RELATÓRIO FINAL - GERADOR DE SINAIS

Nome: Giovanni Pasa, João Pedro de Souza, Luiza Araújo, Matheus Soares
Subsistema: Controle
Data: 09/01/2021

1. CIRCUITO

Inicialmente, visualizamos exemplos de circuitos de geradores de sinais disponíveis na internet para ter uma noção geral de como eles são realizados. Então optamos por utilizar os resistores em escada do tipo R-2R para converter as saídas digitais do Arduino em sinais analógicos capazes de formar as ondas desejadas. Decidimos também alterar o formato da onda através de um botão, alterar a frequência por meio de um potenciômetro e imprimir essas duas informações em um display 16x2.

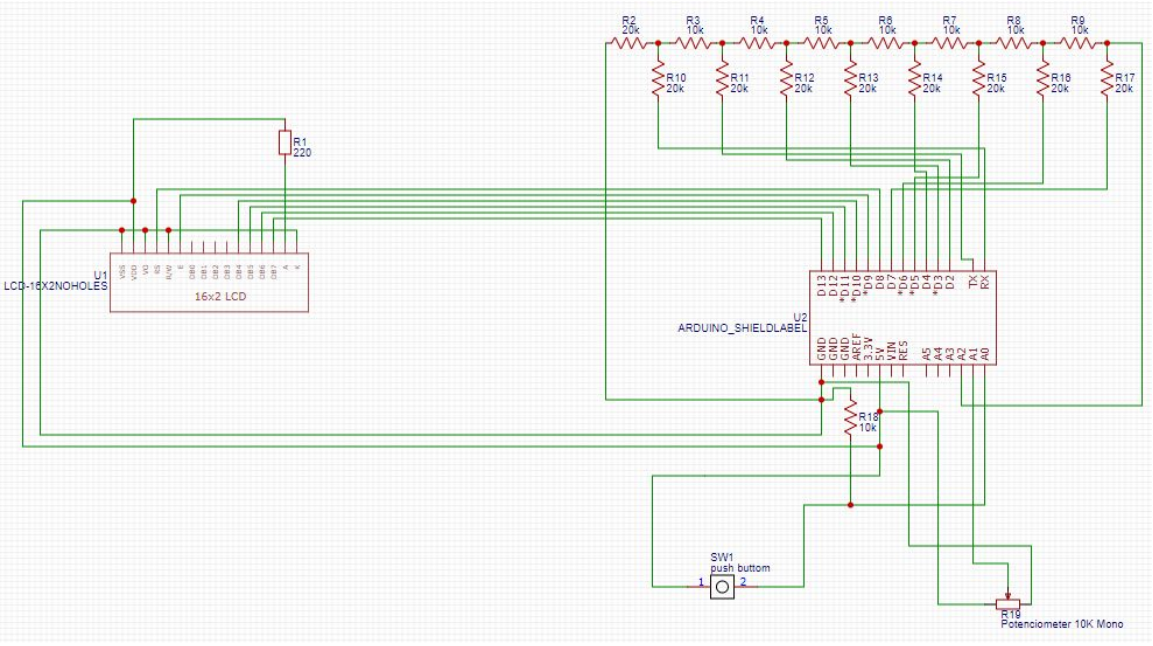

Figura 1: Circuito Gerador de Sinais

Os resistores em cadeia R-2R funcionam como um DAC ao receber 8 bits de informação através das saídas digitais do Arduino e transformá-los em um sinal

analógico de 0V a 5V. Essa conversão obedece à seguinte equação: $V_{out} = V_{ref} * \frac{Value}{2^8}$, em que $V_{ref} = 5V$ e Value é igual ao valor digital transmitido pelo Arduino (0 a 255). Foram utilizados 7 resistores de $10k\Omega$ e 9 resistores de $20k\Omega$ para essa montagem.

O botão foi utilizado para alternar o formato da onda na seguinte ordem: quadrada, triangular, dente de serra e senoidal. Também utilizamos um potenciômetro de $10k\Omega$ para alterar a frequência da onda entre $1Hz$ e $200Hz$.

A fim de exibir o formato da onda atual e a frequência escolhida, utilizamos um LCD 16x2 conectado nos pinos 8 a 13 do Arduino.

2. PCB

Utilizando o software easyEDA, a criação da Placa de Circuito Impresso tornou-se possível. A montagem foi baseada no circuito projetado e tem apenas uma camada (TopLayer).

As configurações do design das trilhas são: Largura de Trilha e Apuramento iguais a 1, Via Diâmetro igual a 0.8 e Diâmetro do Furo da via igual a 0.4. Essas configurações citadas acima garantem uma melhor eficiência e condutividade da PCB. É recomendado utilizar trilhas mais grossas em circuitos de baixas frequências, já em circuitos com frequências altas é aconselhado que as trilhas sejam o mais finas possíveis, pois pode surgir a capacitância parasita.

Logo após ter sido feito o roteamento das trilhas, utilizando a ferramenta CopperArea, foi selecionada toda a área da PCB para que o cobre cubra todo o aterramento da placa. Essa etapa é bastante importante para que ela tenha uma melhor condutividade e seja menos sujeita a ruído.

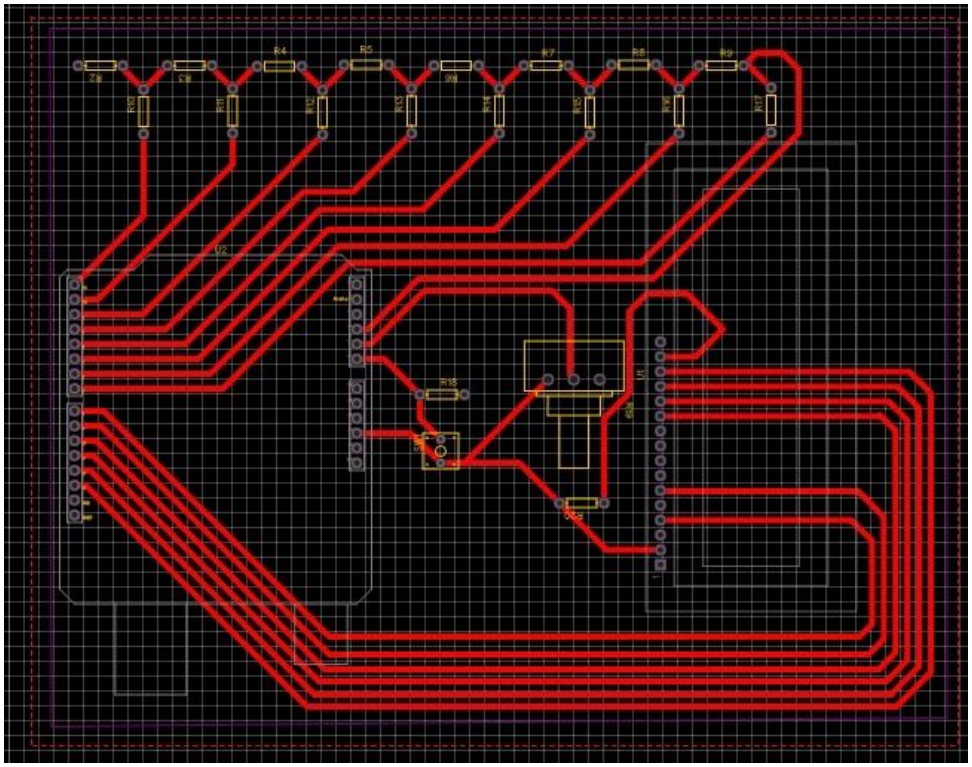

Figura 2: PCB

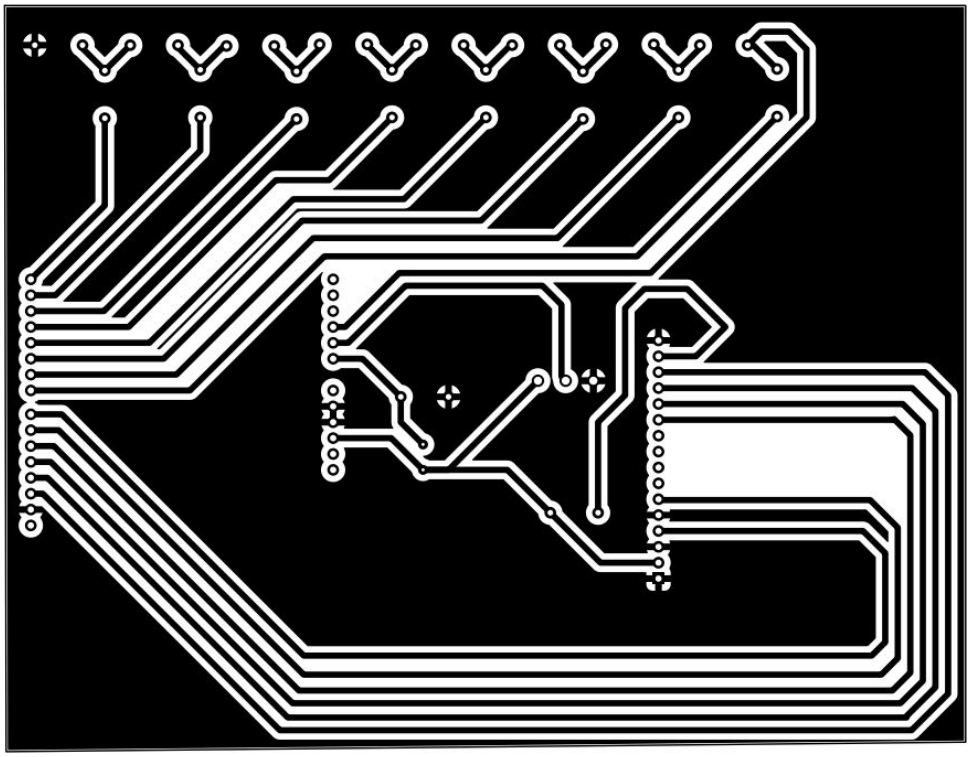

Figura 3: PCB

3. SIMULAÇÃO E PROGRAMAÇÃO

A simulação e a programação foram feitas em conjunto no Tinkercad, a fim de que toda modificação no programa pudesse ser imediatamente testada. O projeto foi então dividido em três partes principais - botão + potenciômetro, display LCD e DAC - que foram reunidas no final.

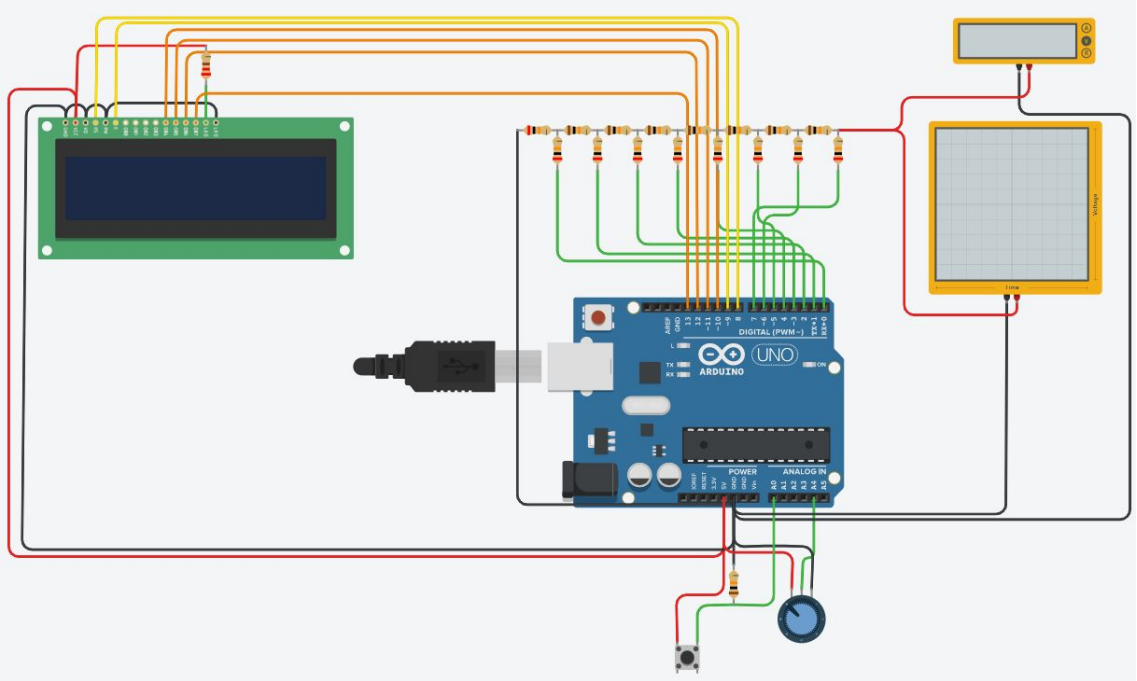

Figura 4: Circuito no Tinkercad

Em primeiro lugar, foram desenvolvidas as lógicas do botão e do potenciômetro. Quando o botão é pressionado, um contador aumenta indicando que deve-se alterar o formato de onda. Porém, caso ele seja mantido pressionado, o contador não continua aumentando, de modo a ser necessário retirar o dedo do botão e apertar novamente para alterar mais uma vez o formato. O potenciômetro, por sua vez, indica um valor de 0 a 1023 que será armazenado dentro de uma variável e esta servirá de base para definição da frequência que será utilizada nas funções de onda.

Em segundo lugar, o display LCD foi configurado. Ao iniciar o programa é mostrada uma mensagem “Formula Tesla” por três segundos e então passa a ser exibida a mensagem “Waveform Gen.” e o tipo da onda. Nossa intenção inicial era exibir

também a frequência escolhida, porém tivemos que abrir mão dessa ideia, uma vez que a função que realizava essa operação acabava causando certo atraso na configuração que escolhemos (configuração essa que será explicada a seguir).

Por fim, foi necessário programar o DAC construído a partir da cadeia de resistores e criar as funções específicas para cada formato de onda. Dois caminhos distintos foram cogitados: utilizar a função delay() e utilizar o Timer1 do Arduino.

A grande vantagem do Timer1 é não precisar travar o programa a cada mudança de valor de saída e permitir um funcionamento mais limpo do circuito. Nós conseguimos implementar a visualização da frequência e a formação das ondas quadrada e dente de serra nessa configuração, porém elas apresentavam irregularidades, além de não termos conseguido implementar a onda triangular.

Por outro, a função delayMicroseconds() nos permitiu gerar ondas quadradas, dentes de serra, triangulares e mesmo senoidais confiáveis e estáveis de $1Hz$ até $200Hz$. Por isso, apesar de suas desvantagens, optamos por esse caminho para a versão final da programação.

Desse modo, foram criadas quatro funções, uma para cada formato de onda. Essas funções são compostas de laços for, dentro dos quais o valor de saída é modificado e enviado às portas digitais 0 a 7 através do comando PORTD, seguido de um delay proporcional ao valor da frequência. Caso a variável serialView seja verdadeira, o sinal passa a ser enviado para o *Serial Plotter* do Arduino através da função Serial.println().

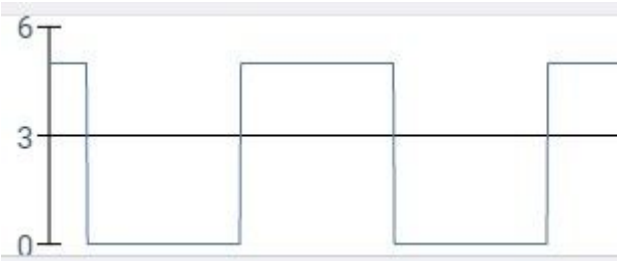

Figura 5: Onda Quadrada

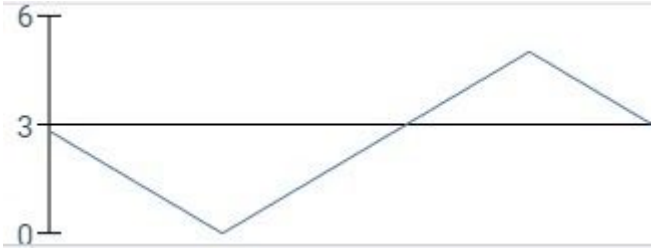

Figura 6: Onda Triangular

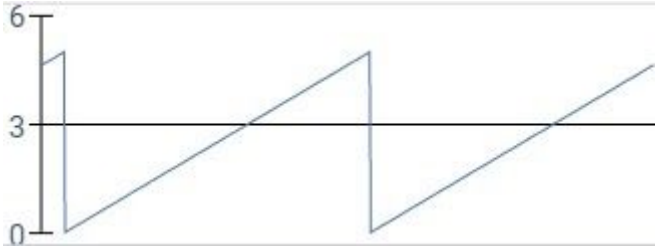

Figura 7: Onda Dente de Serra

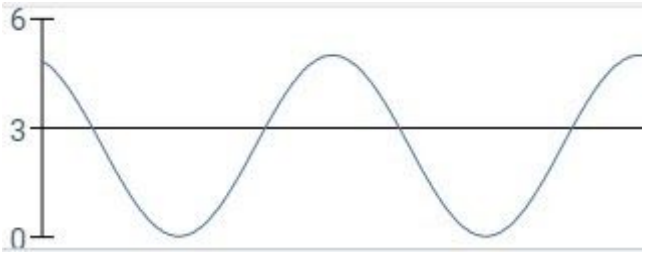

Figura 8: Onda Senoidal

4. MATERIAIS

A maioria dos materiais será adquirida na loja Eletrogate, devido a sua confiabilidade, preço e localização (BH) que possibilitará uma entrega rápida. Apenas os resistores de $20k\Omega$, a placa de fenolite e o percloreto de ferro não são vendidos na loja e, por isso, serão adquiridos na loja Eletrônica Santa Efigênia, no centro de Belo

Horizonte.

Tabela 1 - MATERIAIS E VALORES				
Componentes	Características	Preço individual	Quantidade	Preço total
Arduino Uno	R3	$R\$\ 59,90$	1	$R\$\ 59,90$
Barra de pinos	Fêmea	$R\$\ 0,0725$	10	$R\$\ 2,90$
Barra de pinos	Macho	$R\$\ 0,0725$	18	$R\$\ 2,90$
Botão	PBS-16A	$R\$\ 5,90$	1	$R\$\ 5,90$
Adaptador P2	$3.5mm$	$R\$\ 7,90$	1	$R\$\ 7,90$
Display Lcd	16x2	$R\$\ 28,90$	1	$R\$\ 28,90$
Jumpers	Macho/Fêmea	$R\$\ 0,445$	8	$R\$\ 8,90$
Placa de Fenolite		$R\$\ 20,00$	1	$R\$\ 20,00$
Potenciômetro	$10k\Omega$	$R\$\ 3,50$	1	$R\$\ 3,50$
Resistor	220Ω	$R\$\ 0,20$	1	$R\$\ 2,00$
Resistor	$10k\Omega$	$R\$\ 0,20$	8	$R\$\ 2,00$
Resistor	$20k\Omega$	$R\$\ 0,20$	9	$R\$\ 2,00$
			Impressão PCB em papel fotográfico	$R\$\ 5,00$
			Solução de Percloreto de Ferro	$R\$\ 15,00$
			Frete (1 dia útil)	$R\$\ 14,90$
			Valor Total	$R\$\ 177,60$

Os resistores são vendidos em pacotes com 10 unidades, os jumpers em pacotes com 20 unidades e a barra de pinos em pacotes com 40 pinos, por isso o preço total é maior do que Preço individual x Quantidade.

5. PLANEJAMENTO DA FABRICAÇÃO

Após a compra dos materiais, o primeiro passo é a construção da placa PCB. O método escolhido consiste em imprimir o circuito da placa em papel fotográfico usando uma impressora a laser (as impressões de gráfica costumam ser a laser). Depois disso, usa-se uma fonte de calor para transferir o toner da tinta do papel fotográfico para a placa de fenolite. Para isso, a forma mais simples e eficaz é usando um ferro de passar roupa. Depois disso, é necessária a corrosão das linhas de cobre usando Percloreto de Ferro, que irá corroer apenas a parte da placa que não estiver coberta por tinta. Finalmente, é necessário apenas lixar com uma lixa bem fina para que as trilhas de cobre se tornem visíveis e a tinta saia por completo da placa. Apesar de ser um método de boa eficácia, há a possibilidade de erro, portanto, o valor estipulado é o de duas impressões em papel fotográfico A4 em uma gráfica.

Após a fabricação da PCB, o próximo passo é conectá-la ao Arduino e testar o funcionamento do circuito. Enquanto uma parte do grupo se concentra em resolver erros que provavelmente ocorrerão envolvendo desde a programação até uma solda incorreta, passando por possíveis falhas na impressão da placa, outra parte estará responsável pela construção do encapsulamento.

A PCB e o Arduino serão armazenados em uma caixa de madeira com buracos para o display LCD, o botão, o potenciômetro, a saída de sinal (através do adaptador P2) e a saída USB (para manipulação do código, caso necessário, e para a visualização de sinal através do *Serial Plotter*).