

Projeto MAC0329

- Artur Alvarez 9292931
- Bruno Arico 8125459
- Gabriel Capella 8962078
- Nicolas Nogueira 9277541

Estrutura do Projeto

A segunda parte do projeto consiste na implementação dos principais elementos de um ciclo de instrução de nosso computador HIPO.

Um ciclo de instrução (*fetch-decode-execute cycle* ou *FDX*) é o ciclo básico de operação de uma computador; ele é constituído das seguintes etapas:

1. busca de uma instrução na memória (*fetch*);
2. decodificação da instrução, determinando as ações exigidas pela mesma;
3. execução das ações determinadas pela instrução;
4. repetição da ação do item 1.

Descrição do ciclo de instrução

Um ciclo de instrução de nosso computador HIPO é constituído das seguintes etapas:

1. leitura, da memória, do par instrução/endereço apontado por PC (*program counter*), armazenando-o no IR (*instruction register*);
2. incremento do valor do PC;
3. decodificação da instrução armazenada no IR;
4. execução da instrução decodificada;
5. se a instrução executada não for STOP, retorna para o item 1.

As etapas do ciclo de instrução são sequenciais e a transição entre as mesmas é dada através de pulsos de um *clock*.

Componentes do ciclo de instrução

Memória. A memória é implementada por um bloco RAM que é constituído de unidades de 16 bits com endereçamento de 8 bits. Para cada endereço de memória, se o mesmo for instrução, o primeiro byte é o código da mesma e o segundo um endereço de memória; caso contrário, os dois bytes constituem um dado.

Acumulador e IR. Ambos são registradores de 16 bits. O acumulador (Acu) armazena um dado; já o IR armazena no primeiro byte um código de instrução e no segundo um endereço de memória.

PC. O PC é um registrador especial e tem 8 bits, ele armazena o endereço de memória lido e que a cada pulso incrementa o seu valor em uma unidade. O PC é implementado por um contador.

Controlador. O controlador é componente central do computador. Ele é responsável pela

decodificação da instrução armazenada no IR, pelo incremento do PC e pela definição do modo de operação (leitura/escrita) da memória e dos registradores. É implementado com o uso de flip-flop.

Entrada e Saida de dados

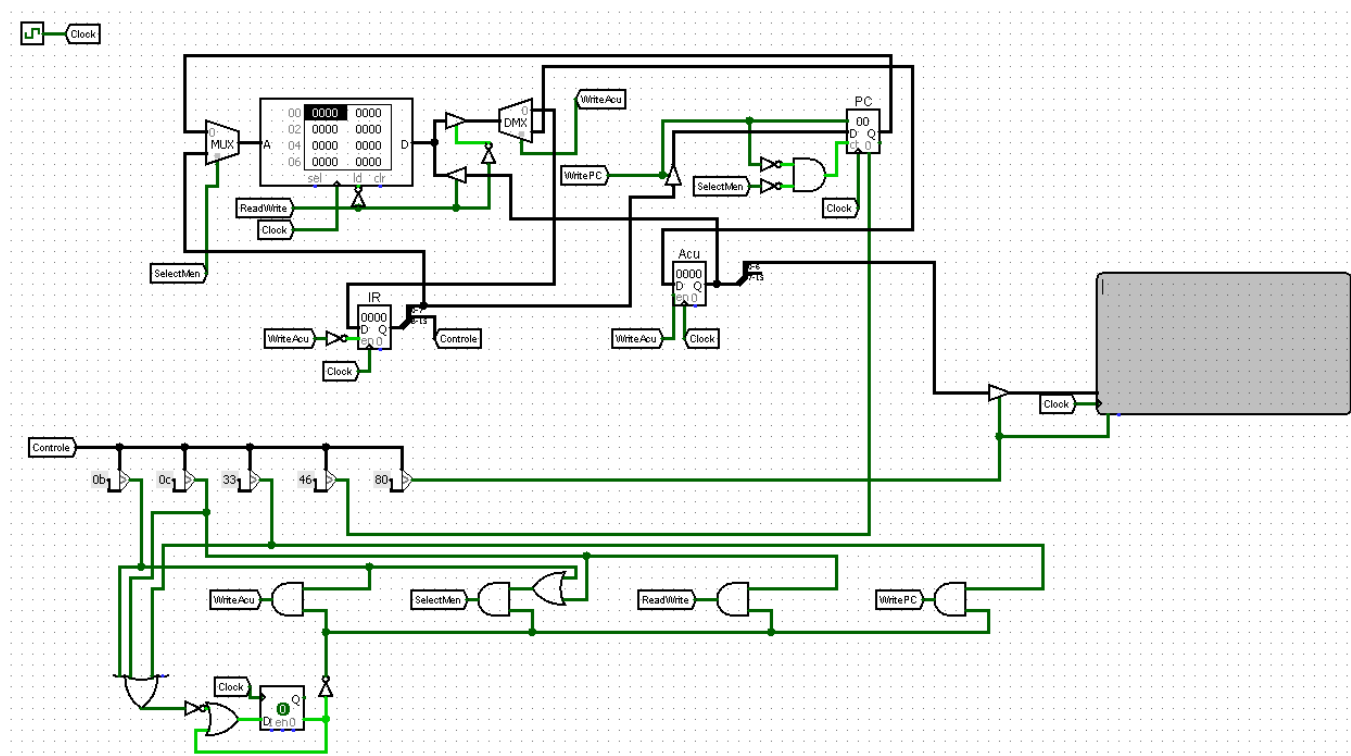
A entrada de dados é realizada via terminal que pode ser testado em conjunto com o arquivo 'mem-image.txt', que esta no anexo, usando o seguinte comando:

```
logisim FDX.circ -tty tty -load mem-image.txt
```

Os dados de saida são impressos no respectivo terminal.

Circuito

O circuito é dividido nos 5 componentes do ciclo de instrução já mencionados: memória, acumulador, IR, PC e controlador.



Breve descrição sobre o funcionamento do circuito para o exemplo em questão:

Os dados inseridos pelo arquivo 'mem-image.txt' são inseridos na SRAM, em seguida o primeiro comando é lido "0b04" que no sistema de controle, por comparação, indica para acionar o circuito que carrega o endereço, no caso, '04' da SRAM no acumulador (Acu), feito isso, no proximo ciclo de maquina, a posição 04 da memoria é lida e então no proximo ciclo o PC é incrementado de 1 e o proximo comando é lido, "8000", que corresponde ao comando que habilita a escrita na saida padrão (no caso terminal), após isso o PC é incrementado e o próximo comando é lido "3303", que corresponde ao salto não condicional para o endereço '03' da SRAM, incrementa-se o PC e o comando "4600" é carregado, ele corresponde ao fim da execução, nesse comando o PC é zerado e se reinicia o programa que esta carregado no endereço '00' da SRAM.

Links úteis/Bibliografia

<http://minnie.tuhs.org/CompArch/Tutes/week03.html>