Tópicos:

1 - Modos de endereçamento do 8051

2 - Pilha e instruções de Pilha

3 - Instruções que usam pilha:
- instrução CALL
- instrução RET

4 - Interrupção

1 - Modos de Endereçamento do 8051

Os modos de endereçamento referem-se às diferentes formas (tipos de instruções) que o microprocessador oferece para definir e acessar dados.

- O microprocessador pode conter modos de endereçamento que facilitam o acesso à lista de dados .

- Dependendo do modo de endereçamento utilizado o programa pode ter maior ou menor número de bytes.

1 - Modos de Endereçamento

Os dados podem ser definidos nas instruções do μP

OU

Os dados podem estar armazenados nas seguintes áreas:

- Área de dados da EPROM
- Área de dados da RAM interna
- Área de dados da RAM externa
- Área de dados na Pilha

1 - Modos de Endereçamento do 8051
RAM interna

1. **Imediato:** o dado é definido na própria na própria instrução, sendo precedido por #.

Exemplo :

```
MOV    A, #dado8

MOV    A, #5FH ;   A = 5FH
```

1 - Modos de Endereçamento do 8051
RAM interna

2. Endereçamento por Registrador: um registrador contém o dado

Exemplo:

MOV 30H, R2 ; uma cópia do dado de R2 é armazenada no endereço 30H da RAM interna

ADD A,R3 ; o conteúdo de R3 é somado com A, resultado em A;

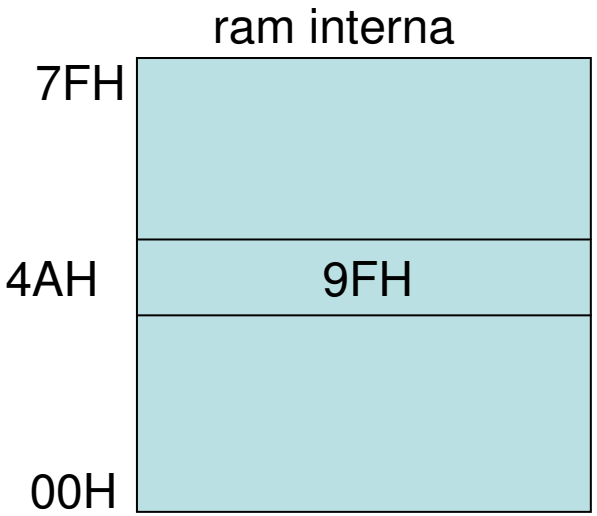
1 - Modos de Endereçamento do 8051
RAM interna

3. Endereçamento Direto: o dado é acessado através de seu endereço

Exemplo:

MOV R7, 4AH ; uma cópia do conteúdo do endereço 4AH da RAM interna, é armazenada em R7

R7← 9FH



1 - Modos de Endereçamento do 8051
RAM interna

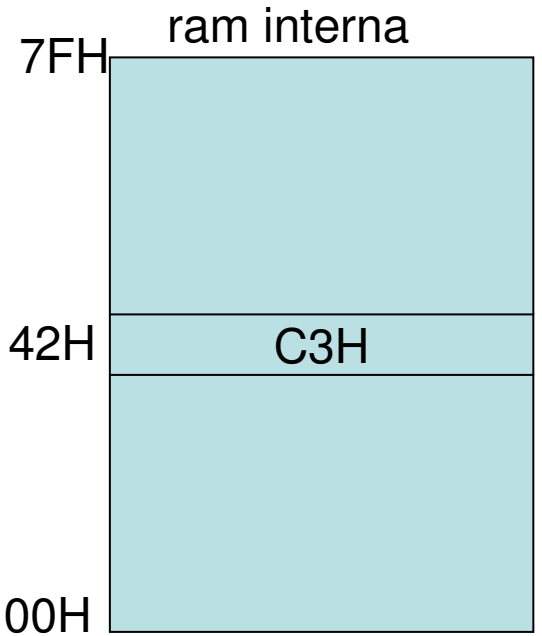
4. Endereçamento Indireto: nesse modo podem ser usados somente **R0** ou **R1,** que são precedidos por @, significando que R0 ou R1 são ponteiros, isto é, contém o endereço do dado.

Exemplo:

MOV A, @R0 ;

Se R0 = 42H, e nesse endereço está armazenado C3H, o acumulador recebe C3H

A←C3H



1 - Modos de Endereçamento do 8051
Memória de Programa (EPROM)

5. Endereçamento Indexado: o conteúdo do ponteiro DPTR é somado ao conteúdo do acumulador. O resultado é o valor do endereço que será acessado pela instrução

Instrução :
MOVC A,@A+DPTR

- Esta instrução acessa área de dados em EPROM
- DPTR é um ponteiro de 16 bits

Exemplo:

```
MOV   DPTR, #0F0BH
MOV   A, #02H
MOVC A,@A+DPTR
```

No exemplo o conteúdo do DPTR será somado ao conteúdo de A:
0F0BH + 02H = 0F0DH
O endereço resultante 0F0DH será acessado na EPROM, o seu conteúdo será lido e armazenado em A.

1 - Modos de Endereçamento do 8051
Memória RAM externa

6. Endereçamento Indireto com ponteiro de 16 bits (DPTR) o conteúdo do ponteiro DPTR é o endereço que será acessado na RAM externa para leitura ou gravação.

Instruções :

MOVX A,@DPTR; leitura da RAM externa

MOVX @DPTR,A ; gravação na RAM externa

2 – PILHA NO 8051

Pilha é uma área da memória RAM INTERNA, onde podem ser lidos ou gravados dados, sob o controle do **ponteiro SP**.

Característica da pilha no microprocessador 8051

É usada para :

- armazenamento de dados de 8 bits, com instruções de pilha (PUSH ou POP)
- guardar um endereço quando é executada uma instrução CALL
- guardar um endereço quando uma interrupção é atendida

❖ O ponteiro **SP** é de 8 bits : é iniciado com o valor 07H da RAM interna ao se fazer "reset"

❖O ponteiro **SP** é incrementado antes de um dado ser armazenado na pilha, portanto a pilha inicia no endereço seguinte (08H)

2 – PILHA no 8051

Instruções de pilha:

```
PUSH    iram ;
POP     iram ;
```

OBS: o endereço iram(8), refere-se a um endereço (8bits) da RAM interna

2 – PILHA NO 8051

Instrução PUSH iram:

- Incrementa o SP e guarda o conteúdo do endereço iram na pilha

PUSH iram

1 . SP ← SP +1

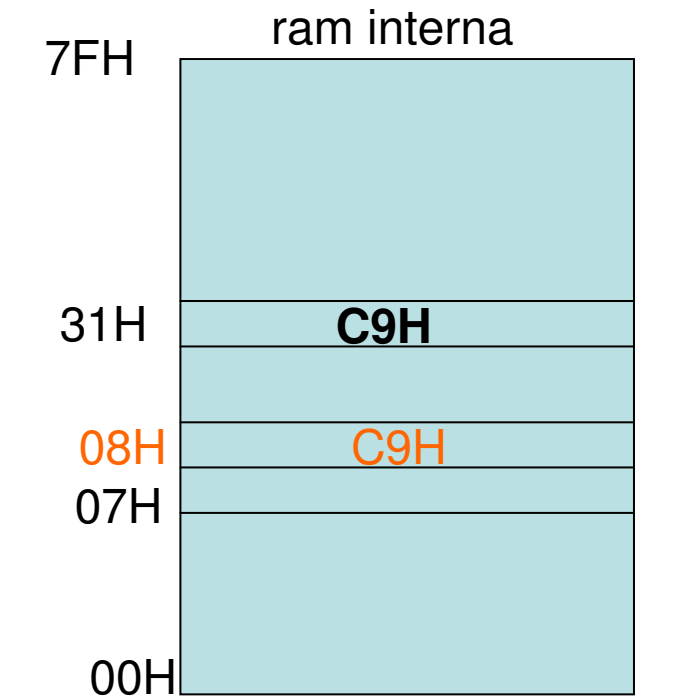
2 . (SP) ← (iram)

Exemplo: PUSH 31H

Valor inicial de SP = 07H

Na execução: SP = 07 + 01= 08H

(08H) ← (31H)



Após a execução de PUSH 31H

2 – PILHA NO 8051

Instrução POP iram:

- Armazena o dado apontado por SP, no endereço **iram**
- em seguida decrementa o ponteiro SP

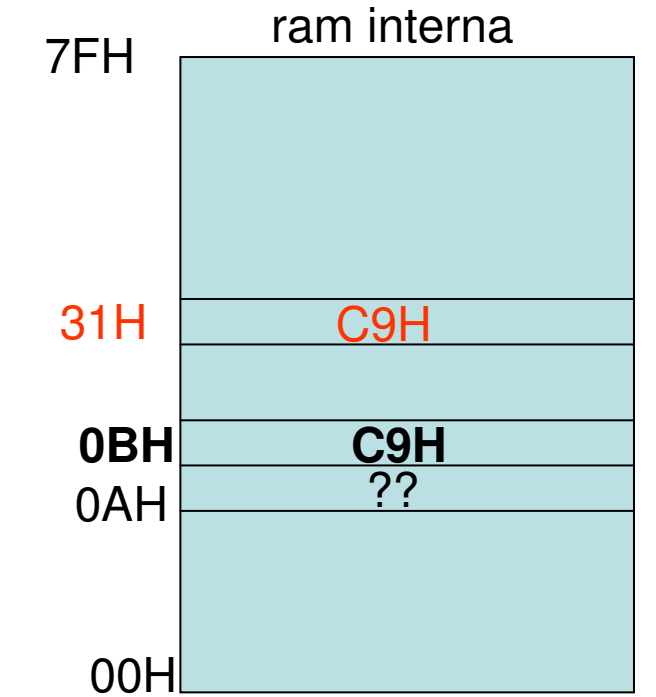
POP iram
(iram) ← (SP)
SP ← SP - 1

Exemplo: POP 31H

Valor inicial de SP = 0BH

Na execução: (31H) ← (SP)

SP = 0B - 01= 0AH



Após a execução de POP 31H

3 – Instruções que usam Pilha

3.1 Instrução CALL

Na execução da instrução CALL:

- o endereço da instrução seguinte (contido no ponteiro PC) é armazenado na Pilha
- o ponteiro de programa PC, é carregado com o endereço especificado na instrução CALL

CALL endereço

((SP) + 1)←(PCL)

((SP) +2)←(PCH)

(SP)←(SP) + 2

(PC)←endereço

3 – Instruções que usam Pilha

3.1 Instrução CALL

Exemplo: valor inicial de SP = 1FH

Endereço instrução

07FEH **LCALL 100BH**

MOV B,A

.....

A instrução LCALL é de 3 bytes e ocupa 3 endereços: 07FEH, 07FFH E 0800H

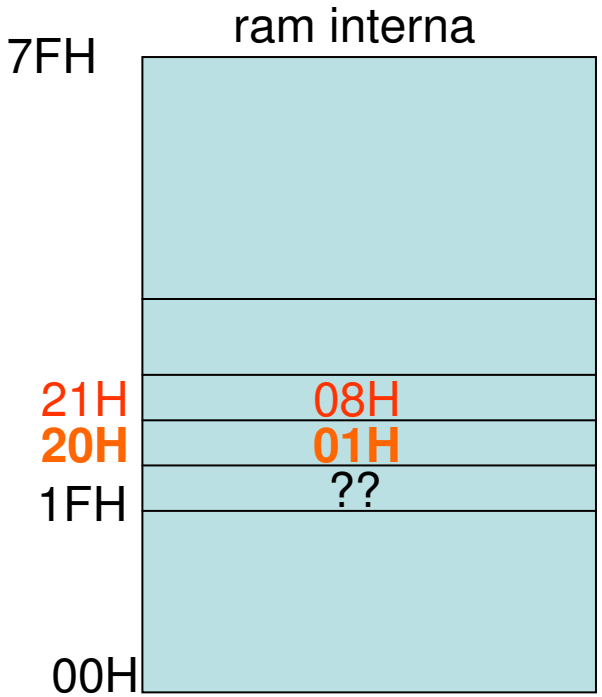
PORTANTO o endereço da instrução MOV B, A é **0801H.**

Ações executadas:

1 . O endereço 0801H é armazenado na pilha

2 . O PONTEIRO DE PROGRAMA PC é carregado com o endereço 100Bh

3. O valor final de SP = 21H



3 – Instruções que usam Pilha

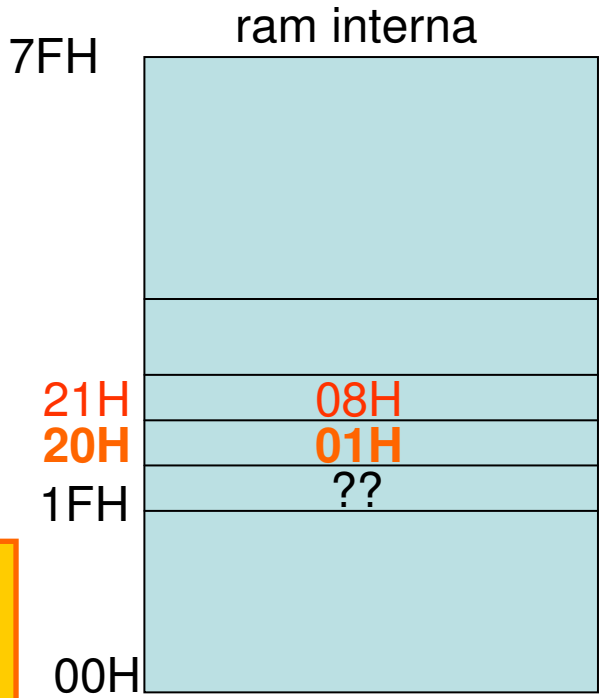
3.2 Instrução RET

- **A instrução RET executa um POP do PC (contador de programa), ou seja, recupera o PC da Pilha**

EXEMPLO: valor inicial do SP =21H

```
O7FEH     LCALL  conv
          MOV    B,A
            ...

          ORG   100BH ;  conv= 100BH
conv:     SUBB  30H ;
          RET
```



Ações executadas:

1 . O ponteiro de programa recebe da pilha o valor 0801H e volta a executar a partir deste endereço.

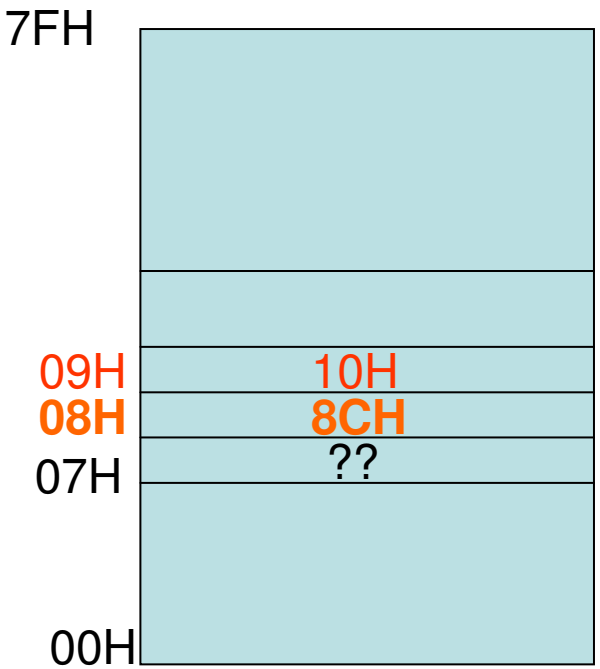
2. SP é decrementado de duas unidades, SP = 1FH

4 – Interrupção

Supondo que ocorreu uma solicitação de interrupção quando estava sendo executada a instrução MOV DPTR , #20FCH, que é uma instrução de 3 bytes.
Dados:
SP = 07H
endereço da subrotina de interrupção: 000BH

Endereço	instrução
1089H	MOV DPTR , #20FCH
	MOVC A,@A + DPTR

Ações executadas pelo microprocessador ao atender a interrupção:
1- conclui a execução da instrução corrente: MOV DPTR , #20FCH
2 - salva na pilha o endereço da instrução seguinte (108CH): MOVC A,@A + DPTR
3 – carrega o PC com o endereço 000BH, da interrupção



4 – Interrupção

A subrotina de interrupção deve ser finalizada com a instrução RETI, que tem o mesmo efeito da instrução RET já explicada anteriormente.

FIM