

# A máquina de Von Neumann

AOC: Arquitetura e Organização de Computadores

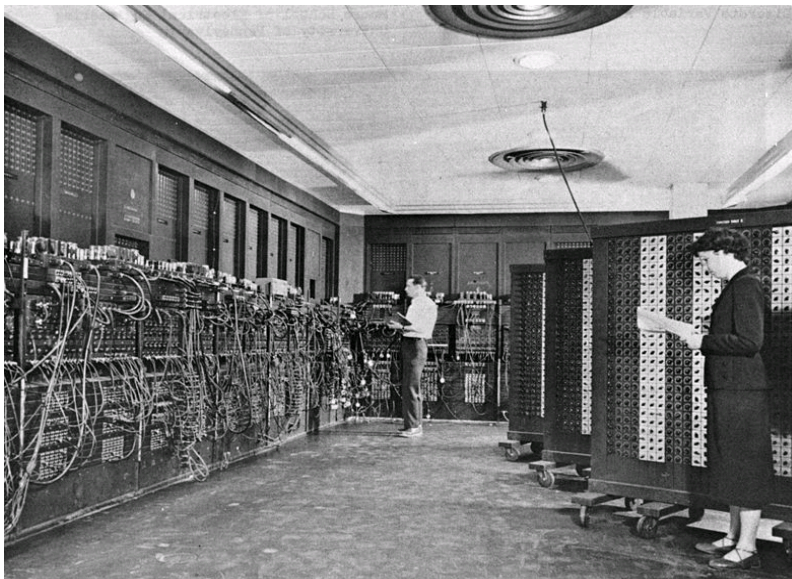
## Evolução dos computadores

- Caracterizado:
  - Aumento da velocidade dos processadores
  - Diminuição do tamanho dos componentes
  - Aumento da capacidade de memória
  - Aumento da capacidade e da velocidade de E/S
- Diminuição do tamanho dos componentes dos microprocessadores é um dos responsáveis pelo grande aumento da velocidade dos processadores
  - Redução da distância entre os componentes e aumento da velocidade

# Primeira Geração: válvulas eletrônicas

- ENIAC (Computador e Integrador Numérico Eletrônico)
  - Primeiro computador eletrônico digital de propósito geral.
  - Projetado, durante a 2ª guerra, para calcular trajetória e alcance de novas armas.
    - 200 pessoas trabalhando com calculadora de mesa por horas ou dias para resolver as equações de balísticas necessárias
    - ENIAC utilizava válvulas (1943). Pesava 30 toneladas, ocupava ~140 m<sup>2</sup>, e tinha + de 18 mil válvulas.
    - Ela era muito mais rápida q qualquer computador eletromecânico, sendo capaz de executar 5 mil adições / segundo
    - Era decimal e não binária. A memória consistia de 10 “acumuladores” cd acumulava um número digital de 10 dígitos
    - Cd dígito era representado por um anel de dez válvulas. A cada instante somente uma válvula ficava no estado ON, representando um dos 10 dígitos
    - Desvantagem: tinha q ser ligado / desligado e conectando e desconectando cabos.
    - Ficou pronto em 46
  - Com o término da guerra, este computador foi utilizado para determinar se a bomba H poderia ser construída. **Propósito diferente do qual foi projetado !**
    - Foi desativado em 55

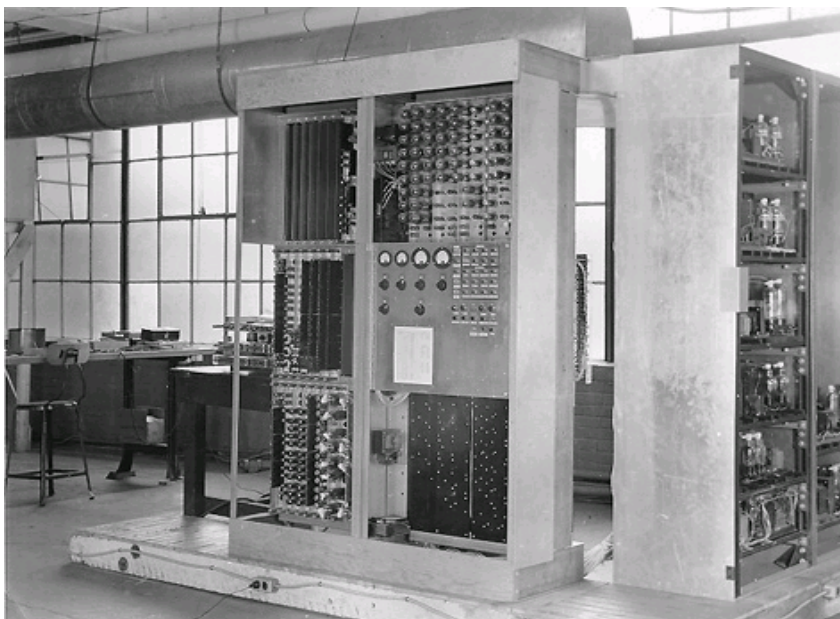
## ENIAC



# Máquina de von Neumann

- O processo de carregar / modificar um programa no ENIAC é complexo.
- O fato de um programa ser armazenado na memória facilitaria muito o processo.
  - Conceito do programa armazenado.
- O primeiro computador baseado neste conceito foi o EDVAC (Computador Variável Discreto Eletrônico). 45
- No entanto, o conceito geral de Computador que conhecemos hoje só foi projetado no IAS.

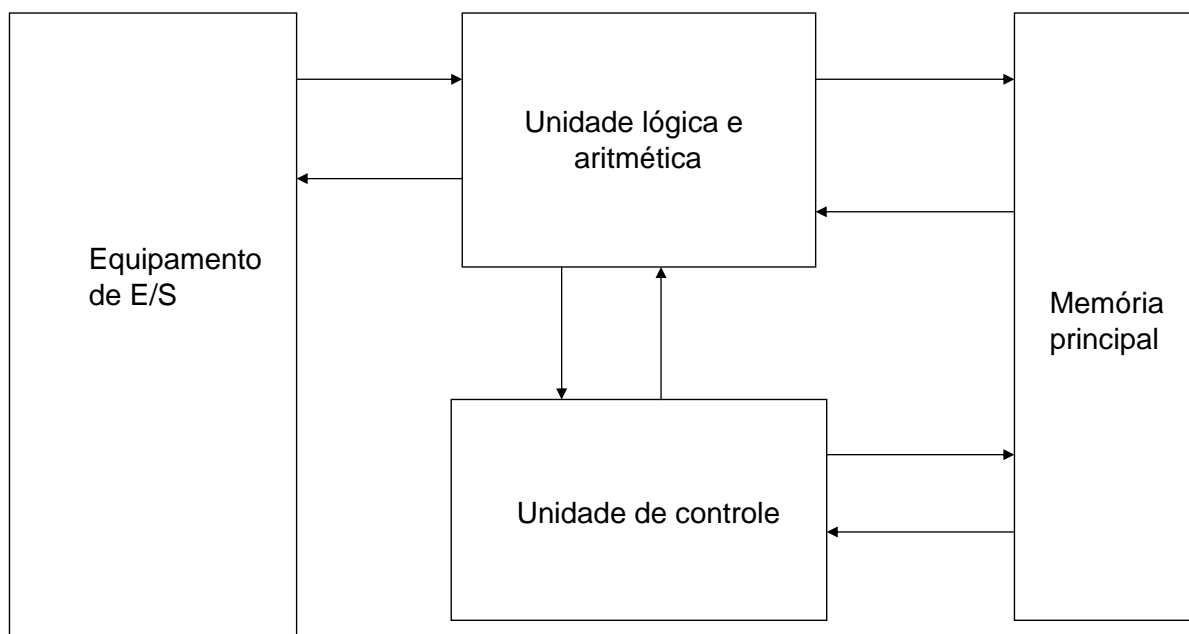
## EDVAC



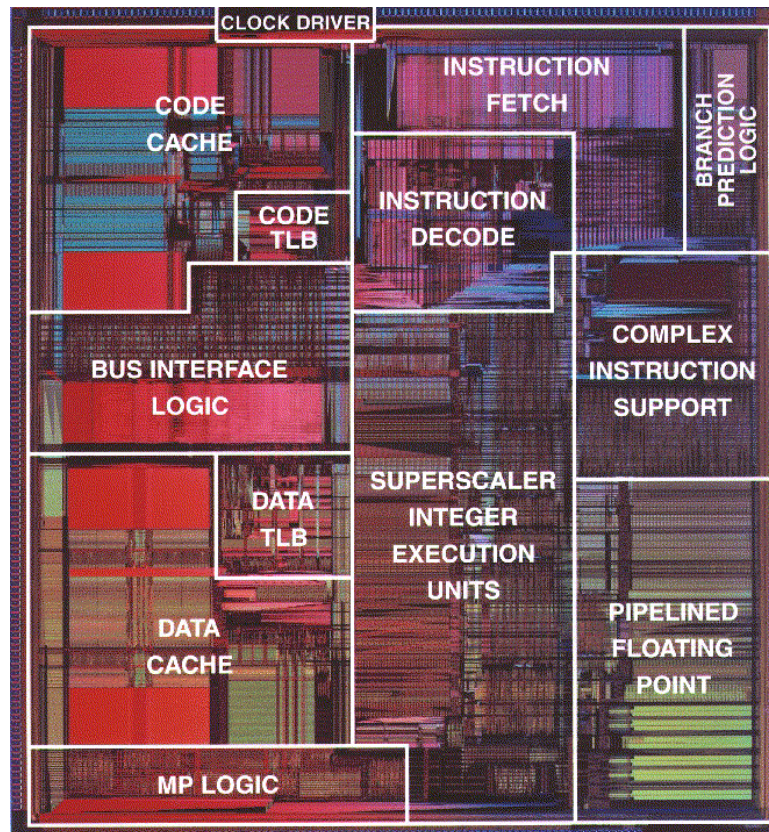
# Estrutura Geral do IAS

- Uma memória principal:
  - Armazena dados e instruções.
- Uma unidade lógica e aritmética (ULA):
  - Capaz de realizar operações com dados binários.
- Uma unidade de controle:
  - Interpreta e executa instruções armazenadas na memória.
- Dispositivos de entrada e saída:
  - Operados pela unidade de controle.

## Estrutura geral de von Neumann



## Um microprocessador (Pentium-PRO) - Complexidade



## Trabalho em grupo – dispositivos de E/S

- Monitores
- Placas de video Apresentação dia 15/6/09
- Impressoras
- teclado, mouses, scanners mesa, scanner móvel.
- leitoras óticas, leitoras de cartões magnéticos, câmeras de vídeo, microfones

Tópicos obrigatórios da apresentação:

- História (como surgiu e evoluiu)
- Funcionamento e tecnologia
- Modelos atuais e relevantes do passado.
- Modelos e preços atuais (comparativo dos benefícios)
- Quais problemas técnicos comuns ao dispositivo.
- Tendência da tecnologia (vai desaparecer? Absorvido?)

# Proposta de von neumann

- Primeiro princípio:
  - Supõem que um computador deva executar mais freqüentemente as operações elementares da aritmética.
  - Considera-se importante a existência de componentes especializados para realizar essas operações.
  - Portanto, deverá existir uma *unidade central de aritmética*.

# Proposta de von neumann

- Segundo princípio:
  - O controle lógico do dispositivo (execução das operações na seqüência ordenada das operações) será mais eficiente se feito por um componente de *controle central*.
  - Para um dispositivo de *propósito geral* será conveniente distinguir o conjunto de **instruções específicas** e os **componentes de controle geral** – estes se encarregam da execução destas instruções
  -

# Proposta de von neumann

- Terceiro princípio:
  - Qualquer dispositivo destinado à execução de longas e complicadas seqüências de operações deve ter uma **memória** considerável.
- Estes três primeiros componentes correspondem aos neurônios associativos do sistema nervoso humano.
  - Resta definir os componentes equivalentes aos neurônios *sensoriais e motores*.

# Proposta de von neumann

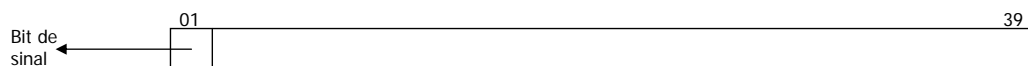
- Quarto princípio:
  - O dispositivo deve possuir elementos para transferir. Informações de A para seus componentes específicos C e M.
  - Constituindo a *entrada, a quarta parte específica do dispositivo: E.*

# Proposta de von neumann

- Quinto princípio:
  - O dispositivo deve possuir elementos para transferir ... de seus componentes específicos C e M para A.
  - Constituindo a *saída, a quinta parte específica do dispositivo: S.*

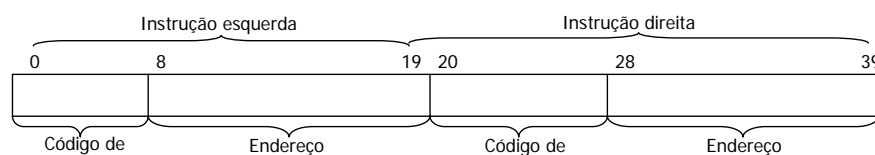
## Estrutura do IAS (memória)

- A memória do IAS consiste em mil posições, denominadas *palavras*, cada uma constituída por 40bits.
- Tanto dados quanto instruções são armazenados na memória.
  - Dados são representados por 1bit (sinal) + 39bits(valor)



Armazenamento de um número

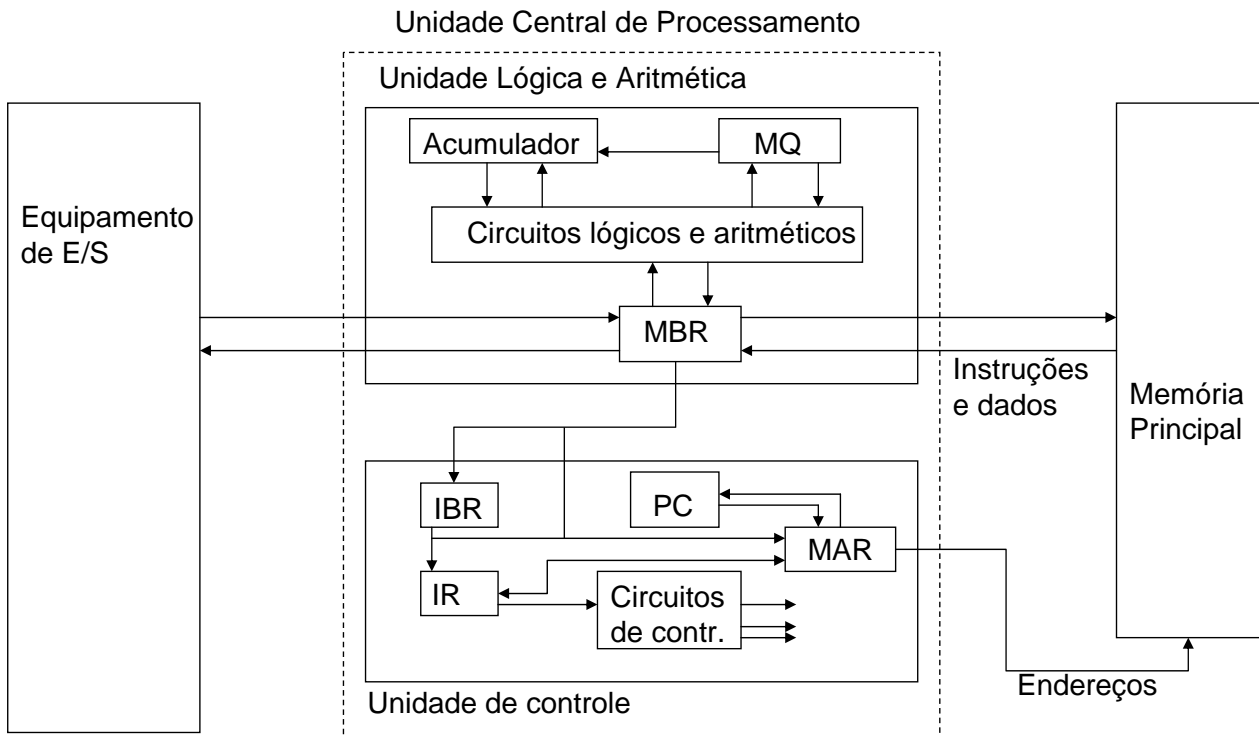
- Instruções possuem 8bits (código da operação) e 12bits (endereço).



Formato de uma palavra no IAS

- Cada palavra pode conter um **dado** ou duas **instruções**.

# IAS detalhado



## Estrutura do IAS (unidade de controle)

- A unidade de controle controla a operação do IAS, efetuando a busca das instruções na memória e executando-as, uma de cada vez.
- Tanto a *unidade de controle* quanto a *unidade aritmética* possuem células de armazenamento, denominadas **registradores**.

# Estrutura do IAS (registradores)

- Registrador temporário de dados (*Memory buffer Register – MBR*):
  - Contém uma palavra com dados a ser armazenada na memória ou é utilizado para receber uma palavra da memória.
- Registrador de endereçamento à memória (*Memory Address register – MAR*):
  - Especifica o endereço, na memória, da palavra a ser escrita ou lida no MBR.
- Registrador de instruções (*Instruction register – IR*):
  - Contém o código de operação de 8 bits que está sendo executado.

# Estrutura do IAS (registradores)

- Registrador de armazenamento temporário de instruções (*Instruction buffer register – IBR*):
  - É utilizado para armazenar temporariamente a instrução contida na porção à direita de uma palavra de memória.
- Contador do programa (*Program counter – PC*):
  - Contém o endereço de memória próximo par de instruções a ser buscado da memória.
- Acumulador (*Accumulator – AC*) e quociente de multiplicação (*Multiplier Quocient – MQ*):
  - São utilizados para armazenar temporariamente os operandos e o resultado de operações efetuadas na ULA.
  - Ex.: o resultado da multiplicação de dois números de 40 bits é um número de 80 bits. Os 40 bits mais significativos são armazenados no acumulador e os 40 bits menos significativos no registrador de quociente de multiplicação.

# Instruções do IAS

- O IAS tinha um total de 21 instruções, as quais são agrupadas da seguinte maneira:
  - Transferência de dados:
    - os dados são transferidos entre a memória e os registradores da ULA ou entre dois registradores da ULA.
  - Desvio incondicional:
    - A unidade de controle executa as instruções na seqüência em que se encontram na memória.
    - Essa seqüência pode ser alterada por uma instrução de desvio.
    - Isto é usado para executar seqüências de instruções repetidamente.

# Instruções do IAS

- Desvio condicional:
  - O desvio é efetuado dependendo do teste de uma condição, o que permite a introdução de pontos de decisão.
- Aritmética:
  - Operações executadas pela ULA.
- Alteração de endereço:
  - Possibilita calcular endereços, utilizando a ULA, para então inseri-los em instruções armazenadas na memória.
  - Isto permite ao programa uma considerável flexibilidade de endereçamento.

# Componentes do Computador

- Conceitos básicos da arquitetura de von neumann:
  - Os dados e as instruções são armazenados em uma única memória de leitura e escrita.
  - O conteúdo dessa memória é endereçado pela sua posição, independentemente do tipo de dados nela contidos.
  - A execução de instruções ocorre de modo seqüencial (exceto quando for explicitamente alterada de uma instrução para a seguinte).

---

## Funções dos Computadores

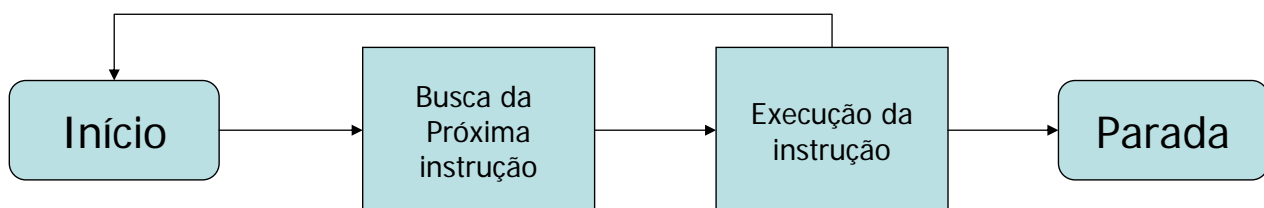
- Básica: executar um programa.
- Como?
  - Seguir o conjunto de instruções armazenadas na memória.
- Quem efetua este trabalho?
  - O processador tem por responsabilidade efetuar a execução das instruções especificadas em um programa.

# Funções dos Computadores

- Processamento de instruções:
  1. O processador lê (busca) as instruções na memória, uma de cada vez, e,
  2. Executa cada uma.
- Execução de um programa consiste na repetição deste processo de *leitura* e *execução* de instruções.
- A execução de uma instrução pode envolver diversas operações, dependendo da natureza da instrução.

## Ciclo de busca e execução

- De uma forma simplificada o funcionamento de um computador pode ser descrito como uma seqüência de ciclos de *busca* e *execução*.



# Busca e execução de instruções

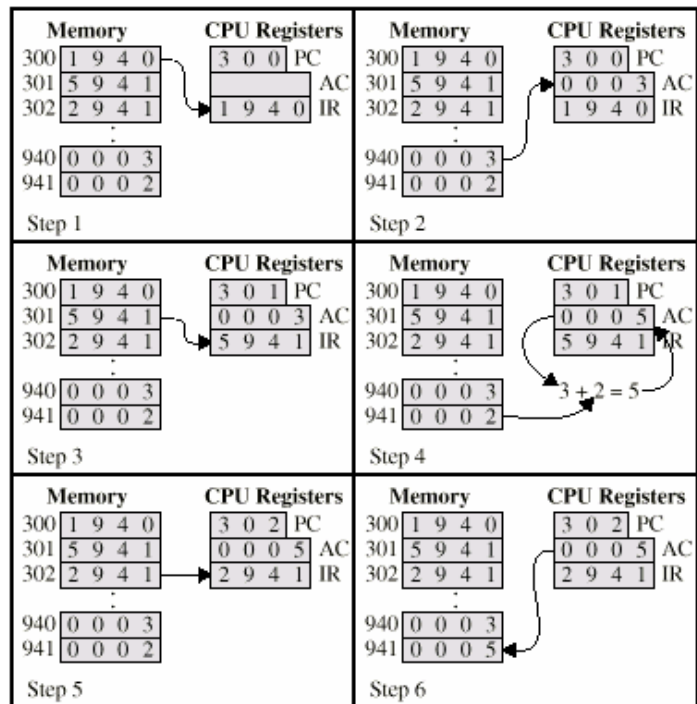
- Contador de programa (PC) guarda o endereço da próxima instrução a ser executada.
- Após a execução da instrução o processador incrementa o PC. Ou seja, se a instrução atual encontra-se na posição 300, as próximas a serem executadas estarão na 301, 302 etc..
- A instrução é armazenada no IR (registrador de instrução)

## Categorias de uma instrução

- Processador-memória:
  - Transferência de dados do processador para a memória e vice-versa.
- Processador E/S:
  - Transferência de dados entre o processador e um dispositivo periférico por meio de um módulo de E/S.
- Processamento de dados:
  - Execução de operações aritméticas ou lógicas sobre os dados
- Controle:
  - Alteração da seqüência das instruções.

# Execução de um programa

- LISTA DE REGISTRADORES
  - PC – Contador de programa
  - AC – acumulador
  - IR – registrador de instruções;
- LISTA DE CÓDIGOS (operação)
  - 0001 (1) = carregar AC a partir do endereço de memória especificado.
  - 0010 (2) = armazenar o valor contido em AC no endereço de memória especificado.
  - 0101 (5) = acrescentar ao valor contido em AC o valor contido no endereço de memória especificado.



# Execução de um programa

1. O conteúdo do PC é 300, o endereço da primeira instrução.
  - Instrução carregada no IR (uso de MAR e MBR)
2. Os quatro primeiros bits (1) do IR indicam que um valor deve ser armazenado no registrador AC. Os 12 bits (940) especificam o endereço 940, de onde o valor deve ser obtido.
3. O PC é incrementado e a próx. instrução é buscada (5941).
4. O conteúdo de AC é somado com o conteúdo da posição de memória de endereço 941 e o resultado é armazenado em AC.
5. O PC é incrementado e a próx. instrução é buscada (2941).
6. O conteúdo de AC é armazenado na posição de memória de endereço 941.

# Exercício:

## Memória

400 - 1802

401 - 7803

402 - 2809

403 - 3504

504 - 5802

505 - 2803

802 - 0025

803 - 0010

809 - ????

## Registadores Internos:

- PC (contador de programa)
- AC (registrador de dados temporário)
- IR (registrador de instrução)

## INSTRUÇÕES:

0001 - carregar AC a partir do endereço de memória especificado.

0010 - armazenar o valor contido em AC no endereço de memória especificado.

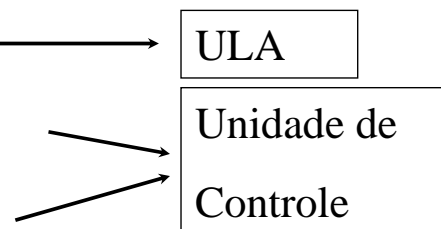
0011 - altera o endereço do PC p/ o endereço especificado.

0101 - = acrescentar ao valor contido em AC o valor contido no endereço de memória especificado.

0111 - subtrai o valor do acumulador do valor contido no endereço especifica

# Tratamento de Instruções

- Registradores Especiais:
- ACC: acumulador
- IR: registrador de instruções
- PC: contador de programa



# Tratamento de Instruções

- Ciclo de Busca
  - carrega a instrução da memória, apontada pelo PC
  - armazena no IR
  - atualiza o PC

# Tratamento de Instruções

- Ciclo de Execução
  - decodifica a instrução contida no IR
  - busca os dados (caso necessário)
  - executa a instrução
  - armazena os resultados (caso necessário)
  - retorna ao passo de busca

## Instruções de Transferência de Dados

<i>Instrução</i>	<i>Opcode</i>	<i>Descrição</i>
■ LOAD MQ	00001010	AC ← MQ
■ LOAD MQ, M(X)	00001001	MQ ← M(X)
■ STOR M(X)	00100001	M(X) ← AC
■ LOAD M(X)	00000001	AC ← M(X)
■ LOAD - M(X)	00000010	AC ← - M(X)
■ LOAD  M(X)	00000011	AC ←  M(X)
■ LOAD -  M(X)	00000100	AC ← -  M(X)

## Instruções de Desvio Incondicional

<i>Instrução</i>	<i>Opcode</i>	<i>Descrição</i>
■ JUMP M(X,0:19)	00001101	próxima instrução <b>M(X,0:19)</b>
■ JUMP M(X,20:39)	00001110	próxima instrução <b>M(X,20:39)</b>

## Instruções de Desvio condicional

<i>Instrução</i>	<i>Opcode</i>	<i>Descrição</i>
■ JUMP +M(X,0:19)	00001111	Se $AC \geq 0$ , então próxima instrução <b>M(X,0:19)</b>
■ JUMP +M(X,20:39)	00010000	Se $AC \geq 0$ , então próxima instrução <b>M(X,20:39)</b>

## Instruções Aritméticas

<i>Instrução</i>	<i>Opcode</i>	<i>Descrição</i>
■ ADD M(X)	00000101	<b>AC</b> ← <b>AC+M(X)</b>
■ ADD  M(X)	00000111	<b>AC</b> ← <b>AC+  M(X) </b>
■ SUB M(X)	00000110	<b>AC</b> ← <b>AC-M(X)</b>
■ SUB  M(X)	00001000	<b>AC</b> ← <b>AC -  M(X) </b>
■ MUL M(X)	00001011	<b>AC, MQ</b> ← <b>MQ×M(X)</b>
■ DIV M(X)	00001100	<b>MQ, AC</b> ← <b>MQ/M(X)</b>
■ LSH	00010100	<b>AC</b> ← <b>AC x 2</b>
■ RSH	00010101	<b>AC</b> ← <b>AC / 2</b>

## Instruções de alteração de endereço

<i>Instrução</i>	<i>Opcode</i>	<i>Descrição</i>
■ STOR M(X,8:19)	00010010	<b>M(X,8:19) ← AC(28:39)</b>
■ STOR M(X,28:39)	00010011	<b>M(X,28:39) ← AC(28:39)</b>

## Como o Computador IAS soma 2 números

- Supondo os números armazenados em memória nas posições (endereços) **100** e **101**
- O resultado da soma deve ser armazenado em memória na posição **102**

<i>Instrução</i>	<i>Opcode</i>	<i>Descrição</i>
LOAD M(100)	00000001	<b>AC ← M(100)</b>
ADD M(101)	00000101	<b>AC ← AC+M(101)</b>
STOR M(102)	00100001	<b>M(102) ← AC</b>