



ESCOLA DE ENGENHARIA DE SÃO CARLOS

Departamento de Engenharia Elétrica e de
Computação

LABORATÓRIO DE SISTEMAS DIGITAIS

Profa.. LUIZA MARIA ROMEIRO CODÁ



LABORATÓRIO DE SISTEMAS DIGITAIS

Utilização de Dispositivos Lógicos Programáveis(FPGA)
circuito de controle de um servomotor

Profa. Luiza Maria Romeiro Codá

OBJETIVO:

- Utilização de Dispositivo Lógico Programável (FPGA) na síntese de Circuito Digital;
- Geração de sinal modulado por de largura de pulso (**PWM** -*Pulse-Width Modulation*) para controlar a posição de um servomotor

Atenção: Ler os seguintes arquivos no link da disciplina no Stoa Moodle:

- ✓ Dispositivos Lógicos Programáveis
- ✓ Guia esquemático do quartus II Altera
- ✓ Roteiro Prática nº2-matriz de LEDs UD

TRABALHO:

Utilizando o software QUARTUSII v.12.OSP2, escolha o dispositivo HCPLD Cyclone IV-E EP4CE30F23C7 e faça um projeto para controle de um servomotor, o qual funciona com modulação por largura de pulso.

SERVOMOTOR

- O servomotor é uma atuador, ou seja, é um motor de posição controlada;
- não tem rotação contínua, trabalha em faixas entre limites de ângulos específicos: Ex de 0° a 180° ;
- Funciona com sinal modulado em largura de pulso(PWM);
- De acordo com a largura do pulso o servomotor vai girar em um certo ângulo específico e se manter na posição até que outro pulso com largura diferente seja aplicado;



SERVOMOTOR

O micro servomotor apresenta 3 fios :

vermelho: Alimentação (5V);

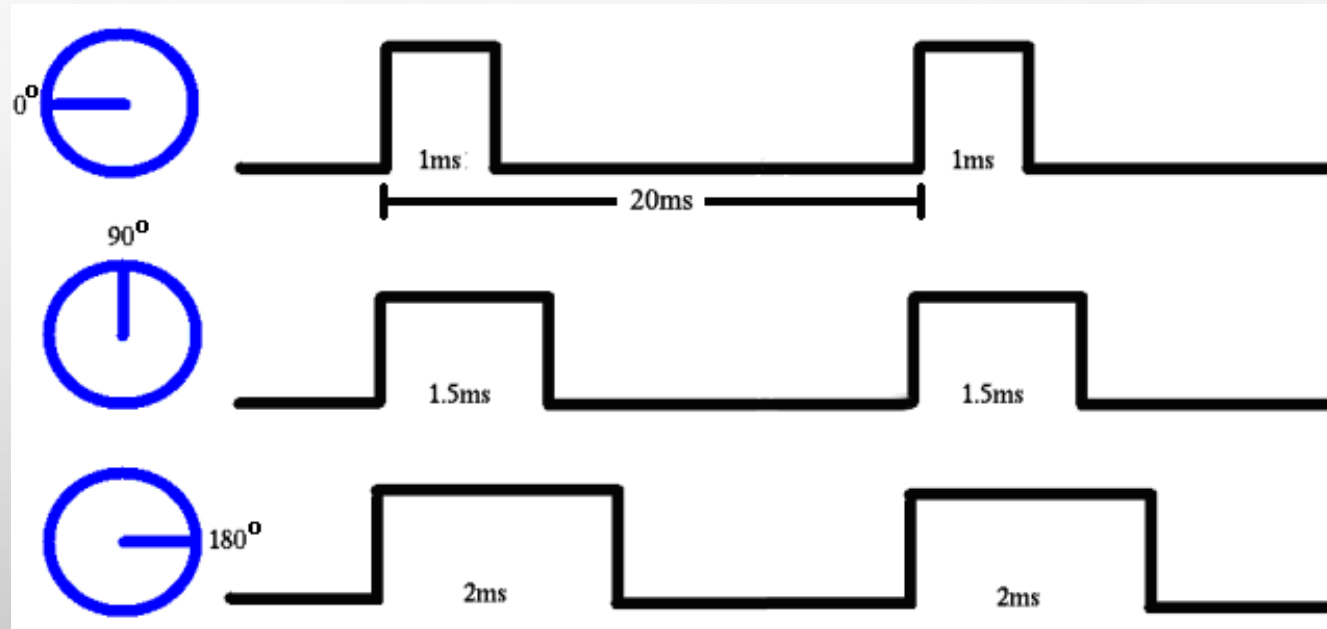
preto (ou marrom): GND ou Terra

Amarelo (ou branco): sinal de controle (PWM)



Funcionamento do Servomotor

Aplicando um pulso no pino do fio amarelo (ou branco):

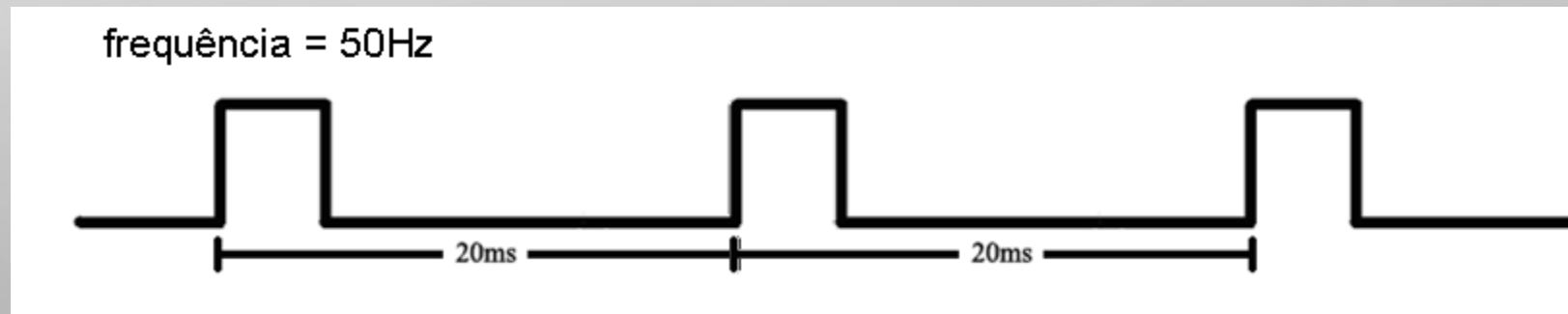
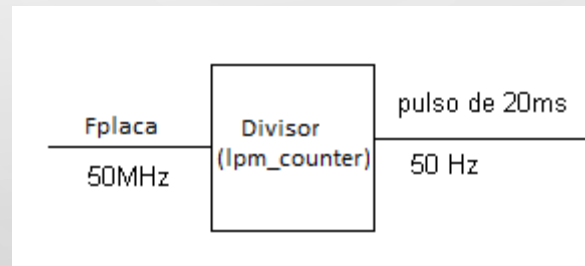


- Dependendo do modelo do servomotor essa largura do pulso pode ser diferente assim como os limites de curso do giro do servomotor;
- O motor tem um potenciômetro ligado ao eixo e utiliza a posição do potenciômetro par criar um controle de malha fechada e ajustar a posição.

Implementação do Circuito para Geração do PWM:

- Gerar o pulso de 20 ms (frequência de 50 Hz)

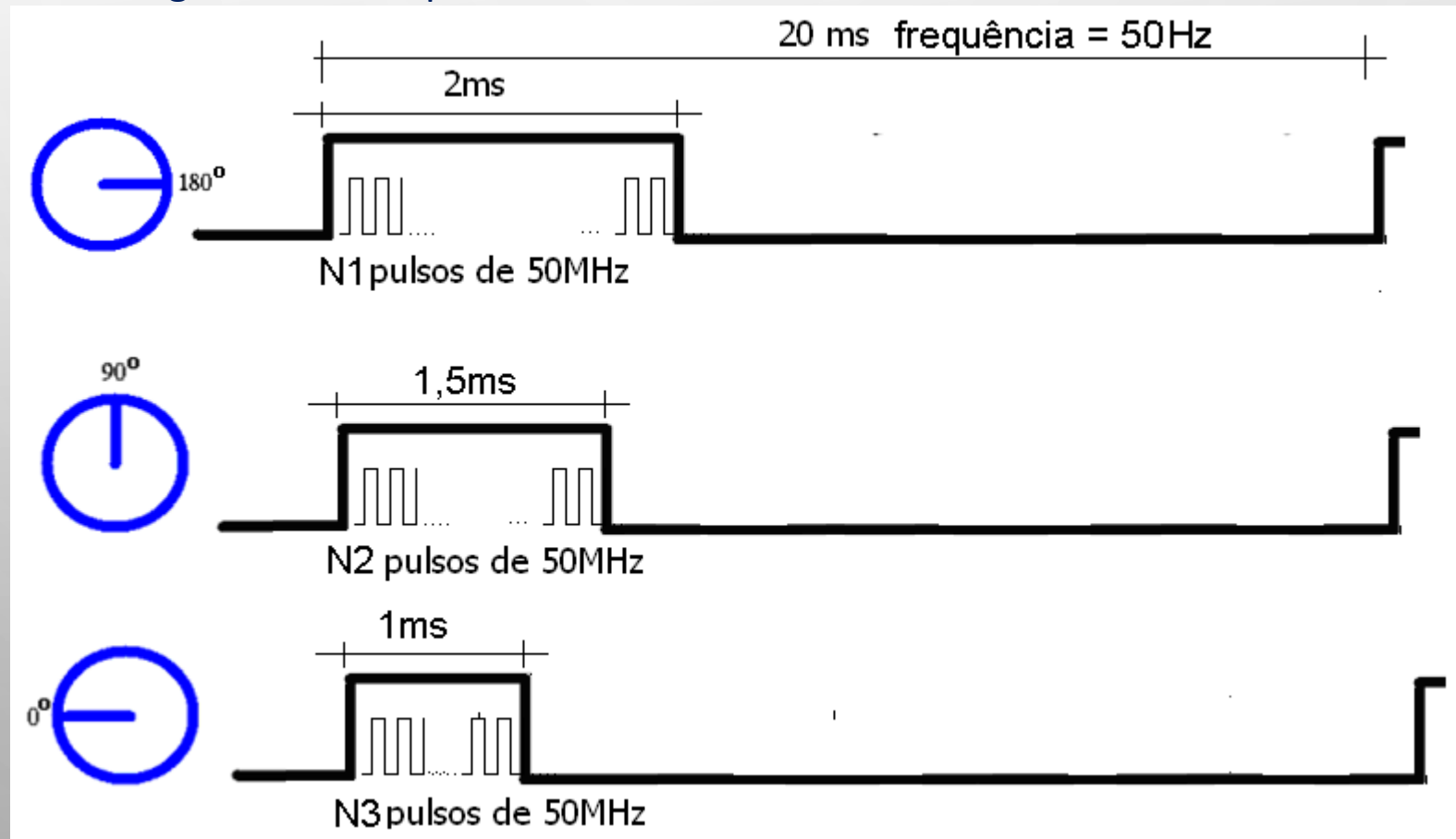
Utilizando a frequência de 50MHz (placa Mercúrio IV) fazer um divisor de frequência e obter 50Hz:



Implementação do Circuito :

Utilizando a frequência de 50MHz como base de tempo, implementar um contador que conte quantos pulsos de 50 MHz cabe em 2ms, 1,5 ms e 1 ms.

Esse contador deve ser capaz de pelo menos o número de pulsos de 50MHz que cabem em 2 ms (N1 pulsos) que é a largura máxima que deve contar.



Implementação do Circuito :

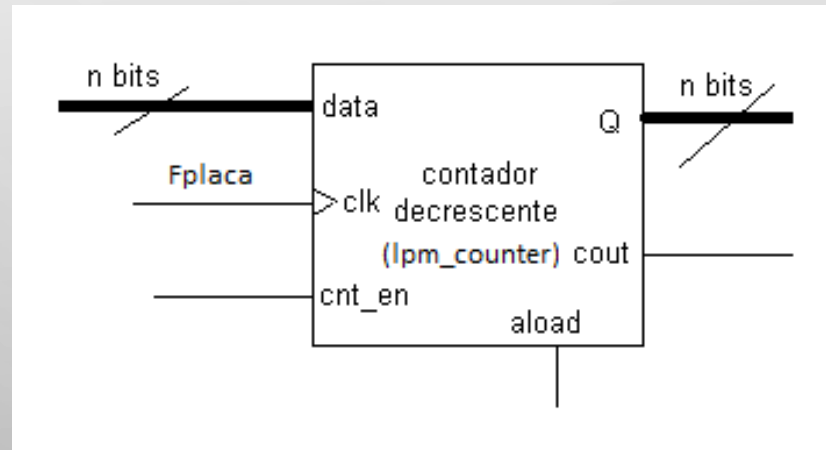
Criar um contador decrescente que conte N1 pulsos,
Transformar o valor N1 em binário e criar um contador com o número de Bits que representa N1 em binário.

Supondo que N1 tenha N bits:

Utilizando o projeto lpm_counter criar um contador decrescente (contagem down) com:

entradas: cnt_en , aload e DATA ;

Saídas: Cout e Q



Implementação do Circuito :

Funcionamento do contador para geração de pulso de 2 ms:

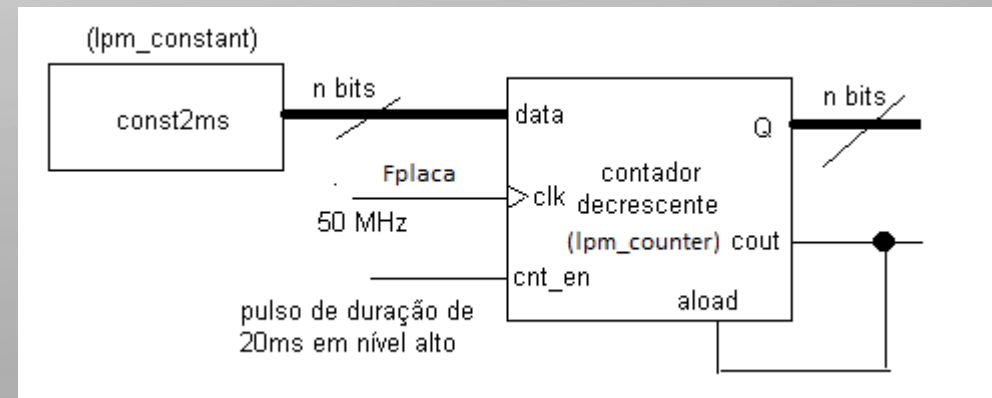
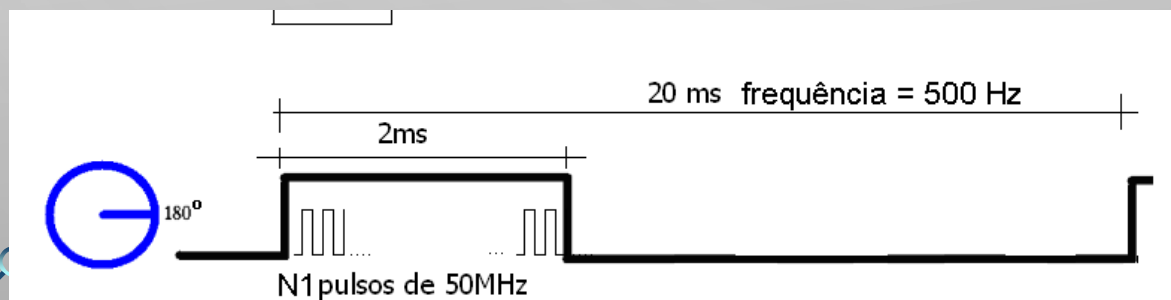
O contador deve receber na entrada **DATA** o número (N1) de pulsos de 50MHz que cabem em 2 ms;

A entrada **CLOCK** é a frequência de 50MHz (a qual será contada)

Na entrada **cnt_en** deve ter um pulso de duração de 20 ms, desta forma o contador vai receber o valor que está na entrada **DATA**, quando a entrada **aload** = '1' e vai começar a contagem decrescente de N1 pulsos, quando a contagem chegar no zero, terá contado o tempo de 2ms';

A saída **Cout** é ligada a entrada **aload** para que toda vez que a contagem chegar ao zero, carregar o valor da entrada **DATA** novamente. (**aload** é a entrada que possibilita o carregamento da entrada **DATA**)

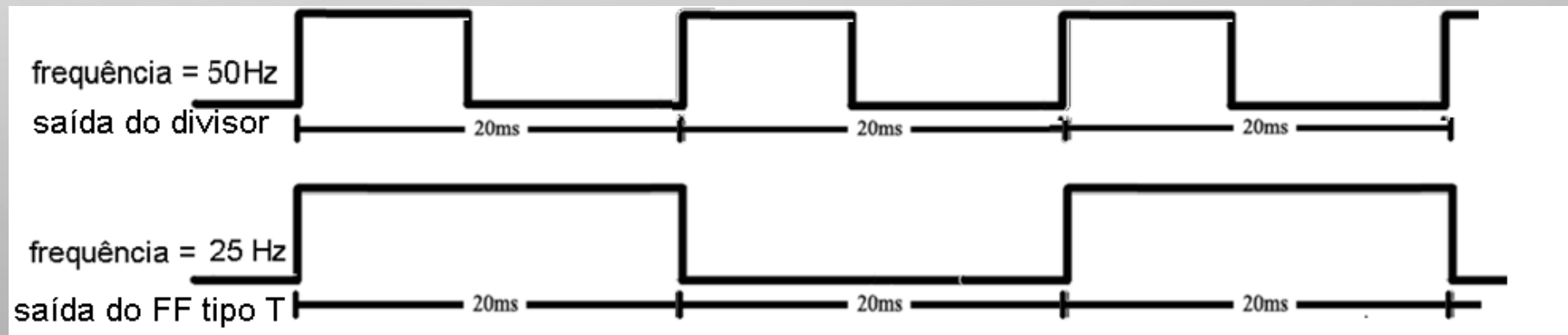
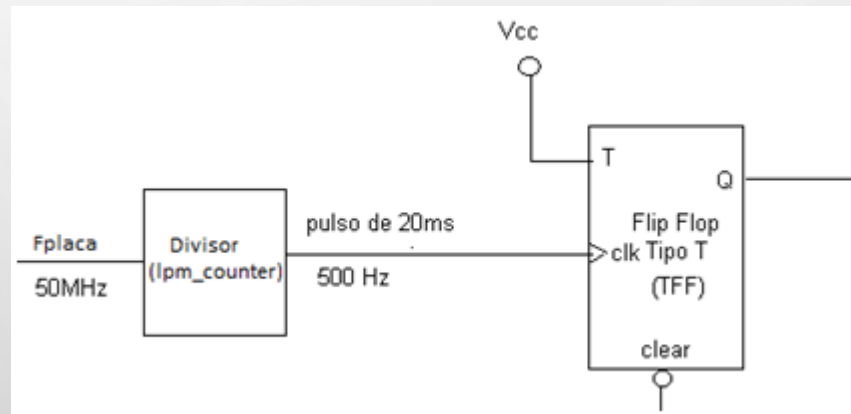
valor N1 deve ser criado como uma constante com o projeto **lpm_constant** (salvar com o nome **const2ms**



Implementação do Circuito :

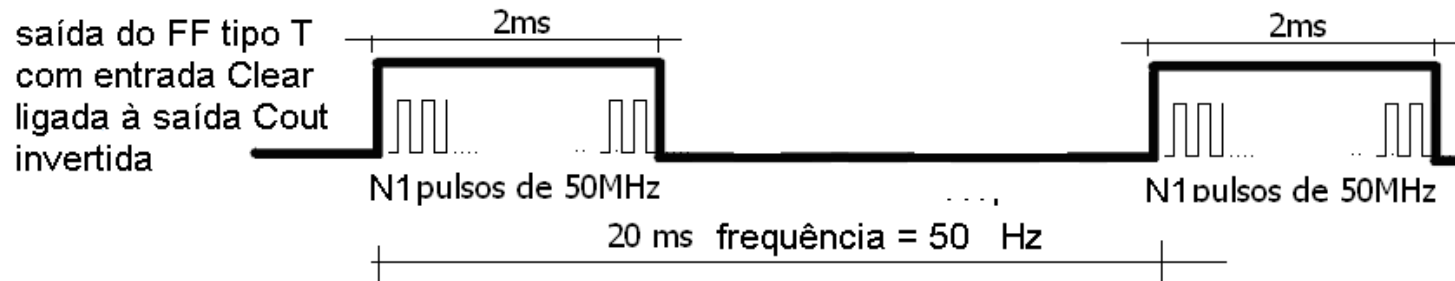
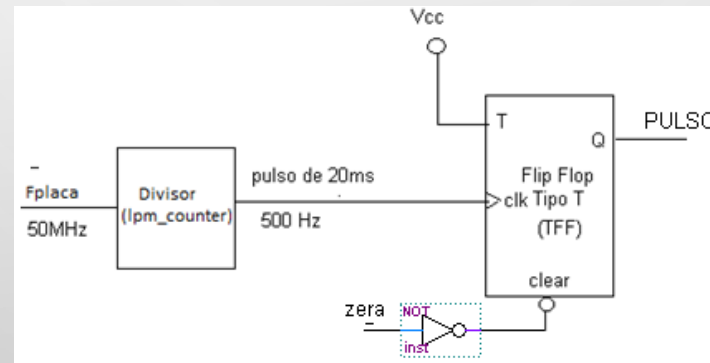
Para gerar o pulso de da entrada **cnt_en** que deve ser um sinal de 20 ms em nível alto:, usar o projeto TFF com a configuração no modo Toggle, ou seja, a entrada T em Vcc.

O pulso na entrada de clock deve ter a frequência de 50Hz (20 ms de período)



Implementação do Circuito :

A saída do FF tipo T é transformado no PULSO PWM correto para o controle do servomotor, qdo a saída do FF tipo T for zerada quando o contador contar a quantidade de pulsos N1 (par gerar 2 ms) Isso é obtido ligando na entrada clear do FF tipo T a saída Cout do contador decrescente invertida. Pois, quando o contador executa contagem decrescente de N1 pulsos, ao chegar ao zero a saída Cout é colocada em nível alto (acusando o final da contagem). Utilizando a saída Cout invertida. Lig-se ela a entrada Clear (ativa em nível baixo) do FF tipo T, colocando assim a saída Q em zero. Gerando o pulso mostrado abaixo

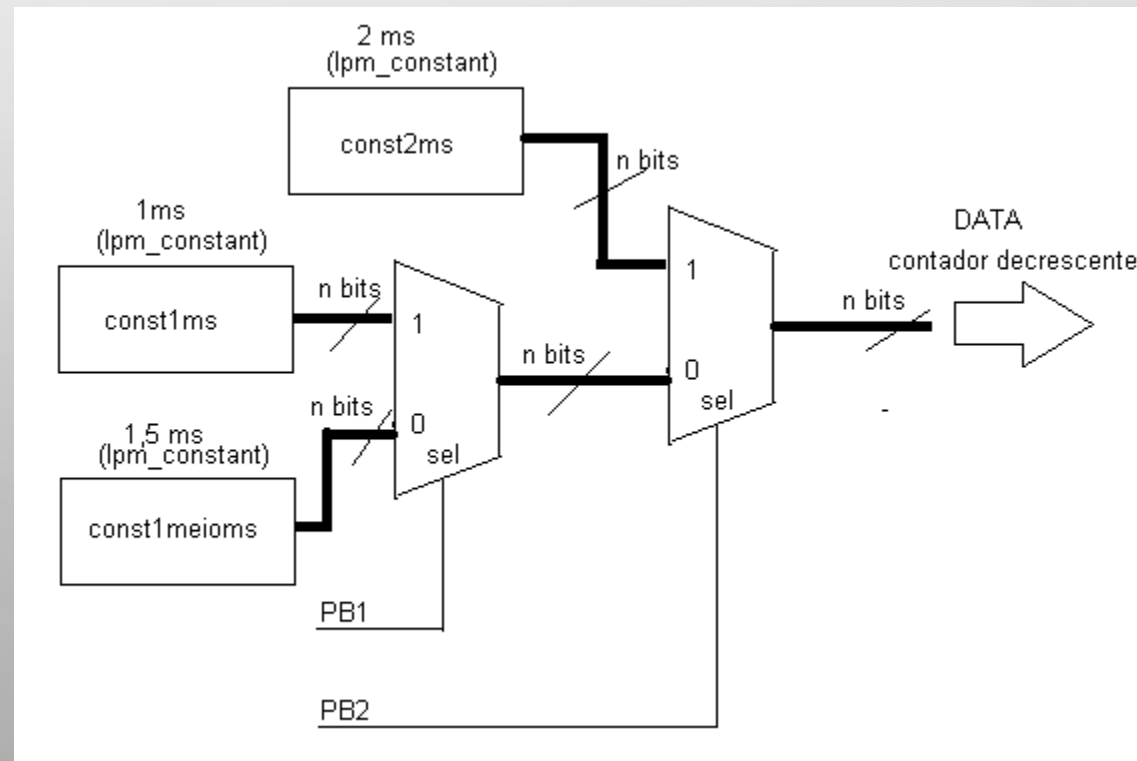


Implementação do Circuito :

- Para criar pulsos PWM de 1ms e de 1,5 ms trocar o valor da entrada **DATA** do contador decrescente para a constante que define a quantidade de pulsos de 50 MHz que cabem em 1ms e de 1,5 ms. Portanto criar mais 2 constantes com o projeto **lpm_constant** e salvar com os nomes **const1ms** e **const1meioms**.

Então, utiliza-se 2 multiplex que vão selecionar qual das constantes será ligada à entrada **DATA**:

- Se a chave PB1 for acionada, a constante **const1ms** é ligada à entrada DATA (posição 0° do servomotor) ;
- Se a chave PB2 for acionada, a constante **const2ms** é ligada à entrada DATA (posição 180° do servomotor); ;
- E se nenhuma chave for acionada a constante **const1meioms** é ligada à entrada DATA (posição 90° do servomotor)



Circuito Final de controle do servomotor:

