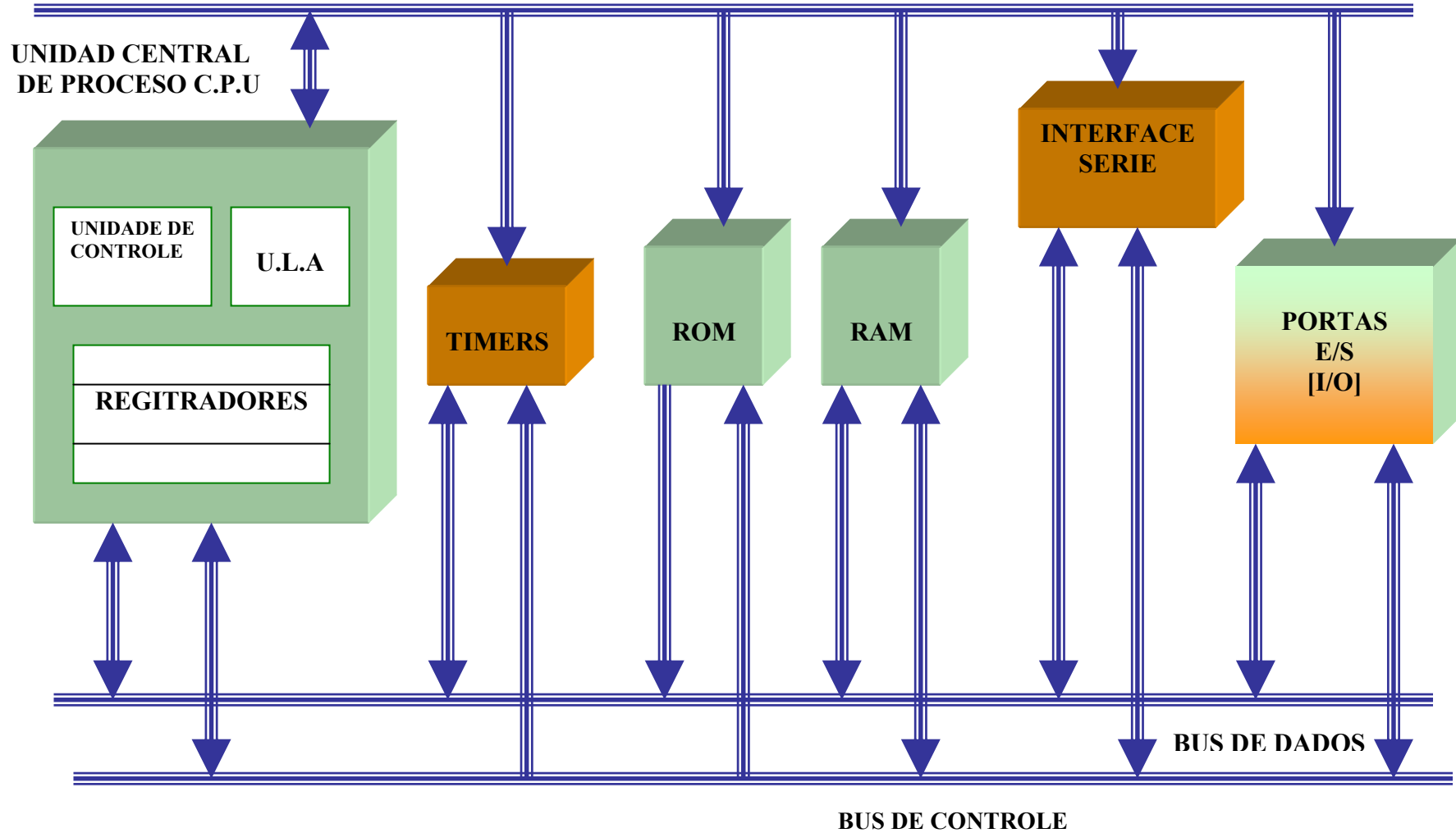


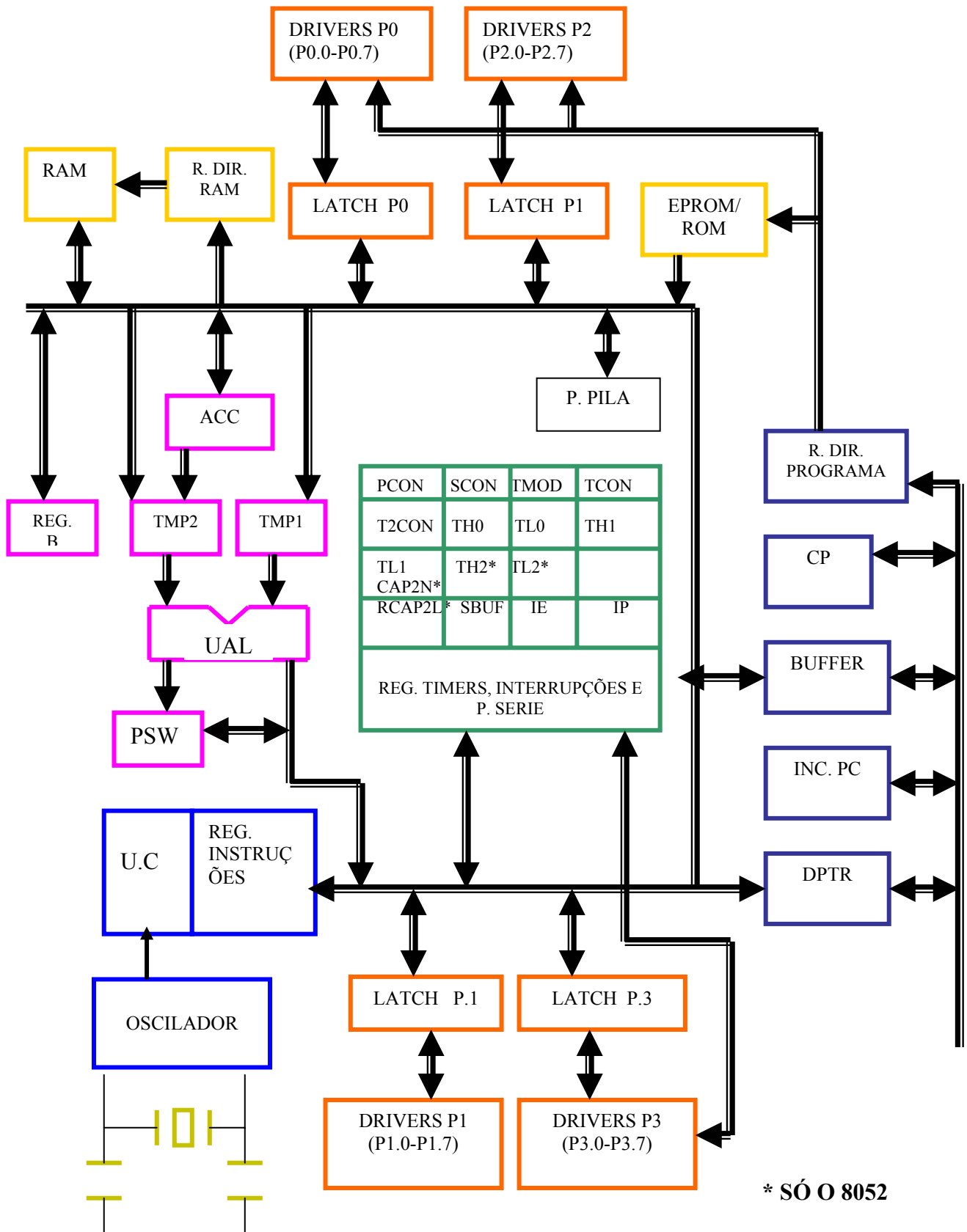
DIAGRAMA DE BLOCOS DE UM MICROCONTROLADOR

BARRAMENTO [BUS] DE ENDEREÇOS



ARQUITETURA DO MICROCONTROLADOR

8051



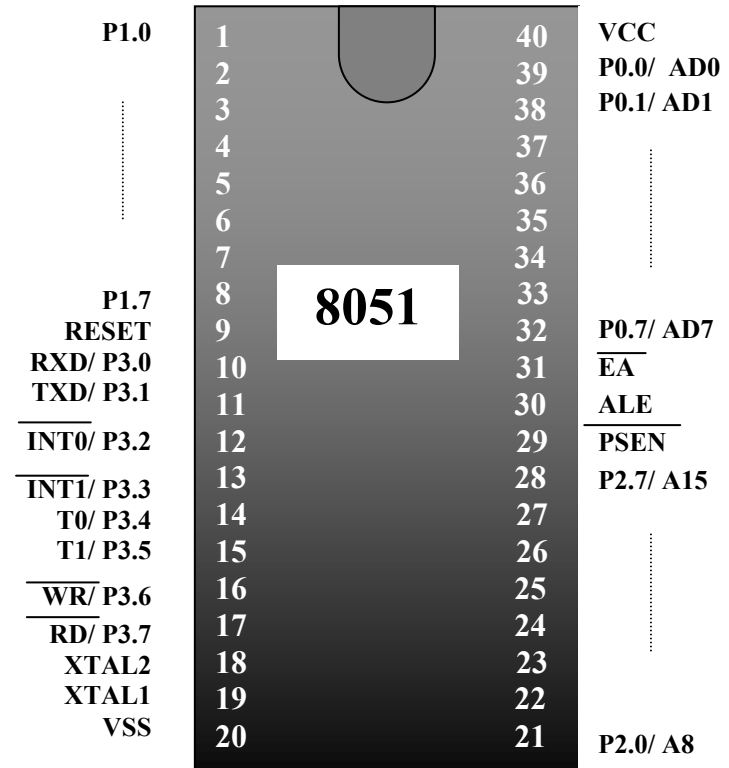
* SÓ O 8052

O MICROCONTROLADOR 8051

Características

- ❖ µC de 8 bits.
- ❖ RAM usuário de 128bytes (256 para o 8052).
- ❖ RAM SFR de 128bytes (128 para o 8052).
- ❖ 4K ROM (8K para o 8052).
- ❖ 32 linhas E/S (4 portas de 8bits).
- ❖ 1 interface série.
- ❖ 2 timers de 16 bits (3 para o 8052).
- ❖ 1 oscilador interno.
- ❖ 64K de memória para programa externo.
- ❖ 64K para dados externos.
- ❖ Frequência 12 MHz.
- ❖ Alimentação +5v.
- ❖ 111 instruções, de 1 a 3 bytes.
- ❖ Ciclo de instrução de 12 ciclos de relógio.
- ❖ 6 fontes de interrupção (2 externas, 3 timers, 1 série).

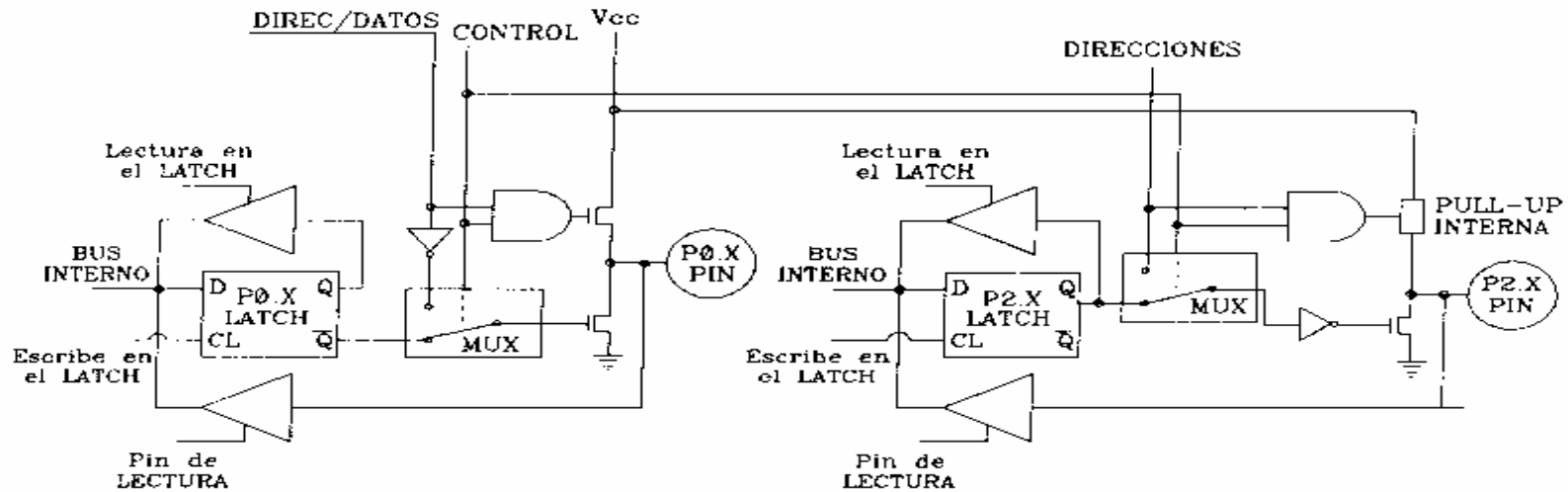
Panagem



DESCRIÇÃO DA PINAJEM DO μ C 8051

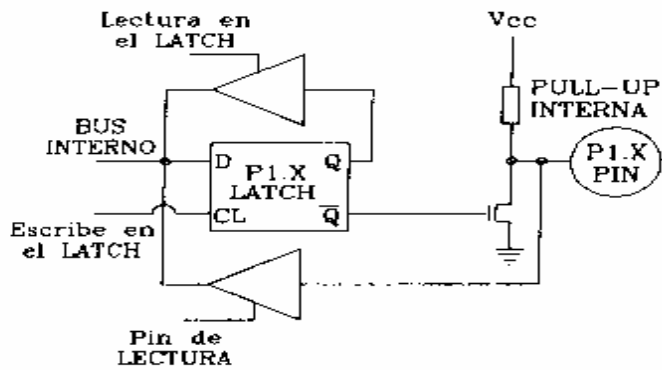
PX.X	Linhas de entrada/saída das portas
RESET	Entrada de inicialização "1"
RXD	Entrada porta série
TXD	Saída da porta série
$\overline{\text{INTX}}$	Entrada para interrupção externa X
TX	Entrada de contagem para o timer X ↓
$\overline{\text{WR}} / \overline{\text{RD}}$	Saídas de escrita e leitura a memória externa
XTAL1-2	Entrada e saída do oscilador
ADX	Bus de dados e parte baixa dos endereços
$\overline{\text{EA}}$	Entrada ativa em "0" que permite executar programas da memória externa ("1" memória interna)
ALE	Saída para multiplexar o bus de endereços (ALE = 0 dados)
$\overline{\text{PSEN}}$	Saída "0" para acionar a memória externa
AX	Parte alta do bus de endereços

HARDWARE DAS PORTAS DO μC 8051

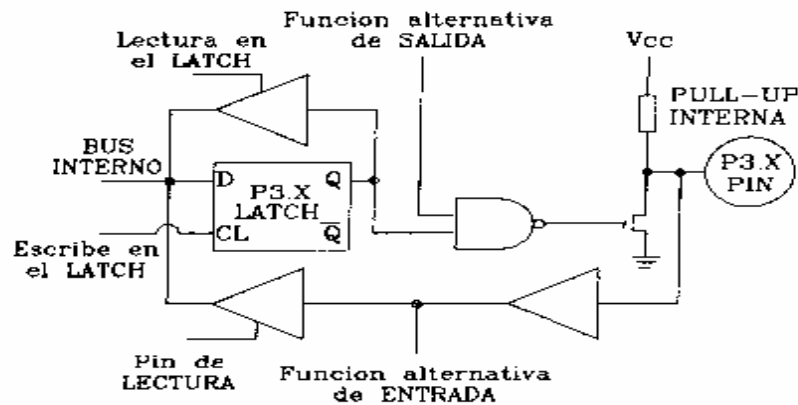


A) Pin del PUERTO 0 (P0.X)

B) Pin del PUERTO 2 (P2.X)



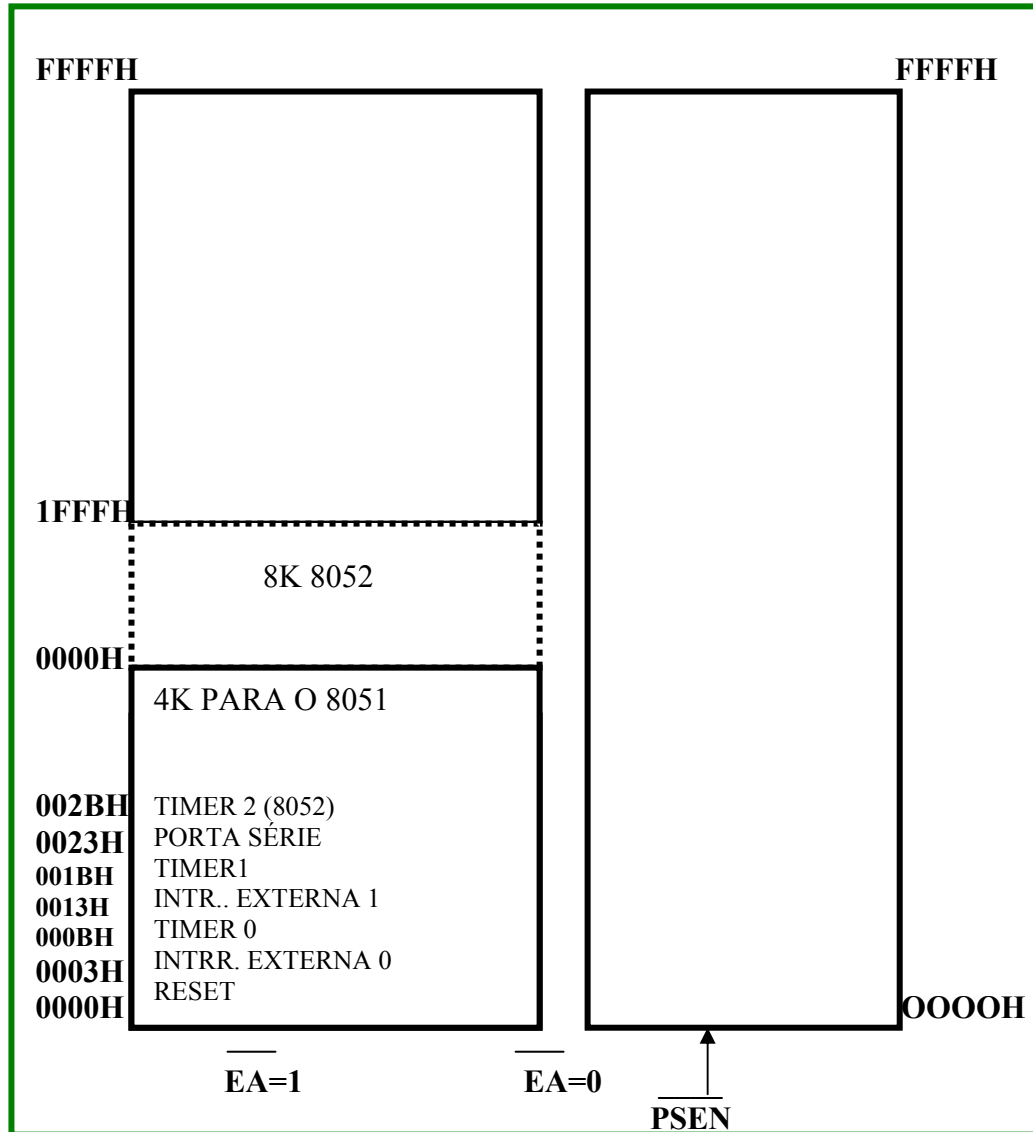
C) Pin del PUERTO 1 (P1.X)



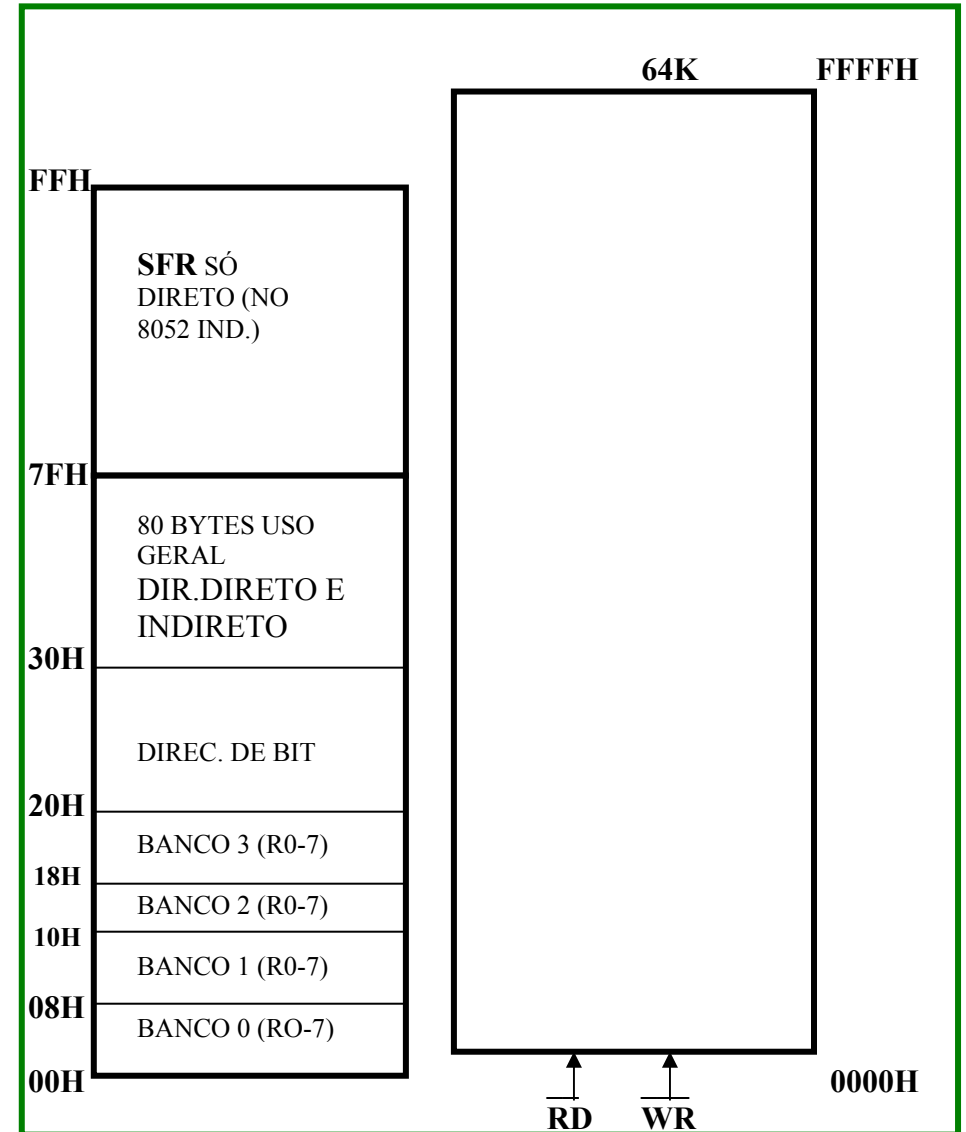
D) Pin del PUERTO 3 (P3.X)

MAPA DE MEMÓRIA DO μ C 8051

MEMÓRIA DE PROGRAMA (64k)

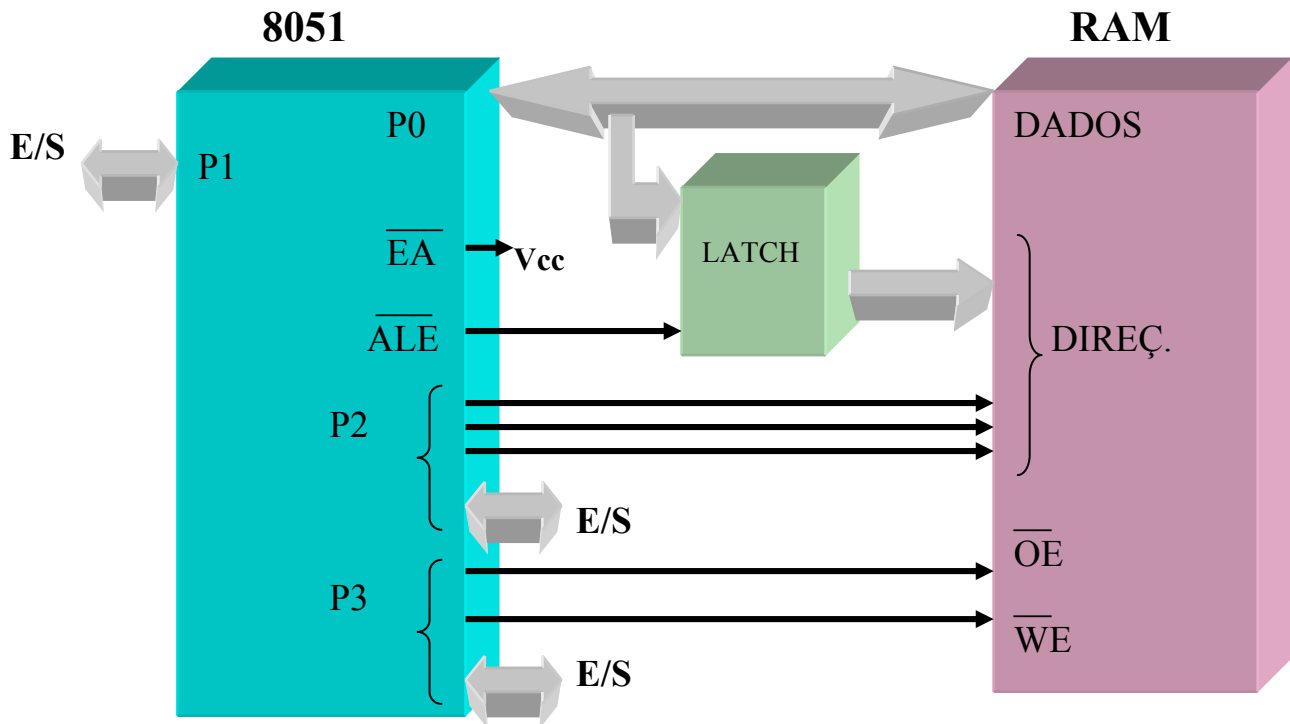


MEMÓRIA DE DADOS



USO DA MEMÓRIA DO μC 8051

MEMÓRIA DE DADOS



REGISTRADORES ESPECIAIS

F8H									FFH
F0H	B								F7H
E8H									EFH
E0H	ACC								E7H
D8H									DFH
D0H	PSW								D7H
C8H	T2CON*		RCAP2L*	RCAP2H*	TL2*	TH2*			CFH
COH									C7H
B8H	IP								BFH
B0H	PS								B7H
A8H	IE								AFH
A0H	P2								A7H
98H	SCON	SBUF							9FH
90H	P1								97H
88H	TCON	TMOD	TL0	TL1	TH0	TH1			8FH
80H	P0	SP	DPL	DPH				PCON	87H

* SÓ PARA O 8251

RESET: ACC=B=PSW=DPTR=TMOD=TCON=THX=TLX=SCON= 00H

SP= 07H

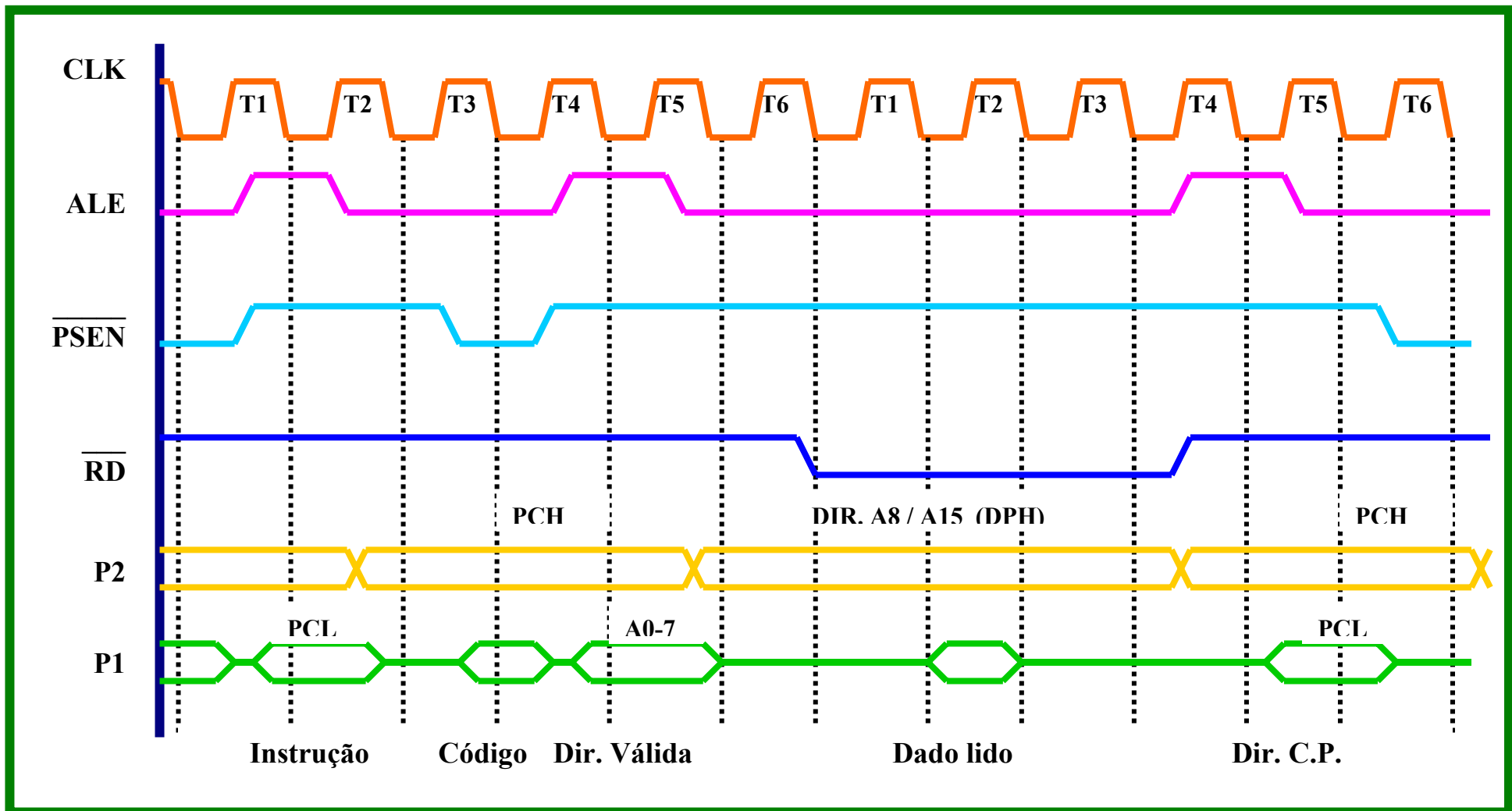
P0-3= 0FFH

IP= XXX00000B

IE= 0XX00000B

PCON= 0XXX0000B

CICLO DE LEITURA NO μ C 8051



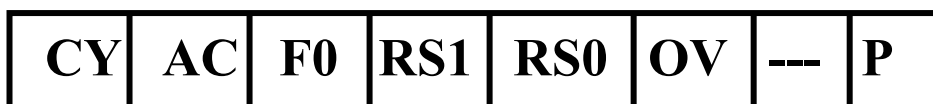
SET DE INSTRUÇÕES (I) DO μ C 8051

ARITMETICAS	
ADD A,Rn	Soma um registro ao acumulador
ADD A,direct	Soma o conteúdo de uma direção ao acumulador
ADD A,@Ri	Soma o conteúdo da direção apontada por Ri (R0 ou R1 de qualquer banco) ao acumulador
ADD a,#dato	Soma o dado imediato ao acumulador
DAC A,Rn	Soma um registro ao acumulador mais o <i>carry</i> (todas as instruções anteriores de soma também existem com o <i>carry</i>)
SUBB a,Rn	Subtrai um registro do acumulador (igual aos anteriores)
INC A	Incrementa o conteúdo do acumulador
INC Rn	Incrementa o conteúdo do registro indicado
INC direct	Incrementa o conteúdo da direção indicada
INC @Ri	Incrementa o conteúdo da direção apontada por Ri
INC DPTR	Incrementa o conteúdo do registro ponteiro DPTR
DEC A	Decrementa o conteúdo do acumulador (todas como o incremento, exceto DPTR)
MUL AB	Multiplica os conteúdos dos registros A e B (em A fica a parte alta e em B a baixa)
DIV AB	Divide os conteúdos dos registros A e B
DA A	Ajusta a decimal o conteúdo do acumulador, sempre que $A \neq 00H$
LÓGICAS	
ANL A,Rn	Operação lógica AND do acumulador com o registro indicado (todas como na somas)
ANL direct,A	Operação lógica AND do conteúdo de uma direção com o acumulador.
ANL direct,#dato	Operação lógica AND do conteúdo de uma direção com um dado imediato
ORL A,Rn	Operação lógica OR do acumulador com um registro (todas iguais a AND)
XRL A,Rn	Operação lógica XOR do acumulador com um registro (todas iguais a AND)
CLR A	Coloca a zero acumulador
CPL A	Complementa bit a bit o acumulador
RL A	Rotaciona à esquerda o acumulador (RLC A , igual com <i>carry</i>)
RR A	Rotaciona à direita o acumulador (RRC A , igual com <i>carry</i>)
SWAP A	Intercambia os bits de maior peso com os de menor peso no acumulador
TRANSFERÊNCIA DE DADOS	
MOV A,Rn	Move o conteúdo do registro indicado para o acumulador (*=também ao contrário)
MOV A,direct	Move o conteúdo do endereço para o acumulador (*, e MOV Rx,direct)
MOV A,Ri	Move o conteúdo do endereço indicado por Ri para o acumulador (*)
MOV A,#dato	Move o dado indicado para o acumulador (* e MOV Rx,#dato)
MOV direct,direct	Move o conteúdo de um endereço para outro (também MOV direct,Rx)
MOV DPTR,#dato16	Move para o ponteiro DPTR um dado imediato de 16 bits
MOVC A,@A+DPTR	Move para o acumulador o conteúdo do endereço de ROM apontado por DPTR+A
MOVC A,@A+PC	Move para o acumulador o conteúdo do endereço de ROM apontada por PC+A
MOVX A,@Ri	Move para o acumulador o conteúdo do endereço de memória externa indicada (*)
MOVX A,@DPTR	Move para o acumulador o conteúdo do endereço de memória externa indicada (*)
PUSH direct	Guarda o conteúdo de um endereço no registro da pilha
POP direct	Recupera o conteúdo guardado em um endereço da pilha
XCH A,Rn	Intercambia os conteúdos (também com <i>direct</i> e <i>Ri</i>)
XCHD A,@Ri	Intercambia o conteúdo da parte baixa do endereço por Ri com A

SET DE INSTRUÇÕES (II) DO μ C 8051

BOOLEANAS	
CLR C	Coloca a zero o bit de <i>carry</i> do registro PSW
CLR bit	Coloca a zero o bit indicado (CLR P2.7, CLR A.0, CLR OV)
SETB C	Coloca a um o bit de <i>carry</i>
SETB bit	Coloca a um o bit indicado
CPL C	Complementa o bit de <i>carry</i>
CPL bit	Complementa o bit indicado
ANL C,bit	Operação lógica AND entre <i>carry</i> e um bit
ANL C,/bit	Operação lógica AND entre <i>carry</i> e o complemento a um de um bit (estas duas instruções também existem com ORL y MOV)
DE CONTROLE DE PROGRAMA	
ACALL direcc11	Chamada absoluta de sub-rotina (para 2K, na memória interna)
LCALL direcc16	Chamada larga de sub-rotina (+de 2K, memória externa)
RET	Retorno de sub-rotina
RETI	Retorno de interrupção
AJMP direcc11	Salto absoluto incondicional (2K)
LJMP direcc16	Salto largo incondicional (+2K)
SJMP rel	Salto relativo (+-128bytes da instrução)
JMP @A+DPTR	Salto indireto relativo al DPTR
JZ rel	Salto se o conteúdo do acumulador é zero
JNZ rel	Salto se o conteúdo do acumulador é diferente de zero
JC rel	Salto se o <i>carry</i> é um (**= ver os bits de PSW)
JNC rel0	Salto se o <i>carry</i> é zero (**)
JB bit, rel	Salto se o bit indicado é um
JNB bit, rel	Salto se o bit indicado é zero
JBC bit,rel	Salto se o bit indicado é um; e limpa o bit <i>carry</i>
CJNE A,direct,rel	Compara e salta para um endereço relativo se o acumulador não é igual ao conteúdo de um endereço (existe também com Rn e Ri)
DJNZ Rn,rel	Decrementa e salta para um endereço relativo se não é zero o conteúdo do registro
DJNZ direct,rel	O mesmo porem com o conteúdo de um endereço
NOP	Não realiza nenhuma operação (se perde um ciclo)

O REGISTRO PSW



CY.... bit de *carry*.

AC.... indicador de *carry* no quarto bit.

F0.... sem função, disponível para usuário.

RS1 y **RS2**... seleção dos bancos de registros (00-banco 0, 11-banco3).

OV.... indicador de transbordamento [*overflow*].

P..... indicador de paridade.

INTERRUPÇÕES DO μ C 8051

- ❖ 5 níveis de Interrupção (6 para o 8052):
 - 2 externas: INT0, INT1.
 - 2 Timers: T0, T1. (T2 para o 8052).
 - 1 série: RXD (TXD).
- ❖ As interrupções são escrutadas em cada ciclo de máquina e são atendidas ao finalizar a instrução em curso, exceto no caso de estar atendendo-se uma de ,maior prioridade.
- ❖ Registro de habilitação de Interrupção **IE** (0A8H de SFR):



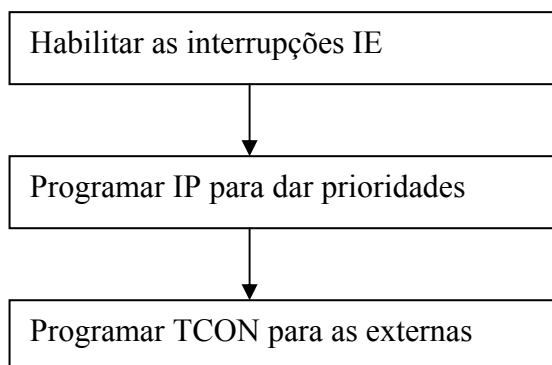
EX0	“0” “ \downarrow ” “ (TCON) habilita a externa INT0	0003H	“0” auto
ET0	“1” habilita o Timer0	000BH	“0” auto
EX1	“0” ” “ \downarrow ” “ (TCON) habilita a externa INT1	0013H	“0” auto
ET1	“1” habilita o Timer1	001BH	“0” auto
ES	“1” habilita a porta serie	0023H	“0” softw
ET2*	“1” habilita o Timer2	002BH	“0”softw
EA	“1” habilita todas as interrupções a “1” “0” não reconhece nenhuma interrupção.		

- ❖ Registro de Níveis de Prioridade **IP** (0B8 de SFR):



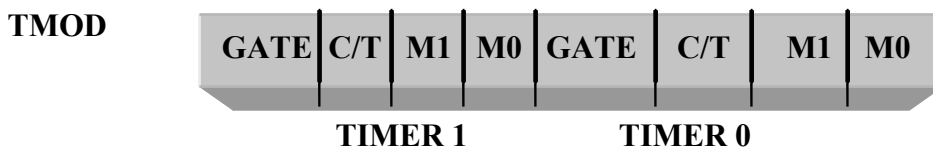
PX0	“1” alta prioridade INT0	Ordem normal de Prioridade INT0 (maior) TIMER0 INT1 TIMER1 P. SERIE
PT0	“1” alta prioridade TIMER0	
PX1	“1” alta prioridade INT1	
PT1	“1” alta prioridade Timer1	
PS	“1” alta prioridade porta serie	
PT2*	“1” alta prioridade Timer2	

- ❖ Processo de Software para as interrupções:



TIMERS / COUNTERS DO μC 8051

- ❖ Cada Timer está dotado de um contador de 16 bits. São acessíveis em forma de registros de 8 bits:
 - TH0 -TL0.... Timer 0.
 - TH1 -TL1.... Timer 1.
- ❖ Configuram-se mediante os registros **TMOD** e **TCON**.



GATE	"1" habilita Timer x, se $\text{INTx}="1"$ e $\text{TRx}="1"$ (TCON). "0" habilita Timerx, se $\text{TRx}="1"$
$\text{C}/\overline{\text{T}}$	"0" o timer atua como temporizador. "1" o timer atua como contador de eventos da entrada Tx. ↓
M1 M0	0 0...contador 13 bits. Modo 0. tempo= $(2^{13}-\text{valor de carga})/\text{freq. Oscilador Hz}$. O valor de carga vezes dado por TLX(5bits)+THX(8bits). 0 1...contador 16 bits. Modo 1. Igual al modo 0 porém com 8 bits de TLX. 1 0...contador 8 bits. THx se carrega em TLx deve ser zero. Modo 2 (auto-recarga). tempo= $(256-\text{TH1})/\text{freq. Oscilador Hz}$. 1 1...Timer0: TL0 atua como contador de 8 bits controlado por Timer 0. TH0 atua como temporizador de 8 bits controlado por Timer 1 (Gate, C/T, TR0, INT0 y TF0). Timer1: permanece desabilitado. Modo 3, igual al modo 2 (autocarga).

- Como temporizador o contador se incrementa automaticamente com cada ciclo máquina.
- Como contador se incrementa ao produzir-se uma transição negativa na entrada do timer.



IT0	Indicador de evento para o disparo de INT0. "0" interrupção por nível "0". "1" interrupção por flanco (descida do sinal) ↓
IE0	Indicador de transição da fonte externa INT0. Coloca-se a "1" ao detectar uma transição no terminal INT0 ↓ Ao processar a informação volta a zero.
IT1	Igual a IT0, porém para INT1.
IE1	Igual a IE0 porém para INT1.
TR0	Bit de inicialização do Timer 0. Programa-se por software "1" arranca, "0" para.
TF0	Indicador de transbordamento do Timer 0. Coloca-se a "1" ao atingir a contagem zero. Está-se permitida uma interrupção, este indicador se coloca a "0" ao executar-se a sub-rotina de interrupção.
TR1	Igual que TR0 para o Timer 1.
TF1	Igual a TF0 para o Timer 1.

MODO DE BAIXO CONSUMO NO μ C 8051

❖ Existem dois modos: Baixa Potência, Irregular.

❖ Seleciona-se mediante o registro PCON.

PCON



IDL	Bit de controle do modo irregular “1”.
PD	Bit de controle do modo de baixa Potencia “1”.
GF0	Bit de uso geral. <i>Flag</i> que pode indicar que a interrupção ocorreu no modo irregular.
GF1	Bit de uso geral. <i>Flag</i> que pode indicar que a interrupção ocorreu no modo irregular.
SMOD	Bit de controle de Velocidade (transmissão) da porta serie. “1” duplica a Velocidade do Timer1.

❖ Modo Irregular:

- O sinal de relógio não aciona a CPU.
- O sinal de relógio controla os Timers, a porta série e interrupções.
- Os estados da CPU são mantidos.
- Os terminais das Portas são mantidos em seus níveis atuais.
- ALE e PSEN, ficam a nível alto.
- Sai-se mediante uma interrupção ou um *Reset*.

❖ Modo Baixa Potência:

- O sinal de relógio fica parado.
- Todas as funções estão paradas.
- Os valores de RAM, SFR e terminais das portas são mantidos.
- ALE e PSEN, estarão em nível baixo.
- A tensão de alimentação pode ser reduzida até 2V, uma vez neste modo.
- Sai-se mediante um *Reset*.

COMUNICAÇÃO SÉRIE NO μC 8051

- ❖ Realiza-se utilizando os terminais **RXD** (entrada) e **TXD** (saída).
- ❖ Funciona em *full duplex*.
- ❖ Os registros de recepção e transmissão são o mesmo e está no endereço **99H** da RAM interna e se denomina **SBUF**. Quando se transmite se escreve e ao receber se lê.
- ❖ O controle da porta se realiza mediante o registro do endereço **98H**, chamado **SCON** :

SCON



RI	Indicador de recepção. “1” quando recebe o 8ºbit em modo zero, assim como o bit de <i>stop</i> nos outros modos. Colocará-se a zero via programa.
TI	Indicador de transmissão. “1” quando se transmite o 8ºbit em modo zero, assim como o bit de <i>stop</i> nos outros modos. Colocará-se a zero via programa.
RB8	É o 9ºbit recebido no modo 2 ou 3. No modo 1, se SM2=0 então RB8 é o bit de <i>stop</i> recebido.
TB8	É o 9ºbit a transmitir no modo 2 ou 3.
REN	“1” habilita a recepção serie.
SM2	“1” há possibilidade da comunicação multi-processor nos modos 2 e 3.
SM1-SM0	Selecionam os modos de funcionamento:

- ❖ Modos de funcionamento:

SM1-SM0

00	Modo 0 , registro de deslocamento. $F = \text{fos}/12$. Os dados entram e saem por RXD. TXD é o relógio para o deslocamento. O bit menos significativo é o primeiro a ser transmitido (8bits). A transmissão começa ao carregar SBUF e acaba quando TI=“1”. A recepção começa estando REN=“1” e RI=“0”. Acaba ao colocar RI=“1”.
01	Modo 1 , UART de 8 bits. Frequência variável (Timer 1). Os dados se transmitem por TXD e se recebe por RXD. 10 bits= 1 arranque (“0”)+ 8 dados + 1 parada (“1”).
10	Modo 2 , UART de 9 bits. $F = \text{fos}/32$ ó $F = \text{fos}/64$. Os dados se transmitem por TXD e se recebe por RXD. 11 bits= 1 arranque (“0”)+ 8 dados + 1 bit programable+ 1 parada (“1”).
11	Modo 3 , UART de 9 bits. Frequência variável (Timer 1). Os dados se transmitem por TXD e se recebe por RXD. 11 bits= 1 arranque (“0”)+ 8 dados + 1 bit programable+ 1 parada (“1”).

PROGRAMANDO O μ C 8051

DIRETIVAS

- ORG** define a origem do programa vem acompanhado de um endereço
- END** define o final do programa
- EQU** associa um valor a uma constante
- ;** define um texto
- :** define uma etiqueta [*LABEL*]

EXEMPLO DE PROGRAMAÇÃO

```
1 $TITLE ( OPERAÇÃO COM MEMÓRIA )
2 ;
3 ;
4 ORG          0000H      ;ORIGEM DA MEMÓRIA DE PROGRAMA
5 AJMP        100H      ;SALTA PARA UMA POSIÇÃO NÃO OCUPADA
6 ORG          100H
7 ;
8 ;
9             CLR        C          ;LIMPA O BIT DE CARRY
10            MOV        70H,#05H   ;CARREGA A POSIÇÃO 70 DA RAM COM O VALOR 05H
11            MOV        A,39H      ;CARREGA O ACUMULADOR COM O CONTEÚDO 39 DA RAM
12            SUBB       A,38H      ;SUBTRAI DO ACUMULADOR O CONTEÚDO 38 DA RAM
13            JC         SOMA       ;SALTA SE EXISTE CARRY NA SUBTRAÇÃO
14            MOV        70H,A      ;CARREGA O ENDEREÇO 70 DA RAM COM O RESULTADO
15            JMP        FIM        ;FINALIZA O PROGRAMA
16 SOMA:
17            MOV        A,39H      ;MOVE PARA O ACUMULADOR O CONTEÚDO DA
18                               ;POSIÇÃO 39 DA RAM
19            ADD        A,70H      ;SOMA O CONTEÚDO DO ACUMULADOR COM A
20                               ;POSIÇÃO 70 DA RAM
21            MOV        70H,A      ;CARREGA O ENDEREÇO 70 DA RAM COM A SOMA
22 FIM:
23            END
```

Prof. Corradi

www.corradi.junior.nom.br