



Departamento de Engenharia Elétrica e de Computação - EESC-USP

SEL-0415 Introdução à Organização de Computadores

Aula 8 : Microcontrolador 8051 – Parte 2

Profa. Luiza Maria Romeiro Codá

MICROCONTROLADOR 8051

Mapeamento de memória interna,
pilha e Interrupção

Memória de dados interna (RAM)

8051 oferece uma memória de dados interna, com um mínimo de 128 bytes:

- **Vantagem 1:** rápido acesso aos dados e, em muitas aplicações, pode eliminar a necessidade da RAM externa \Rightarrow custo menor e maior velocidade;
- **Vantagem 2:** áreas de RAM interna acessíveis bit a bit \Rightarrow útil para operações booleanas.

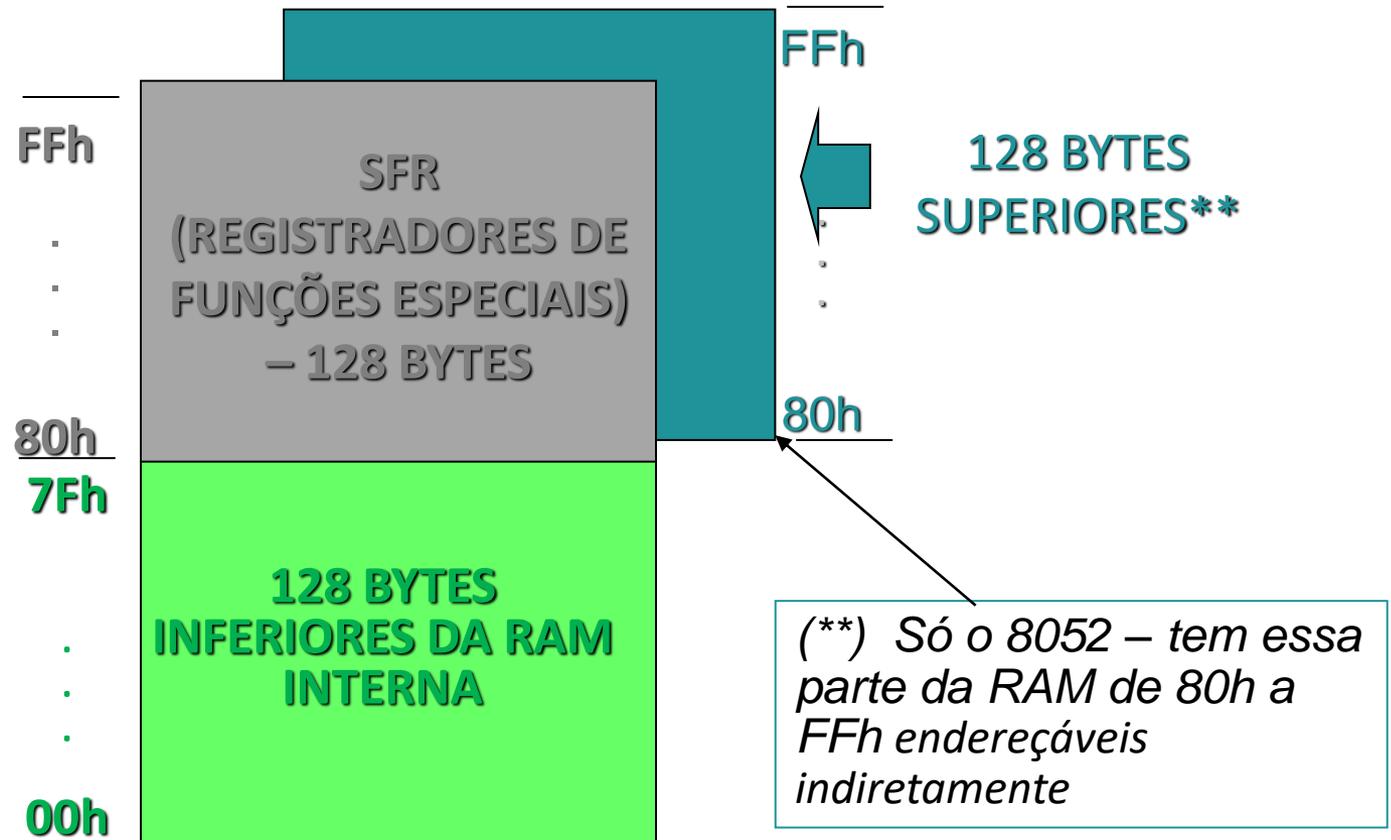
Espaço de endereçamento para acessar a RAM interna = 8 bits \Rightarrow máximo de 256 bytes (8052 \Rightarrow mais 128 bytes = 384 bytes);

RAM interna \Rightarrow 4 bancos de 8 registradores (**R0 – R7**) que podem ser utilizados pelo usuário

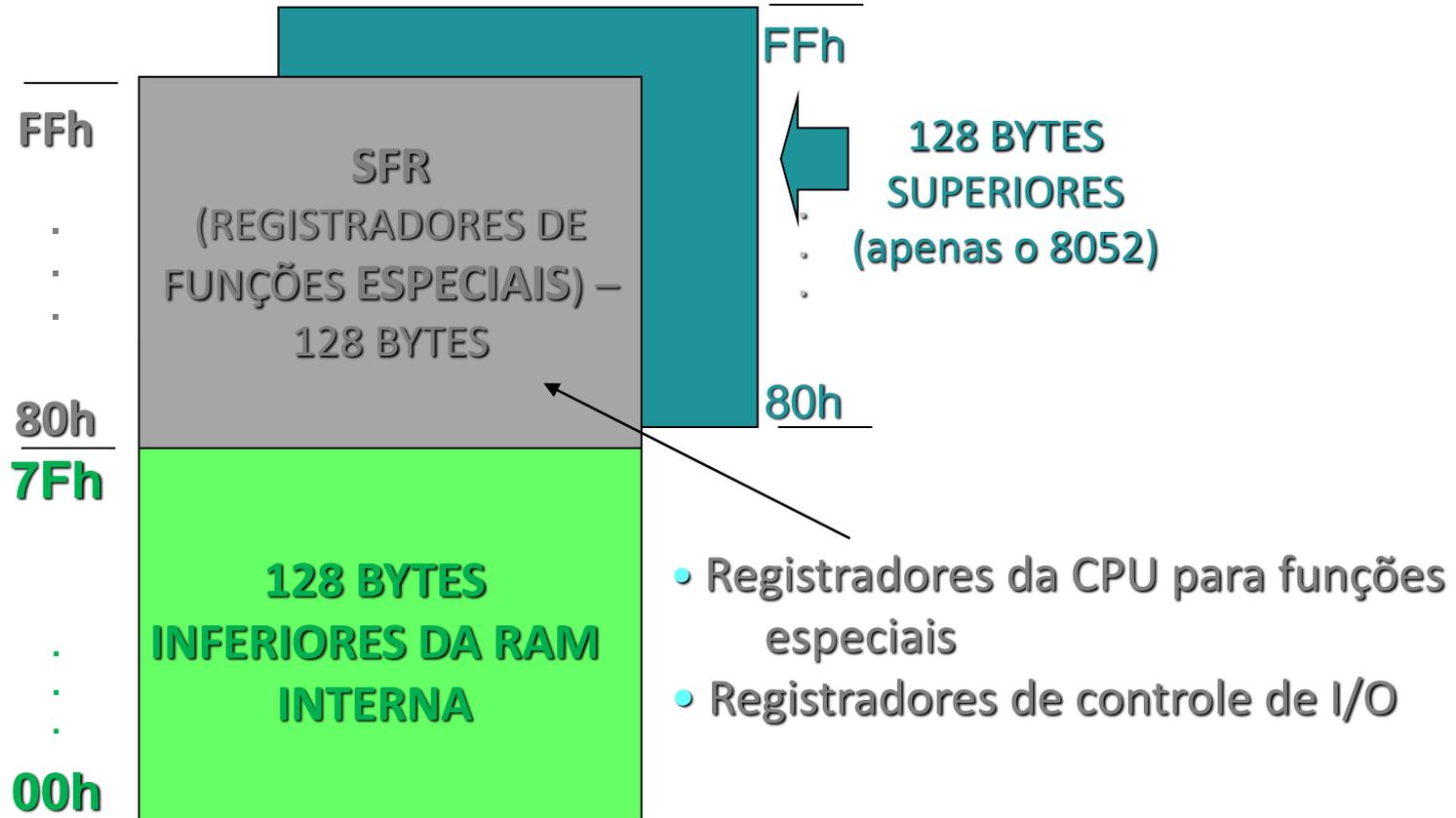
Memória de dados interna (RAM)



Memória de dados interna (RAM)

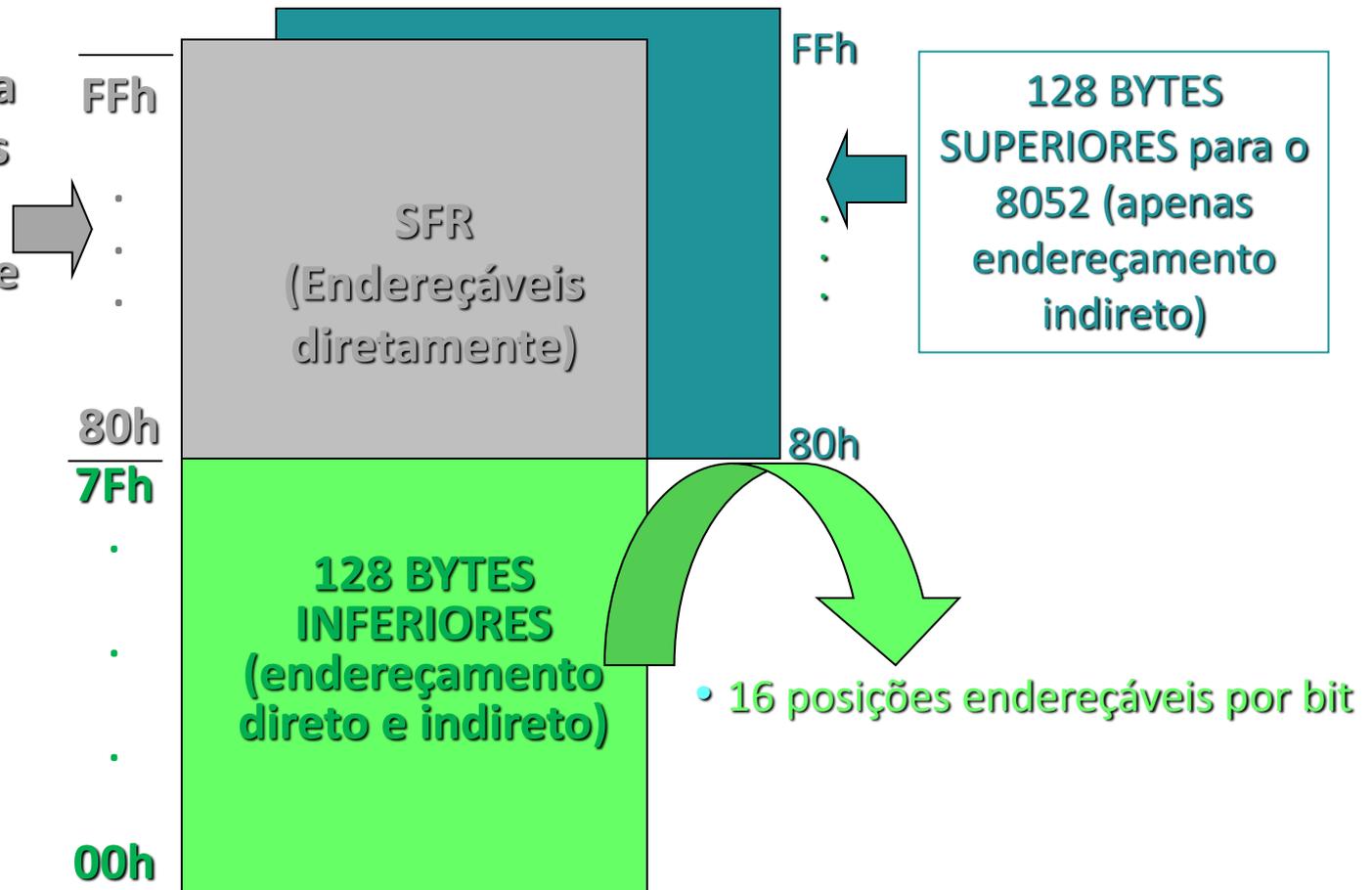


Memória de dados interna (RAM)



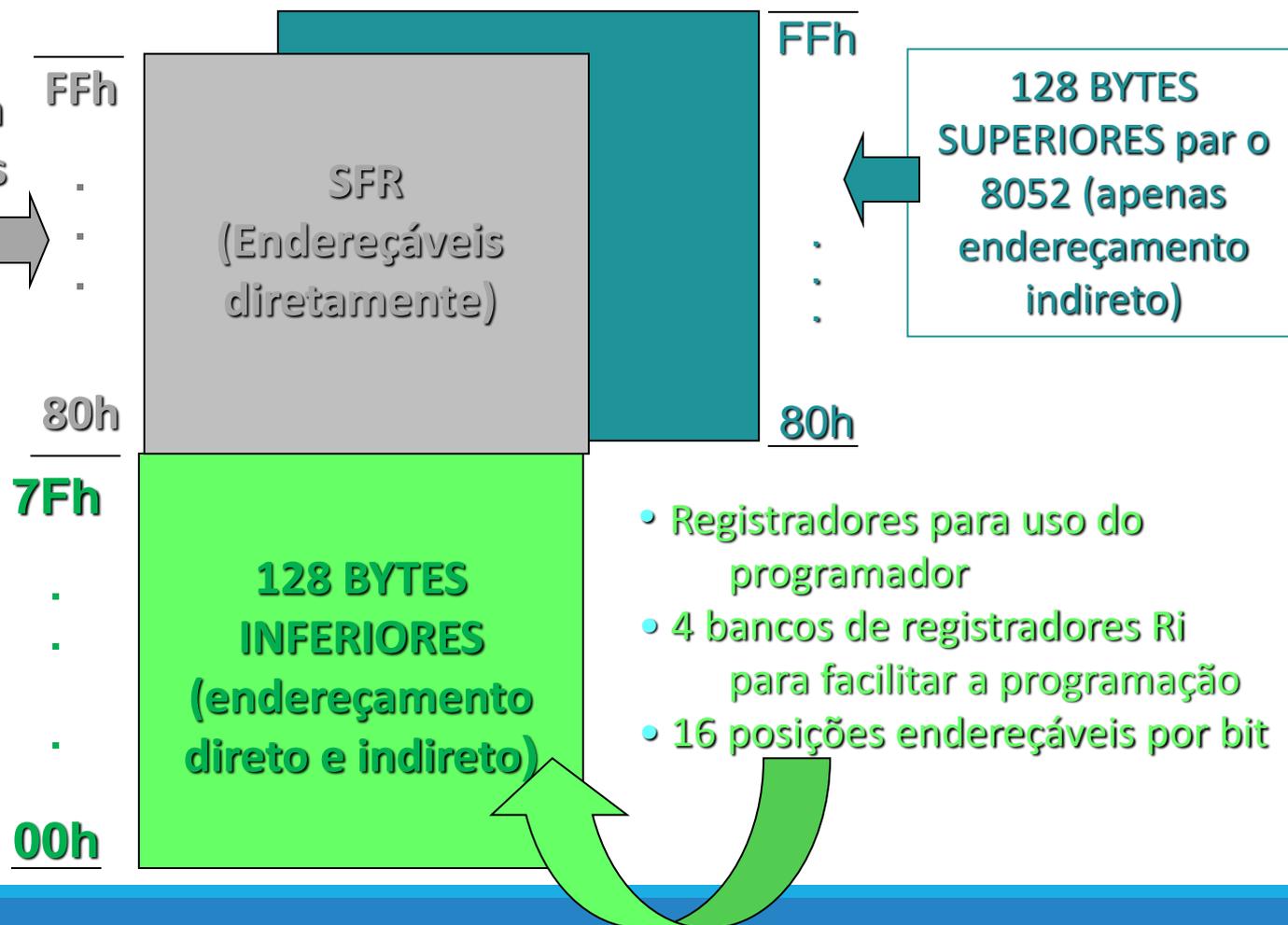
Registadores de Funções Especiais (SFR)

- Registradores da CPU para funções especiais
- Registradores de controle de I/O



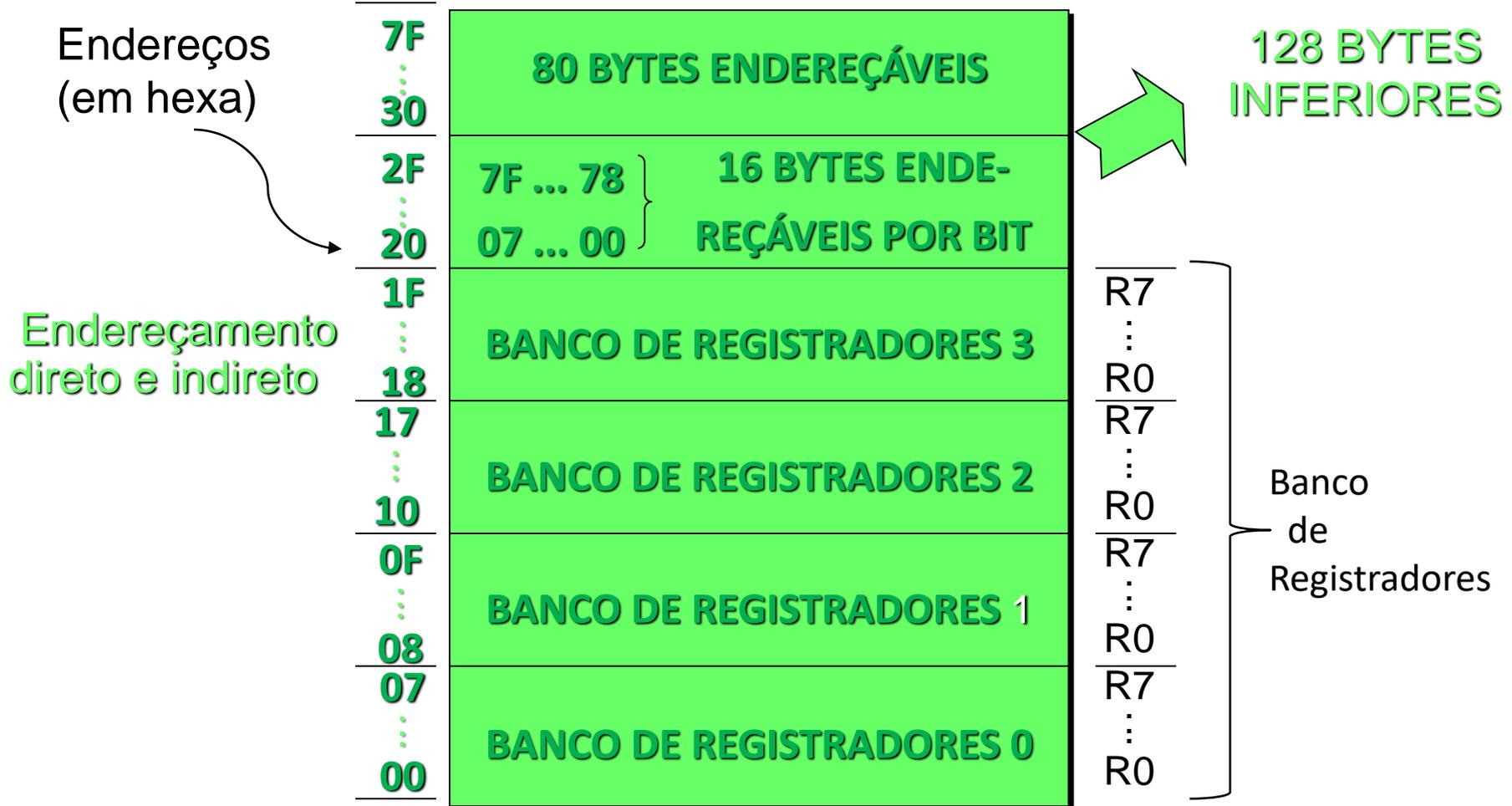
Registadores de Propósito Geral (GPR)

- Registradores da CPU para funções especiais
- Registradores de controle de I/O



Memória de dados interna (RAM)

Endereçamento feito com 8 bits



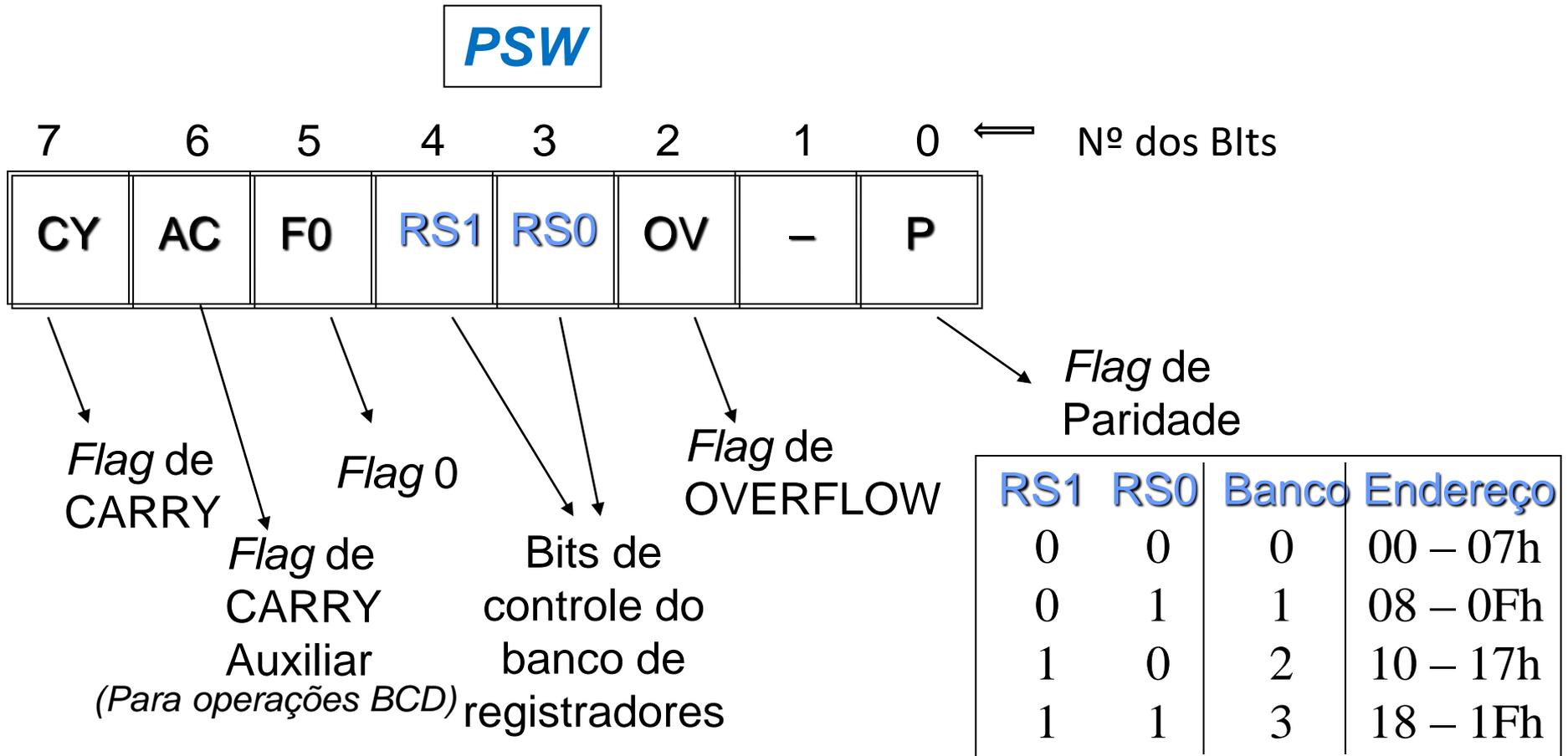
Memória de dados interna (RAM)

BANCO DE REGISTRADORES

- Formados pelos registradores R0 a R7
- R0 a R7 são posicionados na RAM interna nos 128 bytes menos significativos
- Seleção entre os Bancos, feita pelos bits 3 e 4 do byte PSW (*program status word*) que é um registrador que se localiza na RAM dos SFRs

Memória de dados interna (RAM)

Seleção do Banco de Registradores



Microcontrolador 80C51

Flags ➔ **bits indicadores de estado:**

- contidos no registrador PSW (palavra de status do programa – “*program status word*”);
- são colocados em “1” ou “0” dependendo do resultado das operações da CPU;
- algumas instruções testam *flags* para definir se a execução do programa prossegue na instrução seguinte, ou se salta para outra parte do programa;
- *flags* típicas: *SIGN*, *CARRY*, *ZERO*, *OVERFLOW*
bit de *flag* usualmente se refere ao estado do acumulador (A);
- bit de sinal = MSB do acumulador (A) após a operação da ULA.

Microcontrolador 80C51

Instruções que alteram *Flags* ➔ bits indicadores de estado:

Instrução	Flags			Instrução	Flags		
	C	OV	AC		C	OV	AC
ADD	X	X	X	CLR C	0		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C,bit	X		
MUL	0	X		ANL C,/bit	X		
DIV	0	X		ORL C,bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C,bit	X		
RLC	X			CJNE	X		
SETB C	1						

Memória de dados da RAM interna

Área de dados

As posições de 30h a 7Fh da RAM interna são disponíveis para leitura e escrita, através de endereçamento direto e indireto.

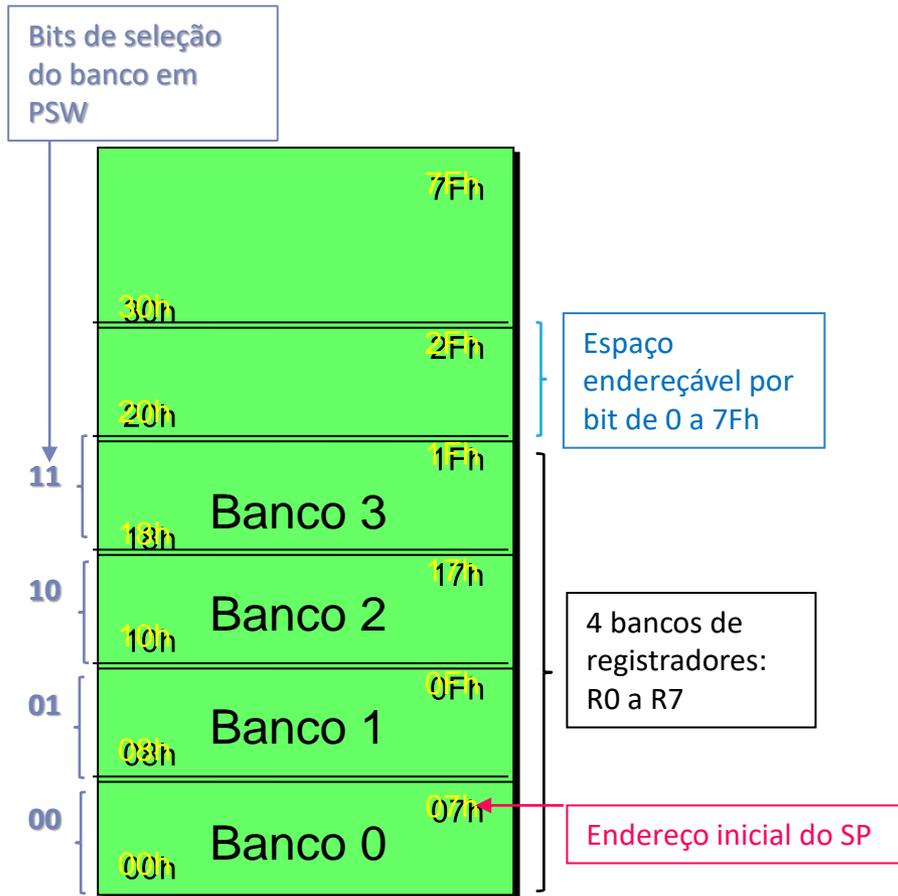
Bits de seleção em PSW

O SP é inicializado no endereço 07H, mas a 1ª. Posição da pilha a ser gravada é 08H (banco 1).



Memória de dados da RAM interna

Bytes endereçáveis



Cada uma dessas posições de memória pode também ser acessada imediatamente ou por registrador .

Ex.:

a) Endereçamento Imediato

mov 20h, #0AAh

O valor do dado é AAH , mas para o compilador interpretar que é um valor e não "string" deve ser colocado o número 0 antes

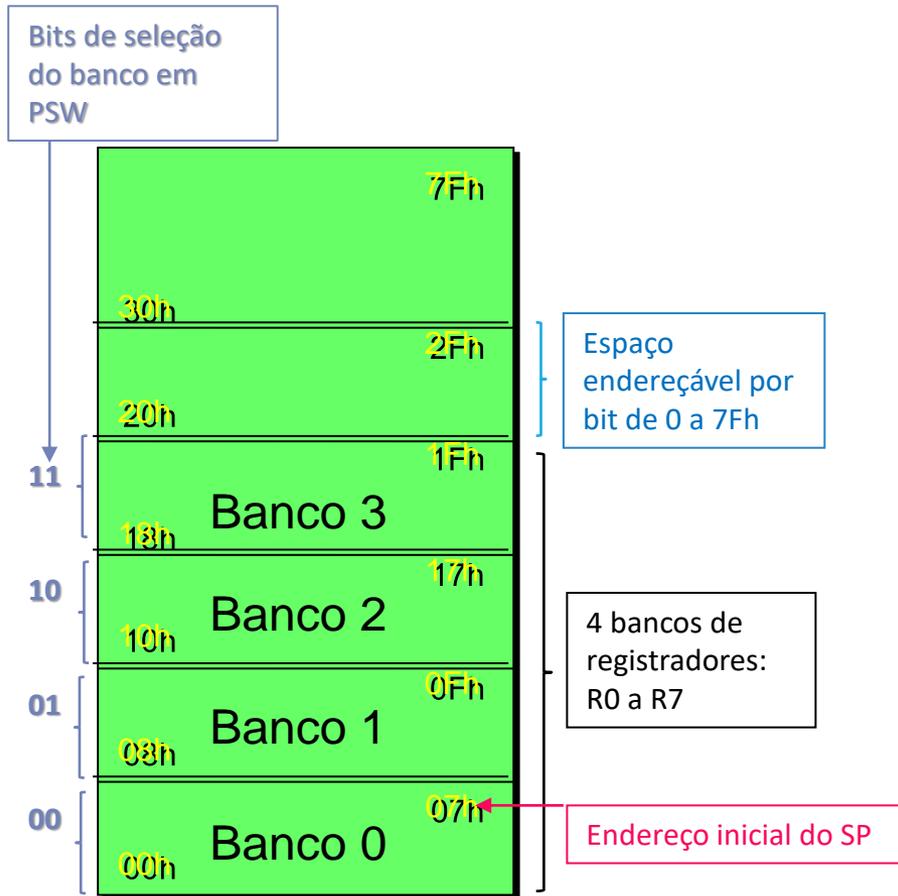
b) Endereçamento por Registrador

mov R0,R2

add R0,R1

Memória de dados da RAM interna

Bytes endereçáveis



Cada uma dessas posições de memória pode também ser acessada direta ou indiretamente por byte .

Ex.:

c) Endereçamento Direto

mov 20H,30H

d) Endereçamento Indireto

mov @R0,#0AAh

ou

mov A, @R1

Memória de dados da RAM interna

**Faixa de RAM interna
endereçável por Bit
(de 20H a 2FH).**

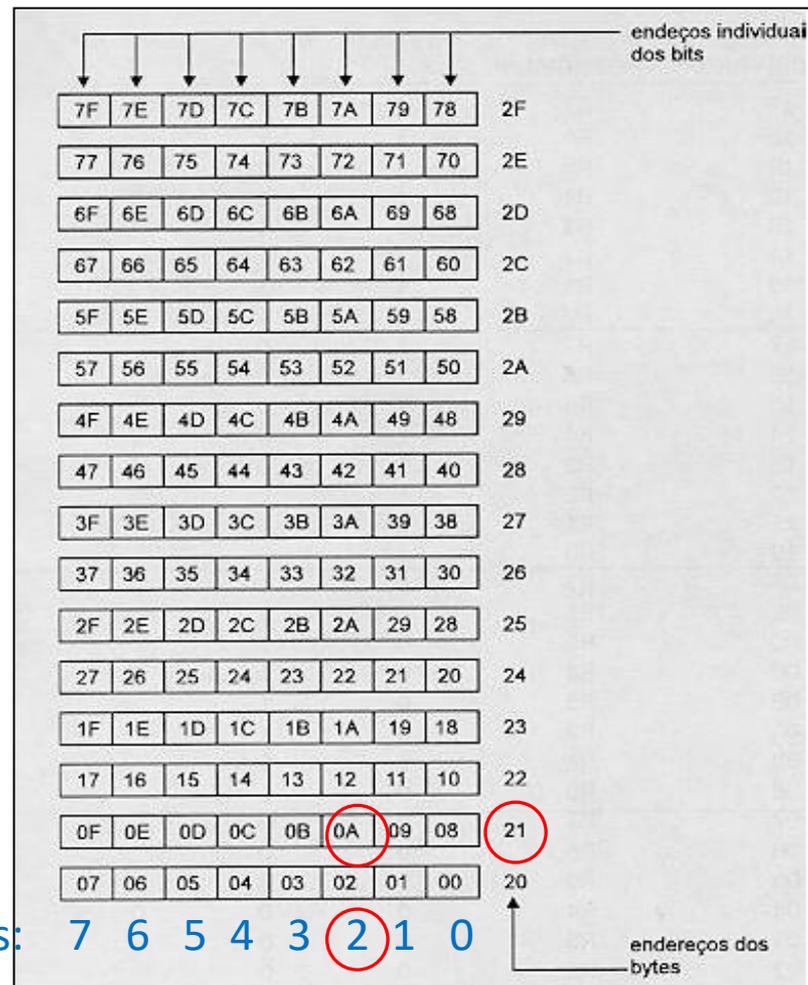
Ex. : Setar o bit 2 da posição 21h

setb 0Ah

ou

setb 21h.2

Existe um conjunto de instruções
específicos para endereçamento a bit.



Memória de dados da RAM interna

**Faixa de RAM interna endereçável por Bit
(de 20H a 2FH).**

Bits de seleção
do banco em
PSW



INSTRUÇÕES DE BIT:

CLR bit → zera o bit diretamente

SETB bit → seta o bit diretamente

CPL bit → complementa o bit diretamente

ANL C,bit → AND entre o bit e o *carry*

ANL C,/bit → AND entre o complemento do bit e o *carry*

ORL C,bit → OR entre o bit e o *carry*

ORL C,/bit → OR entre o complemento do bit e o *carry*

MOV C,bit → move o bit para o *carry*

MOV bit,C → move o *carry* para o bit

JB bit,rel → jmp para rel se bit = 1

JNB bit,rel → jmp para rel se bit = 0

JBC bit,rel → jmp para rel se bit = 1 e zera o bit

RAM interna dos Registradores de funções Especiais (SFRs)

Faixa de 80H a FFH endereçável diretamente

F0h	B
E0h	ACC
D0h	PSW
B8h	IP
B0h	P3
A8h	IE
A0h	P2
99h	•
98h	•
98h	SBUF
98h	SCON
98h	•
98h	•
98h	•
90h	P1
8Dh	TH1
8Ch	TH0
8Bh	TL1
8Ah	TL0
89h	TMOD
88h	TCON
87h	PCON
•	•
•	•
•	•
83h	DPH
82h	DPL
81h	SP
80h	P0

Registradores uso geral

Contém flags sinalizadores da ULA

Registradores para configuração de Interrupções

Porta Paralela P3 cada bit tem uma função diferente

Porta Paralela P2 contém 8 bits mais significativos do endereço A15-A8

Registradores Para comunicação serial

Porta Paralela P1 para propósito geral

Temporizadores e registradores para configuração dos mesmos

Porta paralela P0 Contém Bits menos significativos do endereço A0-A7 ou Dados D0-D7

DPTR (DPH 8 bits + sig DPL 8 bits menos sig)

Ponteiro SP

Registradores de Funções Especiais (SFR) – Mapa

Coluna Bit endereçável



80h	P0	SP	DPL	DPH				PCON	87h
88h	TCON	TMOD	TL0	TL1	TH0	TH1			8Fh
90h	P1								97h
98h	SCON	SBUF							9Fh
A0h	P2								A7h
A8h	IE								AFh
B0h	P3								B7h
B8h	IP								BFh
C0h									C7h
C8h									CFh
D0h	PSW								D7h
D8h									DFh
E0h	ACC								E7h
E8h									EFh
F0h	B								F7h
F8h									FFh

Função dos SFRs

- **Stack Pointer (SP):** Registrador de 8 bits, contém o endereço do topo da pilha;
- **Acumulador (A ou ACC):** Registrador de 8 bits, é o registrador usado nas operações lógicas e aritméticas;
- **Registrador B (B):** Registrador de 8 bits, usado nas operações de multiplicação e divisão. Pode ser usado como um registrador de uso geral;
- **Data Pointer (DPTR):** Registrador de 16 bits formado pela junção de dois registradores de 8 bits, DPH e DPL. Este registrador é usado para acessar a memória externa;
- **Ports (P0, P1, P2 e P3):** Registradores de 8 bits que contém o valor presente nos pinos de entrada e saída do 8051 quando estes pinos estão sendo usados como pinos de entrada e saída;
- **Program Status Word(PSW):** contém flags sinalizadores da ULA e possibilita alterar banco de registradores
- **Interrupt Enable(IE):** Registrador de 8 bits, usado para habilitar ou desabilitar as interrupções do 8051;
- **Interrupt Priority (IP):** Registrador de 8 bits que determina o nível de prioridade (alto ou baixo) das interrupções do 8051;
- **Power Control (PCON):** Registradores de 8 bits, controla os modos de economia de energia do 8051;
- **Timer Control (TCON):** Registrador de 8 bits, usado para configurar e controlar os temporizadores/ contadores TCO e TC1;
- **Timer Mode (TMODE):** Registrador de 8 bits, define o modo de operação dos temporizadores/ contadores TCO e TC1;
- **Timers (TH0, TL0, TH1 e TL1):** Registradores de 8 bits que contém o valor da contagem dos temporizadores/ contadores 0 e 1;
- **Serial Control (SCON):** Registradores de 8 bits, controla o modo de operação da interface serial;
- **Serial Buffer (SBUF):** Registrador duplo de 8 bits, contém o valor sendo transmitido e recebido pela interface serial. São dois registradores acessados pelo mesmo endereço;

Registradores de Funções Especiais (SFR)

- Qualquer dos SFR pode ser endereçado a byte **diretamente ou imediatamente**, através do endereço de cada um;
- Ou pode ser endereçado através do nome (**endereçamento por registrador**).

Exemplo:

mov P0,#3Fh ou *mov 80h,#3Fh*

mov DPL,DPH ou *mov 82h,83h*

Registadores de Funções Especiais (SFR)

SFR endereçáveis a bit

- Os SFR's cujos endereços terminam em 0 ou 8h podem também ser endereçados a bit

Modos de acesso ao bit:

(I) por endereço do Bit dentro do Byte:

- setb 80h.1**; seta o bit 1 do endereço 80h (Porta 0)
- clr 80h.2**; zera o bit 2 do endereço 80h (Porta 0)

P0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
80h	80.7	80.6	80.5	80.4	80.3	80.2	80.1	80.0

80	P0
88	TCON
90	P1
98	SCON
A0	P2
A8	IE
B0	P3
B8	IP
C0	
C8	
D0	PSW
D8	
E0	ACC
E8	
F0	B
F8	

Registadores de Funções Especiais (SFR)

SFR endereçáveis a bit

- Os SFR's cujos endereços terminam em 0 ou 8h podem também ser endereçados a bit

Modos de acesso ao bit:

(II) por nome :

- setb P0.1** ; set no bit 1, do endereço 80h (Porta 0)
- clr P0.2** ; zera o bit 2, do endereço 80h (Porta 0)

P0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
80h	80.7	80.6	80.5	80.4	80.3	80.2	80.1	80.0

80	P0
88	TCON
90	P1
98	SCON
A0	P2
A8	IE
B0	P3
B8	IP
C0	
C8	
D0	PSW
D8	
E0	ACC
E8	
F0	B
F8	

Registadores de Funções Especiais (SFR)

SFR endereçáveis a bit

- Os SFR's cujos endereços terminam em 0 ou 8h podem também ser endereçados a bit

Modos de acesso ao bit:

(III) pelo endereço absoluto do bit :

- setb 81h** ; seta o bit 1 do endereço 80h (Porta 0)
- clr 82h** ; zera o bit 2 do endereço 80h (Porta 0)

P0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
80h	80.7	80.6	80.5	80.4	80.3	80.2	80.1	80.0
	87	86	85	84	83	82	81	80

80	P0
88	TCON
90	P1
98	SCON
A0	P2
A8	IE
B0	P3
B8	IP
C0	
C8	
D0	PSW
D8	
E0	ACC
E8	
F0	B
F8	

Registadores de Funções Especiais (SFR)

SFR endereçáveis a bit

Modos de acesso ao bit:

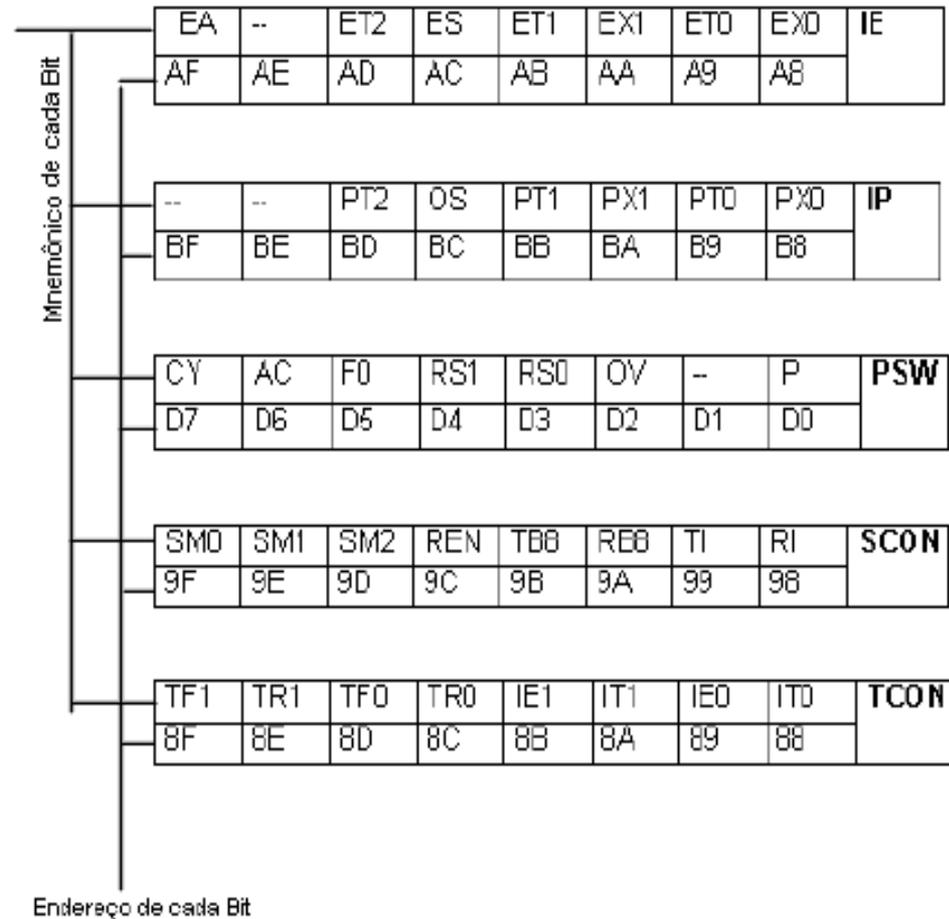
(IV) pelo nome do bit :

Os SFRs endereçáveis a bit que determinam funções, podem ser endereçados através do nome de cada bit:

Ex.:

setb EA ; faz o bit 7 de IE=1

setb 0AFh ; idem



Microprocessador de 8 Bits

PILHA

Pilha ➡ parte da RAM interna dedicada para armazenar dados sob o controle do ponteiro SP.

- Solicitação de Interrupção e chamada de subrotina usam a pilha.
- Existem instruções para armazenar e retirar dados da pilha (ex: PUSH PSW e POP PSW).
- **PONTEIRO DE PILHA** (“*Stack pointer*”) ➡ aponta para o topo da pilha onde o último dado foi armazenado.
- No armazenamento os dados são empilhados sequencialmente.

Microprocessador de 8 Bits

ALGUMAS DEFINIÇÕES ÚTEIS

Uso mais importante da Pilha é na chamada de sub-rotina:

- **CALL** ➔ instrução que diz à CPU para ir ao endereço de início de uma sub-rotina e executá-la

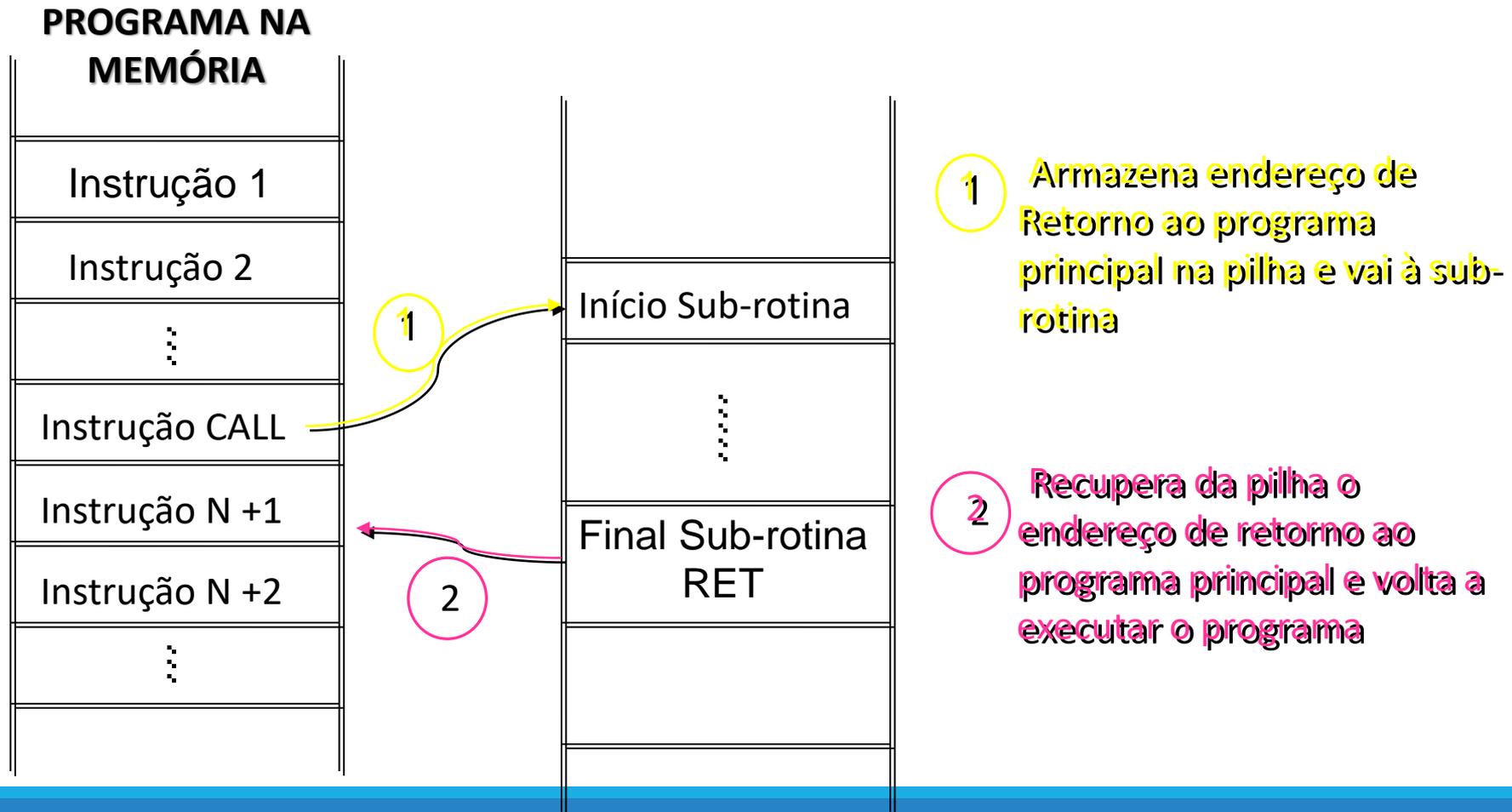
Guarda automaticamente o endereço de retorno (instrução seguinte ao CALL) na pilha, antes de ir para a sub-rotina

- **RETURN** ➔ última instrução

resgata da pilha o endereço de retorno

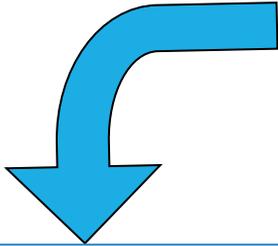
Microprocessador de 8 Bits

ALGUMAS DEFINIÇÕES ÚTEIS



Memória de dados interna (RAM)

PILHA

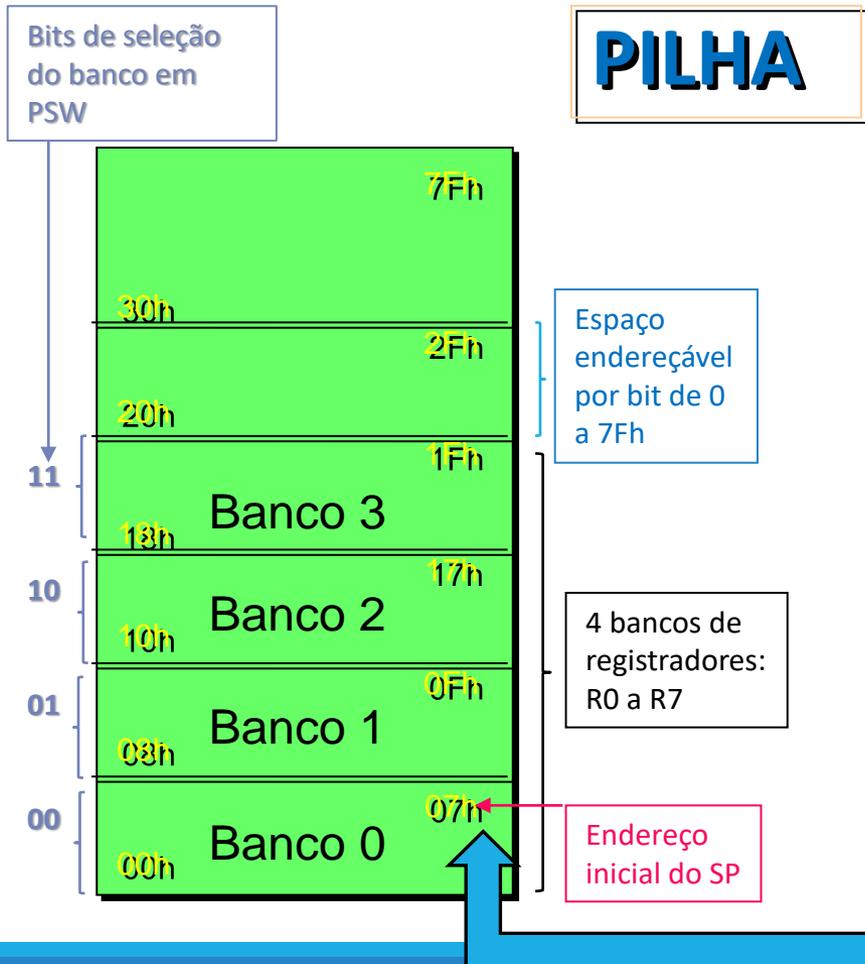


pilha começa na posição 08H (banco 1 de registradores)

- O reset inicializa o **PONTEIRO DA PILHA** (*Stack Pointer – SP*) na posição 07h, e é incrementando a cada vez que é usado.
- como o **SP** aponta para uma região de 8 bits, são necessárias 2 posições para armazenamento de cada endereço da memória de programa (PC=16bits)
- a pilha cresce através da memória (SP é incrementado antes dos dados serem armazenados)
- o ponteiro da pilha pode ser inicializado em qualquer endereço na RAM interna (a escolha do programador ex: `MOV SP, # 20H`)

Lembrete: PILHA é o conteúdo da posição para onde o ponteiro SP aponta na RAM interna

Memória de dados interna (RAM)



- Ao se aplicar reset na CPU, RS1 e RS0 = 0 (bits da registrador PSW)
- o banco de registradores 'default' é o **Banco 0**.

*Para usar mais de um Banco de Registradores, **SP** deve ser inicializado em outra posição da RAM (por ex: posição 30h pela instrução `MOV SP,#30H`)*

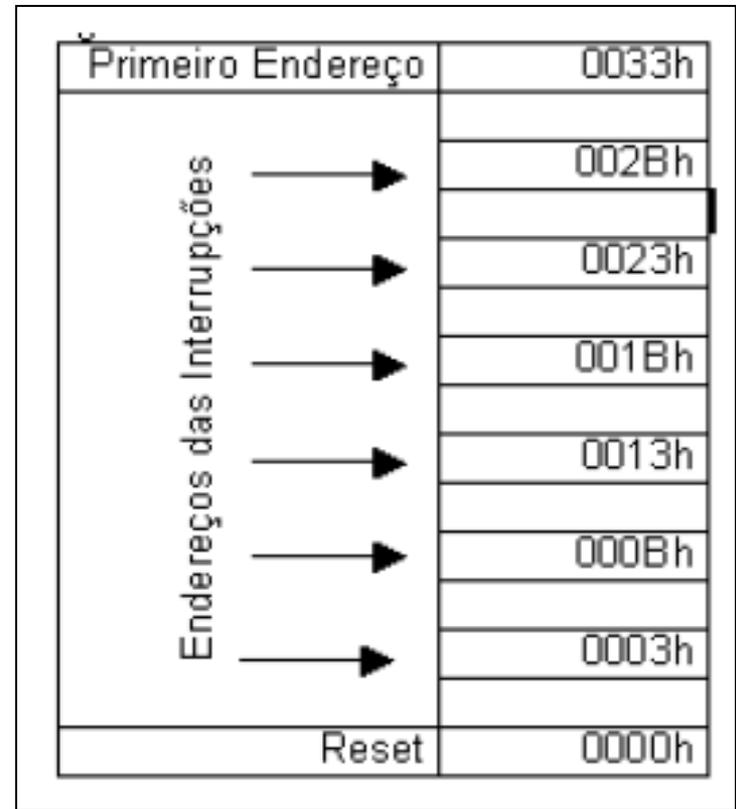
Microcontrolador 80C51

Interrupção

MEMÓRIA DE PROGRAMA

Endereço das interrupções:

- Cada interrupção causa um salto para um endereço fixo na memória de programa (ROM , EPROM ,...) a partir do endereço 0003 H
- 5 interrupções:
 - 2 externas, 2 timers/contadores e 1 porta serial;



Interrupção

Interrupção é uma característica de um computador que permite ao mesmo parar a execução de um determinado programa e passar a executar uma sub-rotina, localizada em um endereço pré-determinado da memória.

A sub-rotina a ser executada é denominada de **Sub-rotina de Atendimento de Interrupção**.

Ao terminar a execução desta sub-rotina o controle volta para o programa inicial no endereço imediatamente abaixo do ponto onde foi interrompido.

Subrotina de Interrupção X Subrotina convencional

Diferença entre uma subrotina convencional e a subrotina de interrupção:

- A **subrotina convencional** é chamada por uma instrução do microprocessador (**instrução CALL**), em posições definidas pelo programador, no programa principal.
- A **subrotina de interrupção** está relacionada à ocorrência de um evento que pode ser imprevisível .

Estrutura de Interrupção

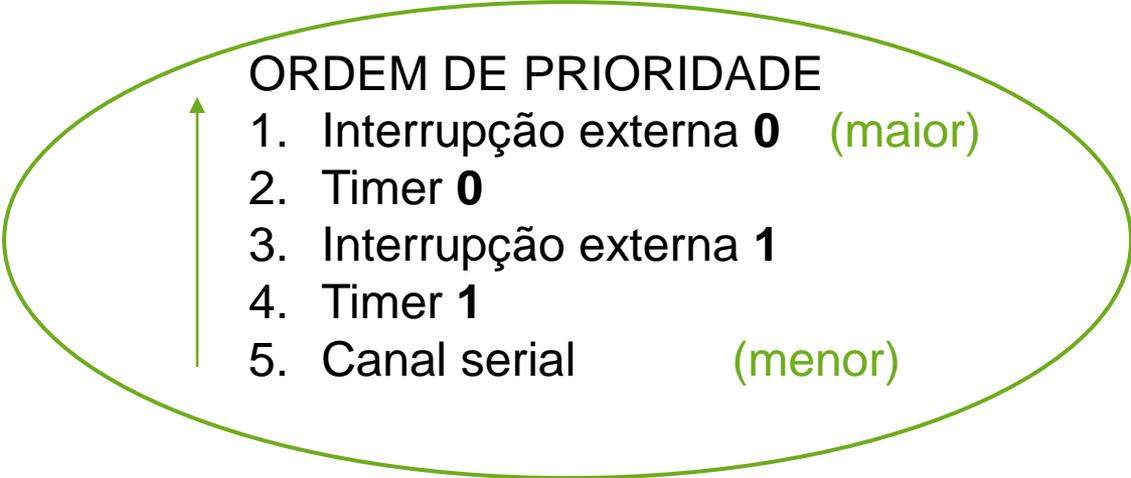
Procedimento:

- ✓ Atendendo a uma interrupção, o microprocessador pára a execução do programa e vai executar uma sub-rotina (*sub-rotina de atendimento de interrupção*)
- ✓ Ao terminar a execução desta sub-rotina, o controle volta para o programa inicial no endereço imediatamente abaixo do ponto onde foi interrompido.

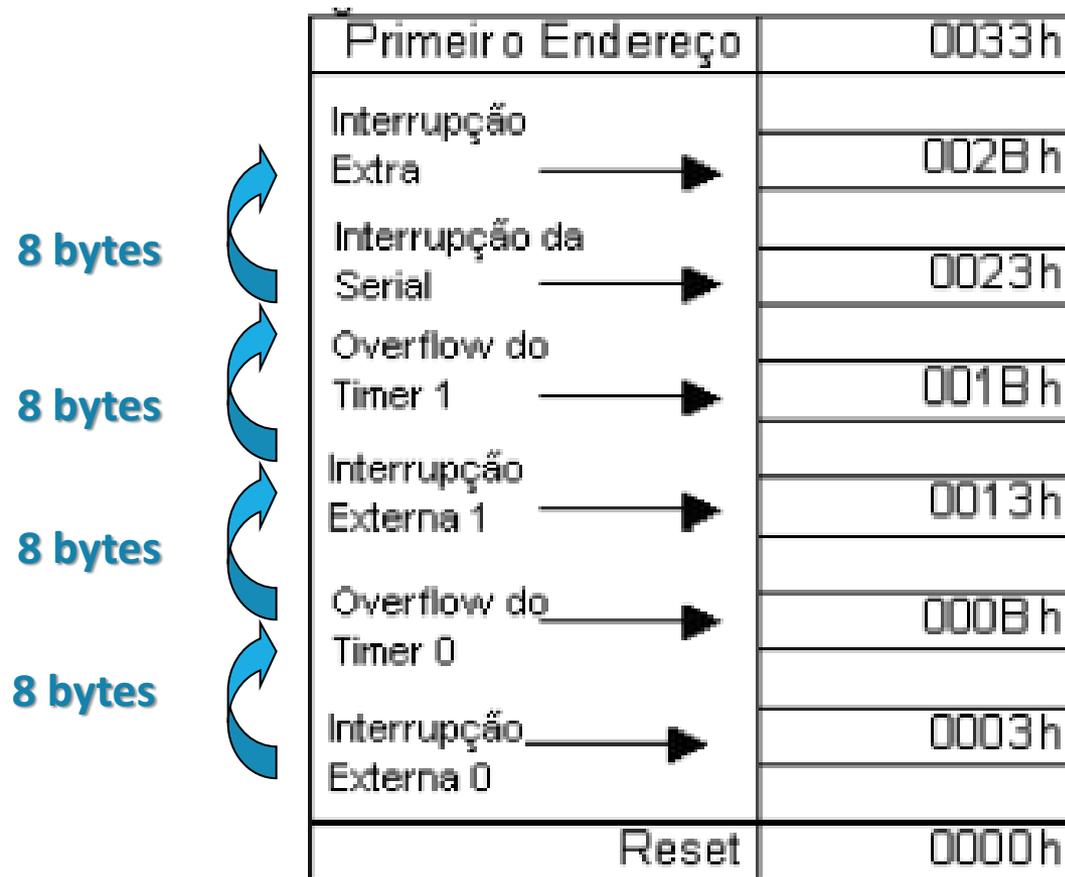
Habilitação das Interrupção

- Cada interrupção pode ser habilitada individualmente ou não e todas podem ser desabilitadas de uma só vez
- Cada interrupção pode ter níveis de prioridade:

ORDEM DE PRIORIDADE

- 
1. Interrupção externa **0** (maior)
 2. Timer **0**
 3. Interrupção externa **1**
 4. Timer **1**
 5. Canal serial (menor)

Endereço das Interrupções



Configuração da Interrupção

O registrador IE configura as Interrupções



IE.7	IE.6	IE.5	IE.4	IE.3	IE.2	IE.1	IE.0
EA	—	ET2	ES	ET1	EX1	ET0	EX0

- EA bit IE.7 : Se EA=0 todas as interrupções são desabilitadas e nenhuma interrupção será reconhecida;
Se EA=1 cada interrupção é individualmente habilitada ou desabilitada setando ou limpando seu bit de habilitação;
- — bit IE.6: Nenhum uso;
- ET2 bit IE.5: habilita ou desabilita *overflow* do Timer 2 (ou interrupção para o 8052);
- ES bit IE.4: ou desabilita a interrupção da porta serial;
- ET1 bit IE.3: habilita ou desabilita interrupção de *overflow* do Timer 1;
- EX1 bit IE.2: habilita ou desabilita Interrupção externa 1;
- ET0 bit IE.1: habilita ou desabilita interrupção de *overflow* do Timer 0;
- EX0 bit IE.0: habilita ou desabilita Interrupção externa 0;

Configuração da Interrupção

O registrador IP define o nível de prioridade
↓
das Interrupções

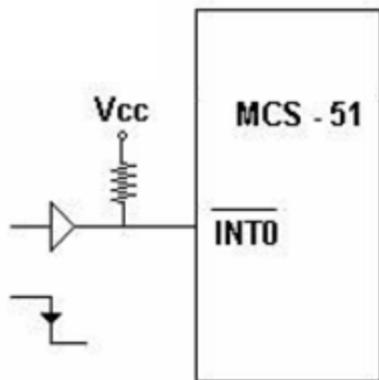
IP.7	IP.6	IP.5	IP.4	IP.3	IP.2	IP.1	IP.0
—	—	PT2	PS	PT1	PX1	PT0	PX0

- — bit IE.7 e IE6: Nenhum uso;
- PT2 bit IE.5: define o nível de prioridade da interrupção do Timer 2 (somente para o 8052);
- PS bit IE.4: define o nível de prioridade da interrupção da porta serial;
- PT1 bit IE.3: define o nível de prioridade da interrupção do Timer 1;
- PX1 bit IE.2: define o nível de prioridade da Interrupção externa 1;
- PT0 bit IE.1: define o nível de prioridade da interrupção do Timer 0;
- PX0 bit IE.0: define o nível de prioridade da Interrupção externa 0;

Configuração da Interrupção

O Bit EA do registrador IE : habilita o uso de interrupções

Exemplo: Programação da Interrupção Externa 0 sensível à borda de



IE0 → 0003h

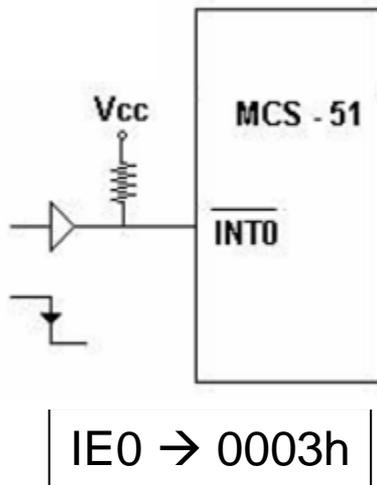
Programa principal:

```
ORG 100H      ; Origem do Programa fora da área de Interrupções

SETB EA      ; Habilita o uso de Interrupções
SETB IE.0    ; Habilita a Interrupção Externa 0
SETB IT0     ; Estabelece que deve ser sensível a descida de borda
...
...          ; Comandos do Programa Principal
...
END
```

Configuração da Interrupção

Exemplo: Programação da Interrupção Externa 0 sensível à borda de descida



Sub-rotina de Atendimento da Interrupção:

```
ORG 0003h      ; Sub-rotina de Atendimento da Interrupção Externa 0.

CLR EA         ; Desabilita as Interrupções para evitar Interrupção da
               ; Interrupção
PUSH PSW       ; Salva os Flags do Programa Principal na pilha
...
...           ; Comandos da Sub-rotina de Atendimento da Interrupção
...
POP PSW        ; Recupera os Flags do Programa Principal
SETB EA        ; Re-habilita as interrupções antes de voltar ao Programa
               ; Principal
RETI           ; Volta para o Programa Principal
```

Software Real

```
ORG 0
MOV A,#00000001B ;faz acumulador = 00000001
MOV P1, A      ;move acumulador para a Porta 1
LEITURA:
    JNB P3.5, LEFT      ;pula para LEFT se P3.5 = 0, senão próx. linha
RIGHT:
    RR A                ;roda byte do Acumulador para direita
    MOV P1, A          ;move Acumulador para a Porta 1
    ACALL TEMPO        ;gasta tempo
    SJMP LEITURA      ;lê bit P3.5 novamente
LEFT:
    RL A                ;roda byte do Acumulador para esquerda
    MOV P1, A          ;move Acumulador para a Porta 1
    ACALL TEMPO        ;gasta tempo
    SJMP LEITURA      ;lê bit P3.5 novamente
TEMPO:
    xxx                ;sub-rotina para gastar tempo
    RET                ;retorna da sub-rotina
    END                ;fim do programa (compilador)
```

Regras para rótulos :

1. o primeiro caracter do rótulo precisa ter uma letras (não pode ser um número);
2. não podem ser utilizadas palavras reservadas (instruções e diretivas).

Como fica o código no 8051

Code Window (Disassembly)

Addr	Opcodes	ASC	Label	Disassembly	Comments
0000	74 01	01		MOV A,#01h	faz acumulador = 00000001
0002	F5 90	90		MOV P1,A	move acumulador para a Porta 1
0004	30 B5 07	07	LEITURA	JNB P3.5,LEFT	pula para LEFT se P3.5 = 0, senão próx.
0007	03		RIGHT	RR A	roda byte do Acumulador para direita
0008	F5 90	90		MOV P1,A	move Acumulador para a Porta 1
000A	11 15			ACALL TEMPO	gasta tempo
000C	80 F6	F6		SJMP LEITURA	lê bit P3.5 novamente
000E	23		LEFT	RL A	roda byte do Acumulador para esquerda
000F	F5 90	90		MOV P1,A	move Acumulador para a Porta 1
0011	11 15			ACALL TEMPO	gasta tempo
0013	80 EF	EF	(EXT1 INT)	SJMP LEITURA	lê bit P3.5 novamente
0015	22		TEMPO	RET	
0016	00			NOP	
0017	00			NOP	

PORTAS de I/O do 8051

O 8051 possui 4 portas de I/O (P0, P1, P2 e P3) de 8 bits cada :

- Quando o microprocessador recebe um RESET todas as portas são configuradas como saída;
- Para usar estas portas P0 como entrada é preciso ligar resistores pull up aos pinos e programar.
- Para usar as portas P1 e P2 como entrada **não** precisa ligar resistores pull up, mas é preciso programar.
- A porta P3 não precisa de resistor pull up para funcionar como entrada, mas geralmente ela é utilizada para suas funções específicas.

PORTAS de I/O do 8051

Exemplo de programa para configurar a porta como entrada:

Com os resistores pull-up conectados a P0, se desejar utiliza-la como entrada é necessário programa-la inicialmente escrevendo 1 para todos os bits. Por exemplo:

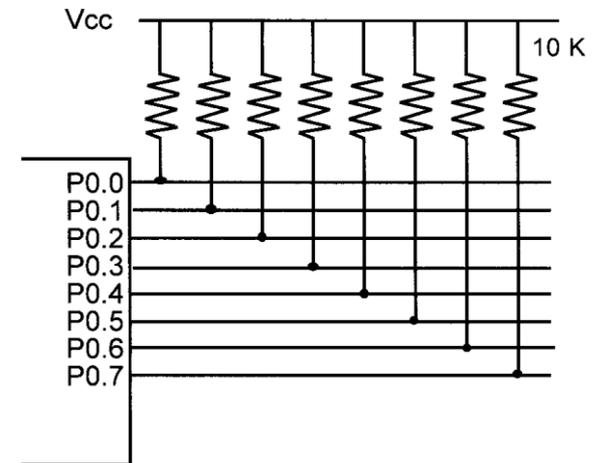
```
MOV A, #0FFH ; A = FFH
```

```
MOV P0, A ; torna P0 entrada
```

```
VOLTA: MOV A, P0 ; pega dados da P0
```

```
MOV P1, A ; envia-os para a P1
```

```
SJMP VOLTA ; retorna
```



FIM