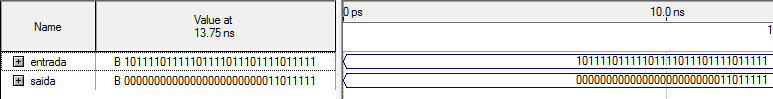
**Entidade: Loadbyte**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da Simulação:

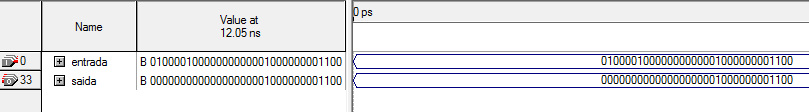
Neste waveform o modulo loadbyte esta recebendo os 32 bits da entrada e selecionando os 8 bits menos significativos para concatenar com 24 zeros e liberar na saída.

**Entidade: Loadhalf**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da Simulação:

Neste waveform o modulo loadhalf esta recebendo os 32 bits da entrada e selecionando os 16 bits menos significativos para concatenar com 16 zeros e liberar na saída.

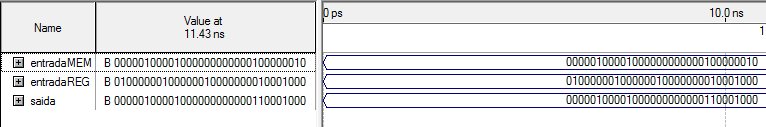
**Entidade: Storebyte**

Descrição das portas:

EntradaMEM: representa a entrada de dados que vem da memória.

EntradaREG: representa a entrada de dados que vem do registrador.

Saida: representa o número de saída no módulo.



Descrição da Simulação:

Neste waveform o modulo storebyte esta recebendo os 32 bits da entradaMEM e da entradaREG e selecionando os 8 bits menos da entradaREG e os 24 mais significativos da entradaMEM e concatenando-os para gerar a saída de 32 bits mostrada a baixo.

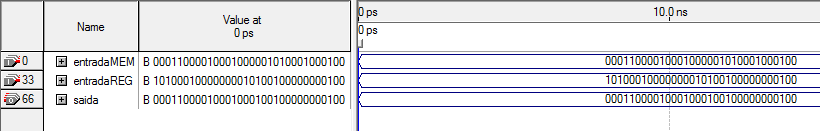
**Entidade: Storehalf**

Descrição das portas:

EntradaMEM: representa a entrada de dados que vem da memória.

EntradaREG: representa a entrada de dados que vem do registrador.

Saida: representa o número de saída no módulo.



Descrição da simulação:

Neste waveform o modulo storehalf esta recebendo os 32 bits da entradaMEM e da entradaREG e selecionando os 16 bits menos da entradaREG e os 16 mais significativos da entradaMEM e concatenando-os para gerar a saída de 32 bits mostrada a baixo.

**Entidade: Caixa1511**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da simulação:

Neste waveform o modulo recebe uma entrada de 16 bits, e seleciona os 5 bits que vão para a sáida.

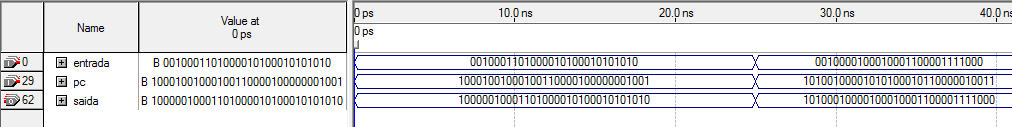
**Entidade: Concat**

Descrição das portas:

Entrada: representa o número principal de entrada no módulo.

Pc: representa o número que vem do pc para ser concatenado.

Saida: representa o número de saída no módulo.



Descrição da simulação:

Neste waveform o modulo recebe os 28 bits referentes à saída do shiftleft2 e os 32 de PC. Concatena os 28 com os quatro mais significativos do PC e libera esses 32 bits de saída.

**Entidade: Concatena**

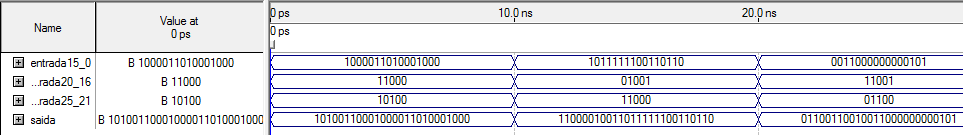
Descrição das portas:

Entrada15\_00: representa a entrada de 16 bits do módulo.

Entrada20\_16: representa uma das entradas de 5 bits do módulo.

Entrada25\_21: representa uma das entradas de 5 bits do módulo.

Saida: representa o número de saída no módulo.



Descrição da Simulação:

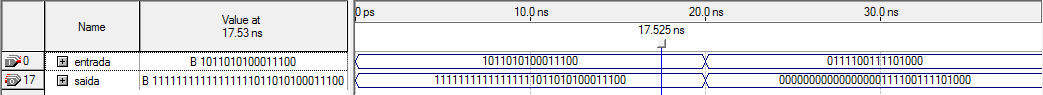
Neste waveform o modulo recebe 3 entradas, que somadas formam 26 bits, e as concatena para mandar para a saída.

**Entidade: Extensaodosinal**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da simulação:

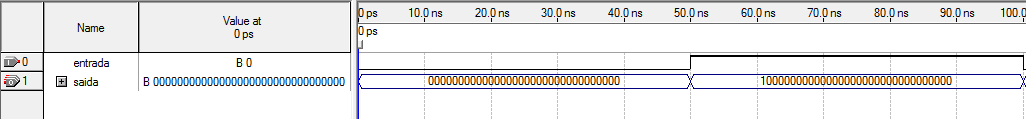
Neste waveform o módulo recebe o número de 16 bits e extende seu sinal, resultando numa saída de 32 bits.

**Entidade: Extensordoslt**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da simulação:

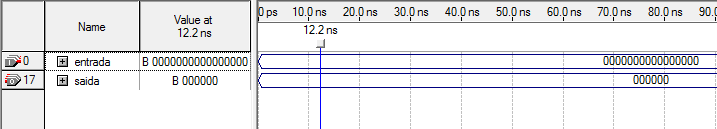
Neste waveform o módulo recebe um numero de 1 bit, e concatena-o com 31 bits zero.

**Entidade: Functparacontrole**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da Simulação:

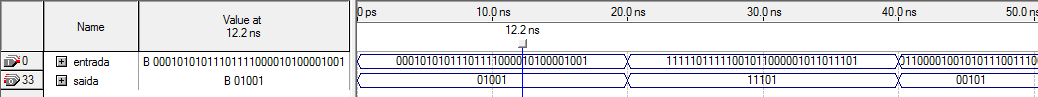
Neste waveform o módulo recebe 16 bits e seleciona 6 bits para a saída, que representa o funct.

**Entidade: Rsparashift**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da Simulação:

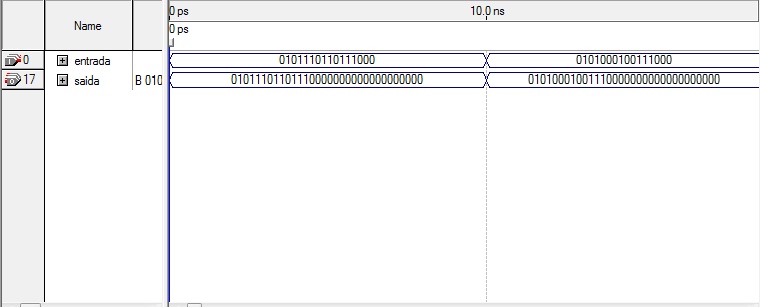
Neste waveform o módulo recebe um numero de 32 bits e seleciona os 5 menos significativos para a saída, para ser liberado para o shift.

**Entidade: Lui**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o número de saída no módulo.



Descrição da simulação:

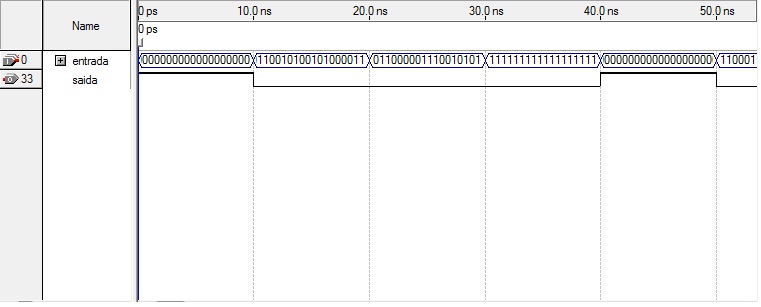
Neste waveform o módulo recebe um numero de 16 bits e o coloca como os 16 bits mais significativo, colocando os 16 menos significativos como zero.

**Entidade: Nop**

Descrição das portas:

Entrada: representa o número de entrada no módulo.

Saida: representa o sinal de saída do módulo.



Descrição da Simulação:

Nesta simulação Quando o valor da entrada são 32 bits zero, ele libera o sinal 1, caso contrário, ele libera o sinal zero.

**Entidade: Multiplicador**

Descrição das portas:

Clk : representa o clock do sistema.

X :Representa o multiplicando.

Y: Representa o multiplicador.

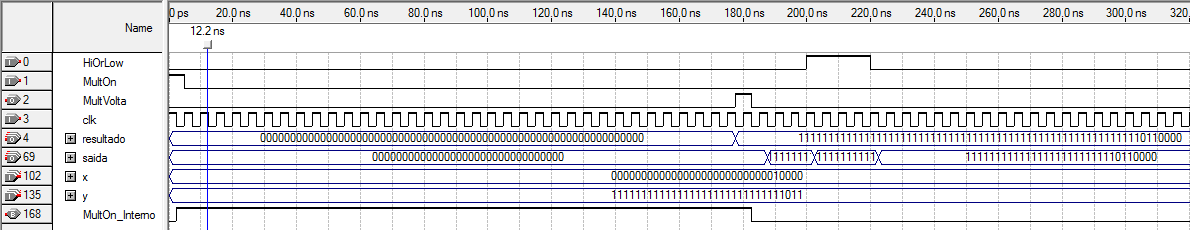
MultOn: Sinal que ativa o multiplicador.

HiOrLow :Sinal que indica indica a liberação de um valor pelo componente.

Saída: Vetor que recebe a saída requisitada.

MultVolta :sinal que indica o fim da multiplicação.

Saida: representa o sinal de saída do módulo.



Descrição da Simulação:

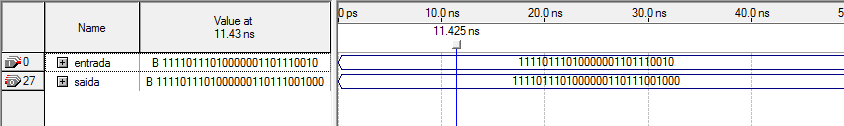
Primeiro ele recebeu as entradas e ativou o sinal do multon, depois ele esperou acabar a multiplicação e então retornou o sinal do multvolta. Após alguns ciclos de clock ele ligou o hiorlow para liberar também a parte alta.

**Entidade: Shiftleft228**

Entradas:

Entrada: representa o número de 26bits de entrada no módulo.

Saida: representa o número de 28 bits saída no módulo.



Descrição da simulação:

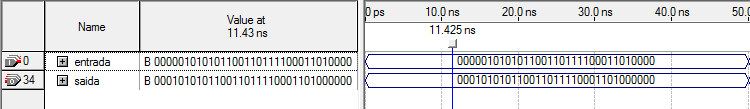
Neste waveform um número de 26 bits entra, e é colocado como os 26 bits mais significativos da saída, sendo concatenado com dois bits zero.

**Entidade: Shiftleft228**

Entradas:

Entrada: representa o número de 32bits de entrada no módulo.

Saida: representa o número de 32 bits (já dado o shift)de saída no módulo.



Descrição da simulação:

Neste waveform um número de 32 bits entra e é dado um shiftleft2, onde os dois primeiros bits são perdidos, e os dois últimos bits passam a ser zero.

**Entidade: Shift**

Descrição das portas:  
Clk: Representa o clock do sistema  
Entrada: Vetor que possui a palavra a ser deslocada,ou seja, que vai sofrer o shift.  
Shift: Representado por 3 bits, indica que operação de deslocamento ou não será efetuada.  
N: Representa o número de vezes que a palavra vai ser deslocada ou rotacionada.  
rs: Na instrução de deslocamento srav ela utiliza o valor que está em RS para usar como a quantidade de vezes que a entrada deve ser deslocada.  
Reset: O reset ele zera a saída quando o seu valor vai para “1”.  
Saida: Vetor que possui a palavra com a instrução existente no shift efetuada.

|  |  |
| --- | --- |
| Shift | Descrição |
| 000 | Shift a esquerda n vezes |
| 001 | Shift a esquerda rs vezes |
| 100 | Shift lógico para a direita n vezes |
| 010 | Shift aritmético para a direita n vezes |
| 011 | Shift aritmético a direita rs vezes |

N:Representado por 5 bits,possibilita deslocamento de 0 bits até 32 bits.  
RS: Representado por 5 bits,possibilita deslocamento de 0 bits até 32 bits.  
Reset:Assume valores entre 0 e 1.Com o valor 0 o reset não faz nada ,porém, quando assume o valor 1 ele reseta(zera) a saída.

Descrição das simulações:

Para o clock(Clk) = 1 e o Reset = 1:





Como pode ser observado enquanto o clock está com 1 e o Reset com 1 o valor da saída é zerado.

Para o clock(Clk) = 1 e o Reset = 0:

Quando o shift = 000 a instrução é para a entrada sofrer um deslocamento aritmético ou lógico N vezes para a esquerda.





Na primeira imagem a entrada 00000000000000000000000000010101 sofre um deslocamento para a esquerda de 5 vezes tendo como saída 00000000000000000000001010100000.

Na segunda imagem a entrada 00000000000000000000000000110101 sofre um deslocamento para a esquerda de 13 vezes tendo como saída 00000000000001101010000000000000.

Quando o shift = 100 a instrução é para a entrada sofrer um deslocamento lógico N vezes para direita.





Na primeira imagem a entrada 0000000000000000000000000000110 sofre um deslocamento lógico para a direita 1 vez gerando a saída 00000000000000000000000000000011.  
Na segunda imagem a entrada 00000000000000000000000000010111 sofre um deslocamento lógico para a direita 5 vezes gerando a saída 00000000000000000000000000000000.

Quando o shift = 010 a instrução é para a entrada sofrer um deslocamento aritmético N vezes para direita.





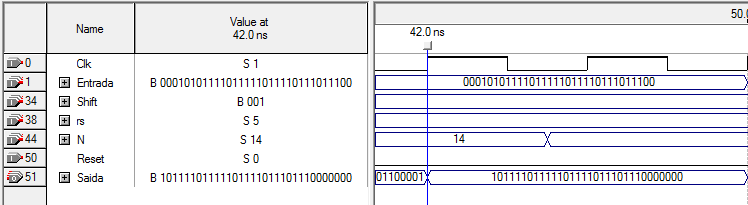
Na primeira imagem a entrada 00000000000000000000000000101001 sofre um deslocamento aritmético para a direita de 10 vezes conservando o sinal 0 na saída 00000000000000000000000000000000.  
Na segunda imagem a entrada 10101111100111011001010100010111 sofre um deslocamento aritmético para a direita 2 vezes conservando o sinal 1 na saída 11101011111001110110010101000101.

Quando o shift = 011 a instrução é para a entrada sofrer um deslocamento aritmético rs vezes para direita.



Na imagem a entrada 00010101111011111011110111011100 sofre um deslocamento aritmético para a direita de 2 vezes(valor que está em RS) conservando o sinal 0 na saída 00000101011110111110111101110111.  
Nesse caso a quantidade de deslocamentos é dita pelo valor que está em RS e não pelo valor que está em N.

Quando o shift = 001 a instrução é para a entrada sofrer um deslocamento lógico para a esquerda rs vezes.



Na imagem a entrada 00010101111011111011110111011100 sofre um deslocamento lógico para a esquerda de 5 vezes(valor que está em RS) 10111101111101111011101110000000.  
Nesse caso a quantidade de deslocamentos é dita pelo valor que está em RS e não pelo valor que está em N.