

Projeto 2 - Processamento de Sinais Digitais

Daniel Tatsch, Nelson Alves, Schaiana Sonaglio

Engenharia de Telecomunicações

IFSC - Instituto Federal de Santa Catarina, São José, SC

Julho de 2018

1 Introdução

Este relatório apresenta a implementação de um detector DTMF (do inglês *Dual-Tone Multi-Frequency*), que utiliza os tons de duas frequências (multifrequência) para representar as teclas de telefones.

As frequências destes tons e suas combinações são mostradas na Tabela 1, onde são combinadas as frequências altas com as baixas, formando o *dual tone* para cada tecla apresentada. A frequência de 1633 *hertz*, utilizada nas letras “A”, “B”, “C” e “D”) é utilizada apenas internamente, para realização de testes.

O tom que é enviado à central é obtido através da soma das senóides das frequências correspondentes da linha e da coluna onde a tecla se posiciona. Na central, o sinal é analisado para detectar a presença destas duas frequências, a fim de saber se alguma tecla foi pressionada.

Tabela 1 - Tabela DTMF

Hz	1209	1336	1477	1633
697	1	2	3	A
770	4	5	6	B
852	7	8	9	C
941	*	0	#	D

Estas frequências foram escolhidas pela baixa probabilidade de a voz humana combiná-las e reproduzi-las.

O detector DTMF é utilizado porque todas as frequências das teclas são transmitidas em um mesmo canal, sendo necessário separá-las em faixas, a fim de identificar corretamente qual tecla foi pressionada.

2 Metodologia

O detector DTMF deste projeto, representado na Figura 1, foi implementado da seguinte forma:

1) Primeiramente, foram gerados áudios, através do software Audacity®, que continham as teclas de um teclado de telefone;

2) Como o sinal gerado neste *software* possuía frequência de amostragem (f_a) de 44100 *Hz* e o detector de DTMF utilizava frequência de 44100/N *Hz*, sendo N = 12 (devido à especificação de projeto), foi necessário incluir um filtro *antialiasing* - *low pass* com frequência de corte $f_c = (44,1/2)/N$ *kHz*;

3) Como as frequências de amostragem citadas no item 2 eram diferentes, foi necessário um circuito (*downsampling*) para subamostrar o sinal com $f_a = 44100$ *Hz*,

este circuito serve para reduzir a frequência de amostragem do sinal, caso a frequência necessitasse ser aumentada, seria utilizado um circuito de *upsampling*;

4) Após passar pelo circuito de *downsampling*, o sinal passa por dois filtros *IIR* (do inglês *Infinite impulse response*) Chebyshev tipo II, um passa baixas e um passa altas, a fim de separar as frequências das linhas do teclado (frequências mais altas) e das colunas (frequências mais baixas); para ambos os filtros, a frequência de corte foi $f_c = 1067$, obtida através da raiz quadrada da maior frequência das linhas (941) multiplicada pela menor frequência das colunas (1209);

5) Depois de passar pelos filtros separando as frequências em altas em baixas, o sinal passou por filtros passa faixa, cada um com a frequência de passagem correspondente à linha que o filtro representava. Os filtros das linhas foram implementados como *IIR* e os da coluna como *FIR* (do inglês *Finite impulse response*). As frequências de passagem foram calculadas através das frequências de cada colunas multiplicadas por $\delta = 1,5\% + 2 \text{ Hz}$, gerando assim a faixa de frequência do filtro;

6) Após passar pelos filtros passa faixa, o sinal passa por um retificador, que transforma a parte negativa da onda em positiva. Esta retificação foi feita utilizando o bloco "*abs*" da *software* Simulink®;

7) Após passar pelo retificador, o sinal de cada linha ou coluna passa por um filtro passa baixas *IIR Butterworth* de ordem 2, a fim de detectar a envoltória do sinal. A frequência de corte deste filtro foi calculada através da Equação 1, onde o valor de τ era de 5 ms.

$$f_c = \frac{1}{2\pi\tau} \quad f_c = 31,83 \text{ Hz} \quad (1)$$

8) Após detectar a envoltória do sinal, foi utilizado o bloco "*relay*", da ferramenta Simulink®, para determinar, através de um limiar de decisão, quais amplitudes (em *volts*) que correspondem a cada tecla pressionada;

9) Depois de cada tecla separada, foram feitas conversões de *volt* para binário e de binário para número decimal, para enfim vermos cada tecla ser representada.

Algumas observações sobre o projeto:

1) Para as teclas 1, 2, 7, 8, 0, que são geradas pelas linhas L1, L3 e L4 e colunas C1 e C2, devido às especificações de projeto, foram implementados filtros passa faixa que precisaram necessariamente atender às especificações de banda de passagem e rejeição, para as demais linhas e colunas, foram gerados filtros com as frequências centrais das linhas e colunas correspondentes, sem nenhum ajuste;

2) Para as conversões de *volt* para binário, as seguintes associações foram feitas: 1 => 0001 a 9 => 1001, 0 -> 1010, * => 1011, => 1100;

3) Para tratamento de erros, os números 13 e 14 foram reservados, o 13 indica se apenas uma frequência do *dual tone* foi detectada, o 14 caso fossem ativados 2 linhas e 1 coluna ou 1 linha e duas colunas;

5) Os cinco sinais de DTMF correspondentes às cinco teclas indicadas para este projeto tiveram duração de 0,5 segundos de execução cada;

6) Quando foi gerado o sinal DTMF, correspondente à sequência de teclas "123456789*0", a duração do tom foi de 200 ms e pausa de 200 ms para cada tecla, conforme a especificação do projeto.

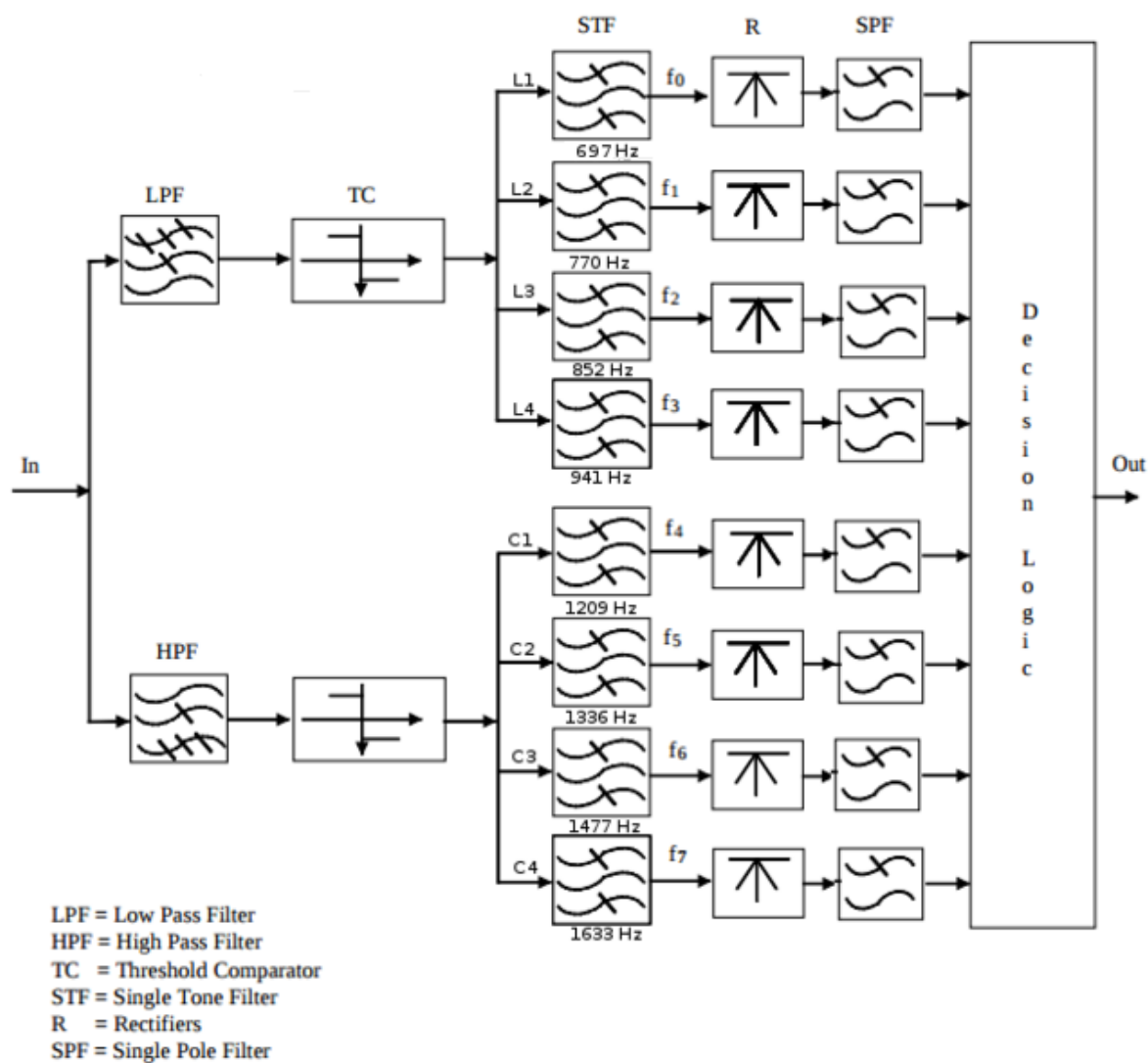


Figura 1 - Especificações do discriminador de frequência. Fonte: página da disciplina.

3 Análise dos Resultados

Após a explicação da metodologia do projeto, será feita uma breve análise dos resultados obtidos, após implementação do detector DTMF.

Os resultados ilustrados nas figuras a seguir são correspondentes às teclas 1, 2, 7, 8, 0, mas o detector funcionou corretamente para todas as demais teclas e para teclas sozinhas.

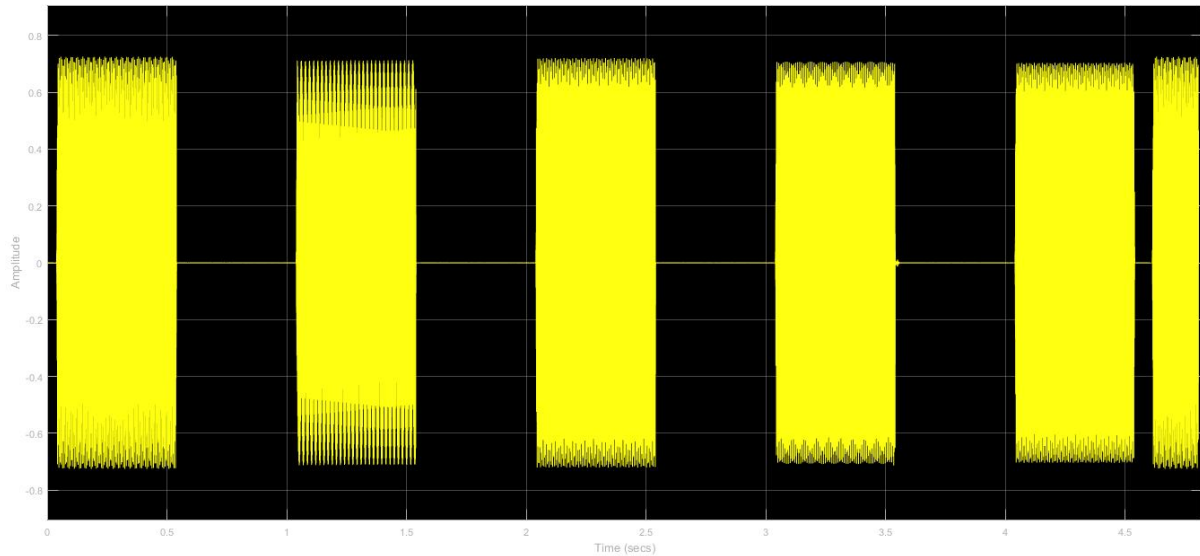


Figura 2 - Circuito antes do circuito de downsample.

Na Figura 2, é mostrado o circuito antes de fazer o processo de *downsample*.

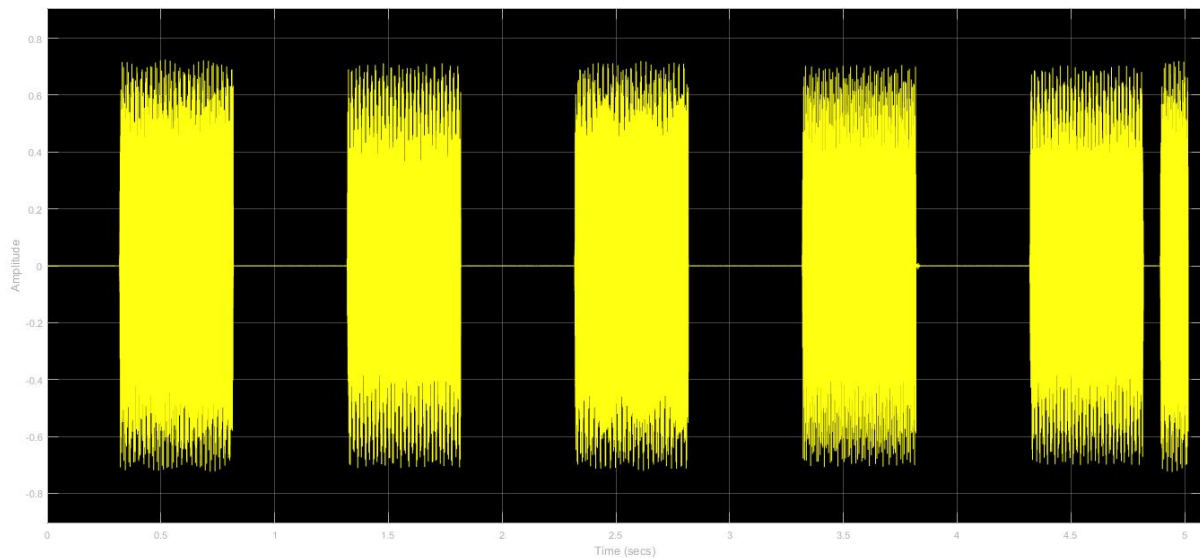


Figura 3 - Circuito depois do circuito de downsample.

Na Figura 3, é mostrado o circuito depois de fazer o processo de *downsample*, onde é possível ver como a redução da frequência de amostragem muda o sinal, que passa a ter menos "pontos".

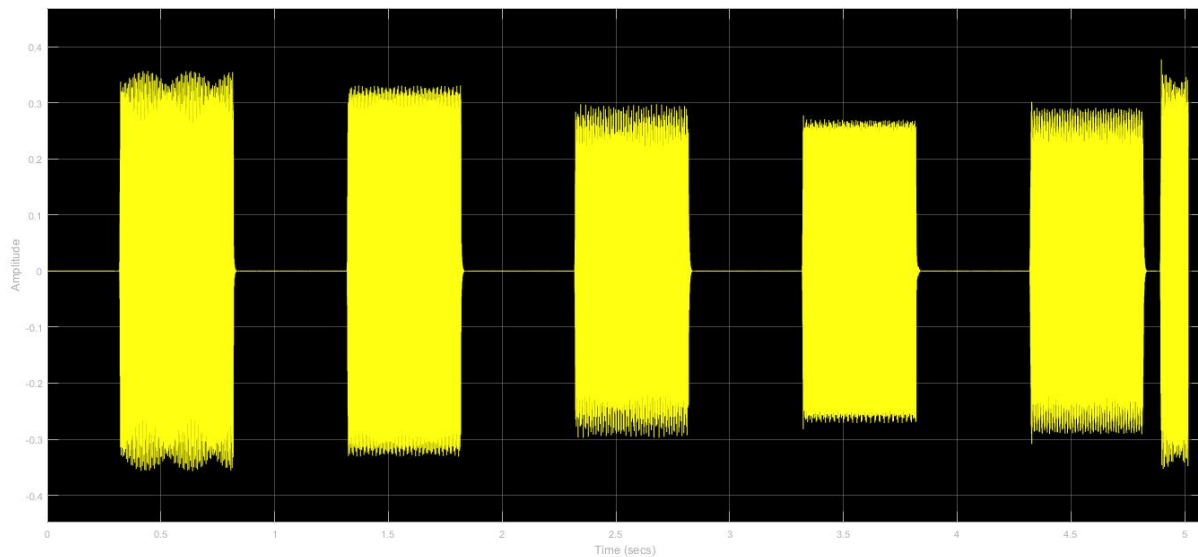


Figura 4 - Circuito depois do filtro passa altas de entrada.

Na Figura 4, é mostrado o sinal após passar pelo filtro passa altas de entrada, para depois ir para os filtros passa faixa das colunas.

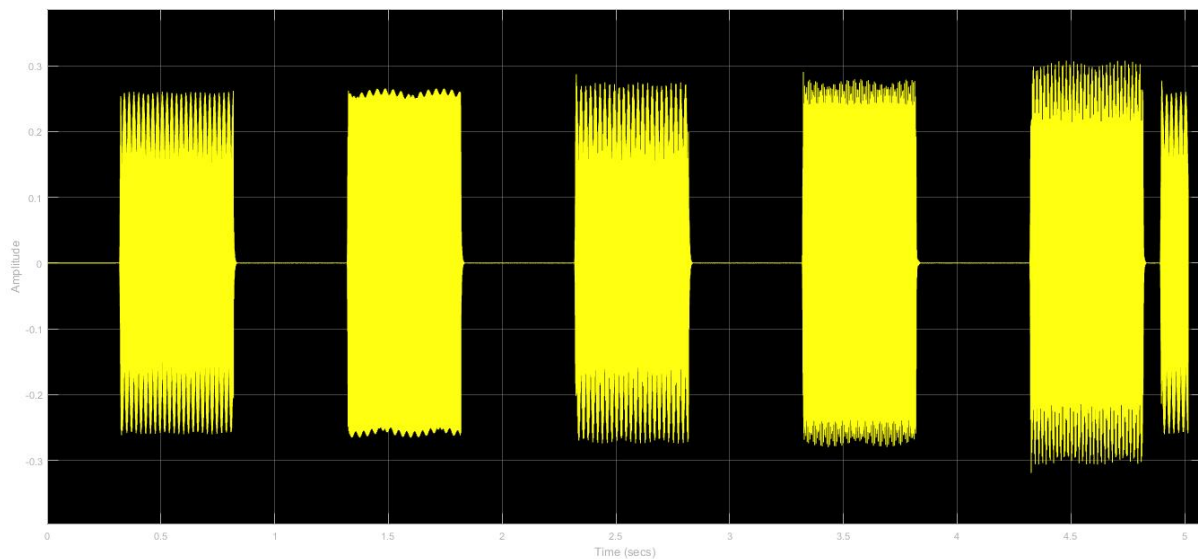


Figura 5 - Circuito depois do filtro passa baixas de entrada.

Na Figura 5, é mostrado o sinal após passar pelo filtro passa baixas de entrada, para depois ir para os filtros passa faixa das linhas.

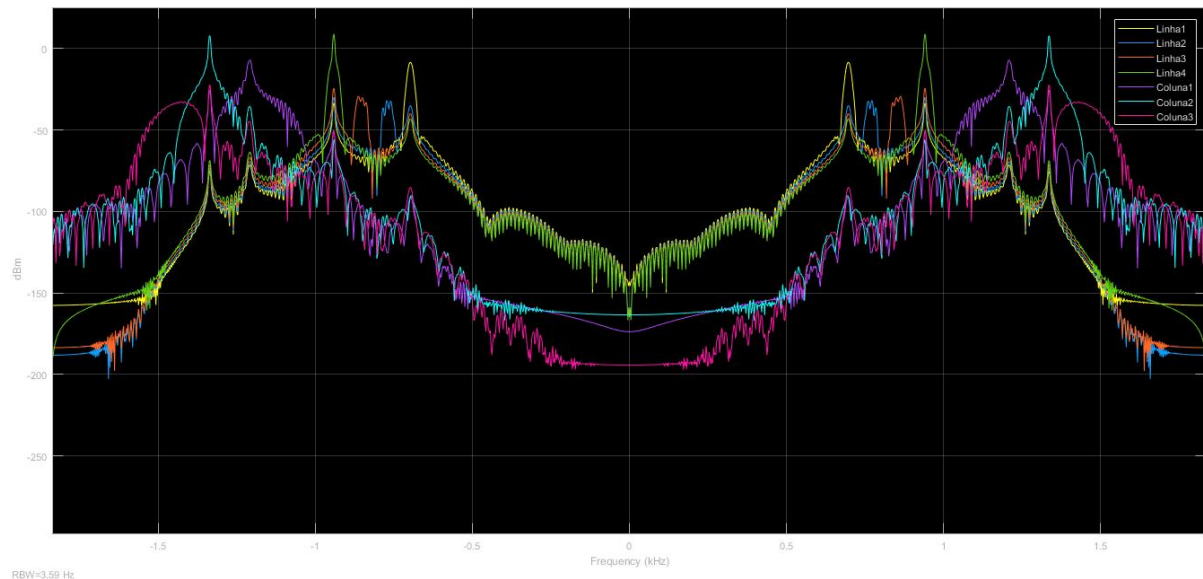


Figura 6 - Espectro depois dos filtros passa faixa.

Na Figura 6, é mostrado o espectro do sinal depois de ele ter passado pelos filtros passa faixa de linha e colunas.

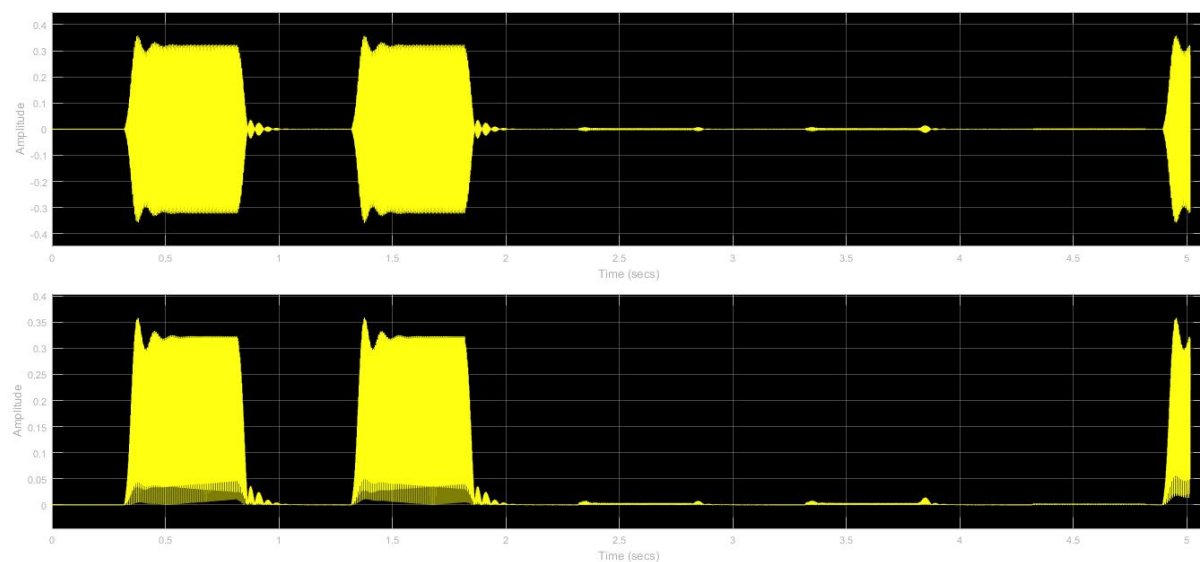


Figura 7 - Circuito depois do retificador.

Na Figura 7, é mostrado o sinal antes e depois de ter sido retificado, onde a parte negativa do sinal também ficou positiva.

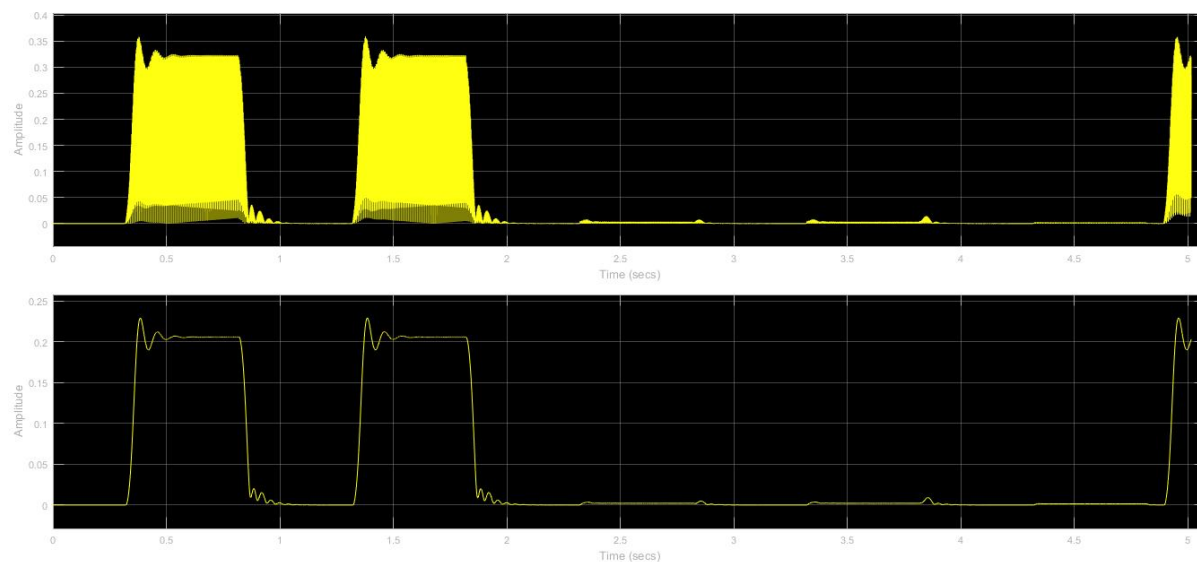


Figura 8 - Circuito antes e depois do filtro passa baixas de saída - detector de envoltória.

Na Figura 8, é mostrado o sinal antes e depois do filtro passa baixas que detecta envoltórias.

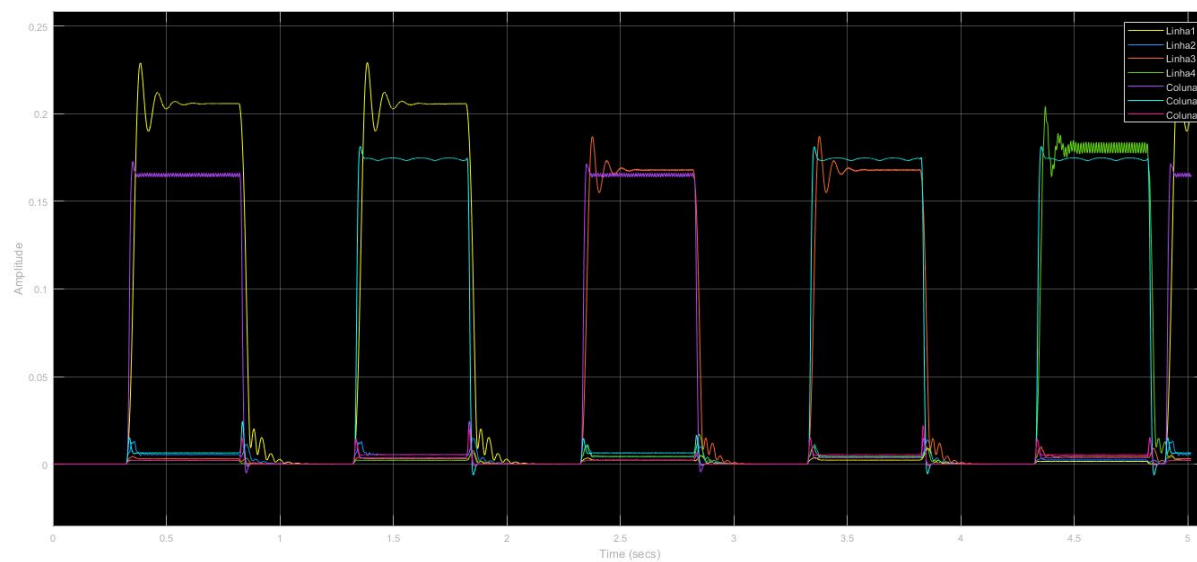


Figura 9 - Soma dos sinais que formam cada tecla.

Na Figura 9, é mostrado a soma dos dois sinais que representam as teclas.

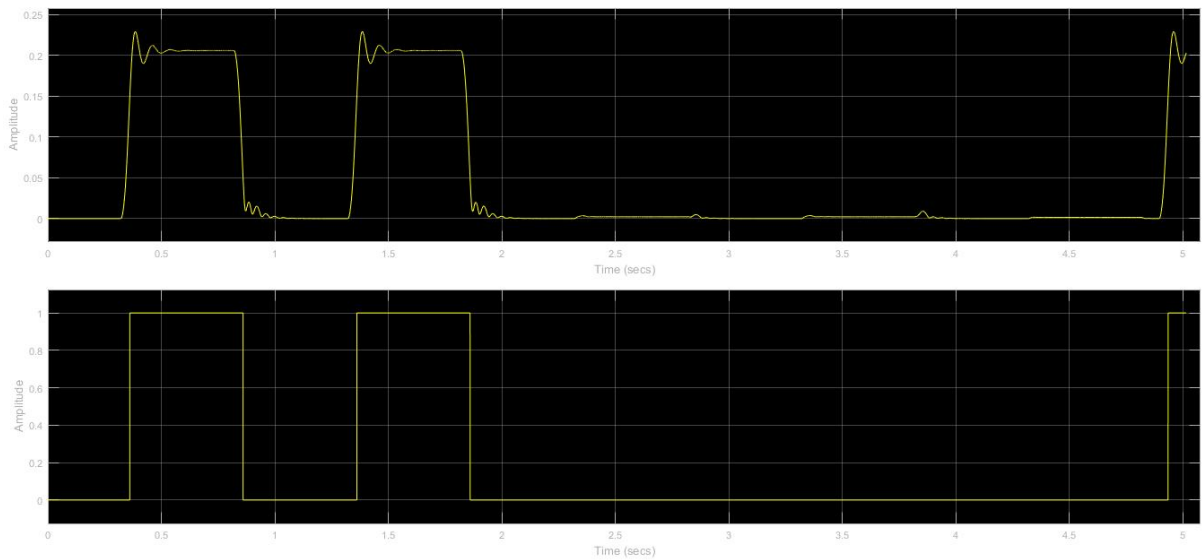


Figura 10 - Circuito antes e depois do relay.

Na Figura 10, é mostrado o sinal antes e depois do *relay*, que define qual é a tensão representada.

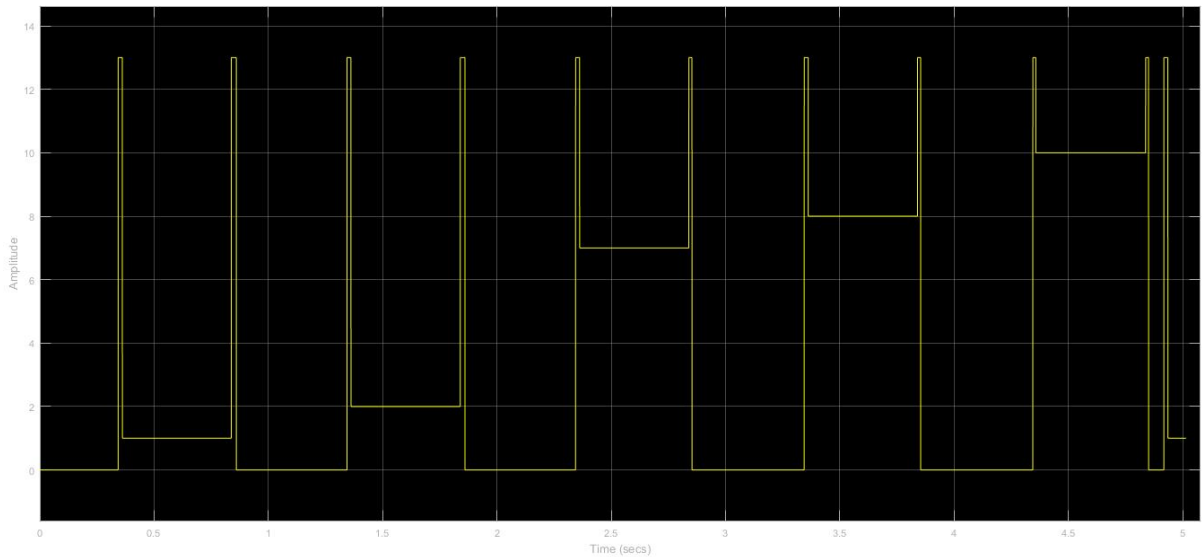


Figura 11 - As teclas 1, 2, 7, 8 e 0.

Finalmente, na Figura 11, são mostradas os sinais referentes às teclas 1, 2, 7, 8 e 0, com a presença de atraso dos filtros em cada um das teclas, sendo que o atraso do filtros *FIR* é constante e atraso dos *IIR* não é constante e, em relação ao *FIR*, é maior.

4 Considerações Finais

Após a implementação do detector DTMF, vários conceitos sobre processamento de sinais digitais foram revisados, além de proporcionar maior conhecimento sobre o software Simulink®. O projeto atendeu todos os requisitos propostos e mostrou a versão digitalizada das teclas pressionadas.