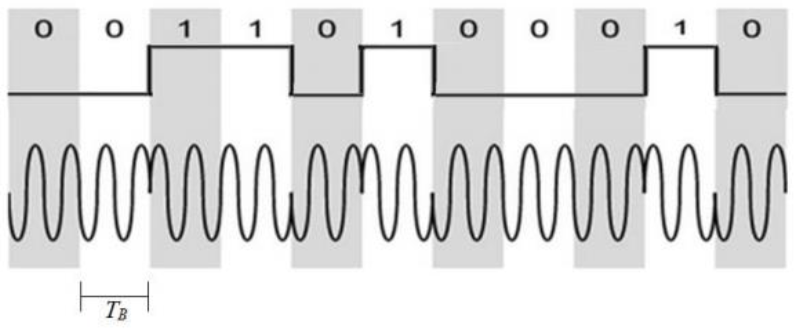
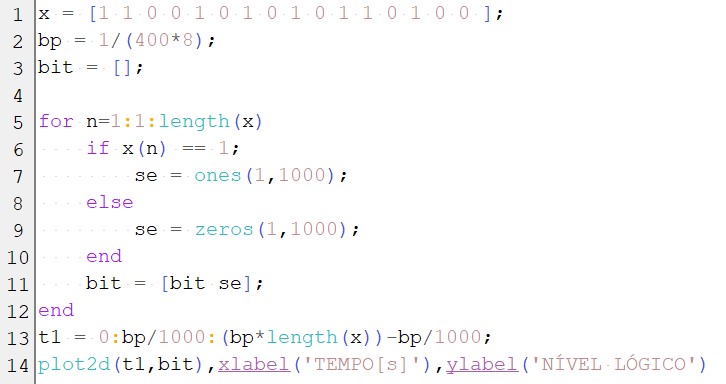
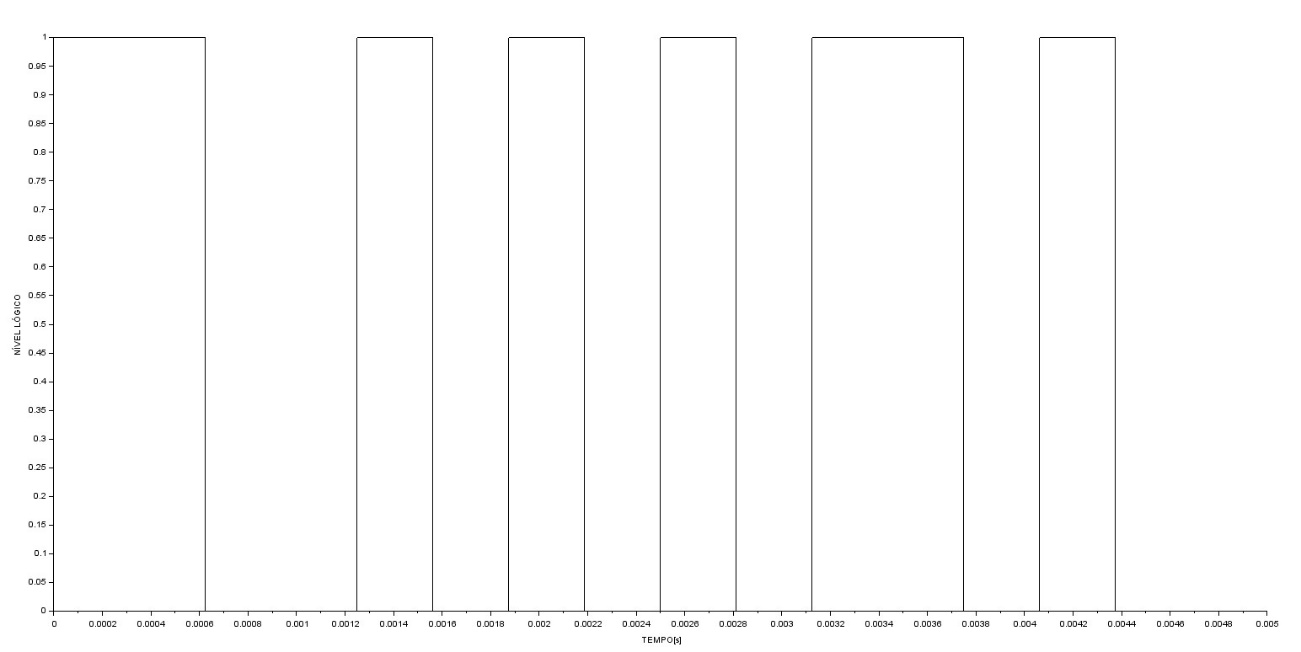
Assim como na etapa 4, para aplicar a modulação digital BPSK escolheu-se uma amostra do sinal PCM criado na etapa 1. A técnica de modulação BPSK terá o seguinte funcionamento:

* Quando o bit for 1 a resposta modulada será:
* E quando o bit for 0 a resposta modulada será:

Para melhorar a visualização do efeito da modulação BPSK, escolheu-se as amostras 84 e 85 do sinal PCM, como mostrado abaixo:  


Construção da sequência no Scilab:





A sequência binária pode ser vista claramente. Cada bit vai possuir um período igual a:

Existem 16 bits na sequência binária, logo o tempo de duração total será de 0,005[s].

Para realizar a obtenção do sinal PWM na modulação BPSK, é necessário primeiramente fazer a amostragem deste sinal modulado. Para descobrir qual é o valor da frequência de amostragem, faz-se necessário saber o valor da banda ocupada pelo sinal modulado com a sequência binária utilizada, com isso será possível encontrar o valor da frequência de amostragem resultante.

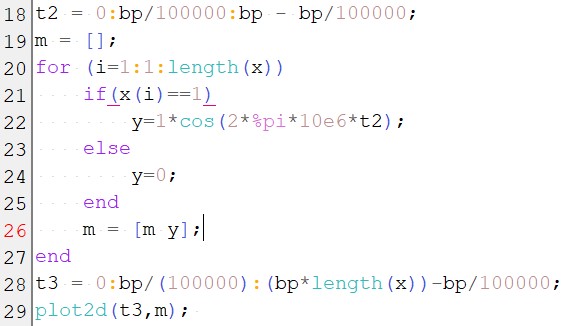
Para determinar a banda ocupada na modulação BPSK, faz-se necessário encontrar o valor do espectro de magnitude do sinal. Para realizar tal ação, é utilizada a lógica da modulação BPSK, feita para cada bit da sequência binária. Sendo assim, o resultado da concatenação da aplicação da modulação para cada bit será o sinal modulado total m.

Foi escolhida uma frequência de amostragem significativamente grande o suficiente para realizar a amostragem de maneira bem sucedida, levando em conta que a frequência da portadora é de 10[MHz]. Tal escolha se faz necessária pois se deseja visualizar o espectro de magnitude da forma mais limpa possível, com isso é evitado perdas de informações acarretadas por uma frequência de amostragem ruim. Aplicando o Teorema de Nyquist, é necessária escolher uma frequência de no mínimo 20[MHz]. Portanto, levando em conta que o período do bit é igual a 0,0003125; foi escolhida uma frequência de amostragem seguindo a seguinte fórmula:

Com isso, pode-se notar claramente que se obteve um valor 32 vezes maior que o da portadora.

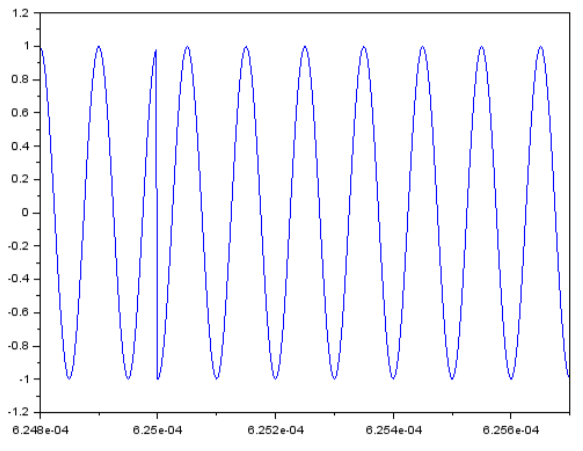
Vale ressaltar que esta não é a frequência de amostragem final, e sim uma frequência provisória para gerar uma boa visualização de como é o sinal modulado e o seu espectro de amplitude. A frequência de amostragem final será determinada aplicando-se o teorema de amostragem passa-faixa.

Agora já se faz possível visualizar o sinal modulado BPSK:



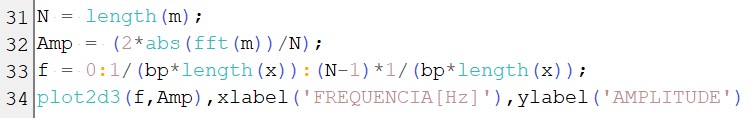
Como período do bit é de:

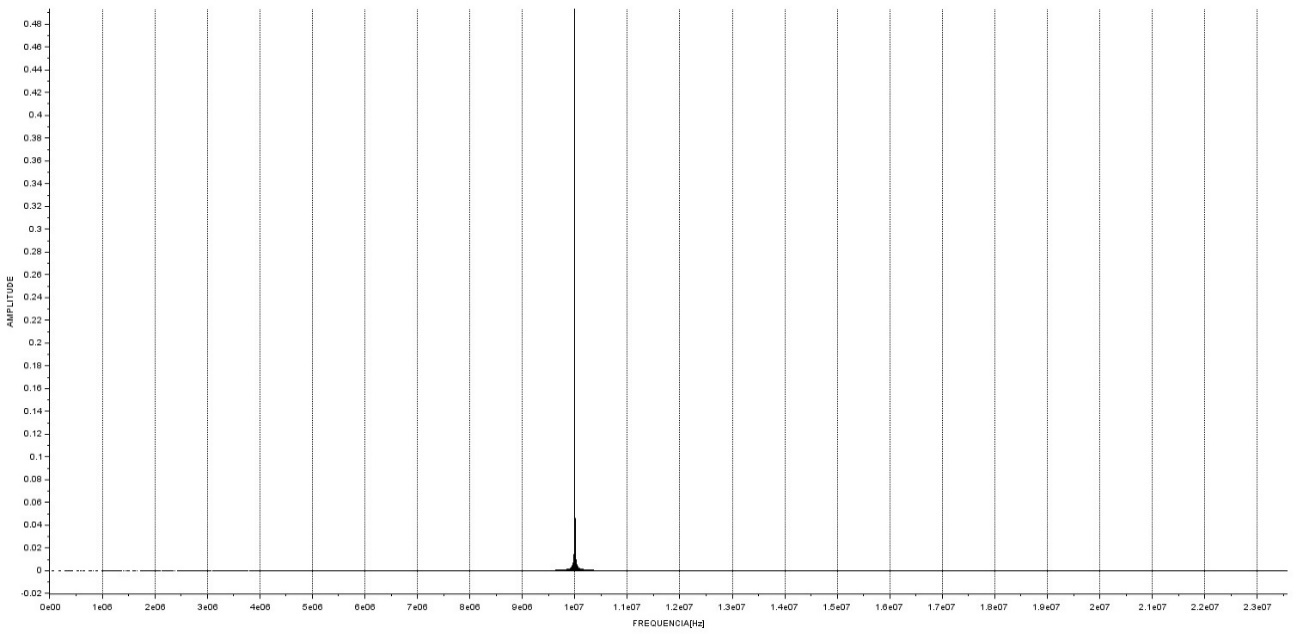
Pode-se observar uma transição clara de 1 para 0 do segundo para o terceiro bit, havendo uma transição nos cossenos no tempo:

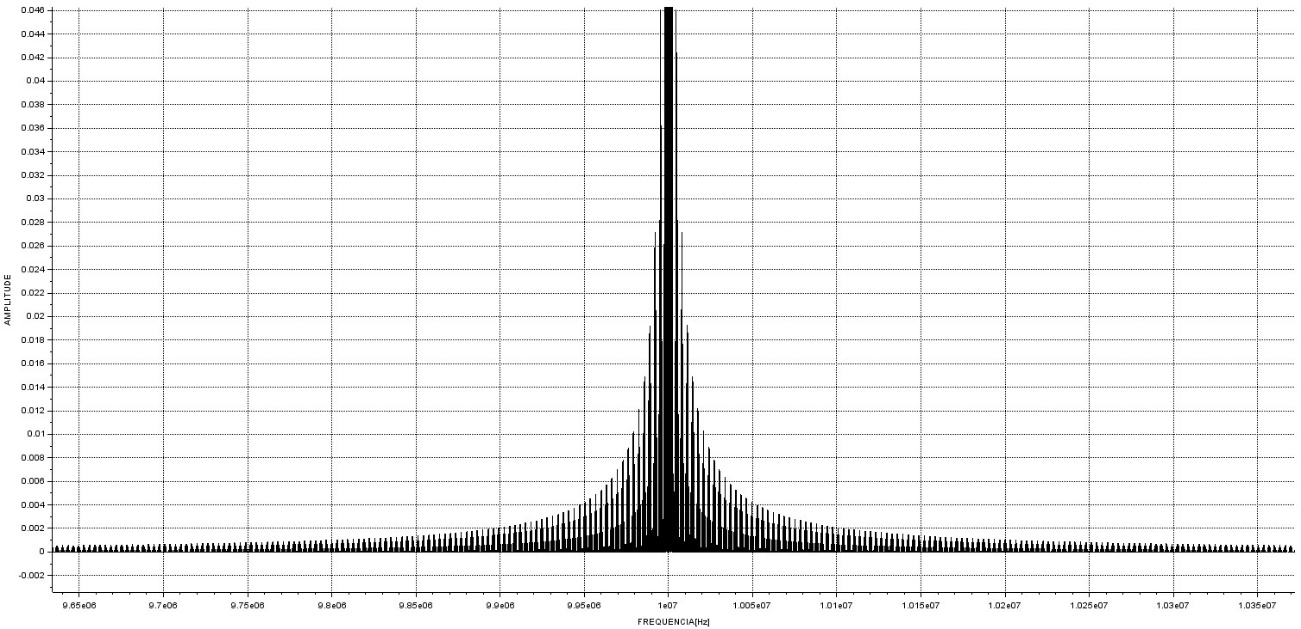


Nota-se claramente os cossenos, bem como sua transição.

O espectro de amplitude é obtido da seguinte forma: tendo as magnitudes do sinal, sendo elas em função das frequências, tem-se que o vetor de amplitude é dado pelo módulo da aplicação da FFT sobre o sinal modulado completo multiplicado por 2 e dividido pelo número de amostras do próprio sinal, no entanto, o vetor de frequência é construído com base nos múltiplos da frequência fundamental, que por sua vez é definida como sendo o inverso do período de observação do sinal, indo até o número de amostras do sinal -1. O resultado obtido fica:





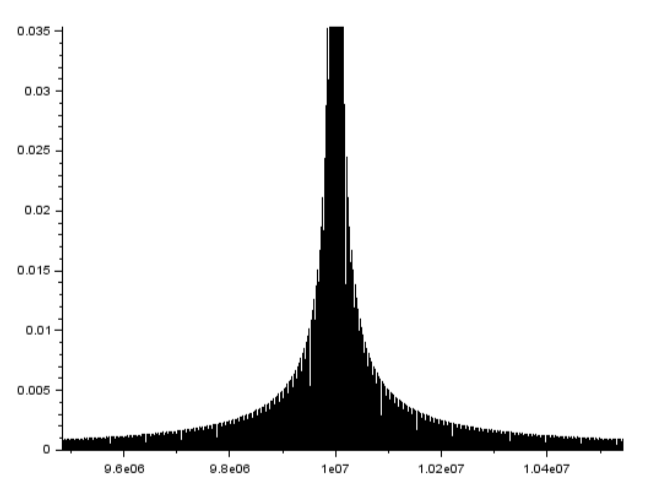


Fazendo a análise do espectro de magnitude, nota-se de forma clara o espectro modulado do sinal, ou seja, a mínima frequência não parte de zero, mas fica centrada a frequência da portadora, que por sua vez, corresponde à 10[MHz], ocupando uma certa banda. Para determinar a frequência de amostragem final do sinal modulado, não é necessário aplicar o teorema de Nyquist, mas sim o teorema de amostragem passa faixa, onde com ele será permitida uma escolha de frequência de amostragem bem menor que a da outra alternativa, além de não gerar sobreposição no espectro e distorção na forma de onda.

Com o teorema da amostragem passa-faixa faz-se possível recuperar o sinal m(t), levando em conta que este foi amostrado com uma frequência dada por:

é dado pelo valor arredondado para baixo, seguindo a divisão abaixo:

Usou-se como referência a amplitude média de 0,1 do diagrama de amplitude para se determinar o valor de como sendo a frequência cuja amplitude é 100 vezes menor do que 0.1, isto é, a frequência máxima onde se tem uma amplitude não desprezível. Esse valor é obtido dando um zoom mais aprofundado no espectro de magnitude:



Portanto tem-se uma amplitude de:

Em aproximadamente 10,5[MHz]/9,5[MHz].

A banda ocupada pelo sinal, contendo as amplitudes não desprezíveis, pode ser definida começando a partir de 9,5[MHz] e terminando em 10,5[MHz].

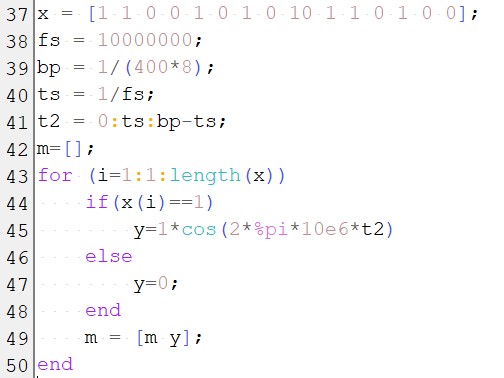
É importante estipular um valor de banda ocupada igual a 20 vezes a banda que o sinal modulado de fato ocupa, com o objetivo de acomodar com sucesso a transição do filtro e também reduzir a distorção gerada no PWM. Tem-se igual a:

A frequência de amostragem final é:

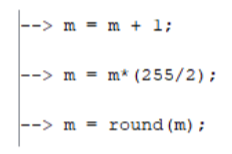
Como esse valor é um múltiplo da frequência da portadora não haverá a necessidade de arredondar o valor calculado. Logo tem-se que:

Percebe-se o quanto foi reduzida a frequência de amostragem necessária para uma recuperação bem sucedida do sinal modulado utilizando o teorema da amostragem passa-faixa, se fosse utilizado o teorema de Nyquist, visando obter o mínimo de perda de informação possível, seria necessária uma frequência em torno de .

Fazendo novamente a construção do sinal modulado BPSK, mas fazendo uso da nova frequência de amostragem. O sinal PWM será obtido com base nesse novo sinal modulado.



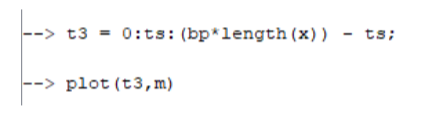
Diferente do sinal BASK da etapa 4, o sinal BPSK não tem valores entra 0 e 1, mas sim entre -1 e 1, logo é necessário dar um Offset no sinal modulado. Além disso, com o novo sinal modulado BPSK já determinado é necessário normalizá-lo, isto é, como foram usados 8 bits de quantização para determinar o vetor de amostras de PCM, o maior valor que uma amostra pode assumir é de 255. Como no sinal modulado BPSK o valor máximo assumido pelo cosseno é de 2, devido ao offset aplicado no sinal, será necessário multiplica-lo por 255/2. Por fim, para não obter valores quebrados será necessário arredondá-los também através do comando round().

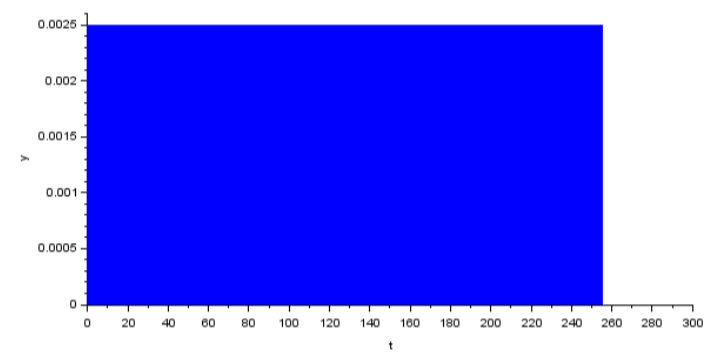


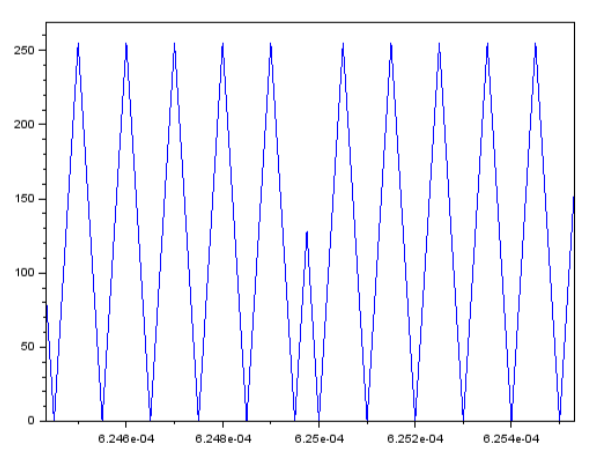
Analisando a construção do sinal modulado BPSK percebe-se que para cada bit, tem-se um número de ciclo de portadora igual a:

Há 8 bits na sequência binária, então o número total de amostras do sinal modulado é igual a:

Pode-se plotar o sinal modulado normalizado resultante, especialmente na região de transição do bit 1 para 0, já identificada anteriormente no momento de 0,000625s, da seguinte forma:







Mesmo que a onda tenha um aspecto triangular ao invés de cossenoidal, pode-se perceber a mesma característica da transição de bits com a modulação BPSK, que não afetará no resultado final do PWM nem na recuperação da onda modulada a partir do sinal PWM, pois o teorema de amostragem passa-faixa foi respeitado. Com a onda modulada BPSK já normalizada e definida, faz-se possível obter o PWM da mesma. Primeiramente, esse processo será realizado na FPGA através do software Quartus, onde os valores das amostras do sinal modulado passados a ele devem estar na forma binária. Essa conversão de decimal para binário é feita utilizando o comando dec2bin ().



O PWM é gerado fazendo a comparação do valor absoluto de cada amostra da onda modulada com uma onda dente de serra. Sendo assim, enquanto a onda dente de serra for menor que o valor da atmosfera da onda modula, a saída terá nível lógico alto, e quando não for menor, a saída terá nível lógico baixo, produzindo assim uma modulação em largura de pulso.

O PWM gerado na FPGA se dará realizando a comparação entre amostra e o valor em contador, que por sua vez, será incrementado por um clock, mas levando em conta que o contador resetará para zero quando ultrapassar o valor máximo, com isso, é visto que se deve atualizar a amostra utilizando a próxima a seguir.

A definição do clock se dará levando em conta o valor da frequência de amostragem, que corresponde à 40[MHz].

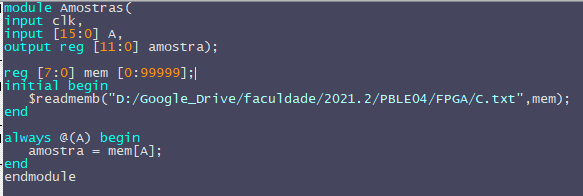
é o máximo valor do contador, cujo valor deve respeitar a seguinte inequação:

Onde N é o número de bits de quantização utilizado. Esta inequação garante que o sinal PWM gerado não possua Duty Cicle superior a 50% da frequência de amostragem. Como foram utilizados 8 bits para quantização, logo:

Para obter um valor de clock redondo, foi escolhido , satisfazendo assim a inequação. Assim, tem-se:

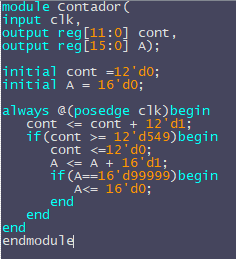
Com todos os parâmetros já definidos, foi feita uma lógica em Verilog para gerar o circuito que criará o sinal PWM. Foi programado um módulo para o contador, amostragem e comparador.

Módulo das amostras:



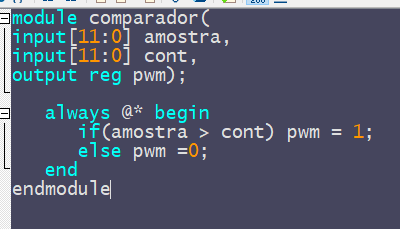
É criado um vetor 100000x8 para então armazenar os valores e preencher cada uma dessas posições com as amostras do sinal modula BPSK normalizado através de um arquivo txt. Vale ressaltar que a saída do módulo via ser controlado pelo valor de entrada A.

O módulo do contador é dado por:



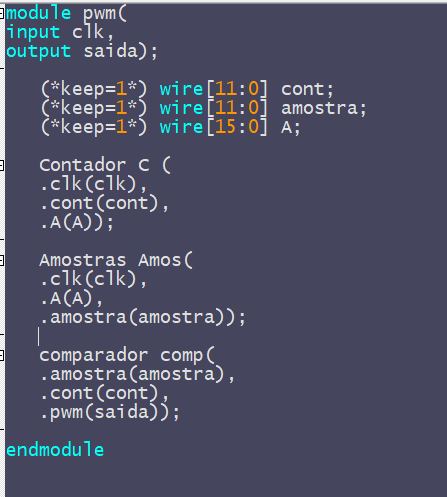
O clock possui valor máximo de 549 e é incrementado de 1 em 1, assim que ultrapassado este valor, ele volta a ser zero, recomeçando todo o processo. O contador é utilizado também para servir de indicação de posição de memória, ou seja, qual amostra deve ser comparada para gerar o PWM. Quando o contador se reinicia, deve-se atualizar a amostra fazendo uso da próxima. Isto é feito realizando a incrementação de 1 na variável de controle da posição da amostra toda vez que o valor do contador é extrapolado. Como existem 100000 amostras, o processo é realizado até chegar a este valor, fazendo com que o valor da posição da amostra volte a 0, reiniciando assim o ciclo.

O módulo de comparação é:



Este módulo apenas realiza a comparação do valor das entradas referentes à amostra atual e do contador, e se o valor absoluto da amostra é maior do que o contador a saída PWM é igual a 1, se não a saída PWM é 0.

Realizando a conexão entre os 3 módulos, tem-se:

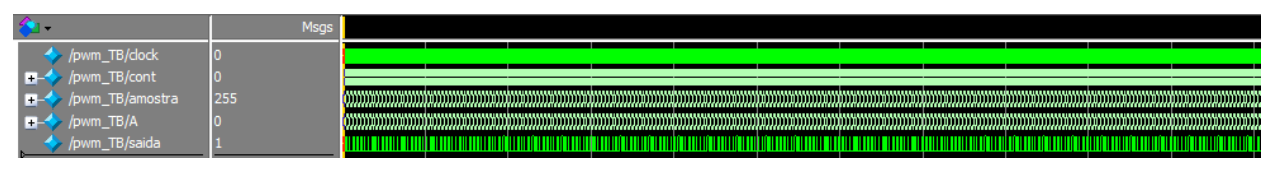


O módulo é conectado da seguinte forma: a saída do módulo contador A vai para a entrada do módulo amostras para indicar qual posição da memória e consequentemente qual amostra deverá ser comparada. Então essa amostra e a saída cont do módulo contador vão para a entrada do módulo comparador onde são comparadas a fim de gerar o sinal PWM.

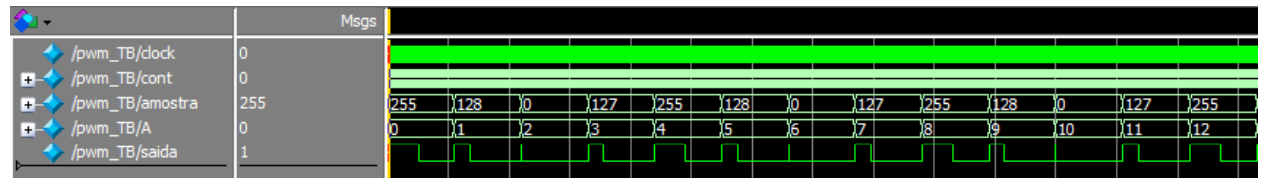
É implementado um clock com período de 0,04544[ns], o que gera uma frequência aproxima de 22000[MHz]:



Tem-se como resultado de simulação:

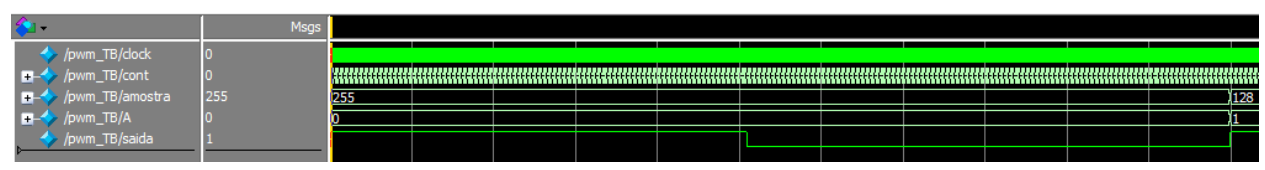


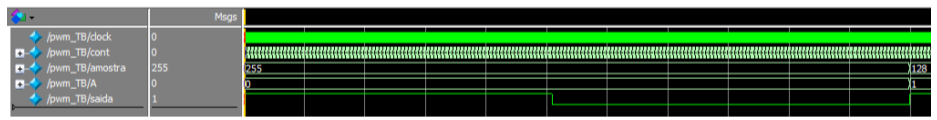
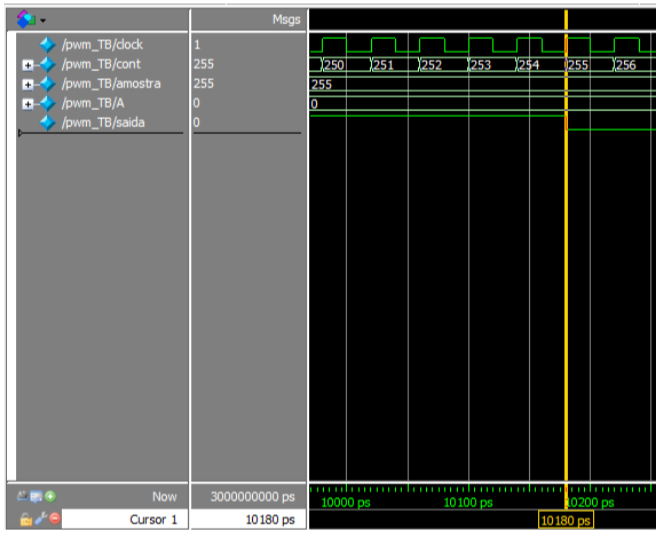
É visualizada as 12 primeiras amostras do sinal modulado BPSK onde era representado o bit 1, e em seguida as primeiras amostras do primeiro bit 0 da amostra PCM escolhida, nota-se que:

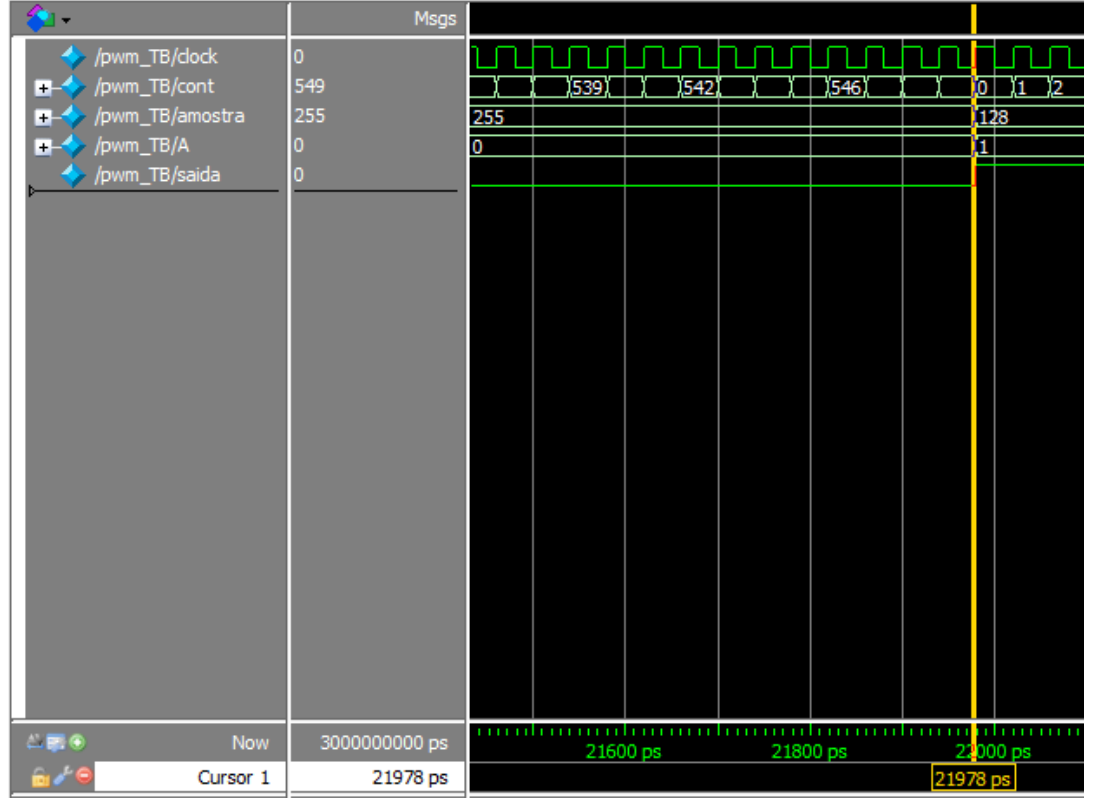


Pode-se ver então que quando a amostra é igual a 0, o sinal PWM resultante também é igual a 0 e quando a amostra é igual a 255, o sinal PWM terá um período em estado alto até o contador passar esse valor e então volta para o estado baixo.

O funcionamento deste módulo se dá analisando o comportamento da onda de PWM em uma única amostra através da comparação entre a proporção do tempo em que a saída fica em 1 e o tempo total do período da onda PWM para essa amostra em específico e a proporção do valor absoluto da amostra com o valor máximo do contador igual a 549. Teoricamente, eles deveriam ser iguais ou muito próximos. Realizando a escolha da primeira amostra para fazer tal teste, é visto:



O sinal PWM muda de nível lógico em 10180[ps]. O contador extrapola no seguinte instante:



Ou seja, em 21978[ps]. Em teoria, o valor ideal seria de:

A simulação foi realizada em Gate Level, sendo assim, os tempos de atrasos inerentes aos componentes foram levados em conta.

As proporções ficam da seguinte forma:

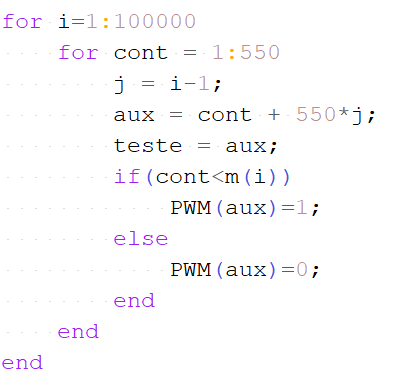
Comprova-se a eficácia do módulo coma base na constatação da mínima diferença entre as proporções.

Após realizada a implementação no Quartus, será executado algo similar no Scilab, empregando-se as mesmas variáveis.

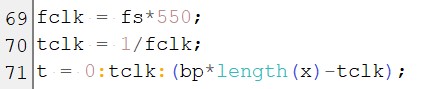
No Scilab, o vetor PWM de saída será:

É feito o algoritmo para realizar a comparação entre o valor de cada uma das amostras da onda modulada BPSK e o contador. É utilizado um ***for*** de 1 a 100000 para varrer cada uma das amostras e outro ***for*** de 1 a 550 para varrer todos os valores possíveis do contador.

Foi feita a comparação entre o valor do contador e o valor da amostra no índice indicado pelo ***for*** através do bloco ***if***. Restando apenas concatenar cada operação de comparação com os valores abstraídos do PWM, afinal, quando extrapolado o valor do contador passa-se para a seguinte posição do vetor de amostras e assim começa um novo ciclo de obtenção dos valores do PWM. Para realizar a solução de tal problema, utiliza-se de uma variável auxiliar a fim de corrigir a posição do PWM, ou seja, conforme o índice da posição do vetor de amostras aumenta em 1, o índice de posição do vetor de PWM deverá ser incrementado em 550, que corresponde justamente ao valor de extrapolação do contador. Assim, tem-se:



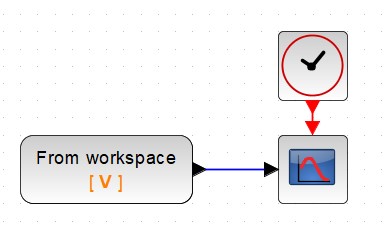
Para averiguar o código, plota-se o gráfico de **t** em função do valor do vetor PWM, onde **t** é o vetor temporal que vai de 0 a 0,0025s (tempo total da sequência binária) incrementado pelo período do clock igual ao inverso de 22000 MHz. Desta forma, tem-se:



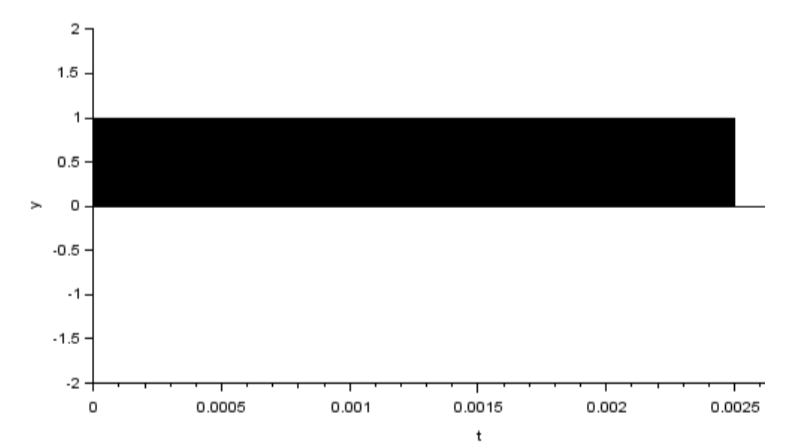
O gráfico do PWM em função do tempo foi feita pelo Xcos, onde os valores do console foram passados a ele através do comando struct().

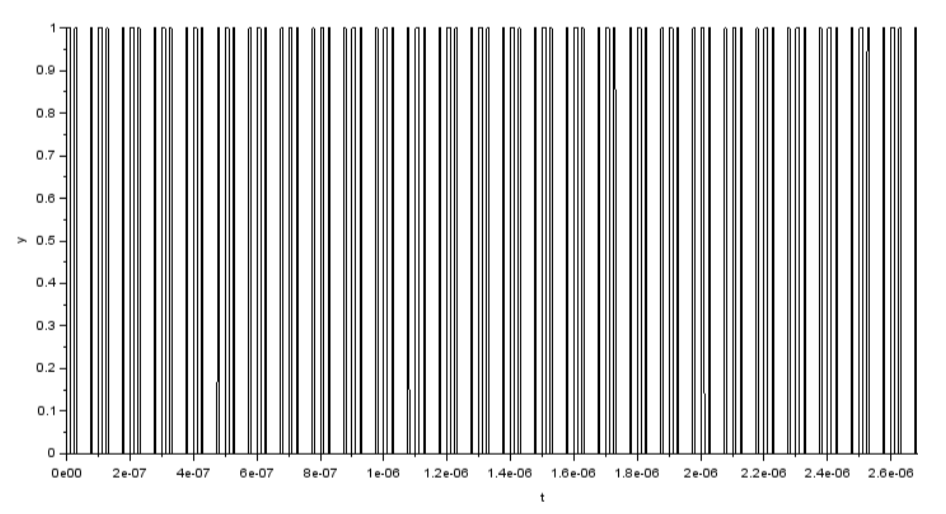


Obteve-se o seguinte diagrama de blocos:

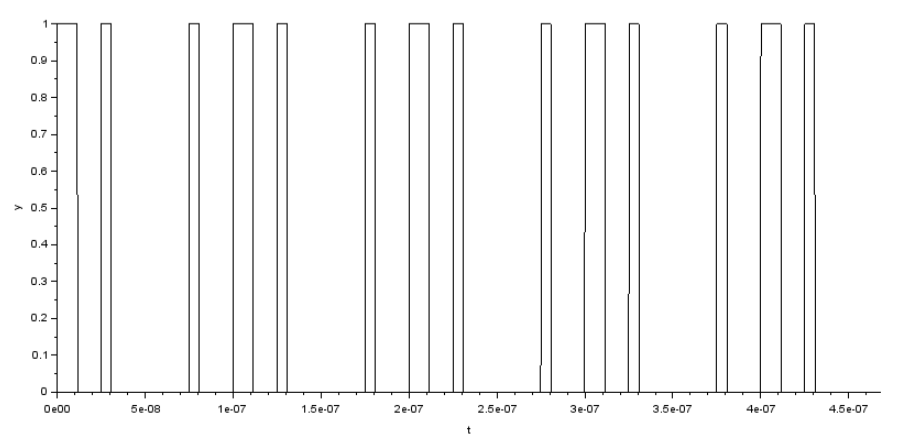


O período de clock foi determinado sendo igual a , valor este inferior ao período do PWM, sendo igual a:

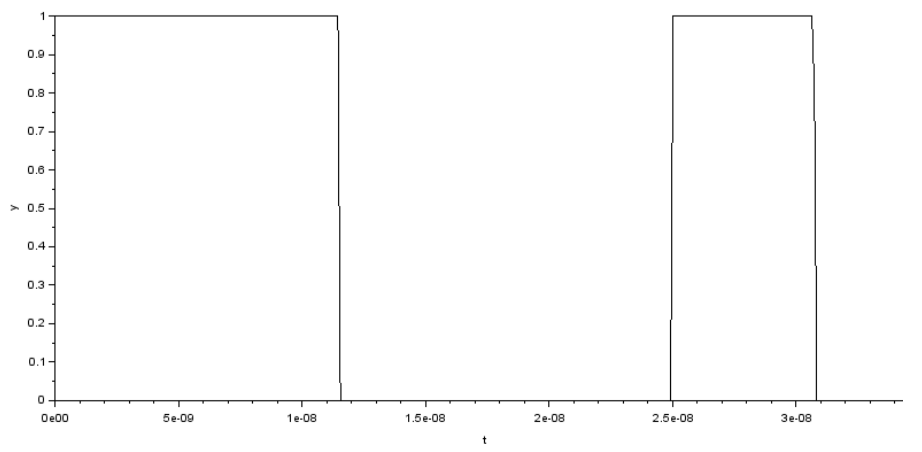
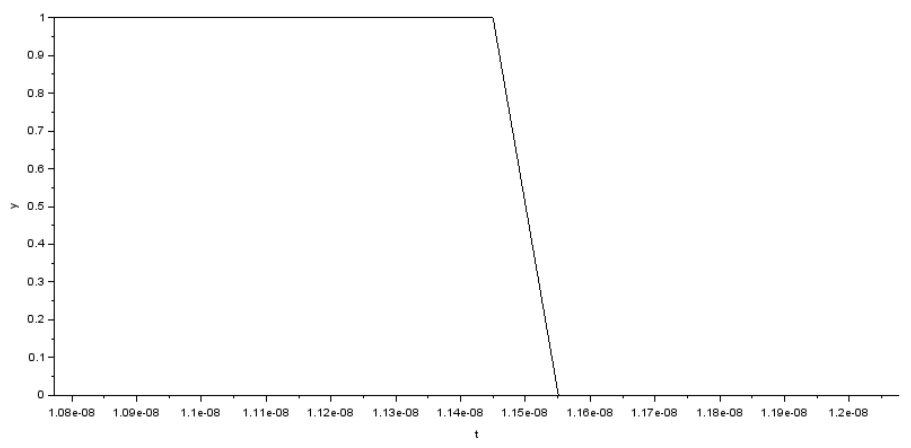
Assim, é obtida uma representação fiel da onda resultante, cujo resultado da simulação no osciloscópio, com tempo total de 0,003[s], é mostrado logo abaixo:



Onde pode-se ver o comportamento geral da onda PWM, em que o comprimento do estado lógico alto varia entre , e 0, isto é, ele será maior de acordo com o tamanho da amostra. Pode-se dar um zoom para ter uma verificação mais clara deste comportamento.

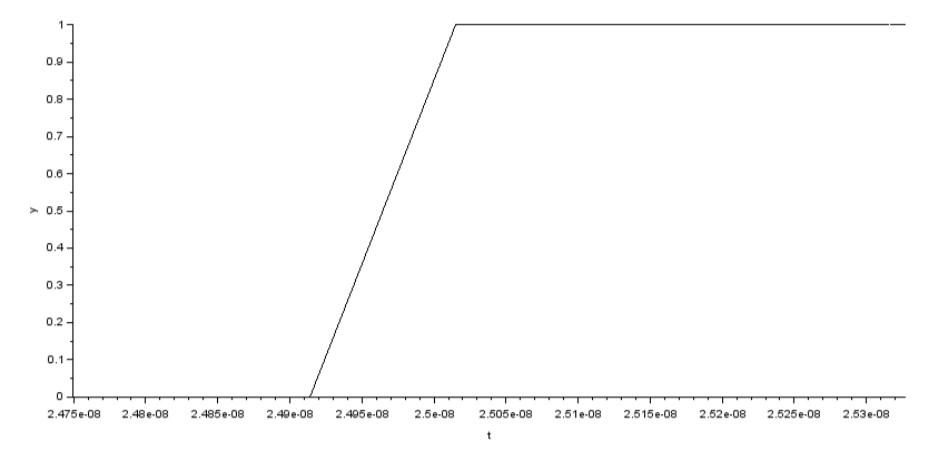
Pode-se observar o comportamento da onda PWM, na qual o comprimento do estado lógico alto varia entre , e 0, ou seja, será proporcional ao tamanho da amostra. Analisando-se com um zoom temos:

Pode-se aplicar novamente a comparação entre as proporções de tempo e valores já explicadas anteriormente. Tomando-se mais uma vez como exemplo a primeira amostra, tem-se que o momento em que é mudado seu nível lógico alto para baixo no pwm é:



O instante corresponde à .

O instante de extrapolação do contador é:



O instante é de aproximadamente .

Assim, fazendo os cálculos das proporções, tem-se:

Com base na análise entre os resultados obtidos no Quartus e Scilab pode-se constatar que houve uma pequena diferença entre as proporções de 0,00003, comprovando-se a eficácia do algoritmo implementado. Além disso, pode-se confirmar a correspondência entre o PWM implementado no Quartus com o implementado no Scilab.