

# Microprocessadores e Aplicações

## *Acetatos de apoio às aulas teóricas*

Ana Cristina Lopes  
Dep. Engenharia Electrotécnica  
<http://orion.ipt.pt> [anacris@ipt.pt](mailto:anacris@ipt.pt)

1. Fundamentos Teóricos;
  - (a) Introdução;
  - (b) Arquitectura básica de um microcomputador;
  - (c) Organização interna de um microprocessador;
  
2. Introdução aos Microcontroladores da família MCS51;
  - (a) Arquitectura básica;
  - (b) Organização da memória;
  - (c) SFRs;
  - (d) Modos de endereçamento;
  - (e) Temporização.

- Introdução aos Microcontroladores

Fundamentos Teóricos

- **Introdução**

- Arquitectura Básica

- Organização Interna

Introdução à família MCS51

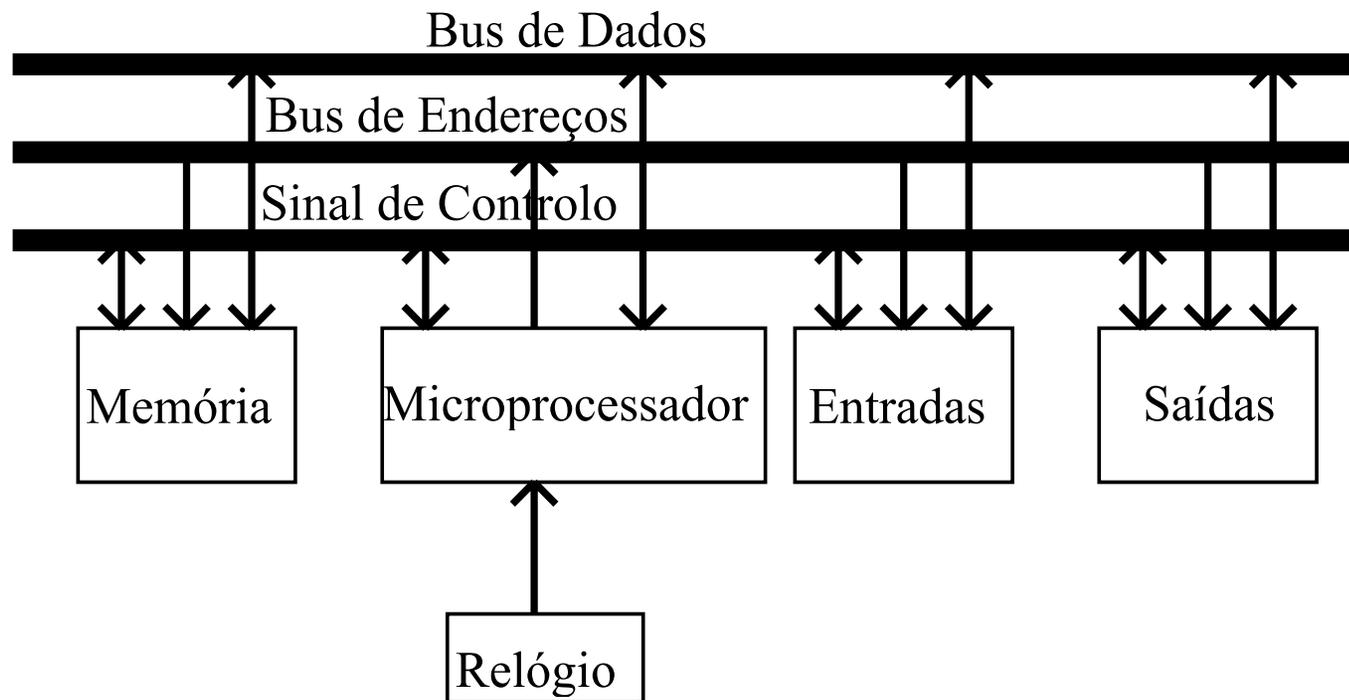
## Microcontrolador vs Microprocessador

Um microcontrolador difere de um microprocessador nos seguintes aspectos:

- **Funcionalidade** - Para que um microprocessador possa ser usado, outros componentes devem-lhe ser adicionados, tais como memória e componentes para receber e enviar dados (microprocessador=coração do computador).
- O microcontrolador foi projectado de modo a ter todos os periféricos necessários integrados nele.

Os Microcontroladores são dispositivos dedicados embebidos em determinadas aplicações. Em suma, um microcontrolador é um microprocessador (CPU) integrado com dispositivos de memória, unidades de E/S e outros blocos de funções especiais, tais como: comunicação série, unidade de temporização (temporizadores e contadores), watchdog, conversor analógico-digital, etc.

A Figura seguinte mostra o diagrama de blocos de um sistema de Microcomputadores.



Existem dois tipos de memória essenciais:

## 1. Memória Primária:

- Utilizada para armazenar dados activos e instruções;
- ROM (*Read Only memory*) - para armazenar informação que não se altera ao longo do ciclo de vida do sistema (Exemplo: PC BIOS (*Basic I/O System*));
- RAM (*Random Access Memory*) - para armazenar informação que será processada pelo microprocessador e que será alterada frequentemente.

## 2. Memória Secundária:

- Pode ou não existir;
- Aplicada quando se requer uma grande capacidade de memória;
- Menor custo/bit e mais lenta;
- Exemplos: diskettes, discos duros, cd, etc.

## Unidade de E/S:

- Uma unidade de entradas designa-se como um porto da CPU que recebe informação do mundo exterior.
- Uma unidade de saídas designa-se como um porto da CPU que envia informação para o mundo exterior.

## Microprocessador:

O Microprocessador é a Unidade Central de Processamento (CPU) do sistema. Este é constituído pela Unidade de Aritmética e Lógica (ALU, registos e circuitos de controlo). Estes items serão discutidos mais adiante.

## BUS:

Os blocos do microcomputador estão ligados por um grupo de "condutores" designado por **BUS**/barramento, o qual consiste no meio de comunicação entre os vários elementos do sistema. Os "condutores" são agrupados de acordo com a sua função.

## 1. Bus de endereços (*Address Bus*):

- Especifica a zona de memória ou porto de E/S para o qual o microprocessador pretende transferir informação (dados ou instruções).
- o N.º de linhas do Bus de endereço determina o N.º de zonas de memória que podem ser endereçadas pelo microprocessador. Exemplo: processador INTEL 8088 possui 20 linhas de endereço  $\Rightarrow 2^{20} = 1048576 = 1024KB = 1M$  byte de zonas de memória endereçadas.

## 2. Bus de dados (*Data Bus*):

- Transferência bidireccional de informação entre o microprocessador e outras partes do sistema, tais como: memória, portos de E/S, etc.

## 3. Bus de controlo (*Control Bus*):

- É através deste Bus que o microprocessador controla as operações das diversas unidades do sistema.
- Exemplos: especificar se a CPU pretende receber ou enviar informação; controlar a função de inicialização, etc.

## Relógio:

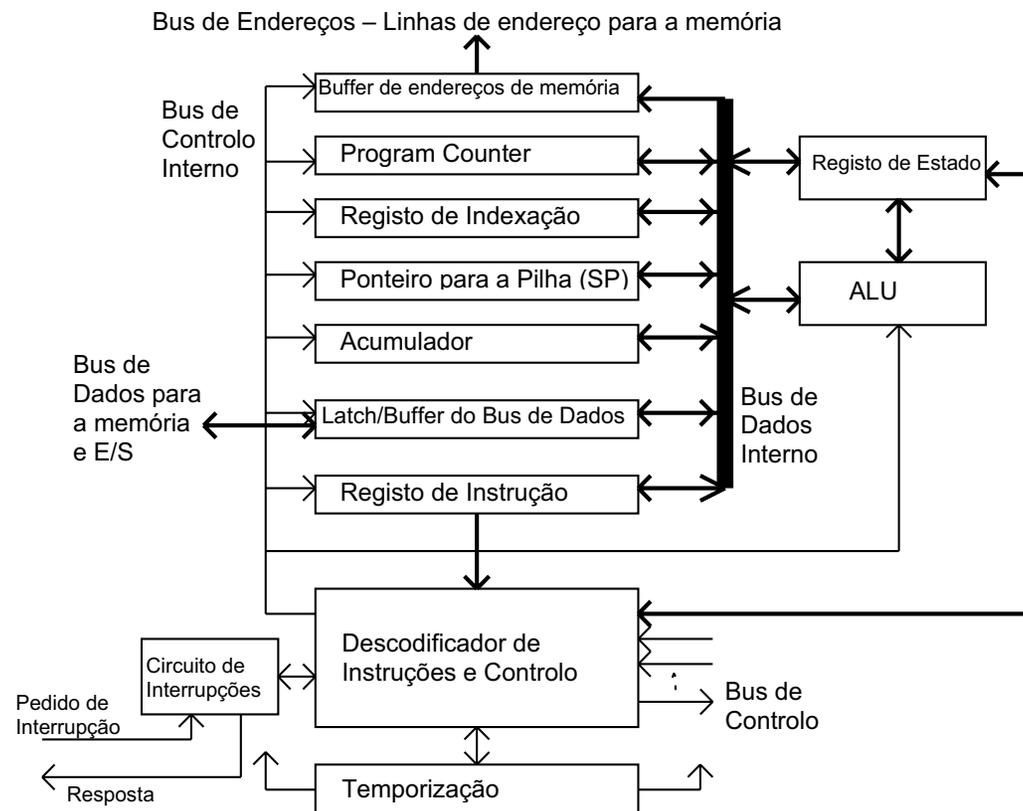
Os microprocessadores necessitam do relógio para operarem, dado que:

- os microprocessadores são circuitos sequenciais síncronos;
- alguns microprocessadores utilizam células de memória dinâmicas para implementar os seus registos. Estes têm de ser refrescados periodicamente por forma a que o seu conteúdo seja retido.

Como é que o sinal de relógio pode ser gerado?

1. Por um gerador externo de sinal de relógio;
2. Implementado dentro da CPU, o que requer um cristal externo ou um circuito R-C.

A Figura seguinte mostra o diagrama de blocos de um sistema de Microcomputadores.



● Introdução aos Microcontroladores

Fundamentos Teóricos

● Introdução

● Arquitectura Básica

● Organização Interna

Introdução à família MCS51

## Program Counter (PC):

- Guarda o endereço da próxima instrução;
- Após a execução de cada instrução, o PC é automaticamente incrementado de modo a apontar para a próxima instrução;
- Regra geral largura do Bus de endereços = largura do PC.

## Buffer/Latch de endereços:

- Guarda o endereço da zona de memória ou portos de E/S a serem acedidos;
- Fornece a corrente necessária para activar o Bus de endereços;
- Regra geral largura do Bus de endereços = largura do buffer.

**Pilha (*Stack*):** É uma zona de memória da RAM interna do processador reservada para armazenamento temporário de dados e devolve o endereço de memória do programa após retorno de uma subrotina. Para além disso:

- Armazena os conteúdos dos registos internos durante a execução das interrupções;
- A pilha tem uma dimensão variável, dependendo da informação que lá é colocada;
- A pilha é do tipo LIFO (*Last In First Out*), sendo os dados da pilha acedidos pelo topo.

**Ponteiro de Pilha (*Stack Pointer (SP)*):**

- É utilizado para apontar a pilha de registos mencionada;
- É incrementado quando se executam instruções como a LCALL e PUSH (instrução que coloca os dados na pilha);
- É decrementado quando se executam instruções RET, RETI ou POP (instrução que retira os dados da pilha).

# Exemplo PUSH e POP

● Introdução aos Microcontroladores

Fundamentos Teóricos

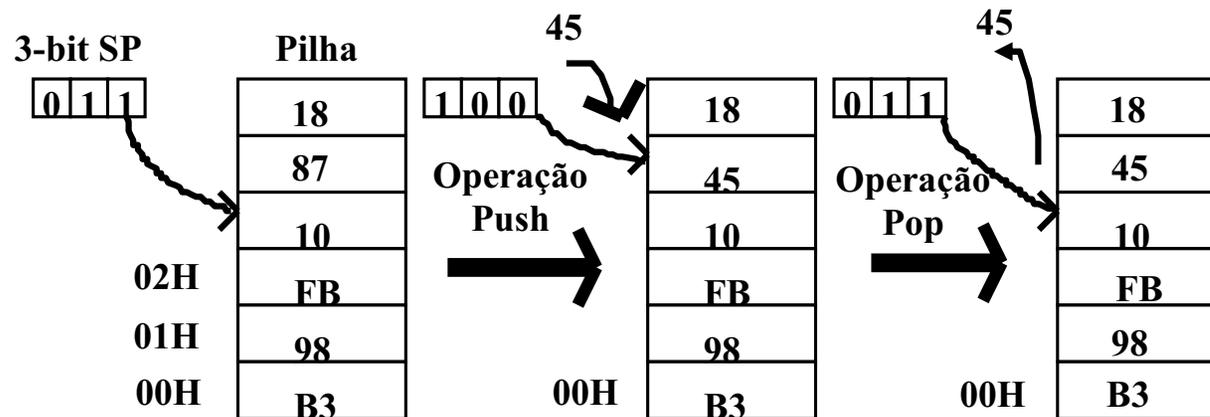
● Introdução

● Arquitetura Básica

● Organização Interna

Introdução à família MCS51

A Figura seguinte mostra como se processam as operações PUSH e POP.

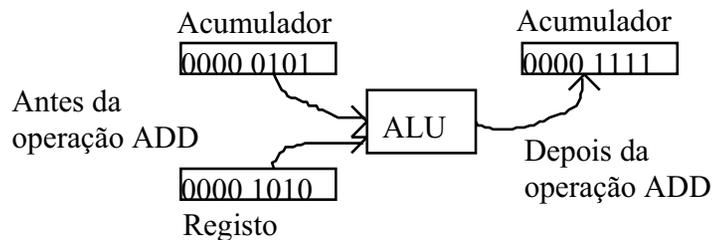


- Inicialmente SP=011B;
- O SP é incrementado de 1 antes da instrução PUSH, ou seja SP=100B;
- O dado 45H é armazenado no endereço 100B;
- Na operação POP, o dado endereçado por SP é lido; neste caso 45H é lido do endereço 100B.
- O SP é decrementado de 1, ou seja SP volta a ser 011B.

## Unidade de Aritmética e Lógica (ALU):

- Circuito combinacional que realiza as operações aritméticas e lógicas nos dados armazenados no acumulador;
- Os resultados da operação executada pela ALU são colocados no acumulador através do BUS interno;
- Operações típicas: ADD, Shift/Rotate, compare, INC, DEC AND, OR XOR, etc.

**Acumulador:** Os resultados das operações executadas pela ALU são colocados no Acumulador, tal como mostra a figura seguinte.



- O acumulador também serve como entrada da ALU;

## Registo de Instruções:

- Guarda a instrução que está a ser descodificada e executada;
- Os conteúdos deste registo são alimentados para o descodificador de instruções. Este último é responsável por receber o OP CODE e gerar saídas que permitem que os circuitos de controlo executem a operação desejada e na sequência correcta.

## Circuito de Interrupções:

- Este circuito inclui um flip-flop que é activado por um sinal de pedido de interrupção, sendo este último requisitado por um dispositivo de entrada ou saída. Este sinal de interrupção faz com que a CPU deixe a sequência do programa que está a ser corrido e inicie a execução de uma série de instruções que fazem parte da rotina de interrupção.
- Uma das aplicações das interrupções é reduzir o tempo de espera da CPU. Sem a possibilidade de interrupções a CPU encontrar-se-ia muitas vezes à espera da informação dos dispositivos de E/S mais lentos, por exemplo: teclado. Com a possibilidade de interrupções, os dispositivos de E/S, mal estejam prontos, enviam um pedido de interrupção à CPU, que assim fica livre para desempenhar outras tarefas.

**Registo de Estado:** Durante a execução de algumas operações aritméticas e lógicas podem aparecer algumas condições que afectam a forma como o programa se encontra a ser executada. É necessário armazenar essas condições num registo de estado (conjunto de flip-flops).

**Flags:** O registo de estado é composto por flags que são colocadas a 0 ou 1 consoante a condição gerada pela última operação da ALU. De seguida apresentam-se algumas das flags que se podem encontrar nos registos de estado:

- Zero - fica a 1 se o resultado da última operação aritmética ou lógica for zero (utilizada para encontrar um determinado valor);
- Sinal (*sign*) - fica a 1 se o resultado da última operação aritmética for negativo;
- Transporte (*carry*) - Armazena o transporte das operações de adição e subtracção;
- Transbordo (*overflow*) - fica a 1 se existir um transbordo após uma operação aritmética. Exemplo: para uma ALU de 8 bits, esta flag fica a 1 se o resultado for superior a +127 ou inferior a -128.

O 8051 foi o primeiro microcontrolador da família MCS51 introduzido pela INTEL. A Figura seguinte mostra as características dos vários microcontroladores da família MCS51.

Versão ROM	Versão EPROM	Versão ROMless	ROM (bytes)	RAM (bytes)	8-bit I/O ports	16-bit Timer/Counters	UART
80C51	87C51	80C31	4K	128	4	2	sim
80C52	87C52	80C32	8K	256	4	3	sim

O MCs 8051 oferece os seguintes recursos:

- 5 interrupções (2 externas, 2 associadas aos temporizadores/contadores e porta série);
- 64 KB endereçáveis de Memória de Programa (PC=16 bits);
- 64 KB endereçáveis de Memória de Dados;
- 111 instruções.

O 8051 não segue a arquitectura de Von Newman, mas sim a arquitectura de Harvard. Por esse motivo a memória do programa encontra-se separada da memória de dados.

# Arquitetura Básica do 8051

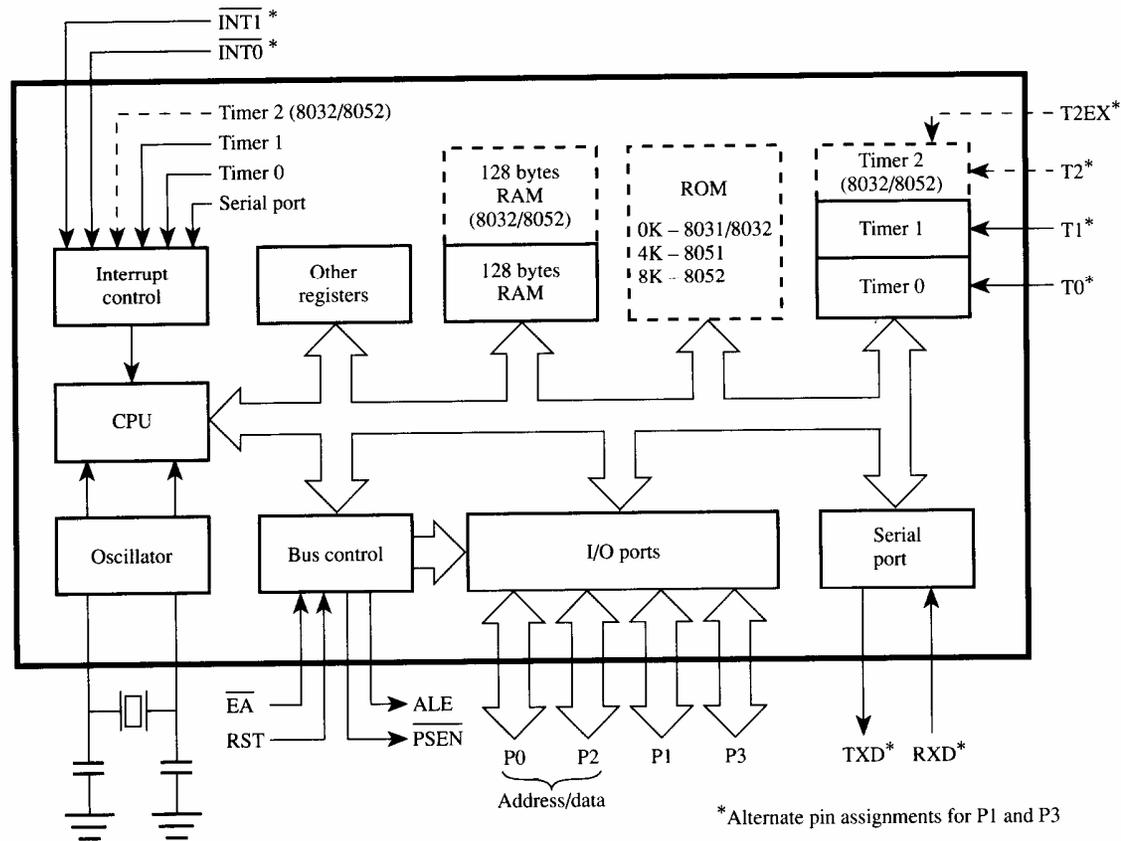
● Introdução aos Microcontroladores

Fundamentos Teóricos

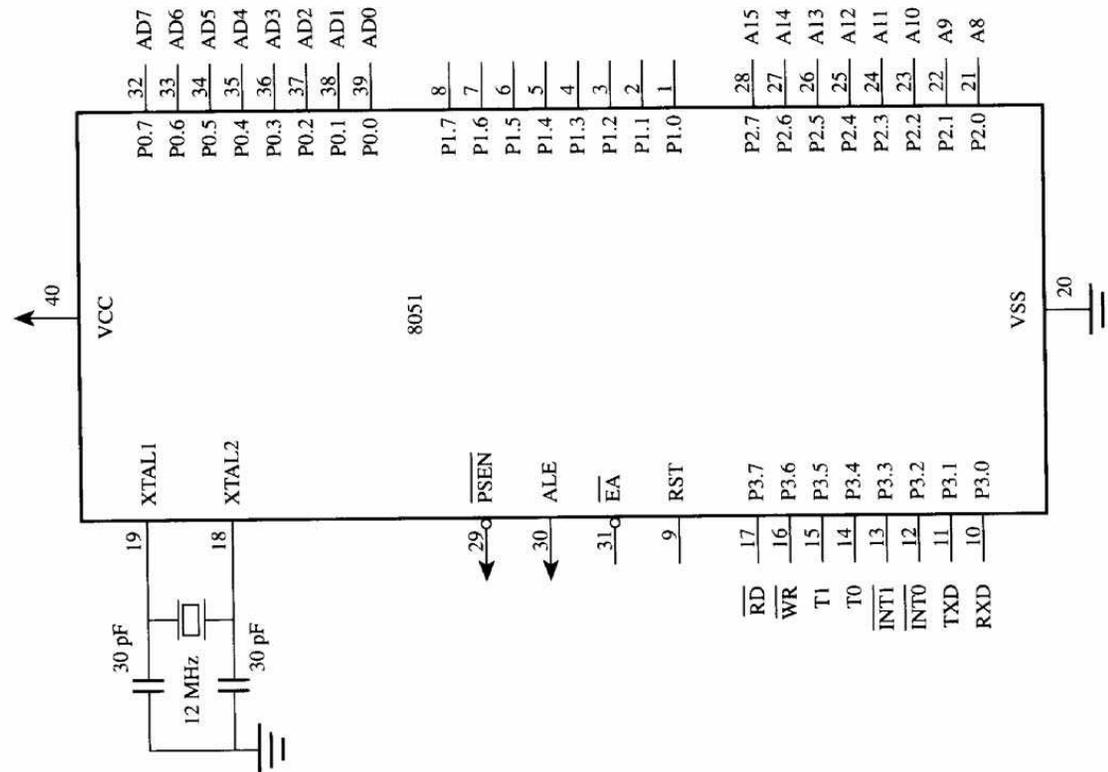
Introdução à família MCS51

● Arquitetura Básica MCS51

A Figura seguinte mostra o diagrama de blocos do microcontrolador 8051.



A Figura seguinte mostra os pinos de ligação do microcontrolador 8051 (40 pinos *dual in-line package (DIP)*).



## Porto 0:

- Função geral - Porto de Entradas/Saídas bidireccional - é um conjunto de oito flip-flops do tipo latch que podem ser acedidos por bit ou byte.
- Este porto não está disponível quando se usa a memória externa de programa, pois os seus pinos são utilizados para transferir dados.
- Suporta a transmissão de dados e endereços. Este porto está multiplexado com o byte menos significativo do bus de endereços e de dados. É necessária a desmultiplexagem externa com o IC74153.

## Porto 1:

- Dedicado quase exclusivamente à função de porto de Entradas e Saídas.
- Para os MCs 8032 e 8052, os dois bits menos significativos podem servir como entradas externas para um terceiro temporizador.

## Porto 2:

- Função geral - Porto de Entradas/Saídas.
- Suporta a transmissão de dados e endereços. Este porto está multiplexado com o byte mais significativo do bus de endereços e de dados.

## Porto 3:

- Função geral - Porto de Entradas/Saídas.
- Outras funções específicas:
  - ◆ P3.0 → RXD, entrada dados no modo de transmissão série;
  - ◆ P3.1 → TXD, saída de dados no modo de transmissão série;
  - ◆ P3.2 → \*INT0, entrada de excitação da interrupção externa 0;
  - ◆ P3.3 → \*INT1, entrada de excitação da interrupção externa 1;
  - ◆ P3.4 → T0, entrada para o timer 0 (contador neste caso);
  - ◆ P3.5 → T1, entrada para o timer 1 (contador neste caso);
  - ◆ P3.6 → \*WR, saída de controlo para operações de escrita na memória de dados externa;
  - ◆ P3.7 → \*RD, saída de controlo para operações de leitura na memória de dados externos;

**/PSEN (*Program Store Enable*):** Saída usada para aceder a memória externa de programa; esta memória é acedida quando surgem impulsos (nível baixo) naquela linha.

**ALE (*Address Latch Enable*):** Saída que suporta os impulsos gerados pelo processador, assinalando a validação do byte menos significativo dos endereços, em operações de acesso à memória externa. Também pode ser utilizada para excitar a entradas de relógio de periféricos (1/6 da frequência do oscilador).

**/EA (*External Access Enable*):** Informa a CPU se o programa está na memória de programa externa ou na ROM interna. Também usado para a gravação da ROM interna.

- /EA = 1 → (0000H a 0FFFH) ROM interna.
- /EA = 1 → (1000H a FFFFH) Memória de Programa Externa.
- /EA = 0 → (0000H a FFFFH) Memória de Programa Externa.

**RST (*Reset*):** Entrada de Reset activada no nível alto.

**XTAL1 e XTAL2:** Entrada e Saída do inversor do oscilador interno.

Grande parte dos microprocessadores implementam a arquitectura de Von Newman para a memória: memória partilhada para dados e programa. Isto é razoável porque normalmente os programas encontram-se armazenados no disco, sendo posteriormente carregados para a RAM quando é necessário executá-los. Desta forma quer os dados quer o programa residem na RAM do sistema.

No caso dos microcontroladores, a memória disponível é muito limitada, e não existem discos, nem outros tipos de memórias secundárias. Por esta razão o 8051 implementa a arquitectura de Harvard para a sua memória: memórias separadas para o programa e para os dados. As memórias, quer de programa, quer de dados, podem ser internas. Porém estas podem ser expandidas até 64K de memória de dados e 64K de memória de programa.

A Figura seguinte mostra a estrutura da memória do 8051.

