisión requerida por el lector para diferenciar una nota de la siguiente: las notas de poco grosor eran confundidas con blancos. Por esta razón se procedió a realizar *wave drive* activando una bobina a la vez y reduciendo el intervalo entre cada paso. Esto requirió conectar el motor a una fuente externa al Arduino Nano, ya que el último no cuenta con la energía suficiente para mantener este modo de funcionamiento.

El "largo" de cada nota en la partitura (es decir, cuánto tiempo permanece la nota bajo el lector) óptimo encontrado para este experimento puntual fue de 5mm. El ancho de la nota es equivalente al ancho del lector, 15mm, aunque para una mayor precisión se recomiendan 20mm.

Una configuración posible que se descartó por su complejidad a la hora de

realizar las partituras es aumentar la cantidad de codificaciones posibles, realizando dos lecturas por nota. Para este modo se planteó agregar un "bit" que indique la duración de la nota (corta o larga), otro que permita indicar si la nota es sostenida y otro que permita modificar la octava en la que se toca.

3. Construcción del prototipo

3.1. Hardware

Los elementos físicos requeridos para su construcción son:

- 1x Arduino Nano o equivalente
- 1x Motor paso a paso 28BYJ-48 con su módulo ULN2003A
- 3x Detectores de objetos infrarrojos KY-032
- 1x Buzzer genérico
- 1x Breadboard
- Piezas hechas con impresora 3D según se detalla a continuación (rodillo, base, lectores)

El esquema de conexión es el siguiente:

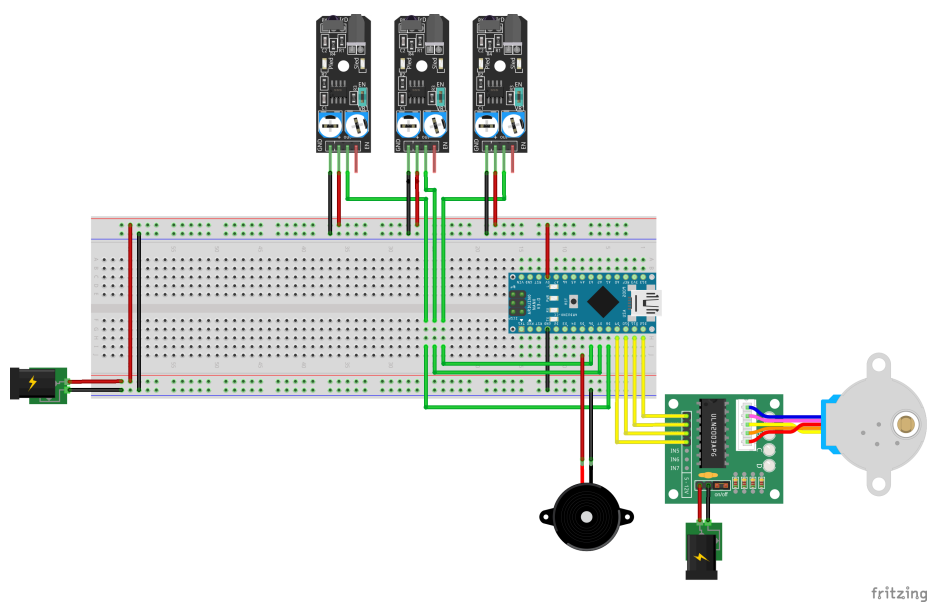


Figura 6: Esquema de conexión del prototipo

3 CONSTRUCCIÓN DEL PROTOTIPO

El rodillo junto al motor se montan en la base de la pianola, según la figura 7, cuyo diseño está disponible en <https://www.tinkercad.com/things/jjR9W5TQoV1-pianola-base>. El rodillo está disponible en <https://www.tinkercad.com/things/6kggc9XXIjA-pianola-roller>. Este último requiere un material que lo recubra, con mayor fricción que el plástico, que le permita arrastrar el papel; por esta razón el rodillo provisto no está en contacto con la base.

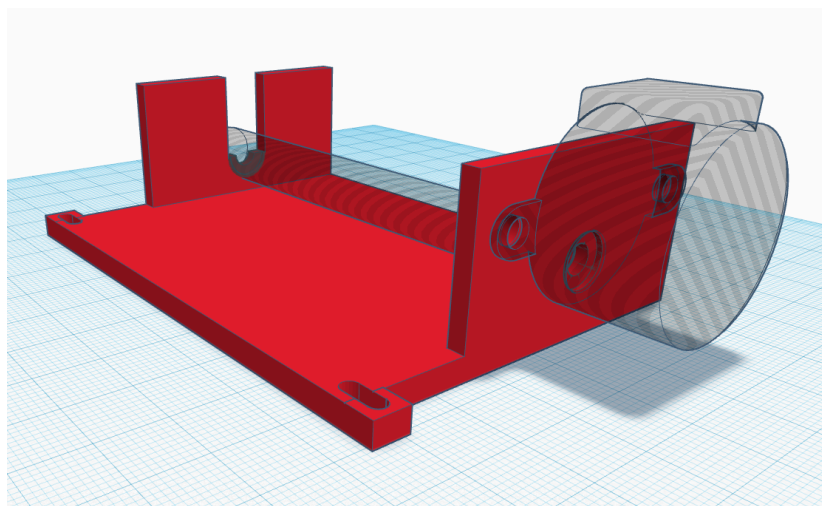


Figura 7: Base de la pianola (modelo 3D)

Luego, se montan los lectores en la pieza de la figura 8, disponible en <https://www.tinkercad.com/things/4Vim890SJKJ-pianola-ir-array>.

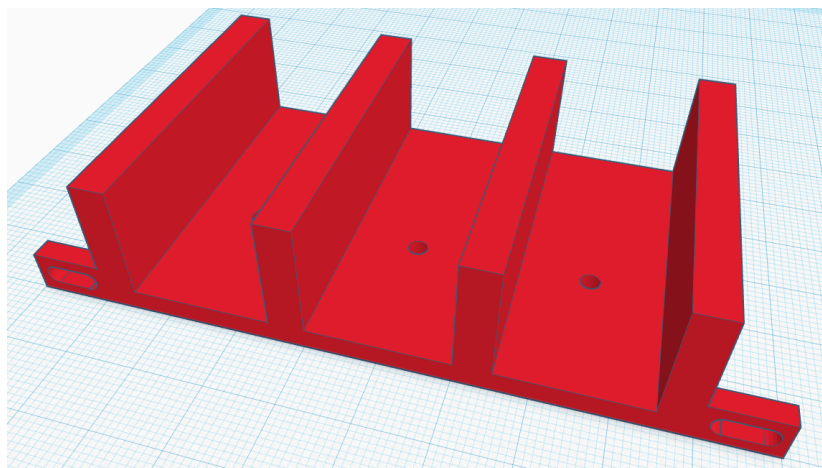


Figura 8: Pieza de lectores IR



3.2. Software

El siguiente código debe compilarse y cargarse en el Arduino utilizando Arduino-IDE¹¹ o similar. En la función *leerNota* es posible ajustar la duración y la octava que se toca por defecto, mientras que *turnOnce* permite indicar el intervalo entre cada lectura. El código completo se encuentra en <https://gitlab.com/tomasjuran/adi-pianola>.

```
int Steps = 0;

void loop() {
  leerNota();
  turnOnce();
}

void turnOnce() {
  for (int i = 0; i < 400; i++) {
    darUnPaso();
  }
}

void darUnPaso() {
  digitalWrite(IN1, Paso[Steps][0]);
  digitalWrite(IN2, Paso[Steps][1]);
  digitalWrite(IN3, Paso[Steps][2]);
  digitalWrite(IN4, Paso[Steps][3]);
  Steps += 1;
  Steps = Steps % 4;
  delay(STEP_DELAY);
}

void leerNota() {
  int duration = 500; // milisegundos
  int octave = 5;
  float note = 0;
  // Cuando la salida del lector es HIGH, es porque no detecta objeto,
  // o en este caso no hay reflexion por ser una superficie oscura
  for (int i = 0; i < READERS_LEN; i++) {
    if (digitalRead(READERS[i]) == HIGH) {
      note += pow(2, i);
    }
  }
  float freq = NOTES[int(ceil(note))];
  tone(BUZZER, pow(2, octave) * freq, duration);
}
```

¹¹<https://www.arduino.cc/en/software>

3.3. Puesta en marcha

Una vez compilado y cargado el código, y siguiendo el esquema de conexión provisto, basta con conectar a una fuente de alimentación (2 entradas USB) al Arduino Nano y al motor paso a paso, introducir una partitura y permitir la lectura, como se puede ver en el video https://youtu.be/_axi65BYuBY.

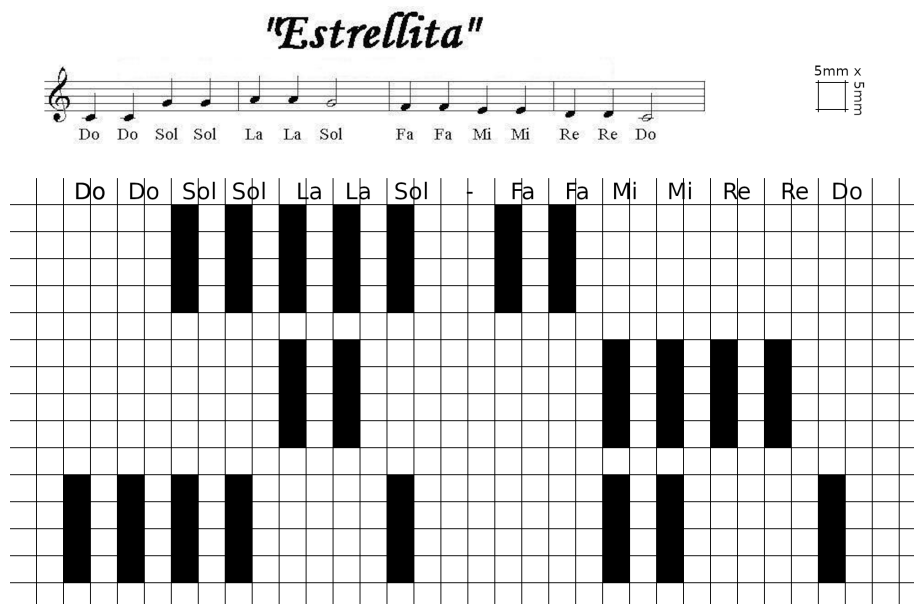


Figura 9: Partitura de muestra "Estrellita"

4. Conclusión

Es importante destacar la dinámica de prototipar con Arduino. En este caso, las limitaciones de los dispositivos de hardware pueden compensarse agregando robustez en el software. Además, gracias a la facilidad de su programación, es posible darle un uso que tal vez no es el óptimo o indicado al hardware disponible para realizar proyectos de cualquier índole.

5. Trabajos futuros

A continuación se enumeran algunas de las mejoras posibles para el prototipo y proyectos futuros:

- Mejorar el aislamiento de los lectores infrarrojos, ya sea cambiando el diseño de la pieza donde se montan o modificando la forma en que se leen



las notas (por ejemplo cambiando la orientación).

- Mejorar el sistema de codificación, implementando múltiples octavas, sostenidos y duraciones.
- Mejorar el sistema de alimentación, con un rodillo o motor más apropiados, para evitar que el papel se atasque o corra y prevenga la correcta lectura de las notas.
- Agregar redundancia en el código para una mejor lectura de las notas, por ejemplo realizando varias lecturas por intervalo y utilizando la lectura más frecuente.
- Cambiar los lectores individuales por un array de sensores IR, tal que se puedan leer más notas (al costo de partituras más grandes).
- Permitir tocar múltiples notas a la vez (acordes).