Universidade Federal de Minas Gerais
Instituto de Ciências Exatas
Departamento de Ciência da Computação

DCC819 - Arquitetura de Computadores

Relatório IV - Unidade de Multiplicação

Iuri Silva Castro
João Mateus de Freitas Veneroso
Ricardo Pagoto Marinho

Belo Horizonte - MG
18 de novembro de 2017

1 Descrição

Até a atual implementação o processador proposto não possuía instrução de multiplicação específica. Deseja-se agora implementar a instrução *MUL*, que fará a multiplicação de dois registros de 16-bit, armazendo o resultado em dois registros *HI* e *LO*, ambos de 16-bit, sendo que o registro *HI* mantém os 16-bit mais significativos e o registro *LO* mantém os 16-bit menos significativos. A fim de melhorar o desempenho de tal instrução, um módulo (hardware) de multiplicação dedicado será implementado, não havendo alterações no funcionamento da ULA. Duas instruções *GHI* e *GLO* foram implementadas para fazer o armazenamento de tais registros no banco de registradores.

A implementação do módulo e das instruções são descritas nas seções seguintes. Textes e discussões são feitas ao final do trabalho.

2 Implementação

Inicialmente, propõe-se a implementação do módulo de multiplicação, que será paralelo a *ULA*. O módulo recebe dois operandos de 16-bit e retorna o resultado em 32-bit, divididos em *HI* e *LO*. O diagrama pode ser visto na figura **??**.

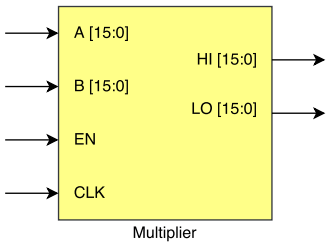

Figura 1: Módulo de multiplicação.

Os sinais de *clock* (*CLK*) e *enable* (*EN*) são utilizados para sincronizar e atualizar os valores dos registros *HI* e *LO*, que são mantidos dentro do módulo.

Para dar suporte a utilização do novo módulo, as seguintes instruções foram adicionadas:

- *MUL -, opA, opB*: Multiplica o operador A com o B, o resultado é armazenado nos registros *HI* e *LO*, interno ao módulo;
- *GHI dest, -, -*: Acessa o registro *HI* dentro do módulo de multiplicação e armazena no registrador *dest* especificado;
- *GLO dest, -, -*: Acessa o registro *LO* dentro do módulo de multiplicação e armazena no registrador *dest* especificado.

A instrução de multiplicação não altera diretamente o banco de registradores, sendo necessário as duas instruções auxiliares. As adições requerem com que o sistema de controle da máquina seja alterado. O diagrama do módulo de controle pode ser visto na figura **??**.

Os novos sinais *IsMulWB*, *HILO* e *HILO-WB* foram adicionados para fazer o encaminhamento dos dados do multiplicador para o banco de registradores e indicar ao multiplicador que ele deve atualizar os registros *HI* e *LO*.

Com o módulo implementado e as atualizações feitas, parte-se para a integração do sistema.

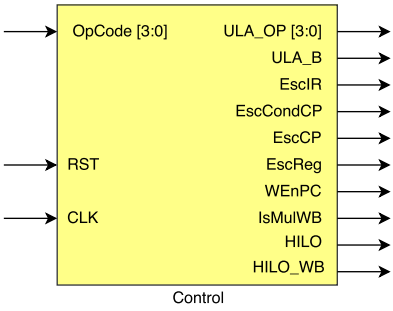

Figura 2: Módulo de controle.

3 Integração

Com a adição do módulo de multiplicação, alterações são necessárias no encaminhamento dos dados de entrada e saída do módulo. Dois multiplexadores foram adicionados para fazer os encaminhamentos. A figura **??** mostra como ficou o sistema.

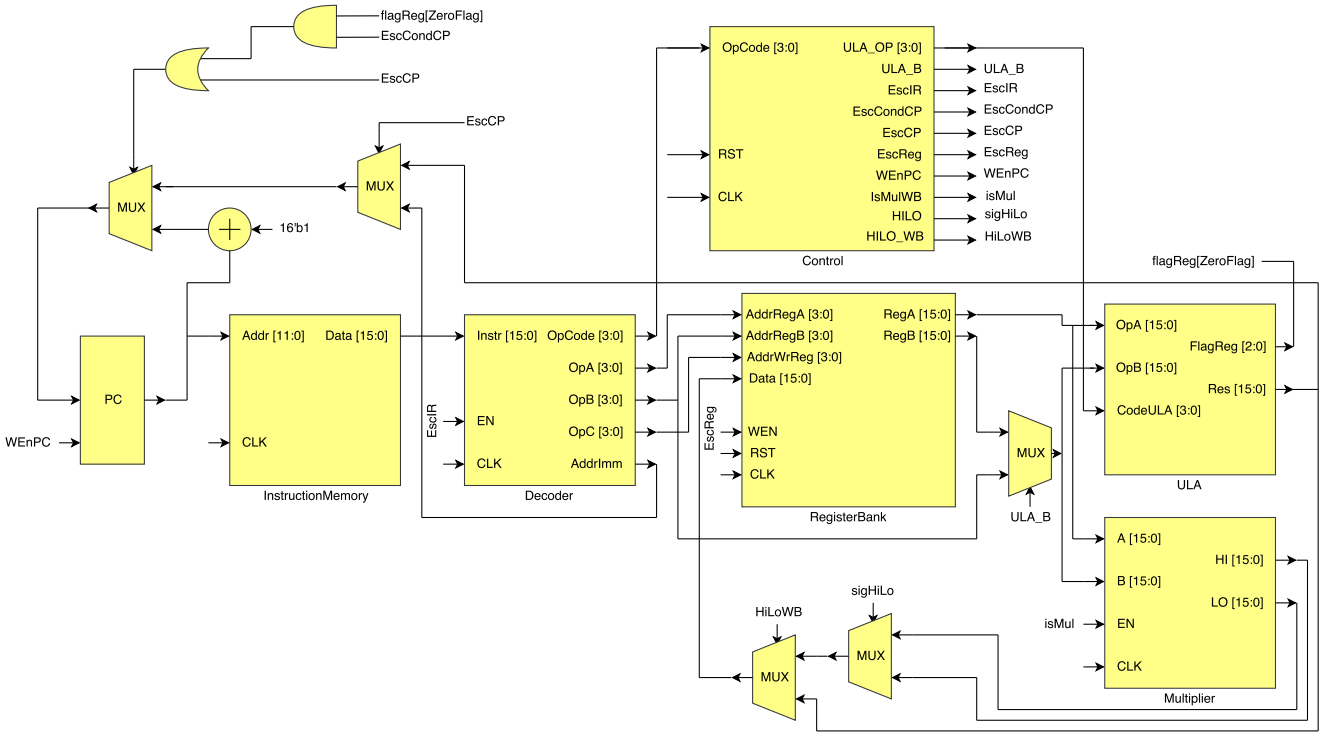

Figura 3: Integração do sistema.

O sinal *HILO* indica qual dos registradores da multiplicação será armazenado no banco de registradores, o sinal *HILO-WB* indica o módulo que ele deve atualizar os registradores *HI* e *LO*, pois uma nova multiplicação foi executada, e o sinal *IsMulWB* indica se está trabalhando com dados da ULA ou do multiplicador.

4 Simulação e Testes

Para a validação, preparou-se um teste com 4 multiplicações para serem simuladas no software ModelSim. No início da simulação, reinicia-se a máquina para zerar os valores dos registros e inicializar a máquina de estados do módulo de controle. Após a reinicialização da máquina, carrega-se os valores a serem multiplicados no banco de registradores, e defini-se as instruções de multiplicação e de armazenamento do resultados, não utilizou-se, dessa vez, o arquivo para inicialização da memória de instruções, as instruções foram carregadas diretamente do *script* de simulação. A figura **??** mostra a saída da simulação.

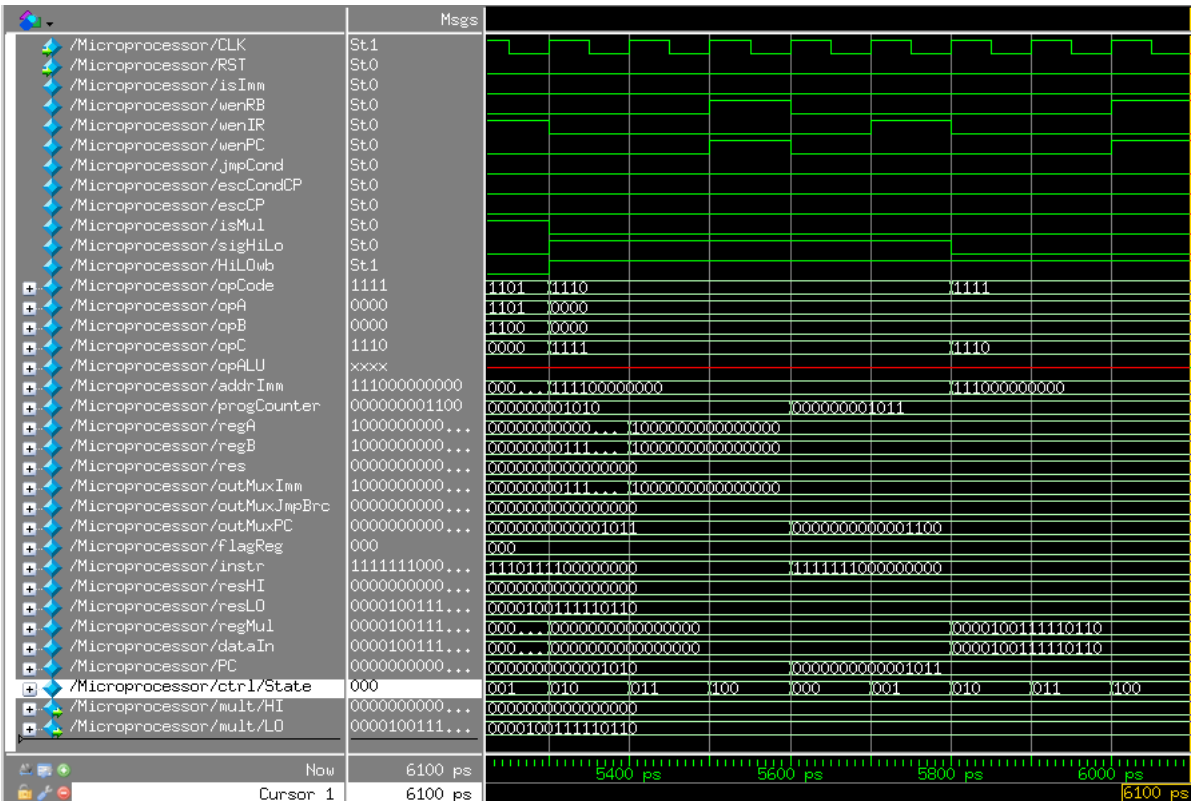

Figura 4: Formas de onda de saída da simulação.

Os valores dos testes foram os seguintes:

- *Teste 1*: 128 (Reg0) x 5 (Reg1). Armazenamento: HI (Reg3) LO (Reg2);
- *Teste 2*: 65535 (Reg4) x 15 (Reg5). Armazenamento: HI (Reg7) LO (Reg6);
- *Teste 3*: 500 (Reg8) x 500 (Reg9). Armazenamento: HI (Reg11) LO (Reg10);
- *Teste 4*: 255 (Reg12) x 10 (Reg13). Armazenamento: HI (Reg15) LO (Reg14).

O resultado dos dados após a simulação pode ser observado no banco de registradores, conforme figura **??**.

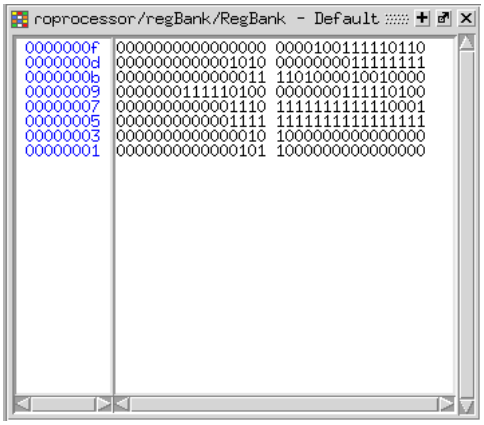

Figura 5: Resultado no banco de registradores.

5 Discussões

A instrução *SLT reg-reg*, adicionada no trabalho passado para auxiliar na ordenação dos registros, foi removida, pois o tamanho de instruções atual só permite a indexação de 16 opcodes, e não haveria espaço para adicionar as 3 novas instruções necessárias.

A instrução de multiplicação implementada trabalha apenas com números inteiros positivos.

Ter um hardware especializado, neste caso em multiplicação, melhora o desempenho do processador ao custo de maior complexidade de projeto e maior área de silício ocupada.