

AE5 - Projeto de um Detector de DTMF

Jessica de Souza, Leticia Aparecida Coelho

Engenharia de Telecomunicações, Instituto Federal de Santa Catarina

jessica.souzajds@gmail.com, leticia.ac23@gmail.com

21 de agosto de 2018

1 Introdução

O objetivo deste relatório é apresentar os resultados obtidos com o projeto de um detector DTMF. A sigla DTMF vem do inglês Dual-Tone Multi-Frequency, que são sinais com dupla frequência utilizados para a transmissão de tons de discagem em telefones. As duplas frequências do sinal DTMF permitem que haja uma combinação de altas e baixas frequências, que são as colunas e linhas mostradas na figura 1, respectivamente.

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

Figura 1 – Tabela DTMF.

Para o projeto do detector DTMF, foram adotadas as seguintes frequências: 770 Hz, 1209 Hz e 1477 Hz, que correspondem na tabela DTMF à linha 2, coluna 1 e coluna 3, respectivamente. Para que haja a identificação de algum dígito, este deve pertencer à linha e coluna especificados pelo projeto. Para isso, iremos utilizar os dígitos 4, 6, 5 e *. De antemão já sabemos que os dígitos 4 e 6 irão ser detectados, pois estão na linha e coluna especificadas para nossa equipe, os dígitos 5 e *, serão detectados apenas uma frequência, o que não será suficiente para detectar qual dígito é, entrando em uma condição de erro. As seções a seguir mostram como foi a realização deste detector.

2 Metodologia

De acordo com a especificação do DTMF ETSI ES 201 235-3 [1], o detector DTMF deve possuir a estrutura conforme mostra a Figura 2.

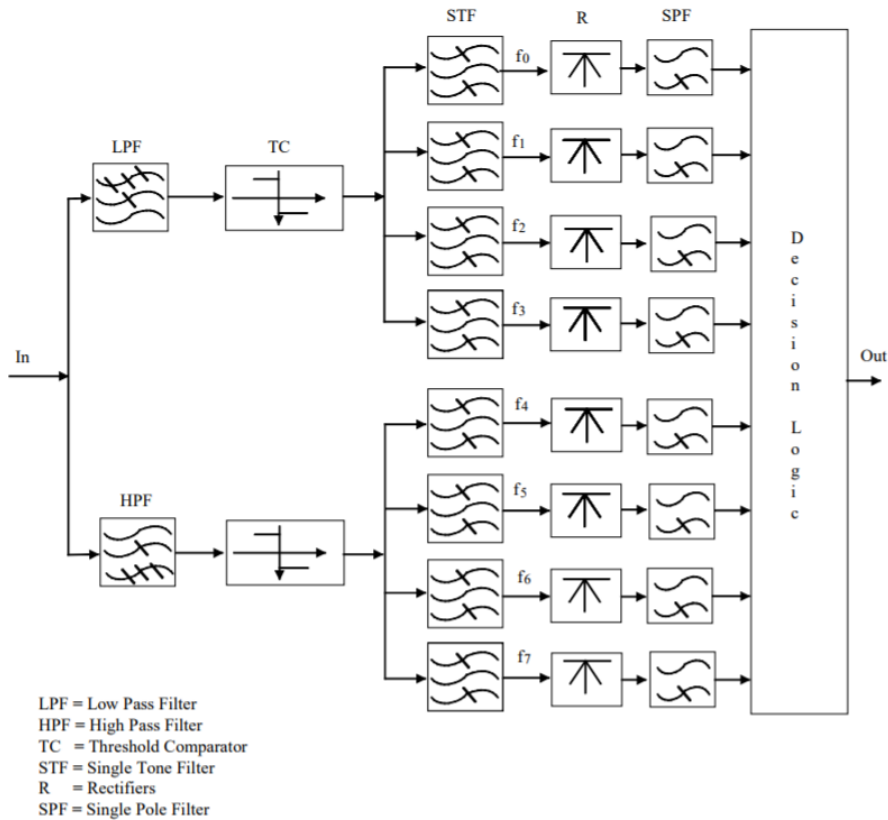


Figura 2 – Detector DTMF. [1]

De acordo com a figura, o detector mostrado nela detecta cada dígito da tabela DTMF, onde são as diversas linhas de f_0 até f_7 . Conforme nossas especificações, iremos ter apenas 3 destas frequências, tendo como resultado um detector mais compacto.

Inicialmente, geramos os sinais DTMF em um formato de áudio utilizando o software Audacity. Para testes iniciais e para o discriminador final, foram utilizados os áudios dos dígitos separados (4, 6, 5 e *), e para a visualização no espectro e no tempo foram utilizados um áudio com os dígitos DTMF "123456789*0" com duração de tom de 65ms e pausa de 65ms para cada tecla. Além disso, foi inserido um bloco AWGN com um ruído branco, de forma a testar posteriormente o comportamento do detector com e sem ruído.

Logo após isso, o restante do projeto foi todo realizado utilizando o software Modelsim, onde iremos implementar o detector semelhante ao da Figura 2.

A primeira parte foi realizar uma filtragem inicial do sinal para a frequência de amostragem do detector, pois os áudios gerados possuíam frequência de amostragem de 80 KHz e a frequência de amostragem do detector é de 4 KHz. Logo após isto, foi realizada a redução de amostras do sinal com taxa de redução = 10, utilizando o bloco downsample do Simulink.

O próximo passo é passar o sinal em um filtro passa baixas e um filtro passa altas, de forma que a próxima etapa, vai realizar a detecção da frequência alta e baixa que compõe o sinal separadamente. O filtro de baixas frequências possui frequência de corte de 950 Hz e o filtro de altas frequências possui frequência de corte em 1200 Hz, ambos são um pouco mais seletivos que a referência citada em [1].

Como o nosso detector é reduzido a apenas uma linha e duas colunas, neste momento é que é inserido o circuito discriminador, que consiste em um filtro passa-faixa, um circuito retificador, um detector de envoltória e um comparador. O filtro IIR passa-faixa é configurado para a frequência específica do dígito: na frequência baixa, iremos detectar a frequência de 770 Hz e na frequência alta iremos detectar em um bloco a frequência de 1209 Hz e em outro 1477 Hz. Os filtros passa-faixa possuem frequências de corte superior e inferior de $\pm 1.5\%$ do valor desejado. Nesta etapa após o filtro passa-faixa é implementado também um circuito retificador, que deixa o sinal apresentar apenas valores positivos por meio da função $\text{abs}()$, e também é realizado um filtro passa-baixa de 10 Hz para poder fazer detecção da envoltória do dígito reconhecido. Para finalizar, inserimos um bloco comparador de amplitude, que irá ajustar a saída do sinal detectado, esta é uma importante fase pois irá nos auxiliar na criação de um algoritmo discriminador com saída binária. A Figura 3 mostra como ficou o circuito em geral.

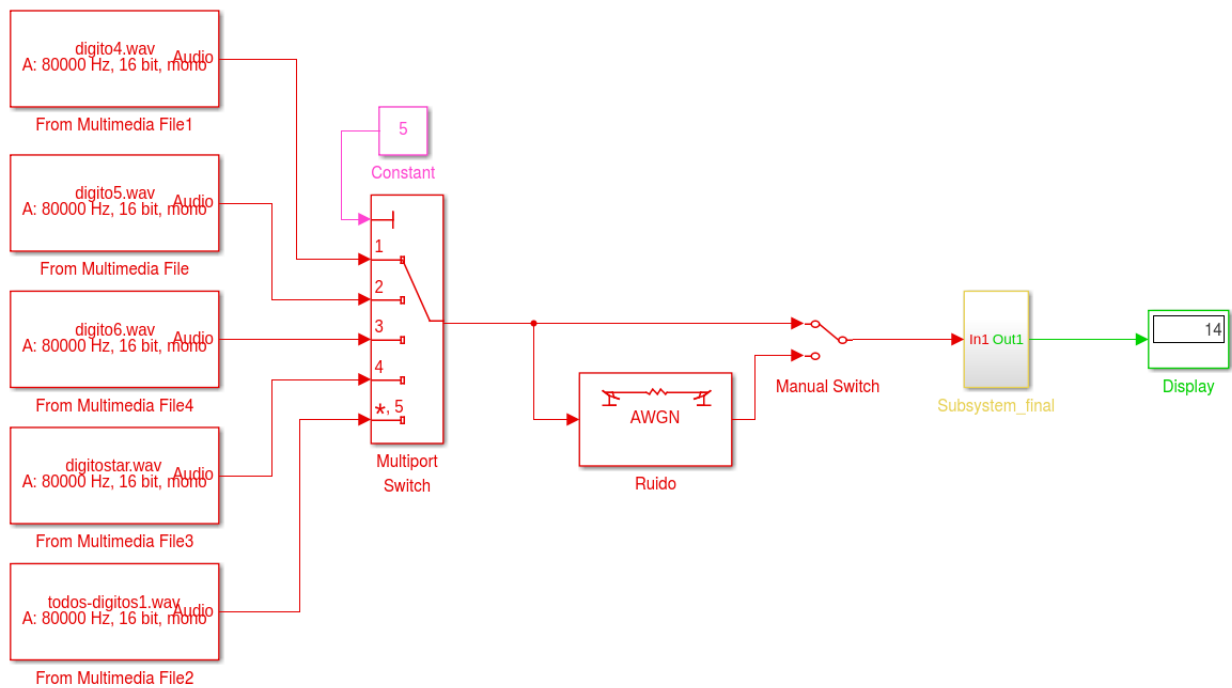


Figura 3 – Projeto no Modelsim.

O algoritmo discriminador, que na Figura 3 está contido dentro do bloco destacado em amarelo, está mostrado na Figura 4 e consiste de um comparador das saídas dos filtros do circuito discriminador. Ele possui uma saída booleana, caso a frequência desejada é detectada ele mostra '1', ou '0' caso contrário. O resultado deste algoritmo é mostrado em um display, que é o final do sistema.

```

1  function y = detector_de_tecla(l2, c1, c3)
2
3  %deteccao dos caracteres 4, 6, 5 e *. Como smente possuímos 1 linha e 2
4  %colunas, nao ha condicoes suficientes para detectar os digitos 5 e *, logo
5  %sera erro
6
7  u = '';
8  if((l2 == 1) && (c1 == 1) && (c3 == 0) )
9      u = 4;
10
11  elseif((l2 == 1) && (c1 == 0) && (c3 == 1))
12      u = 6;
13
14  else
15      u = 14; %14 significa E em ascii conversao decimal, de erro
16  end;
17
18  y = u;

```

Figura 4 – Projeto no Modelsim.

3 Resultados e Discussão

Foram realizados alguns testes com os sinais de entrada, para verificar o funcionamento isolado do detector DTMF. Os resultados mostrados aqui utilizaram o sinal DTMF com todos os dígitos.

3.1 Detecção das teclas

Colocando o sinal com o tom DTMF completo no discriminador da linha 2, ele mostra todos os dígitos esperados, conforme mostra a Figura 5.

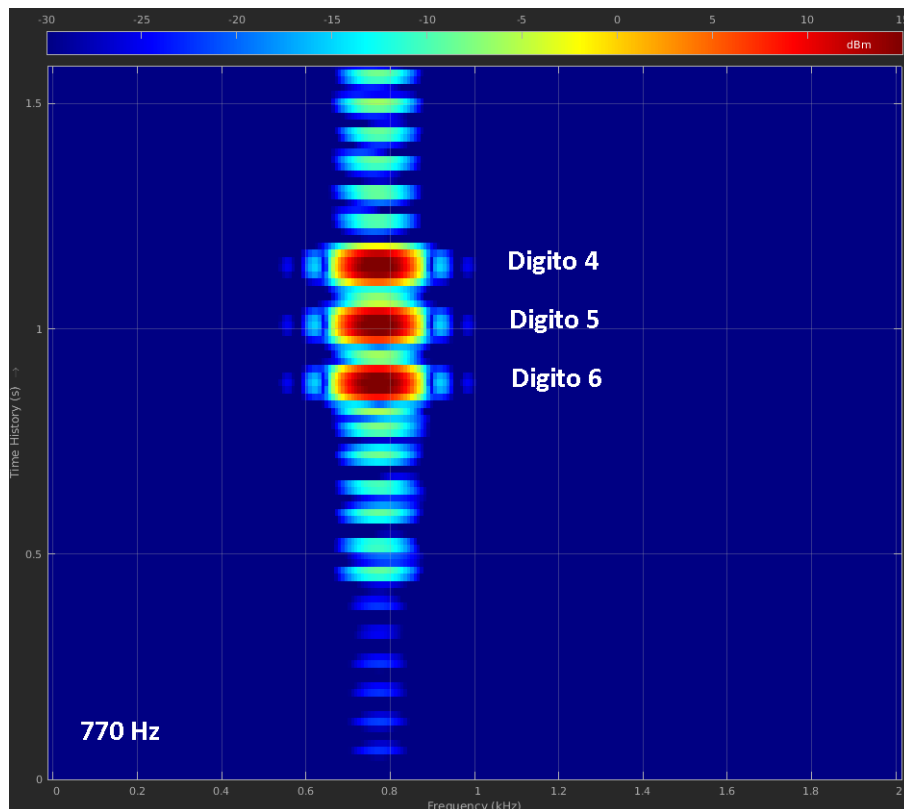


Figura 5 – Espectrograma do discriminador da linha 2.

Podemos observar também no espectrograma, a detecção das frequências das colunas 1 e 3, conforme mostram as Figuras 6 e 7, respectivamente.

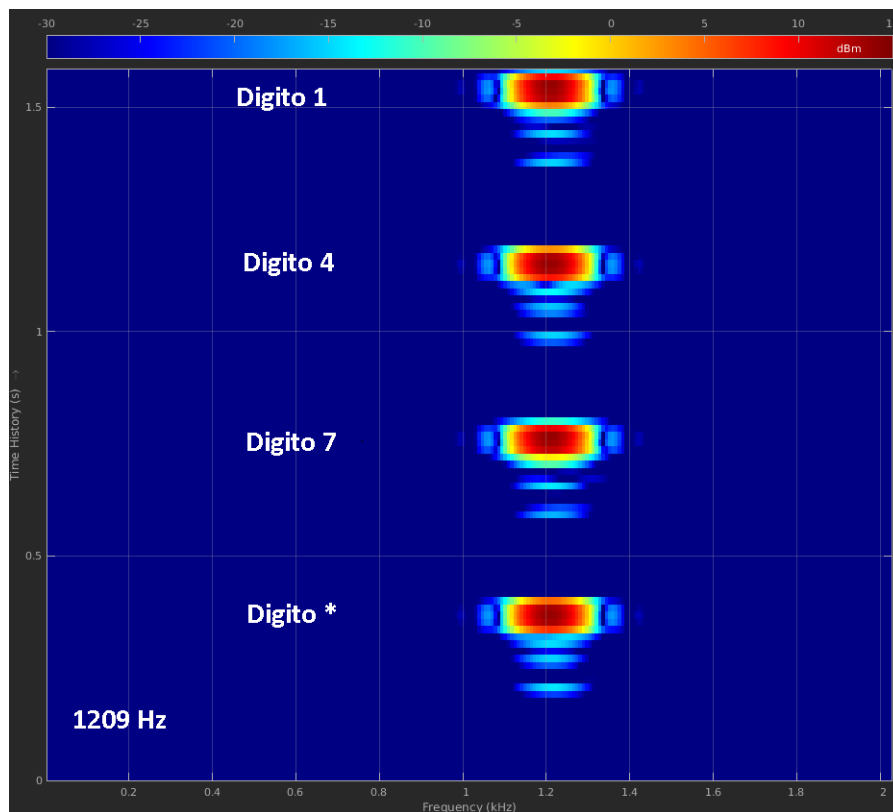


Figura 6 – Espectrograma do discriminador da coluna 1.

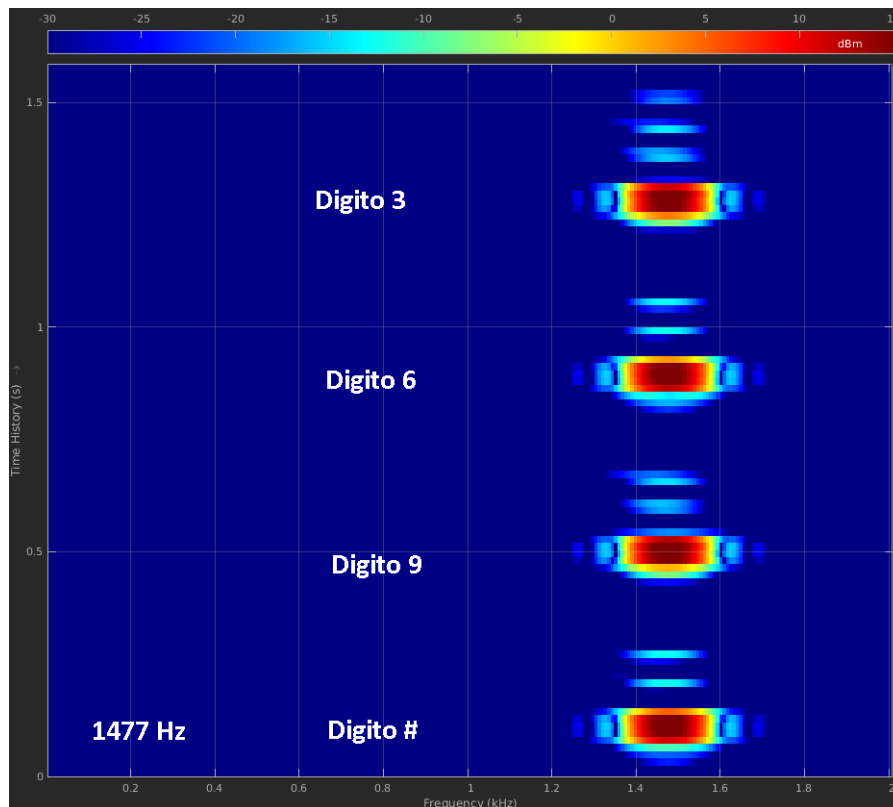


Figura 7 – Espectrograma do discriminador da coluna 3.

3.2 Variação de ruído na entrada

Além do espectrograma, realizamos testes colocando o sinal DTMF com todos os dígitos, somados a um ruído branco, de modo a verificar a capacidade do discriminador de reconhecer as frequências desejadas.

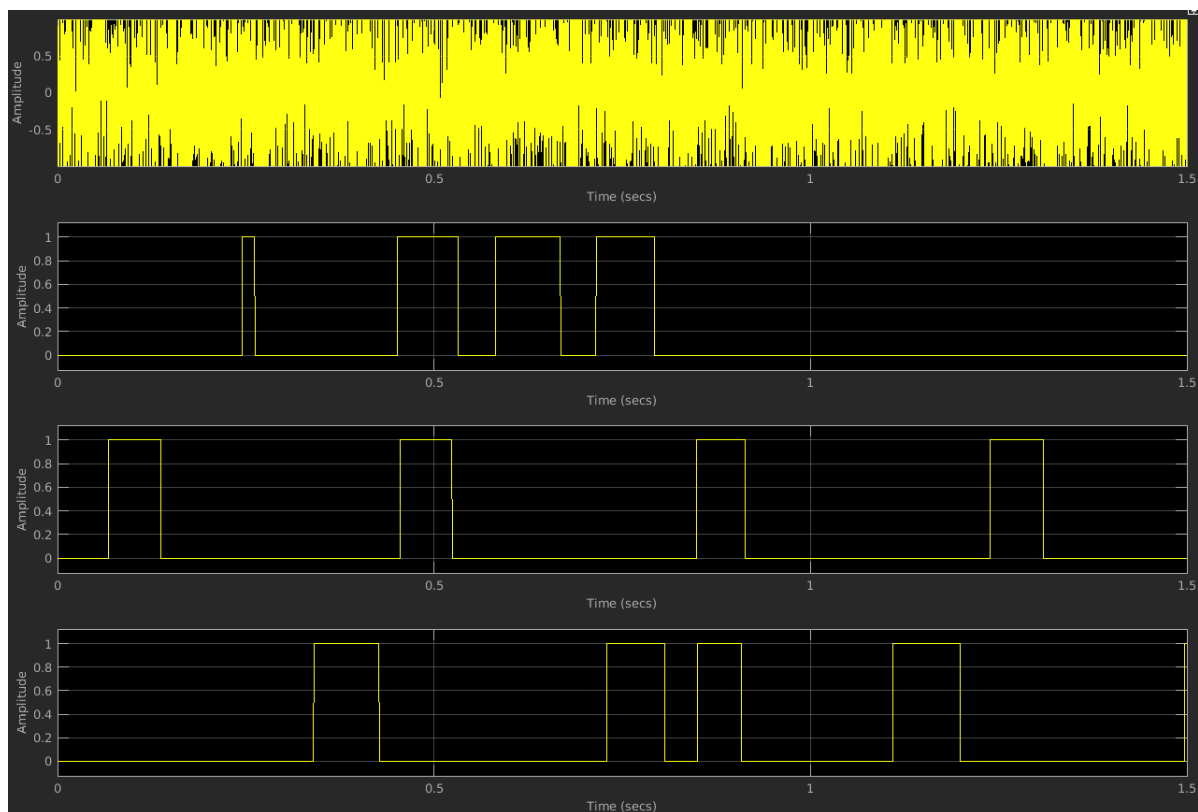


Figura 8 – Time scope com SNR=0.

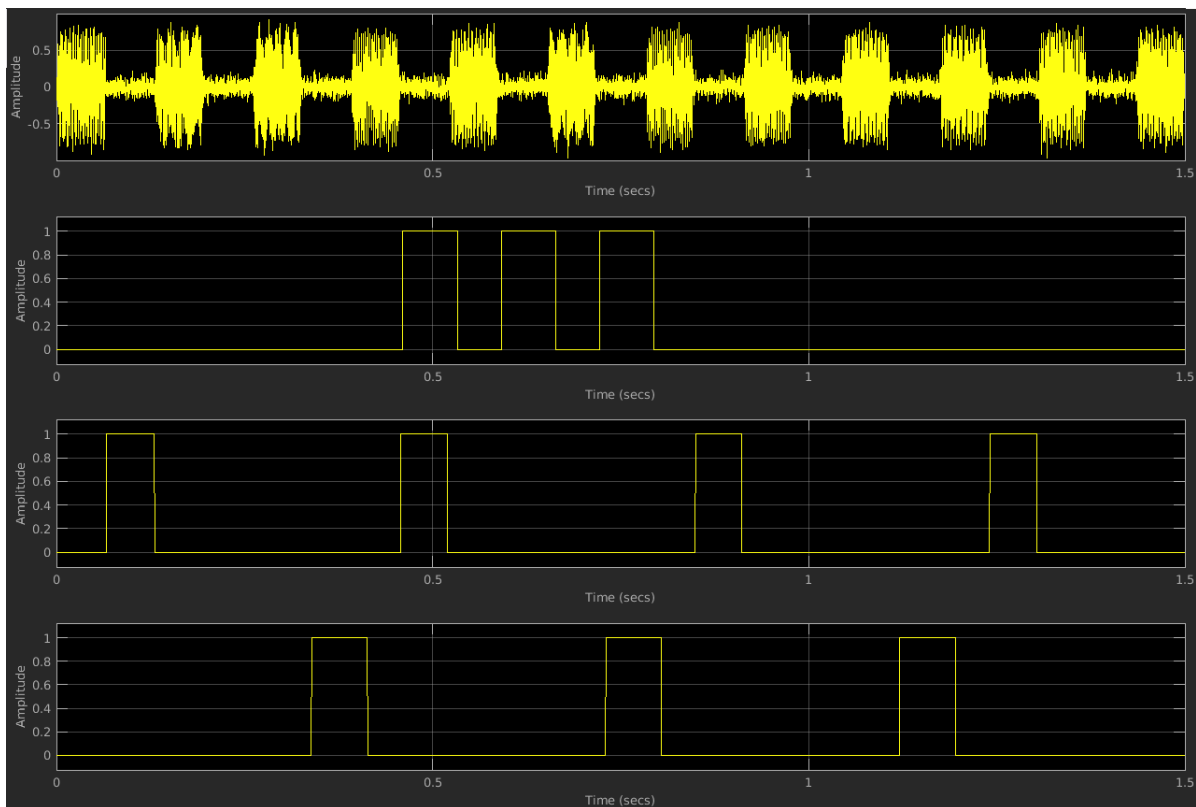


Figura 9 – Time scope com SNR=20.

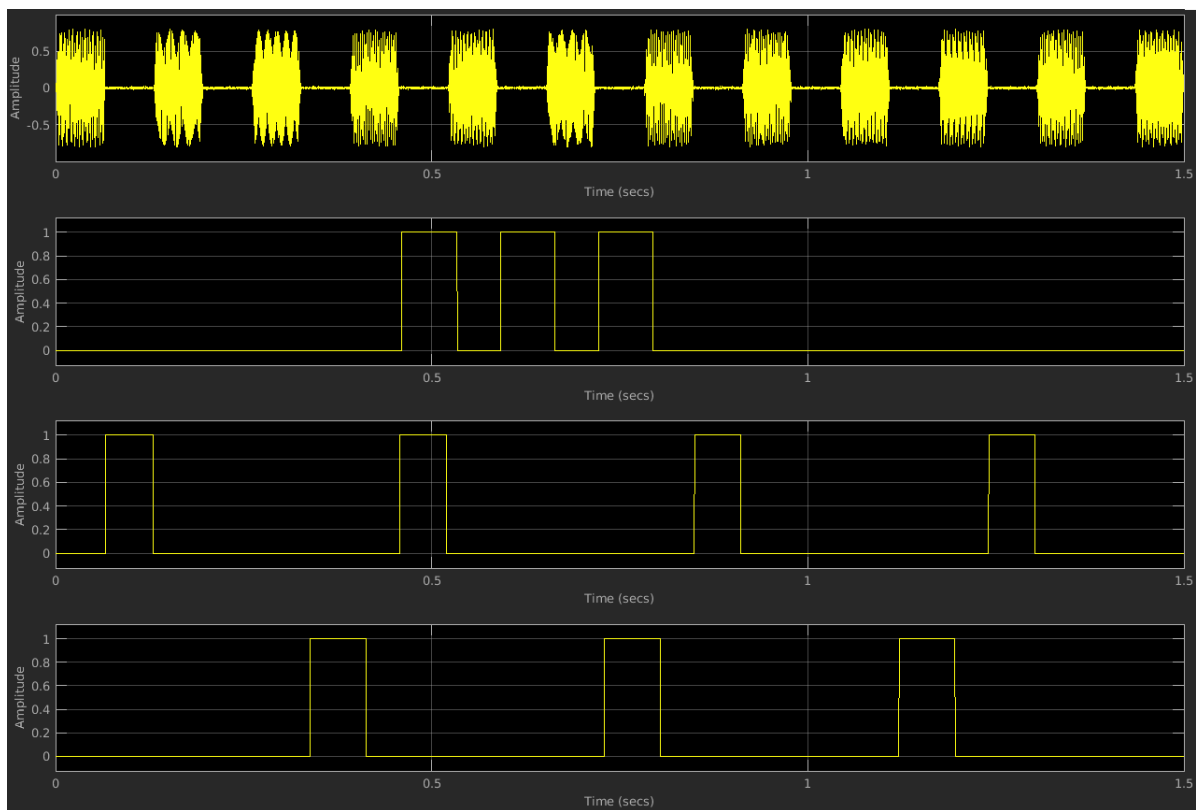


Figura 10 – Time scope com SNR=40.

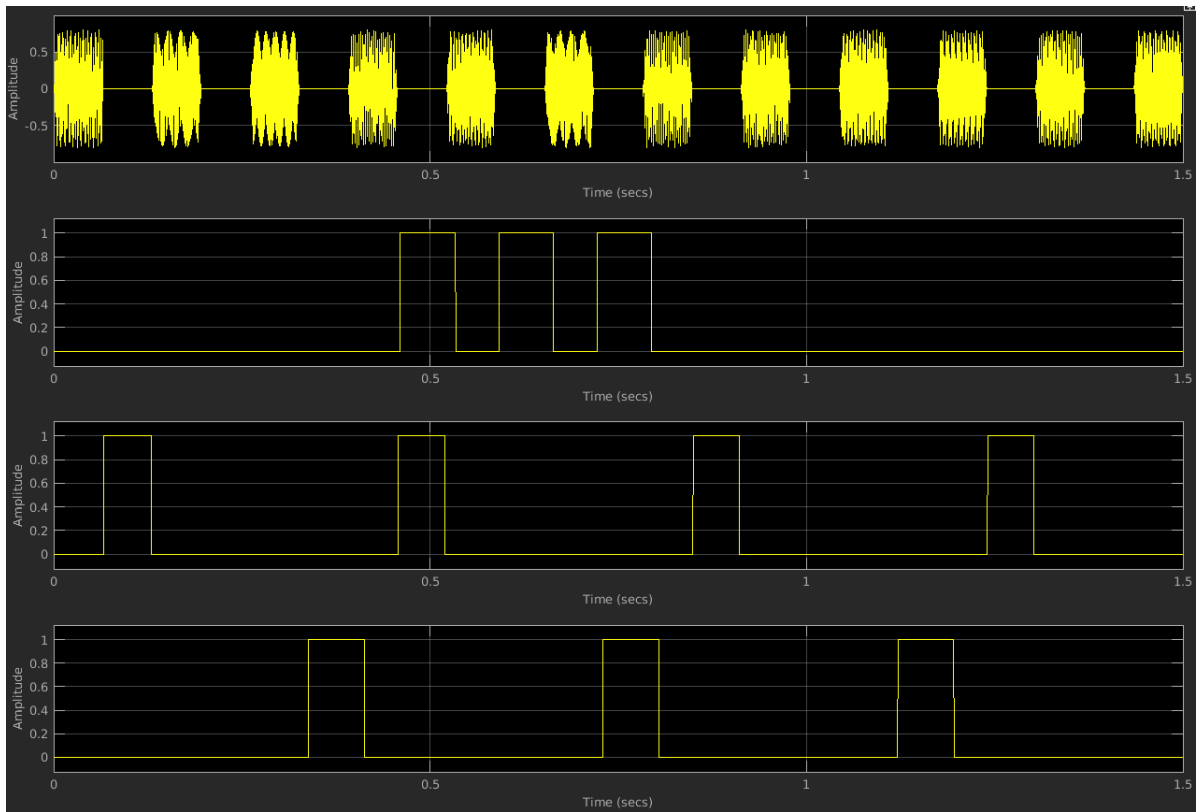


Figura 11 – Time scope com SNR=60.

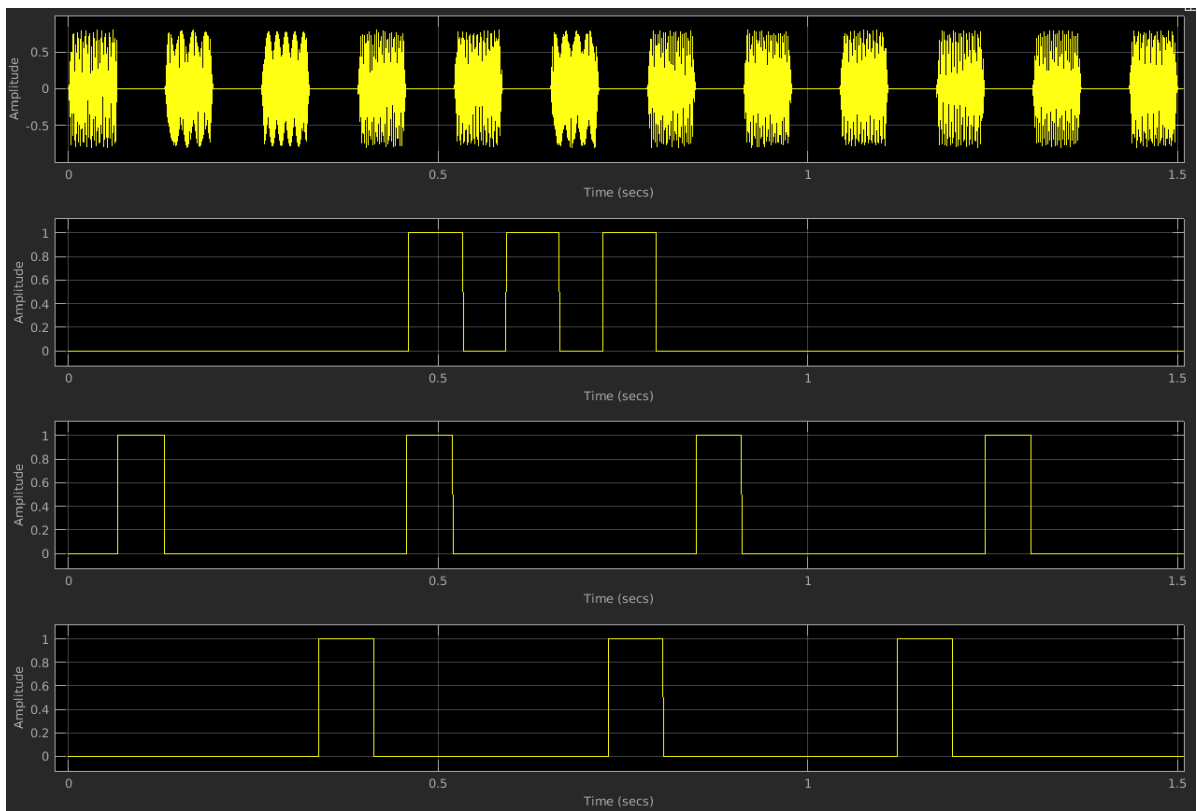


Figura 12 – Time scope com SNR=80.

4 Conclusão

Podemos observar que o projeto obteve êxito ao detectar os dígitos 4 e 6, que eram os dígitos cujas frequências superiores e inferiores estavam no discriminador. Além disso, podemos observar que ao adicionar o ruído branco, quando a SNR é zero, houveram distorções do sinal e até reconhecimento de frequências em locais que não eram esperados. Ao variar a SNR acima de 20, o detector funcionou normalmente.

Além disso, não foi possível realizar a implementação do código VHDL por problemas na geração do código para o Modelsim, sendo esta parte removida como requisito do projeto.

Referências

1. ETSI Standard. Specification of dual tone multi-frequency (dtmf) transmitters and receivers: Receivers for use in terminal equipment for end-to-endsignalling, 2000.