

# Organização e Arquitetura de computadores



**Educação a Distância**  
Cruzeiro do Sul Educacional  
*Campus Virtual*



# Material Teórico



## Unidade de Controle e Clock

### **Responsável pelo Conteúdo:**

Prof. Ms. Vagner Silva

### **Revisão Textual:**

Prof<sup>a</sup>. Vera Lidia de Sá Cicaroni



# UNIDADE

## Unidade de Controle e Clock



- Unidade de Controle
- O relógio (clock)
- Registrador de Instruções
- Contador e decodificador de Instruções



**Objetivo de  
APRENDIZADO**

Nesta quarta aula, você poderá se aprofundar, mais ainda, nos conceitos relacionados ao processador e começará a estudar a unidade de controle, que terá algumas de suas funções e características detalhadas. Logo após, você aprenderá o que é e para que serve o clock de um computador. Algumas características e particularidades desse componente serão descritas. Você conhecerá também, ao estudar o texto, algumas instruções assembly e seu significado.

O primeiro material a ser lido encontra-se em Material Didático. Trata-se da contextualização, na qual você poderá conhecer a técnica usada pela Intel para aumentar o desempenho do processador.

Logo após, você deverá ler o conteúdo teórico sobre a aula. Nesse documento, você encontrará, como primeiro assunto, a unidade de controle. Você verá uma descrição mais detalhada sobre a função da unidade de controle e sobre como funcionam os principais blocos. É importante que entenda bem como esses blocos se comunicam. Caso não consiga entender em uma primeira leitura, leia novamente, quantas vezes sejam necessárias para compreender bem. Estes conceitos são importantes para as demais aulas. Logo após, você estudará como se dá a sincronização e como as instruções são executadas tendo como referência o relógio (clock). A explanação abordará o relógio, a velocidade e o grande problema encontrado em relação ao aumento da taxa de velocidade. Por último você encontrará mais detalhes do registrador de instruções em relação à decodificação e à execução das instruções.

## Contextualização

Aumentar a velocidade de processamento é também objetivo das indústrias. No conteúdo desta aula, você irá estudar a técnica chamada de overclock, que se usa com o objetivo de aumentar o desempenho do processador.



### Explore

Veja, no link abaixo, uma técnica usada pela Intel para aumentar esse desempenho sem prejudicar a vida útil do processador.

<http://informatica.hsw.uol.com.br/arquitetura-processador-nehalem4.htm>

A unidade de controle tem funções pré-determinadas dentro do processador, as quais são executadas pelas instruções que são estabelecidas no projeto. Sendo assim, ela tem os circuitos necessários para acionar a memória ou os dispositivos de entrada e saída e ainda controlar se os dados serão armazenados ou lidos. Nesta aula você irá estudar, de forma um pouco mais detalhada, as funções da unidade de controle.

## 1. Unidade de Controle



A Unidade de Controle (UC) faz parte de um dos blocos do processador; ela é responsável pela decodificação e execução das instruções, fornecendo os sinais de sincronização adequados para as diversas partes do processador e do próprio computador. Além das funções citadas acima, ela contém registradores para armazenamento da Informação Binária (dados, endereços e instruções).

Ela coordena o computador na execução das instruções armazenadas em um programa. A unidade de controle contém os circuitos eletrônicos arranjados de forma que possa executar as instruções da CPU.

A unidade de controle é como um maestro de uma orquestra, pois comanda algumas partes do sistema. Para isso, o conjunto de instruções, embutido nos circuitos da unidade de controle, nada mais é que uma lista de todas as operações que a CPU é capaz de executar.

As CPUs, que são fabricadas por empresas diferentes, têm conjuntos de instruções diferentes, e até mesmo modelos diferentes de CPU fabricados pela mesma empresa podem apresentar conjuntos de instruções também diferentes.



Quando uma nova CPU é desenvolvida, seu conjunto de instruções tem as mesmas instruções de sua antecessora e, geralmente, são incluídos mais alguns comandos novos. Essa estratégia é conhecida como **Compatibilidade ascendente**.

### 1.1 Função Controle

Desempenhada pela unidade de Controle, esta função coordena o andamento de todas as atividades da CPU. A Unidade de Controle (UC) é a responsável pela execução da Função Controle e é, também, um dos elementos mais complexos da CPU.

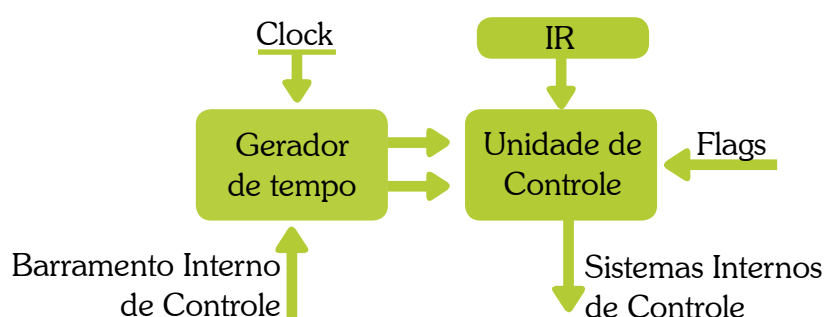


Figura 1: Diagrama da unidade de Controle

A imagem apresentada na figura acima refere-se à unidade de controle e, fisicamente, encontra-se dentro da CPU. Abaixo você estudará as características de cada bloco.

O registrador de instruções (IR), através do processo de busca, recebe as instruções da memória principal (RAM) e passa-as para a unidade de controle já preparada para ser executada.

A unidade de controle, recebendo essas instruções no formato binário, aciona os circuitos correspondentes para que a ação seja executada. Dentre as ações temos: acionamento da escrita na memória ou dispositivo de entrada e saída, acionamento da leitura da memória ou dispositivo de entrada e saída, e acionamento da memória ou dispositivo de entrada e saída para qualquer outra atividade que necessite desses componentes.

O Gerador de Tempo, também conhecido como sincronização, conforme pode ser observado na figura acima, tem como função gerar e enviar para a unidade de controle (UC) a temporização necessária para a sincronização das operações. Esse bloco toma como base de tempo o relógio (clock) do sistema, que é a referência de tempo do sistema. O relógio é expresso em valores de frequência Hertz (Hz) e é largamente utilizado para definir a velocidade da CPU.

Para melhor entendimento, vamos considerar que, a cada ciclo de instrução, a unidade de controle (UC) deve coordenar a execução de uma instrução completa, que consiste na execução de uma tarefa básica da CPU (Ex: Incrementar acumulador ACC).



### Atenção

Algumas instruções do processador precisam de mais que um ciclo para poderem ser totalmente executadas.

Cada instrução de máquina é subdividida em um conjunto de operações ainda menores, denominadas “micro-operações”. A micro-operação é a menor ação que pode ser realizada pelo processador. Dessa forma, cada ciclo de relógio é dividido pelo Gerador de Tempo em intervalos de tempo menores, nos quais é realizada a micro-operação.

Exemplo: Micro-operações realizadas em um ciclo de busca:

t0: MAR  $\leftarrow$  PC

t1: PC  $\leftarrow$  PC + i

MBR  $\leftarrow$  MEM(MAR)

t2: IR  $\leftarrow$  MBR

O pequeno trecho de código acima significa o seguinte: no tempo zero (t0), o contador de programa (PC) insere, no registrador de endereço (MAR), o seu conteúdo. No tempo um (t1), o contador de programa é incrementado de um (PC + 1), uma informação é solicitada de um endereço da memória (MEM(MAR)) e inserido no registrador MBR. No tempo (t2) o registrador MBR insere a instrução no registrador IR.



## 1.2 Tipos de Controle em uma CPU

Como já foi descrito acima, a principal função de uma unidade de controle (UC) é decodificar uma instrução e sequenciar as micro-operações correspondentes. Cada micro-operação deve ser previamente definida na unidade de controle, seja através de hardware (Programação Direta no Hardware) ou de software (Microprogramação). Vamos detalhar um pouco mais cada uma delas.

### 1.2.1 Controle Programado no Hardware

Na implementação através de Programação Direta no Hardware, a unidade de controle é implementada através de circuitos digitais combinacionais, de forma que os sinais de controle são gerados diretamente por hardware.

A principal vantagem deste tipo de implementação é o ganho de desempenho na execução das instruções, pois elas são executadas diretamente pelo hardware sem a necessidade de interpretadores e a principal desvantagem está no maior custo, pois todas as instruções devem ter um circuito que a execute. Outra desvantagem está na inflexibilidade para se fazer qualquer alteração, pois, por ser implementada diretamente no hardware, fica complexo qualquer tipo de alteração. Este tipo de controle é utilizado em máquinas RISC, por possuírem um conjunto de instruções reduzido.

### 1.2.2 Controle por Microprogramação

No controle por Microprogramação, cada instrução de máquina deve ser decodificada em uma série de microinstruções menores para ser executada de acordo com um hardware. As microinstruções constituem um processo prático de definir as micro-operações.

Para fazer isto, a unidade de controle deve possuir, em sua estrutura interna:

- Memória de controle – tipicamente uma ROM, que armazena os códigos das microinstruções;
- MPC – Microcontador de instruções (análogo ao PC);
- MIR – Registrador de microinstrução (análogo ao IR).

O Microprograma é considerado o conjunto de microinstruções necessárias para a realização de uma instrução de máquina (Firmware).

A cada ciclo de instrução, a unidade de controle deve localizar e executar o microprograma correspondente à instrução a ser executada. Um problema para executar esses microprogramas estaria em localizar o início de cada um deles, considerando que temos vários, de forma correta, na memória de controle. Uma solução é utilizar o próprio valor do OpCode da instrução, ou seja, parte da instrução, para fazer parte desse endereço para identificar o início da microinstrução. No esquema abaixo, cada OpCode aponta para um espaço de 4 palavras. No caso de mais microinstruções serem necessárias, pode-se incluir, nesse espaço, um desvio para o espaço entre os endereços 1000000 e 11111111, reservados às microinstruções excedentes. Nesse espaço poderiam estar alocados também os microprogramas fundamentais, como o ciclo de busca, por exemplo.

		OPCODE	
Endereço Micromemória	0	0011	00

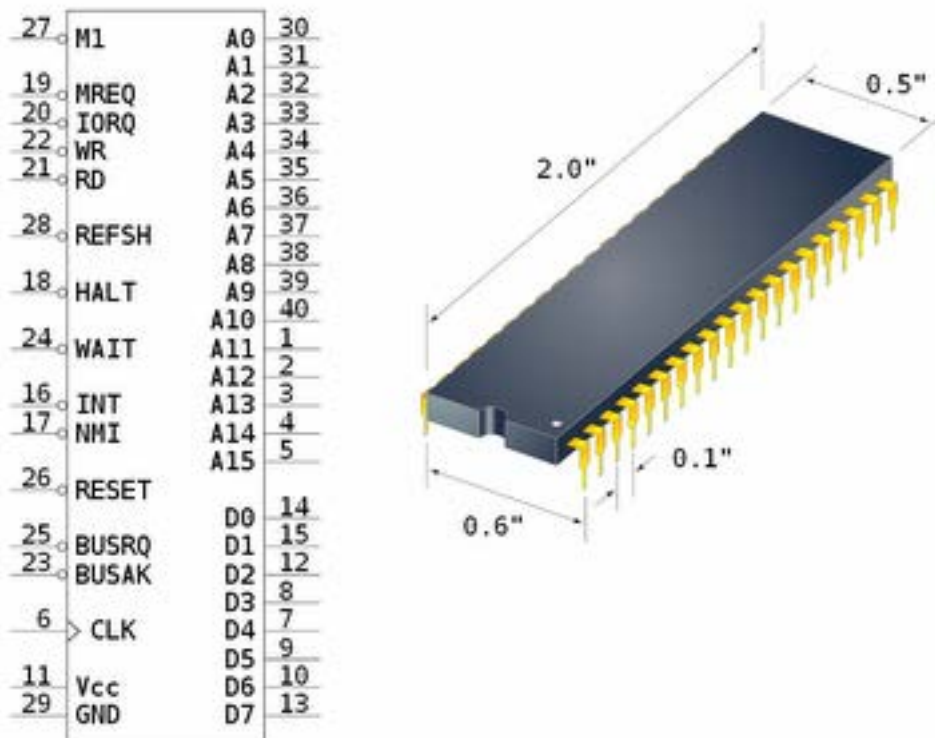


Figura 2: Pinagem do microprocessador Z80

Conforme apresenta a figura acima, os terminais MREQ, IOREQ, WR e RD pertencem a funções da unidade de controle, sendo que o MREQ tem a função de enviar um sinal para habilitar a memória, o IOREQ envia um sinal para habilitar um dispositivo de entrada e saída, notificando-o de que está pronto para enviar ou receber uma informação, e o terminal WR, quando acionado, habilita a memória ou o dispositivo de E/S para escrita e o RD habilita a memória ou o dispositivo de E/S para escrita para leitura. Com esses terminais, é possível ler e escrever, na memória ou nos dispositivos de entrada e saída, da seguinte forma: ao habilitar o terminal MREQ e o WR juntos, a unidade de controle está indicando que será feita, na memória, uma escrita, ou seja, alguma informação será enviada para a memória. Caso os terminais IOREQ e RD sejam acionados juntos então a unidade de controle está indicando que será efetuada uma leitura de dados de um dispositivo de entrada ou saída, ou seja, o processador irá receber uma informação de um dispositivo de Entrada e Saída.

Conforme já descrito até agora, podemos chegar à conclusão de que o computador é composto por diversos componentes, dentre os quais o processador é o mais importante, pois é ele que recebe instruções para controlar grande parte dos componentes. Abaixo você irá estudar como é feito o sincronismo entre esses componentes e como a velocidade de processamento pode ser afetada. Você irá estudar também algumas instruções em assembly. Essa linguagem é usada com instruções que ficam bem próximas do hardware, portanto ela permite que o processamento seja feito mais rápido que uma linguagem de alto nível como Vb.NET, C++, Java, Python e outras.

## 2. O relógio (clock)



Quando decidimos comprar ou trocar um computador, a primeira coisa em que pensamos é o desempenho da máquina. Mas será que quanto mais veloz, melhor? Nem sempre. São muitas variáveis que afetam o desempenho do computador. Ao analisar os componentes de um computador, deparamo-nos com diversos termos técnicos, como é o caso do Clock. Abaixo você terá uma descrição sobre esse componente e qual sua função no computador.

Pelos estudos que já realizamos, chegamos à conclusão de que tudo gira em torno do processador, considerado o “cérebro” do sistema computacional. Esse componente, também chamado de microprocessador ou CPU (Central Processing Unit), é o responsável por processar os dados e as instruções. A maneira como essa operação será processada dependerá do programa, que pode ser desde um editor de textos até mesmo um jogo. Para o processador, o tipo de aplicativo a ser executado não faz a menor diferença, pois ele apenas obedece às ordens (chamadas de comandos ou instruções) contidas nos softwares, as quais serão convertidas em uma ou mais instruções de que o processador dispõe. Portanto qualquer instrução recebida de um programa de alto nível será interpretada para uma instrução que o processador possa entender. Essas instruções podem ser desde um simples cálculo matemático chegando até a dados complexos para a placa de vídeo, por exemplo.

### 2.1 Mais o que é o Clock?

Em um sistema computacional, para que os resultados das execuções das atividades sejam perfeitas, todas elas irão necessitar de um sincronismo entre as comunicações dos componentes. O relógio, também conhecido como clock, serve justamente para isso; ele atua como um sinal digital para sincronizar essas atividades. Ele é gerado por um componente chamado cristal de quartzo, que vibra alguns milhões de vezes por segundo, com uma precisão quase absoluta, quando nele é inserida uma voltagem.

Esses sinais gerados pelas vibrações do cristal são inseridos em um chip que irá digitalizá-los para sincronizar os ciclos dos componentes da placa-mãe, que sabe que, a cada sinal digital gerado no cristal, deve gerar um determinado número de ciclos de processamento das informações. O funcionamento de todos os periféricos, da placa de vídeo ao disco rígido, é coordenado por esse “relógio”, que faz com que eles trabalhem de forma sincronizada.



Figura 3: Cristal de quartzo

Portanto, a cada pulso de clock, os dispositivos executam suas atividades, param e vão para o próximo ciclo de forma sincronizada. Tecnicamente falando, o pulso gerado pelo cristal, após passar por um chip, é uma onda quadrada passando de “0” a “1” a uma taxa fixa, em que o início de cada ciclo é quando o sinal passa de “0” a “1”. A unidade de medida usada para determinar a velocidade de vibração do quartzo é dada em Hertz (Hz), que significa o número de pulsos que o quartzo vibra por segundo. Portanto, um processador de 100 MHz indica que, em um segundo, há 100 milhões de pulsos de clock.

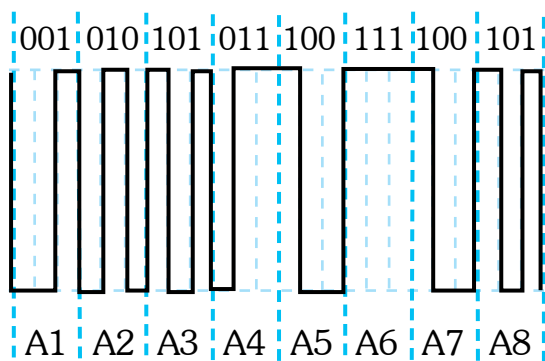


Figura 4: Sinal digital

Dentro do processador, todas as instruções precisam de um determinado número de ciclos para serem executadas. O processador conhece quantos pulsos cada instrução vai requerer, pois ele tem uma tabela em que estão registrados todos os tipos de instruções suportados por ele. Portanto, se há duas instruções para execução e a primeira vai levar sete ciclos de clock, a segunda iniciará, automaticamente, no oitavo pulso de clock.

É lógico que essa é uma explicação para fins didáticos, tomando como exemplo apenas uma unidade de execução sobre o funcionamento do processador, pois os processadores atuais possuem mais que um núcleo trabalhando em paralelo, capazes de executar várias tarefas em paralelo e muitas instruções necessitam de mais que um ciclo de clock para que sejam totalmente executadas.



É importante ressaltar que cada processador lançado no mercado tem um projeto diferente e conta com características que determinam seu desempenho.

**No caso de dois chips completamente idênticos**, o que estiver rodando a uma taxa de clock mais alta, teoricamente, será o mais rápido. Nesse caso, com uma taxa mais alta, o tempo entre cada ciclo será menor, e as tarefas serão desempenhadas em menos tempo, resultando em uma execução maior de informações por segundo.

## 2.2 Multiplicação de Clock

Podemos notar que as velocidades de clock dos computadores estão aumentando, mas essa evolução da velocidade passou por algumas restrições. Como os processadores atuais atingem frequências mais elevadas, muito superiores às frequências das placas-mãe, os fabricantes de chips começaram a usar um novo conceito chamado multiplicação de clock. O objetivo desta técnica é evitar que os processadores fiquem limitados à frequência da placa-mãe e, para que isso fosse possível, foram desenvolvidos e implementados os conceitos de clock interno e o clock externo.

Dentro de qualquer computador, os dados são transmitidos e gerenciados na forma de sinais digitais. Para que você tenha uma ideia, o processador é pequeno, mede em torno de 1,5 centímetros quadrados. A placa-mãe, por sua vez, é muito maior que isso, com várias trilhas (ou caminhos). Essas trilhas são fios, geralmente colocados em paralelo para conectarem vários componentes do computador. O problema é que, com taxas de clock muito altas, esses fios começam a funcionar como antenas interferindo um ao outro.

Como a velocidade está restrita à placa-mãe, pelos motivos descritos acima, a frequência do processador não é fixada. Ela pode ser maior ou menor do que o especificado pelos fabricantes; isto depende de como a placa-mãe está configurada e é, neste ponto, que algumas pessoas, com conhecimentos mais avançados, costumam alterar as velocidades de vibração do quartzo, implementando o que chamamos de overlocks, o que faz com que o processador trabalhe com taxas superiores às recomendadas pelos fabricantes.

Desde o desenvolvimento dos computadores 486 até os dias atuais, os processadores usam o clock externo para transferir dados para a memória RAM e dispositivos de entrada e saída. É usado um chip controlador na placa-mãe, chamado de “ponte norte”, que tem a finalidade de adequar a velocidade entre diferentes barramentos da placa-mãe. Por exemplo, em um Pentium 4 de 3 GHz, a velocidade de 3 GHz refere-se ao clock interno do processador, que é obtido quando multiplicamos por 15 seu clock externo de 200 MHz.

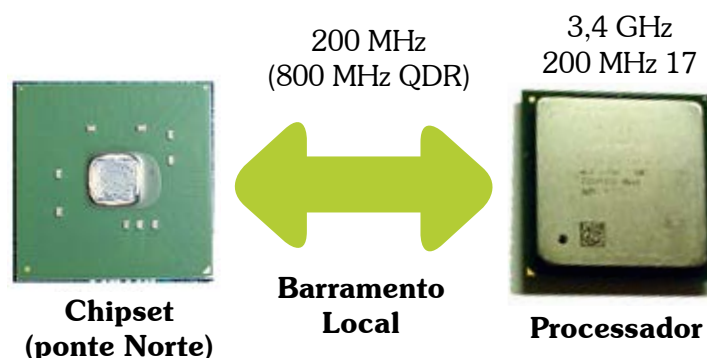


Figura 5: Chipset controlador de velocidade

Por terem valores de velocidade muito distintos, diversas técnicas são usadas para minimizar o impacto da diferença de clock interno e externo. Uma delas é o uso de quantidade maior de memória cache dentro do processador. Outra técnica, considerada mais comum, é fazer a transferência de mais de uma informação por pulso de clock. Tanto os processadores AMD como os processadores da Intel, que são os fabricantes mais conhecidos de chips, usam esse tipo de recurso. Os processadores da AMD transferem dois dados por ciclo de clock e os da Intel, quatro dados por ciclo de clock.

Com a transferência de dois dados por ciclo, considerando um clock externo de 100 MHz, podemos concluir que, na verdade, o processador está trabalhando a uma velocidade de 200 MHz, ou seja, o dobro da velocidade. Como podemos ver, não é somente o clock que influencia o desempenho do computador.



## Pense

Atualmente, o recurso de multiplicação de clock é indispensável, pois sem ele seria impossível desenvolver processadores muito rápidos, já que não é possível aumentar a frequência das placas-mãe e dos demais periféricos na mesma proporção do aumento do clock dos processadores.

### 3. Registrador de Instruções



Mesmo os microprocessadores mais simples e para uso específico têm um conjunto de instruções que determinará como serão tratadas as informações passadas a eles. As instruções são executadas de acordo com bits padrões e circuitos combinacionais determinados para cada microprocessador; cada instrução tem um significado diferente e, antes que sejam decifradas, elas são armazenadas em um registrador chamado registrador de instrução (RI). Nós, seres humanos, não somos rápidos o suficiente para compreender com clareza uma instrução em forma de bits, portanto, ao longo do tempo, para facilitar o entendimento de um programa em bits, foi desenvolvida a linguagem Assembly. Há alguns programas que convertem a linguagem assembly em 0 e 1's correspondente às instruções de um microprocessador. Abaixo encontram-se algumas instruções em Assembly.

Assembly	Descrição
LOADA mem	Carrega para o registrador A o endereço de memória
LOADB mem	Carrega para o registrador B o endereço de memória.
CONB con	Coloca um valor constante no registrador B
SAVEB Mem	Salva registrador B do endereço de memória
SAVEC Mem	Salva registrador de endereço de memória C
ADD	Adiciona A e B e armazena o resultado em C
SUB	Subtrai A e B e armazena o resultado em C
MUL	Multiplica A e B e armazena o resultado em C
DIV	Divide A e B e armazena o resultado em C
COM	Compara A e B e armazena o resultado no FLAG
JUMP addr	Vai para um endereço
JEQ addr	Salta, se há igualdade, para o endereço
JNEQ addr	Salta, se não for igual, para o endereço
JG addr	Salta, se for maior que, para o endereço
JGE addr	Salta, se for igual ou superior, para o endereço
JL addr	Salta, se for menor, para o endereço
JLE addr	Salta, se for igual ou inferior, para o endereço
STOP	Para a execução



## 4. Contador e decodificador de Instruções.



As instruções a serem executadas por um microprocessador estão em forma de códigos de instruções. Essas instruções devem ser processadas de forma correta e na sequência lógica correta e, para que isso seja possível, deve haver um controle preciso sobre essas instruções. A sequência lógica é feita de acordo com o que foi estabelecido nos programas desenvolvidos por nós, sejam eles em qualquer linguagem de programação. A unidade de controle é a responsável pelo controle da sequência de eventos necessários para a execução das instruções, ou seja, de acordo com a definição feita na programação, esta irá auxiliar a execução das instruções. Cada instrução é executada de acordo com o clock, conforme já explanado acima.

O contador de programas indica onde o próximo byte de instrução está localizado na memória. Uma instrução recebida da memória pode alterar o contador de programa para que outra posição da memória seja lida ou escrita. O sincronismo do contador de programas também está relacionado ao relógio (Clock) e pode indicar que uma ou mais instruções foram carregadas da memória para o microprocessador.

O microprocessador utiliza o registrador de instruções para armazenar as últimas instruções obtidas da memória. O primeiro byte de uma instrução é passado para o decodificador de instruções, que tem a função de interpretá-las para determinar qual operação deve ser efetuada com os dados.

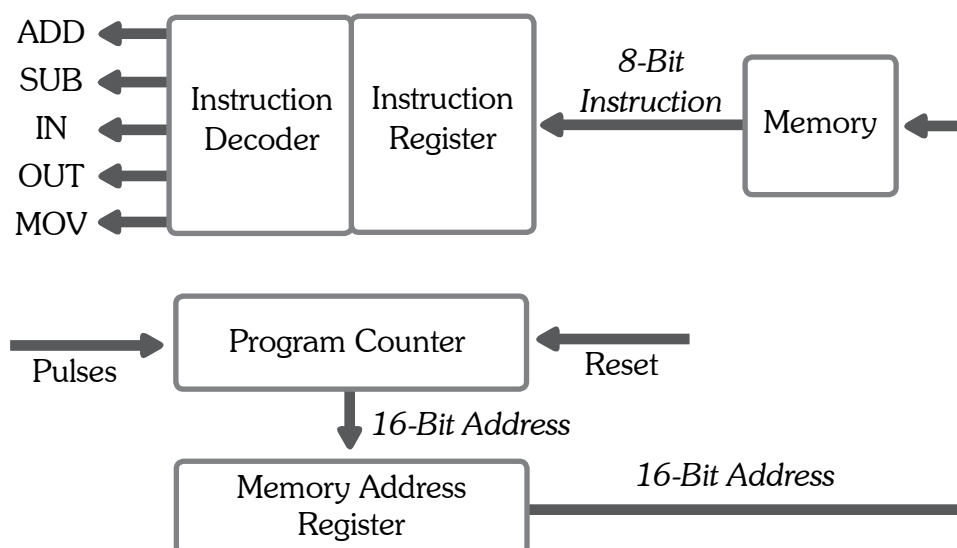


Figura 6: Diagrama do registrador de instruções

Qualquer linguagem de programação de alto nível tem seu programa traduzido para uma das instruções que o processador pode interpretar. Por esse motivo é que colocamos o interpretador e compiladores como influenciadores no desempenho da máquina, pois, se o programa estivesse desenvolvido em linguagem de máquina, não haveria a necessidade de interpretá-los. Como exemplo, entre as operações decodificadas, estão as que foram descritas acima.

A unidade de controle tem um papel fundamental na execução das instruções dos processadores; ela tem circuitos específicos para acionar a memória e os dispositivos de entrada e saída. A grande maioria dos códigos de operações (OPCODES) passa para ser decodificada por ela. De acordo com as instruções que chegam até a unidade de controle, uma ação é executada.



## Material Complementar

Para saber mais procure os livros abaixo

STALLINGS, W. **Arquitetura e Organizacao de Computadores**: Projeto Para o Desempenho. 5. ed. Sao Paulo: Prentice Hall, 2004. (Biblioteca Digital)

TANENBAUM, A. S. **Organização Estruturada de Computadores**. 5 ed. São Paulo: Pearson Prentice Hall, 2007. (Biblioteca Digital)

## Referências

STALLINGS, W. Arquitetura e Organização de Computadores: Projeto Para o Desempenho. 5. ed. Sao Paulo: Prentice Hall, 2004.

B. Marshall. Como Funcionam os Microprocessadores. Disponível em <http://computer.howstuffworks.com/microprocessor4.htm> Acessado em 04.08.2012.

## Anotações







**Educação a Distância**  
Cruzeiro do Sul Educacional  
*Campus Virtual*

[www.cruzeirodosulvirtual.com.br](http://www.cruzeirodosulvirtual.com.br)  
Campus Liberdade  
Rua Galvão Bueno, 868  
CEP 01506-000  
São Paulo SP Brasil  
Tel: (55 11) 3385-3000

