

# 1 Microprocessador 8086/8088

## 1.1 História

Em 1968 a empresa Intel foi fundada por Robert N. Noyce, Gordon E. Moore e Andrew Grove. Robert N. Noyce foi o inventor do circuito integrado.

Em 15 de novembro de 1971 nascia o processador 4004 de apenas 4 bits e grande capacidade para realizar operações aritméticas. Esse microprocessador possuía 2.300 transistores para processar 0,06 milhões de instruções (60.000) por segundo e não tinha o tamanho de um selo de carta. Para se ter uma idéia, o ENIAC, primeiro computador de que se tem notícia, construído em 1946 para fins bélicos, ocupava sozinho 1.000 metros quadrados e fazia o mesmo que o 4004.

O 4004 foi usado apenas para cálculos poucos complexos (4 operações), ele era um pouco mais lento que Eniac II mais tinha a vantagem de possuir a metade do tamanho, esquentar menos e consumir menos energia.

Surgiu em 1972 o 8008, primeiro processador de 8 bits, com capacidade de memória de 16 Kbytes (16.000 bytes), enquanto o 4004 possuía apenas 640 bytes.

Em 1974 é lançado o 8080, com desempenho seis vezes maior que o anterior com um clock de 2MHz, rodava um programa da Microsoft chamado Basic, possuía apenas led's. Além de 16Kb de memória Rom onde ficava o sistema, possuía 4Kb de memória Ram, seus controles eram através de botões, possuía drive de disquete 8" com capacidade de 250 Kb.

O 8086 foi o primeiro processador feito pela Intel para ser usado com os PC's. Ele contava com um barramento de dados interno e externo de 16 bits. E foi este o motivo de não ter sido o processador mais utilizado. Inicialmente ele foi distribuído em versões de 4,77MHZ. Posteriormente vieram versões turbinadas de 8 e 10 MHZ.

A história do 8086 é bem simples. Quando ele foi lançado, a maioria dos dispositivos e circuitos disponíveis eram de 8 bits. Era muito caro adaptar todo o resto do computador por causa do processador. E foi isso que acabou com o 8086. Para adaptar-se a este mercado a Intel lançou o 8088, com barramento externo mais lento, de 8 bits. Deixando a diferença de barramento externo, ambos eram idênticos.

Quando este chip, o 8086, veio a ser utilizado já era tarde demais. Ele chegou até a fazer parte de uns poucos clones do IBM PC e posteriormente em dois modelos do IBM PS/2 e de um computador Compaq. Mas sua destruição veio com um processador mais poderoso, o 80286.

Outro possível fator para a pouca aceitação deste processador pode ter sido a falta de unidades devido à demanda. Nunca havia chips suficientes

para produzir computadores em grande escala.

## 1.2 Visão preliminar

Tanto o 8086 como o 8088 utilizam o conceito de fila de instruções para melhorar a velocidade do computador. Uma área no interior da pastilha denominada fila de instruções retém diversos bytes de uma instrução. Quando o computador estiver pronto para a próxima instrução, ele não precisa pegar muitos bytes na memória, uma vez que toda instrução poderá já se encontrar na fila. O conceito de fila aumenta o numero de operações realizadas por segundo uma vez que o processador vai estar utilizando o bus de dados e endereços por menor período de tempo, disponibilizando este para outros dispositivos. A fila do 8086 tem 6 bytes de largura e a do 8088 tem 4 bytes.

O 8086 pode acessar 1 megabyte de memória de leitura escrita (220 bytes). Entretanto ele utiliza um esquema de endereçamento de memória denominado segmentação, em que determinados registradores de segmento fornecem um endereço básico que é automaticamente acrescentado a cada endereço a cada endereço de usuário de 16 bits na máquina.

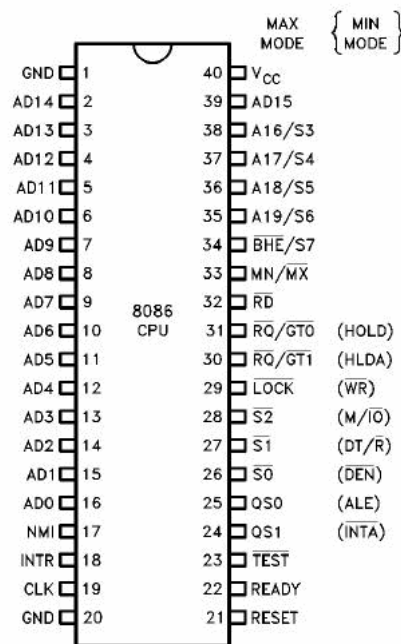


Figura 1: Pinagem do IA-PX 86

A parte do endereço e todas as vias de dados são multiplexados em 16 pinos (os 16 pinos do barramento de dados é o que o classifica como um

microprocessador de 16 bits). Os restantes 4 bits de endereço são implementados por quatro pinos adicionais de endereço, que também são utilizados para status (como mostra a figura 1). É requerido um clock externo à pastilha e é utilizado um controlador de via externo à pastilha para demultiplexar a via de dados e de endereço.

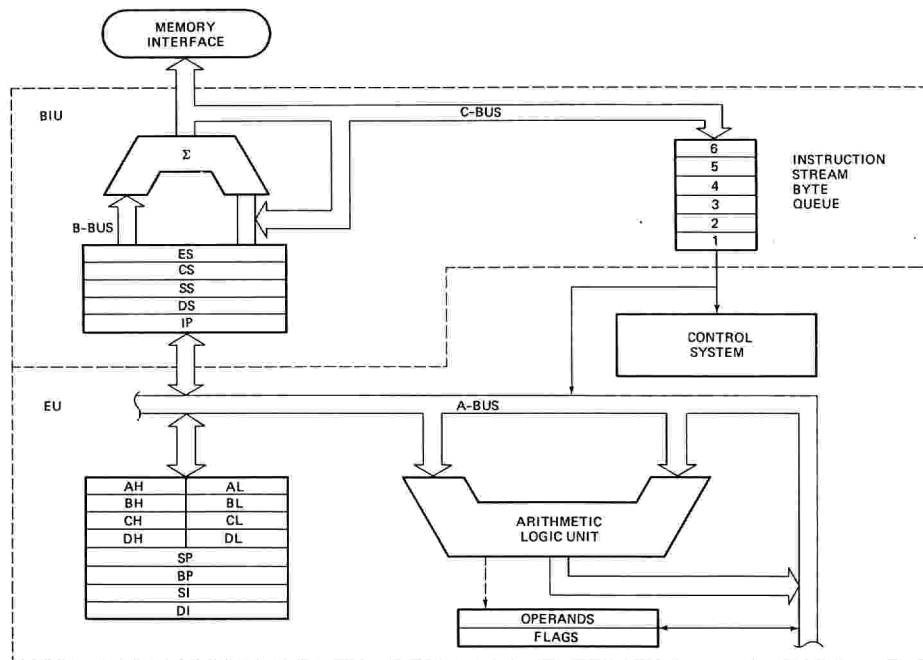


Figura 2: Arquitetura do 8086/8088

O 8086 tem uma estrutura de interrupção poderosa. Quase todos os microprocessadores de 8 bits requerem pastilhas externas adicionais para permitir operações de interrupção adequadas. No 8086, cerca de 1000 bytes são colocados de lado para conter até 265 apontadores de vetores (lembrando que cada apontador é um endereço contendo offset:seletor, ou seja, 4 bytes). O 8086 executa operações de E/S (ou I/O) em um espaço separado da memória denominado espaço de E/S. Tem um total de 64k bytes. Para ser utilizado pelos co-processadores é fornecido um pino de entrada TEST especial (pino 23, figura 1) para permitir ao 8086 saber quando é que o co-processador completou a tarefa. Quando uma instrução WAIT é acionada, o 8086 pára e aguarda do co-processador externo, ou qualquer outro hardware, um sinal para que continue pela alteração do pino TEST.

### 1.3 Memória

A largura de memória "vista" por um microprocessador é ditada pela quantidade de bits que o microprocessador pode acessar por vez. Esta quantidade é determinada pela largura do seu barramento de dados externo. O microprocessador 8086 possui um barramento de dados de 16 bits o que permite acessar dois bytes da memória consecutivamente (cada referência à memória acessa 2 bytes), enquanto, o 8088 possui um barramento de dados de 8 bits, ou seja, uma referência à memória acessa 1 byte.

Os microprocessadores 8086/8088 podem acessar (ler ou escrever) bytes ou palavras (2 bytes) que se localizam tanto em endereços pares como em endereços ímpares da memória. No entanto, dependendo do microprocessador e do tamanho do dado a ser acessado, pode ser necessária que o microprocessador efetue uma ou duas referências para a memória. A tabela 1 resume o número de referências necessárias para os microprocessadores 8086/8088 acessarem dados de 8 bits e 16 bits em endereços pares e ímpares da memória.

Tamanho do dado	Endereço	Numero de referências	
		8086	8088
Byte	Par	1	1
	Ímpar	1	1
Palavra	Par	1	2
	Ímpar	2	2

O espaço de endereçamento da memória do microprocessador 8088 é organizado como um vetor linear de 1 MByte, sendo que cada localização deste espaço é referenciada por meio de um endereço único de 20 bits chamado endereço físico.

No microprocessador 8086 o espaço de endereçamento da memória é organizado em dois bancos de 512KBytes cada, chamados bancos par e banco ímpar, respectivamente. O banco par é conectado ao 8086 por meio de 8 bits menos significativos do barramento de dados, já o banco ímpar é conectado por meio dos 8 bits mais significativos do barramento de dados.

### 1.4 Arquitetura do microprocessador

A unidade de execução (EU) executa as operações aritméticas e lógicas, além de controlar a maioria dos registros internos e manipular os dados. Em contraste, a unidade de interface de barramento (BIU representado por um símbolo de somatória na figura 1) executa as operações de barramento, incluindo a transferência de dados, e controla os registros restantes do mi-

coprocessador.

As duas unidades de processamento são capazes de executar suas operações de forma independente, isto é, cada unidade pode fazer suas tarefas sem a assistência da outra unidade.

Embora a unidade de execução EU esteja isolada do barramento do sistema, pela unidade de interface de barramento (BIU), ela ainda pode acessar o barramento para certas operações. A EU ganha o acesso do barramento requisitando que a BIU temporariamente suspenda suas atividades. A EU então assume o controle da BIU de modo a poder usar o barramento para enviar ou receber informações.

Da figura 1 nota-se que a EU consiste de duas seções principais que são os registros de propósitos gerais e a unidade aritmética e lógica - ALU. Os registros de propósitos gerais do 8086/8088 presentes na EU são áreas de armazenamento com capacidade de manter dados binários que foram ou serão usados pelo microprocessador. A grande vantagem destes registros se deve ao fato de se poder acessá-los com maior facilidade e de forma muito mais rápida do que uma determinada localização da memória.

Todos registros de propósitos gerais possuem capacidades aritméticas e lógicas e dados podem ser armazenados através da BIU ou transferidos para memória. Os registros de propósitos gerais são todos de 16 bits, entretanto os bytes menos e mais significativos podem ser usados separadamente como registros de 1 byte, dessa forma tem-se os seguintes registros AH, AL, BH, BL, CH, CL, DH e DL.

A maioria das operações que se pode executar em um dos registros de propósitos gerais podem também serem executadas nos demais registros. Contudo, os registros de propósitos gerais tem uso específico para algumas poucas instruções, ou seja, algumas instruções utilizam de forma específica determinados registros. Devido a este fato os registros de propósitos gerais recebem nomes descritivos que são: Acumulador (AX), Base (BX), Contador (CX), Dado (DX), índice fonte (SI), índice destino (DI), ponteiro base (BP) e ponteiro da pilha (SP).

A unidade aritmética lógica, ALU, recebe instruções e então executa sobre os dados especificados pela instrução uma operação aritmética, como soma ou subtração ou lógica como OR ou AND.

A seção de lógica de controle de barramento é responsável por todas as operações de barramento do microprocessador como, por exemplo, a busca de dados para a unidade aritmética lógica (ALU). Quando necessário a seção de lógica de controle de barramento acessa localizações particulares na memória, de modo que a EU possa enviar ou receber informações para ou destas localizações.

A seção de lógica de controle de barramento também controla o sentido

do fluxo de informação no barramento. Quando uma informação tiver que ser enviada à memória, esta seção assegura que os sinais de controle sejam os apropriados para a transmissão. O mesmo ocorre quando for necessário receber uma informação. A fila de instruções age como um "encanamento" onde os bytes das instruções trazidos da memória são armazenados antes do seu uso pela EU. No 8086 esta fila é composta de 6 localizações de 8 bits cada, enquanto no 8088 a fila de instruções é composta de 4 localizações de 8 bits. Pode-se dizer que estas localizações servem como áreas para o armazenamento temporário (buffer) das instruções trazidas da memória.

No microprocessador 8086/8088 é a seção lógica de controle de barramento BIU, que busca os bytes das instruções do programa na memória e os coloca na fila de instruções. Esta fila mantém estes bytes até que a EU esteja pronta para aceitá-las.

Devido as características do microprocessador 8086, a BIU sempre busca as instruções acessando palavras (16 bits) que se encontram armazenadas em endereços pares. A única exceção ocorre quando existe um desvio (JUMP) para uma instrução que se encontra armazenada na memória em um endereço ímpar. Quando isso ocorre o 8086 traz para a fila de instruções um único byte da instrução e a seguir continua acessando palavras que se encontram armazenadas em endereços pares. Este fato não ocorre em um 8088 visto que seu barramento de dados é de 8 bits. É importante salientar que as instruções de um microprocessador 8086 poder ter de um a seis bytes de comprimento.

Independente do microprocessador, caso ocorra um desvio na seqüência de execução das instruções, a fila de instruções é automaticamente esvaziada e a BIU passa a buscar as instruções a partir da nova localização de memória para o qual se deu o desvio.

## 1.5 Endereçamento da memória

Ao contrário dos microprocessadores que utilizam um modelo de memória lineal, ou seja, que enxergam o seu espaço de endereçamento de memória de forma lineal, o 8086/8088 utiliza um modelo de memória denominado segmentada. Neste modelo o microprocessador enxerga o espaço de endereçamento de memória dividido em vários segmentos.

Um segmento nada mais é do que uma região continua do espaço de endereçamento de memória que é tratada pelo microprocessador como uma unidade lógica. Por serem unidades lógicas e não físicas, os segmentos podem localizar-se em qualquer parte do espaço de endereçamento linear. Conseqüentemente, dois ou mais segmentos distintos poder ser: adjacentes, parcialmente sobrepostos, totalmente sobrepostos ou desconexos.

No modelo de memória segmenta do 8086/8088, o microprocessador somente pode acessar as localizações de seu espaço de endereçamento por meio de um determinado segmento. Assim para este microprocessador, um segmento funciona como uma "janela móvel" sobre o seu espaço de endereçamento lineal, através do qual ele acessa as localizações do seu espaço de endereçamento.

Para o microprocessador 8086/8088, os segmentos podem se localizar em qualquer parte do seu espaço de endereçamento de memória. Entretanto, devido a arquitetura deste microprocessador, um seguimento somente pode começar em uma localização do espaço de endereçamento de memória, cujo endereço físico seja múltiplo de 16 (10H).

O endereço físico do início de um segmento do 8086/8088 é designado por endereço base, os 16 bits mais significativos do endereço base corresponde a um endereço chamado endereço de segmento ou seletor. Conseqüentemente, cada segmento do 8086/8088 é identificado por um endereço de segmento ou seletor de 16 bits.

Dentro de cada segmento o endereçamento se dá de forma linear, porém relativo ao início do endereço do segmento. Cada localização do espaço de endereçamento de memória dentro do segmento é identificado por meio de um endereço de 16 bits chamado endereço efetivo (Effective Address - EA) ou endereço de offset (offset).

Devido a segmentação do espaço de endereçamento de memória e ao endereçamento relativo dentro do segmento, cada localização do espaço de endereçamento de memória do microprocessador é identificado por meio de um endereço de 32bits chamado de endereço lógico (seletor : offset).

A forma como o microprocessador converte um endereço lógico de 32 bits em um endereço físico de 20 bits, faz com que vários endereços lógicos identifiquem uma mesma localização do espaço de endereçamento de memória. A conversão de endereço lógico em endereço físico se dá multiplicando o seletor por 10h e em seguida somando o offset. O endereço físico pode ser representado na forma normalizada para obter-se o endereço lógico, os 4 bits menos significativos do endereço físico correspondem ao endereço de offset e os 16 bits mais significativos do endereço físico correspondem ao endereço de segmento.

## 1.6 Conjunto de registros

Embora os registros SI, DI, BP e SP, possuam capacidade aritmética e lógica de 16bits, como os demais registros de propósitos gerais de 16 bits, estes registros geralmente são usados para manter o endereço efetivo (offset) de localizações de memória ou para apontar estruturas de dados na memória.

Particularmente, o registro SP é utilizado para manter o endereço efetivo

do topo de uma estrutura de dados na memória que funciona como uma pilha (stack), onde o último dado a ser armazenado nesta estrutura deverá ser o primeiro a ser retirado (LIFO - Last Input/first output). Uma vez que esta estrutura é de vital importância para o funcionamento do microprocessador, a utilização do registro SP em operações aritméticas ou lógicas não é aconselhada.

Como o microprocessador 8086/8088 utiliza um modelo de memória segmentada, o acesso as localizações do seu espaço de endereçamento de memória é feito através de segmentos mediante endereços lógicos de 32bits. Por isso, para poder acessar as localizações dentro de um determinado segmento é necessário que o 8086/8088 conheça o endereço deste segmento, ou seja, o seu seletor. Para isso, o microprocessador 8086/8088 dispõe de um conjunto de registros especiais chamados registro de segmentos, cuja finalidade, como o próprio indica, é matar endereços de segmentos.

Para acessar uma localização do espaço de endereçamento de memória que não é abrangido por um dos segmentos apontados pelos registros de segmentos, é necessário alterar o conteúdo de um dos registros de segmento, de modo que este registro aponte para um segmento que venha a abranger a localização que se deseja acessar.

Um programa, geralmente, é composto por três partes ou segmentos que são: o segmento de códigos, o segmento de dados e o segmento de pilha. As partes ou segmentos de um programa podem residir em qualquer ordem e em qualquer lugar do espaço de endereçamento de memória que tenha memória física.

Parte do programa	Registro de segmento
Código do programa	CS
Dados do programa	DS
Pilha do programa	SS
Área extra da memória	ES

O ponteiro de instruções (IP) é um registro de 16 bits, que sempre matam o endereço efetivo da localização de memória onde está armazenado o código de máquina da próxima instrução a ser executada. Este registro é automaticamente incrementado pelo microprocessador de acordo com o tamanho desta instrução.

A representação de um endereço lógico se dá na forma seletor:offset. Quando os conteúdos de dois registros são usados para especificar um endereço lógico, o endereço lógico geralmente é escrito na forma RS:RO onde RS corresponde ao nome do registro que mantém a parte do endereço lógico



que se refere ao endereço de segmento e RO corresponde ao nome do registro que mantém a parte do endereço lógico que se refere ao endereço efetivo (offset). No caso os registros que podem ser utilizados para apontar localização dentro do segmento de dados são os registros BX, SI e DI.

No segmento de stack o ponteiro de stack (SP) mantém o endereço efetivo da localização de memória que corresponde ao topo da pilha. O endereço de segmento é mantido no registro de segmento SS.

O poder real de um microprocessador está na sua capacidade de tomar decisões. O 8086/8088 baseia as suas decisões no conteúdo de um registro de 16 bits chamado registro de flags. Este registro é automaticamente atualizado para manter informações a respeito da ultima operação aritmética ou lógica que o microprocessador executou. Apenas 9 flags do registro de flags são utilizados, são eles: OF - overflow, DF - direção, IF - interrupção, TF - armadilha, SF- sinal, ZF - zero, AF - carry auxiliar, PF- paridade e CD - carry.

## 1.7 Instruções

Cada instrução (cartão de referencia do microprocessador 8086/8088 em anexo) possui uma representação binária única que é conhecida por código de máquina. Este modelo binário ou código de máquina da instrução, quando for aplicado aos circuitos internos do microprocessador faz com que ele execute uma operação particular. Uma instrução de um modo geral pode ser dividida em duas partes: a do código de operação (opcode) e a dos seus operandos.

O código de operação (opcode) é a parte da instrução que identifica a operação básica a ser executada pelo microprocessador. Enquanto, os operandos identificam os dados que devem ser utilizados pelo microprocessador na operação identificada pelo código de operação.

Dependendo da instrução ela pode ter 2 operandos, 1 operando ou nenhum operando. Na representação das instruções com dois operandos, o operando destino é sempre especificado em primeiro, e este é separado do operando fonte por uma vírgula. Quando um operando se refere a um registro de 8 ou 16 bits ele é chamado de operando registro, e quando ele refere a uma localização de memória ele é chamado operando memória. Por outro lado, um operando imediato se refere a um dado de 8 ou 16 bits que é especificado na própria instrução, ou seja, é especificado no código de máquina da própria instrução.

Um operando pode se encontrar em um registro (operando registro), numa localização de memória (operando memória), num dispositivo periférico de entrada/saída, ou até mesmo estar codificado no próprio código de máquina da instrução (operando imediato). A maneira na qual a localização de um

operando é especificada chama-se modo de endereçamento. O 8086/8088 utiliza duas categorias de endereçamento geral que são, o modo de endereçamento registro ou modo registro e o modo de endereçamento memória ou modo memória. Pode-se dizer que uma instrução do 8086/8088 utiliza o modo de endereçamento registro quando nenhum dos operandos da instrução se refere a memória. Por outro lado, uma instrução utiliza o modo de endereçamento memória quando um dos seus operandos da instrução se refere a um operando memória.

Quando um operando se refere a um operando memória até 3 valores de 16 bits podem ser somados para especificar o endereço efetivo (offset) do operando memória. Nesta soma qualquer carry que venha a ocorrer é ignorado pelo microprocessador, por um endereço efetivo (offset) no 8086/8088 é representado por um número de 16 bits.

Sobre as 2 categorias gerais de endereçamento existem 7 modos específicos de endereçamento.

#### **1.7.1 Endereçamento por registro**

Uma instrução utiliza o modo de endereçamento por registro quando o operando destino e o operando fonte desta instrução forem ambos operando registros.

Exemplo: MOV AX, BX.

#### **1.7.2 Endereçamento imediato**

Uma instrução utiliza o modo de endereçamento por registro quando o operando fonte desta instrução for imediato.

Exemplos: MOV CX, 1234h (modo registro); MOV [2011H], 1234H (modo memória).

#### **1.7.3 Endereçamento Direto**

Uma instrução utiliza o modo de endereçamento direto quando o operando destino ou operando fonte da instrução se refere a uma localização de memória, cujo endereço efetivo é especificado na própria instrução.

Exemplos: MOV CX, [1234H]; MOV [1234H], DX

#### **1.7.4 Endereçamento indireto por registro**

Uma instrução utiliza o modo de endereçamento indireto por registro quando o operando destino ou o operando fonte da instrução se refere a

um operando memória, cujo endereço efetivo se encontra armazenado num registro.

Exemplo: MOV CX, [BX];

### 1.7.5 Endereçamento por base

Uma instrução utiliza o modo de endereçamento por base quando o operando destino ou operando fonte da instrução se refere a um operando memória, cujo endereço efetivo (EA) é especificado pela soma do conteúdo do registro BX ou BP com um número de 8 ou 16 bits chamado deslocamento. No caso do deslocamento de 8 bits o microprocessador estende o sinal até se obter um número binário sinalizado de 16 bits, quando ocorrer um deslocamento de 16 bits o microprocessador interpreta como um número absoluto de 16 bits.

Quando o conteúdo do registro BP é utilizado no cálculo, o 8086/8088 automaticamente associa o endereço efetivo do operando memória com o conteúdo do registro de segmento de stack (registro SS), de modo a formar o endereço lógico do operando memória. Neste caso, portando, o 8086/8088 acesa o operando memória no segmento da pilha.

Exemplo: MOV AX, [BX + 1000H]

### 1.7.6 Endereçamento Indexado

Uma instrução utiliza o modo de endereçamento indexado quando o operando destino ou o operando fonte da instrução se refere a um operando memória, cujo endereço efetivo é especificado pela soma do conteúdo do registro SI ou DI, com um número binário de 8 ou 16 bits chamado deslocamento. No caso do deslocamento de 8 bits o microprocessador estende o sinal até se obter um número binário sinalizado de 16 bits, quando ocorrer um deslocamento de 16 bits o microprocessador interpreta como um número absoluto de 16 bits.

Exemplo: MOV AX, [SI + 2000H]

### 1.7.7 Endereçamento por base indexado

Uma instrução utiliza o modo de endereçamento por base indexado quando o operando destino ou o operando fonte da instrução se refere a um operando memória, cujo endereço efetivo é especificado pela soma do conteúdo do registro BX ou BP, com o conteúdo do registro SI ou DI e opcionalmente um número binário de 8 ou 16 bits chamado deslocamento.

Exemplo: MOV AX, [BX + SI + 2000H]