

Universidade Federal do Rio Grande do Sul
Instituto de Informática

INF01062 - Computação e Música - Tópicos Especiais em Computação XVI

Prof. Marcelo de Oliveira Johann

Prof. Marcelo Soares Pimenta

Construção de Theremin com
uso sensores fotoelétricos

João Luiz Grave Gross

Porto Alegre, 13 de dezembro de 2013.

1 Theremin

O theremin é um instrumento musical eletrônico criado pelo russo Lev Sergeivitch Termen, físico e ex-agente da KGB [1]. Ele foi desenvolvido dentro de um programa de pesquisa de sensores de proximidade financiado pelo governo russo, sendo finalizado em 1920. Porém, a patente só foi realizada em 1928, após León Theremin se refugiar nos EUA [8].

A ideia para o instrumento surgiu a partir de sua percepção da interferência do som em um aparelho de rádio ao aproximar a mão da antena. Assim, baseando-se neste conceito, construiu uma caixa de válvulas – como as comuns aos rádios da época –, adaptando-lhe duas antenas: uma do lado esquerdo, na horizontal, e outra do lado direito, na vertical. Dessa forma, a partir da maior ou menor proximidade das mãos relativamente a estas antenas, altera-se o campo eletromagnético destas, de forma que se pode variar a intensidade através da primeira antena, e portanto, controle de volume, e através da segunda pode-se variar a frequência, ou seja, tonalidades graves e agudas.

O som do theremin foi bastante utilizado na indústria cinematográfica, principalmente em filmes de ficção científica e de drama, por produzir um som estranho e misterioso. Também esteve presente na televisão, em jogos de videogame e na música (bandas de rock psicodélico principalmente).

2 Proposta

O theremin, em sua concepção original, possui duas antenas metálicas e osciladores, responsáveis por controlar a tonalidade e amplitude sonoras. Porém, propõem-se uma forma alternativa de construção do Theremin utilizando sensores fotoelétricos ao invés de antenas metálicas. A diferença das duas implementações é que na segunda, as variações de frequência e amplitude serão realizadas de acordo com a exposição de luz nos sensores, e de igual forma ao Theremin clássico, o theremista (performista) não terá qualquer contato com o instrumento musical para controlá-lo.

Os osciladores de tonalidade, por sua vez, serão substituídos por uma unidade microcontrolada programável, que oferece recursos para emular o comportamento dos osciladores. Já o controle de amplitude será realizado com um circuito amplificador controlado por tensão, ou seja, a medida que o sensor fotoelétrico para controle de amplitude for exposto a diferentes intensidades luminosas, uma tensão corresponde à exposição será injetada no amplificador, alterando seu ganho.

2.1 MSP430G2 LaunchPad

A implementação proposta utiliza o kit de desenvolvimento MSP430G2 LaunchPad da Texas Instrumental, um circuito microcontrolador programável que permite emular os osciladores do Theremin [2]. Esse kit possui uma plataforma open-source para prototipação de circuitos eletrônicos chamada Energia, que utiliza o compilador mspgcc e é baseada nos frameworks Wiring e Arduino.

Energia ainda inclui um ambiente de desenvolvimento integrado (IDE), baseada em Processing – o mesmo no qual Arduino se baseia – que facilita a escrita e compilação dos programas e também sua gravação no circuito microcontrolador.

Optou-se em utilizar este kit de desenvolvimento, ao invés do Arduino, por possuir os mesmos recursos necessários ao desenvolvimento do projeto que o Arduino oferece e também por ser mais barato – US\$ 9,99 na época.

2.2 Implementação

Tanto para a frequência quanto para a amplitude foram utilizados sensores fotoelétricos para capturar variações de tensão nas malhas do circuito de acordo com os movimentos das mãos do theremista. Cada sensor está presente em uma configuração de circuito diferente uma da outra. O circuito do sensor de controle de tonalidade é como segue:

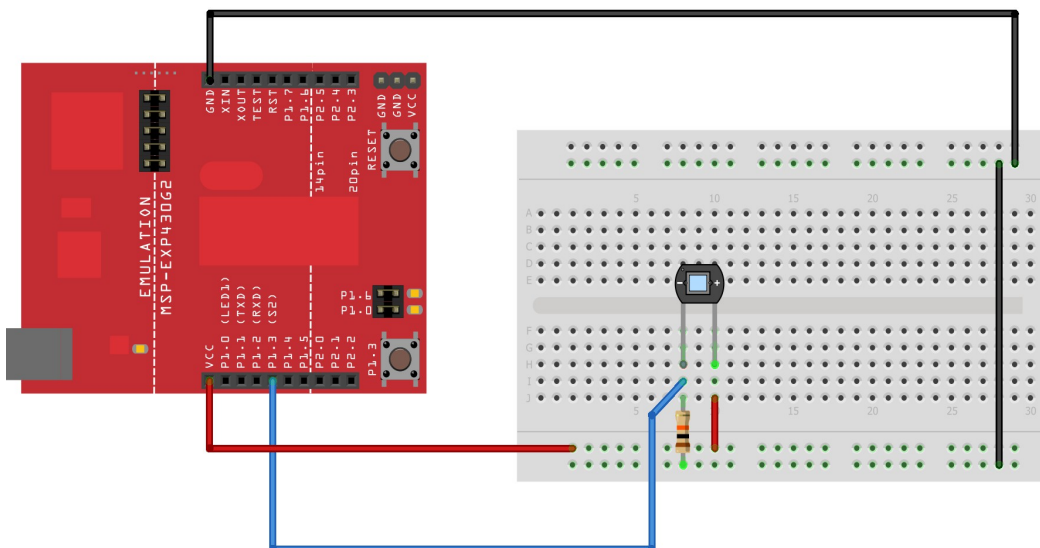


Figura 1: Circuito com sensor fotoelétrico [6].

Neste circuito, a medida que o theremista aproxima a mão do sensor fotoelétrico o microcontrolador programável do kit de desenvolvimento, previamente configurado, percebe as variações de tensão e as mapeia para tons de frequência.

O circuito para capturar as variações de amplitude é semelhante ao mostrado na figura 1, porém as variações de tensão sobre o sensor não são capturadas pelo microcontrolador. Ao invés disso, o sinal é aplicado a uma das entradas de amplificador operacional, com o qual conseguimos construir um amplificador controlado por tensão, ou seja, um amplificador com ganho variável.

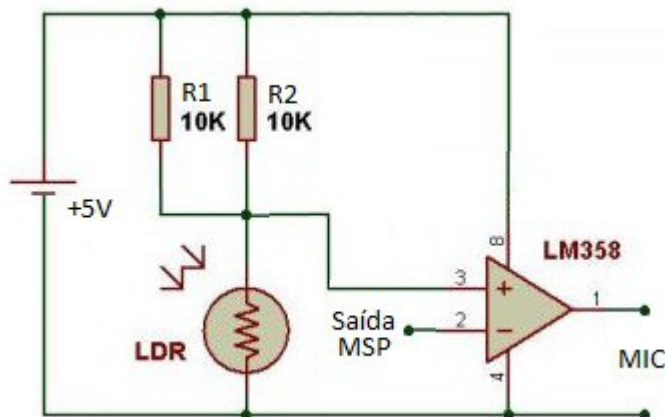


Figura 2: Amplificador controlado por tensão [3 e 4].

A tensão aplicada sobre o sensor fotoelétrico para controle de amplitude é aplicada em Vgain (pino 3), enquanto que o sinal de frequência é aplicado no pino 2 (saída do MSP). O sinal aplicado ao pino 2 é amplificado e sai no pino 1 do amplificador operacional, ao qual é conectado um alto-falante.

A seguir podemos observar o esquema do circuito completo, com as duas partes integradas (controle de frequência e amplitude).

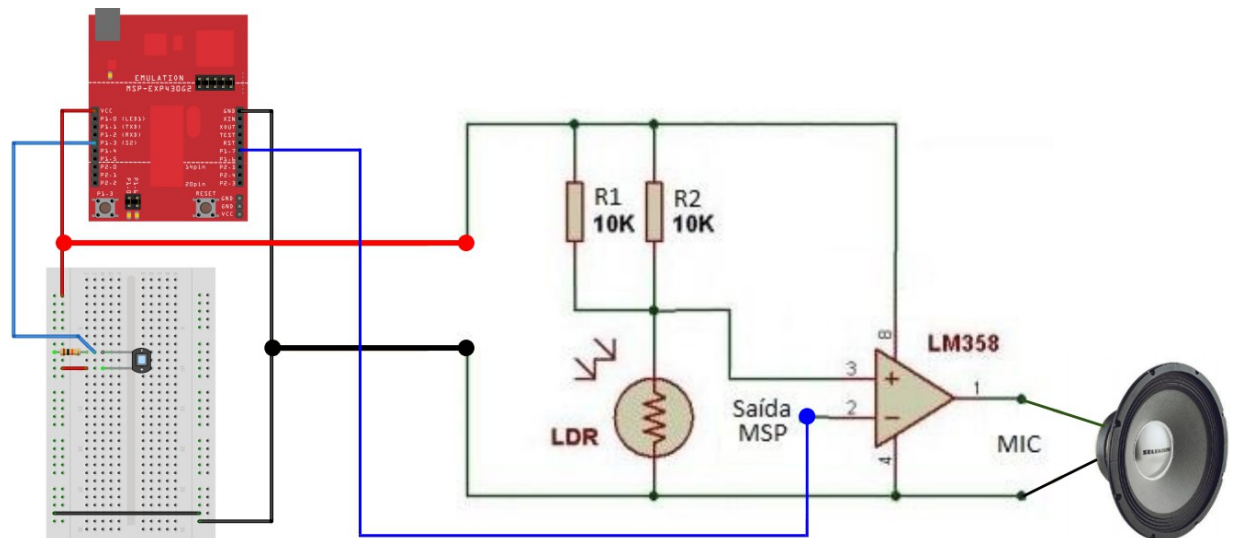


Figura 3: Circuito completo.

2.2.1 Programa para controle de frequências

Como comentado nas sessões anteriores a geração dos tons foi realizado pelo microcontrolador MSP430G2. Para este foi desenvolvido um programa responsável por capturar os níveis de tensão aplicados em uma porta do microcontrolador e mapeá-los para tons de frequência.

A porta do microcontrolador, nada mais é do que um pino, setado como pino de input ou output de dados. Para a captura de dados, o pino 2 foi utilizado como pino de input (como mostrado na figura 3). Nele foram capturamos as variações de tensão aplicadas sobre o sensor fotoelétrico responsável pelo controle de frequência.

O pino 15, por sua vez, foi setado como pino de output. Neste pino são colocados mapeamentos de frequência que posteriormente são amplificados.

O programa responsável este mapeamento de níveis de tensão em tons, é constituído de 3 partes: criação de variáveis, inicialização e laço de execução. Todas são mostradas e explicadas a seguir.

```
1  /* Variáveis */
2  const int sensorFreq = A0; // seleciona pino de input para frequencia
3  const int freqPin = 15;    // saída de frequencia
4
5  int freqHigh = 0;
6  int freqLow = 999999;
7  int freqValue = 0;        // armazena valor capturado no sensor de frequencia
```

Figura 4: Declaração de variáveis.

Nesta etapa variáveis são criadas para que se possa acessar os dados disponíveis nos pinos, bem como variáveis para a execução dos trechos de código que seguem.

```
9  void setup() {
10      // seta pinos como input ou output
11      pinMode(sensorFreq, INPUT);
12      pinMode(freqPin, OUTPUT);
13
14      // Calibra o sensor de frequencia por 5 segundos
15      while(millis() < 5000) {
16          freqValue = analogRead(sensorFreq);
17
18          // Registra o maior valor do sensor
19          if(freqValue > freqHigh) {
20              freqHigh = freqValue;
21          }
22
23          // Registra o menor valor do sensor
24          if(freqValue < freqLow) {
25              freqLow = freqValue;
26          }
27      }
28  }
```

Figura 5: Inicialização.

Nesta etapa os pinos definidos na inicialização são setados como pinos de input ou output. Também é realizada a calibração do sensor de frequência, para estabelecer o range de tensões que serão capturadas na entrada.

```
31 void loop() {  
32     // Lê a frequência do pino A0  
33     freqValue = analogRead(sensorFreq);  
34  
35     //map the sensor values to a wide range of pitches  
36     //Adjust the values below to conform the maximum and  
37     //minimum numbers you get from the sensor  
38     int pitchFreq = map(freqValue, freqLow, freqHigh, 20, 1000);  
39  
40     //toca tom  
41     tone(freqPin, pitchFreq, 20);  
42  
43     //espera  
44     delay(10);  
45 }
```

Figura 6: Laço de execução.

Por fim, um laço fica em execução, capturando os níveis de tensão no pino de entrada e realizando o mapeamento destes para tons, pela função map. A função tone é utilizada para colocar no pino de saída a frequência mapeada.

3 Resultados Obtidos

A montagem se mostrou bastante simples. Tive certa dificuldade em encontrar os componentes corretos, porém com eles em mãos a montagem foi rápida. Também não tive que realizar um alto investimento para a construção do circuito. Considerando o microcontrolador, resistores, sensores fotoelétricos, o amplificador operacional e o alto-falante, juntos eles chegaram a cerca de R\$ 30,00.

A responsividade dos sensores fotoelétricos foi satisfatória. O sensor para controle de frequência, por exemplo, teve uma responsividade muito boa, capturando um bom espectro de frequências, de modo a ser fácil destoar os tons reproduzidos pelo alto-falante a medida que se aproxima ou se afasta a mão do sensor. Porém o sensor de controle de amplitude possui uma sensibilidade muito alta. Ele responde ou maximizando o som no alto-falante ou cortando ele por definitivo. Pelo menos podemos observar que o sensor funciona, embora com sensibilidade muito alta.

4 Conclusões

Com este trabalho tive a oportunidade de trabalhar com hardwares que até então não tive contato no curso de ciência da computação. Há muito havia feito um curso técnico em eletrônica e com este trabalho pude pôr novamente em prática alguns dos conceitos explorados no curso técnico.

Não apenas isso, mas também a possibilidade de poder simular um instrumento musical dos anos 30 com apenas alguns componentes de hardware e pequenos trechos de código, nos mostra como são versáteis as tecnologias que hoje temos à disposição e facilidade de se ter acesso a elas.

Referências

- [1] Theremin. Disponível em: <http://en.wikipedia.org/wiki/Theremin>. Acesso em: 01/11/2013.
- [2] MSP430G2 LaunchPad – Texas Instrumental. Disponível em: <http://www.ti.com/tool/msp-exp430g2>. Acesso em: 01/11/2013.
- [3] Amplificador controlado por tensão. Disponível em: <http://hackmeopen.com/2011/04/voltage-controlled-amplifier/>. Acesso em: 01/11/2013.
- [4] JFET. Disponível em: <http://en.wikipedia.org/wiki/JFET>. Acesso em: 01/11/2013.
- [5] Energia. Disponível: <http://energia.nu/>. Acesso em: 01/11/2013.
- [6] Sensor fotoelétrico. Disponível em: <http://en.wikipedia.org/wiki/Photoresistor>. Acesso em: 01/11/2013.
- [7] Função “Tone”. Disponível em: <http://arduino.cc/en/Tutorial/Tone#>. Acesso em: 01/11/2013.
- [8] Construção de um Theremin. Disponível em: http://www.ifi.unicamp.br/~lunazzi/.../MarianaC_Kleinke_F609_RF1.pdf%E2%80%8E. Acesso em: 04/11/2013.