

Arduino Punk Console

Idea

Crear un secuenciador de 8 tiempos con un generador de tonos(Atari Punk Console) y un arduino que se encarga de la lógica de modificar que tono suena en cada tiempo, la duración y el tempo.

¿Que es un secuenciador?

Un secuenciador es un dispositivo electrónico físico o una aplicación informática que permite programar y reproducir eventos musicales de forma secuencial mediante una interfaz de control físico o lógico conectado a uno o más instrumentos musicales electrónicos. La interfaz de control más extendido es el estándar MIDI.

¿Qué es un Atari Punk Console?

La Consola Punk Atari (en inglés Atari Punk Console) es un simple circuito productor de ruidos “hazlo tú mismo” (DIY) que es relativamente barato y fácil de fabricar. Tiene dos controles, uno para la frecuencia y otro para controlar la duración.

Componentes

- 4 Potenciómetros lineales de 100k Mono
- 8 Pulsadores
- 1 Switch
- 1 Salida de audio (Mono o Estéreo, depende de la salida a la que se quiera conectar)
- 1 Placa perforada
- 9 Resistencias de 10k
- Todo empaquetado felizmente dentro de un tupper de cocina, para un acabado profesional

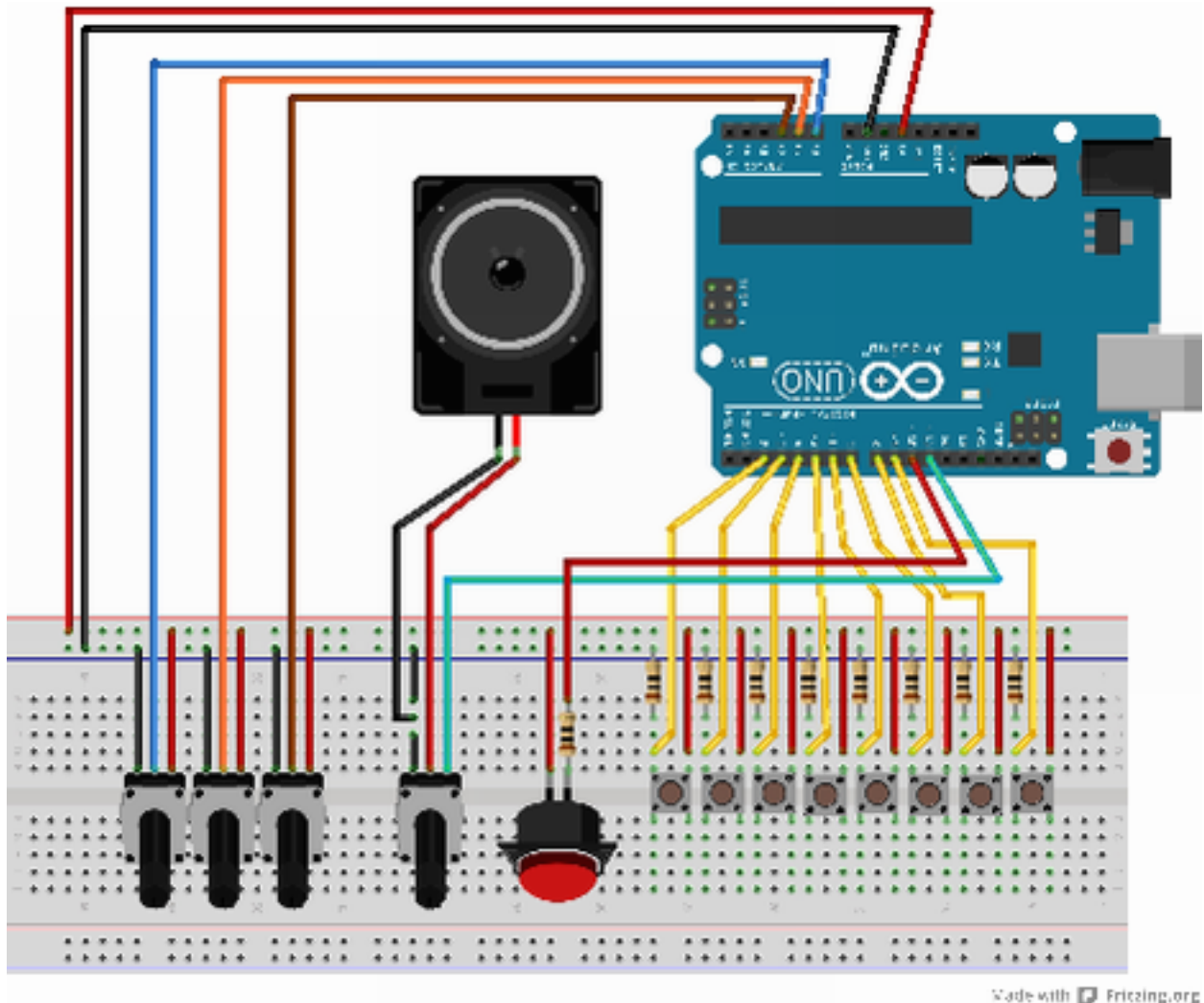
Sobre los controles

Hay cuatro potenciómetros: uno para la frecuencia, uno para la duración, uno para el tempo y otro para el volumen. Los primeros tres son potenciómetros lineales cableados como divisores de voltaje. El que controla el volumen simplemente maneja la intensidad del canal saliente. Los botones sirven para guardar la frecuencia que está marcada en el potenciómetro en el paso correspondiente del secuenciador, por ejemplo, si tenemos el potenciómetro al mínimo y presionamos el botón 2, en el segundo paso de la secuencia, sonará la nota equivalente al potenciómetro al mínimo.

El potenciómetro de la duración determina cuánto tiempo suena cada nota de la secuencia y el del tempo determina el tiempo entre cada paso.

Después de cada secuencia de 8 tonos, el secuenciador vuelve a comenzar repitiéndose indefinidamente hasta que se cambie algún tono o se apague mediante el switch.

Esquema de conexiones



Código

```
/*=====
  Arduino Punk Console
  =====*/
// Mapeo de los pins para INPUT y OUTPUT
#define AnalogInTempo 0
#define AnalogInDuration 1
#define AnalogInFrequency 2
```

```

#define DigitalInSwitch0 2
#define DigitalInSwitch1 3
#define DigitalInSwitch2 4
#define DigitalInSwitch3 5
#define DigitalInSwitch4 6
#define DigitalInSwitch5 7
#define DigitalInSwitch6 8
#define DigitalInSwitch7 9

#define DigitalInStartStop 10
#define DigitalOutSignal 11

// Inicializacion del array de frecuencias
int steps[] = {100,120,140,160,180,200,220,240};
// Setup inicial
int duration = 50;
int pitchval = 1;
int fPlayMode = true;
int lastPushedStep = -1;
// Inicializacion del tempo
int tempo = 900;

void setup()
{
  // Seteo de los pines
  pinMode (DigitalInSwitch0, INPUT);
  pinMode (DigitalInSwitch1, INPUT);
  pinMode (DigitalInSwitch2, INPUT);
  pinMode (DigitalInSwitch3, INPUT);
  pinMode (DigitalInSwitch4, INPUT);
  pinMode (DigitalInSwitch5, INPUT);
  pinMode (DigitalInSwitch6, INPUT);
  pinMode (DigitalInSwitch7, INPUT);

  pinMode (DigitalInStartStop, INPUT);
  pinMode (DigitalOutSignal, OUTPUT);

  Serial.begin(9600);
}

void loop() {
  // Secuencia principal de loopeo
  for (int i=0; i<8; i++) {
    // booleano para la reproduccion del sonido
    fPlayMode = digitalRead (DigitalInStartStop);
  }
}

```

```

// lectura de botones presionados
readSwitches();
// lectura de potenciómetros (duración y tiempo)
readPots();

// Si está encendido
if (fPlayMode) {
// freqout va a emitir la frecuencia de sonido almacenada en la posición i
  freqout (steps[i], duration);
}
// Pausa entre loops
delay (tempo);
}
}
// Lee los valores actuales de los potenciómetros de Duración y de Tiempo
void readPots () {
  tempo = (analogRead (AnalogInTempo) * 1.9);
  duration = (analogRead (AnalogInDuration));
}

// Lee el valor actual de los switches (HIGH o LOW).
// Si alguno está presionado, va a reemplazar en la posición que le corresponde
// por el valor que lee del potenciómetro de frecuencia
void readSwitches() {

if (digitalRead (DigitalInSwitch0) == HIGH) {
  steps[0] = analogRead(AnalogInFrequency);
  lastPushedStep = 1;

} else if (digitalRead (DigitalInSwitch1) == HIGH) {
  steps[1] = analogRead(AnalogInFrequency);
  lastPushedStep = 2;

} else if (digitalRead (DigitalInSwitch2) == HIGH) {
  steps[2] = analogRead(AnalogInFrequency);
  lastPushedStep = 3;

} else if (digitalRead (DigitalInSwitch3) == HIGH) {
  steps[3] = analogRead(AnalogInFrequency);
  lastPushedStep = 4;

} else if (digitalRead (DigitalInSwitch4) == HIGH) {
  steps[4] = analogRead(AnalogInFrequency);
  lastPushedStep = 5;
}

```

```

} else if (digitalRead (DigitalInSwitch5) == HIGH) {
  steps[5] = analogRead(AnalogInFrequency);
  lastPushedStep = 6;

} else if (digitalRead (DigitalInSwitch6) == HIGH) {
  steps[6] = analogRead(AnalogInFrequency);
  lastPushedStep = 7;

} else if (digitalRead (DigitalInSwitch7) == HIGH) {
  steps[7] = analogRead(AnalogInFrequency);
  lastPushedStep = 8;
}
}
}

```

```

void freqout(int freq, int t) {

  int hperiod;
  long cycles, i;

  hperiod = (500000 / ((freq - 7) * pitchval));

  cycles = ((long)freq * (long)t) / 1000; /
  for (i=0; i<= cycles; i++) {
    digitalWrite(DigitalOutSignal, HIGH);
    delayMicroseconds(hperiod);
    digitalWrite(DigitalOutSignal, LOW);
    delayMicroseconds(hperiod - 1);
  }
}
}

```

Proyecto de referencia

Este es el proyecto en el que nos basamos para construir la nuestra:

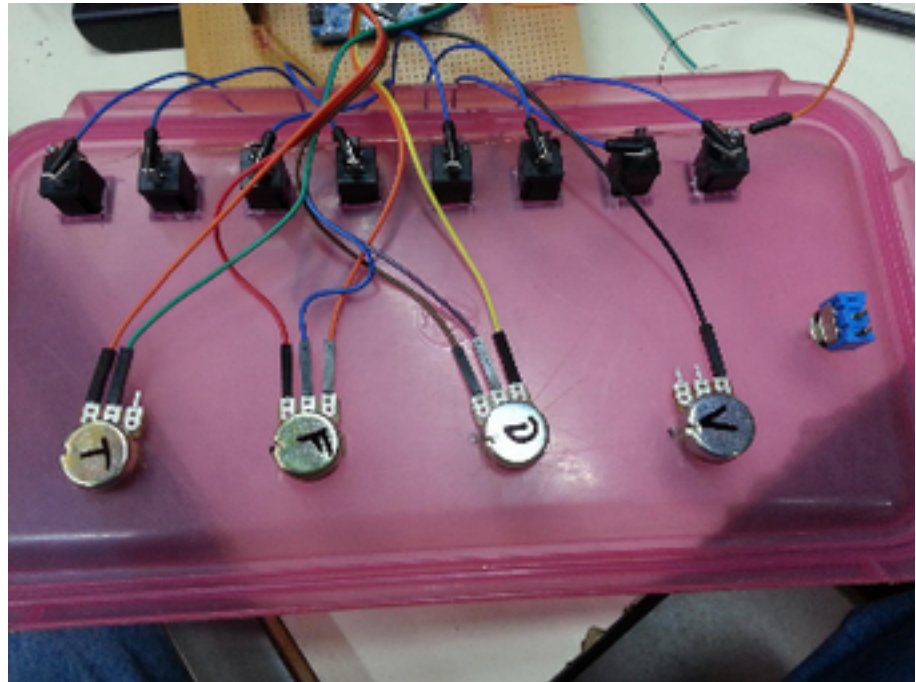
<http://www.beavisaudio.com/projects/digital/ArduinoPunkConsole/>

Dificultades

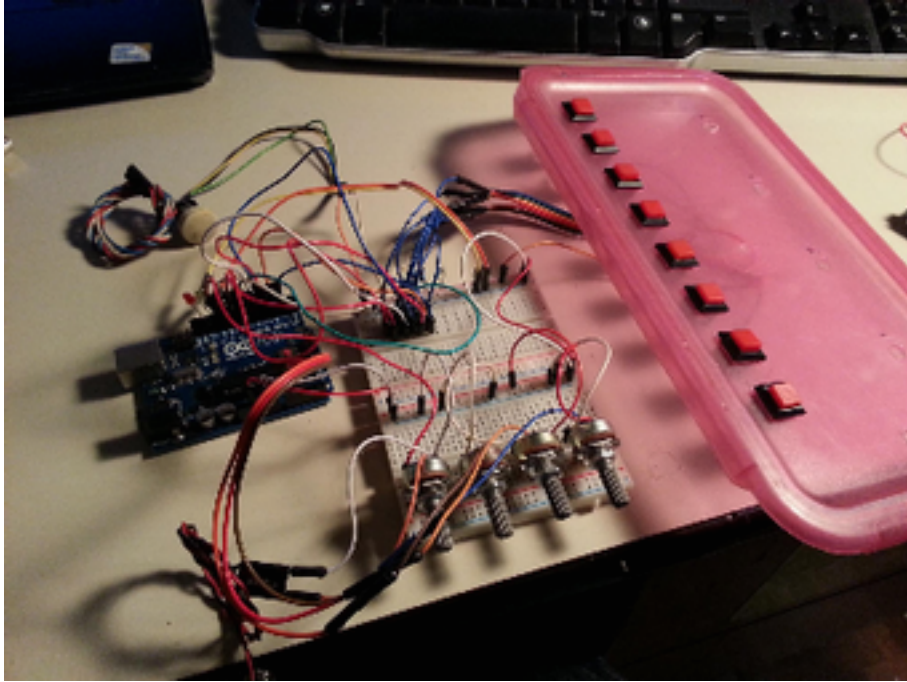
- Coordinarnos tanto en tiempo como en espacio
- Discernir entre errores de código y errores de hardware, saber cuando está fallando cada cosa
- Controlar tensiones para evitar quemar componentes
- Nuestros pulsadores funcionaban a la inversa que los del ejemplo, tuvimos que hacer cambios en el código (Invertir booleanos) para que funcionaran con la misma lógica

Fotos

*Presentando los botones y los potenciómetros al tupper.
Nunca más se iban a separar.*



"Pero, esto lo probaron antes de soldar?". Gracias Jose Luis :)



*Siempre tomar medidas de seguridad.
Evite accidentes.*



Y finalmente, el APC concebido...

