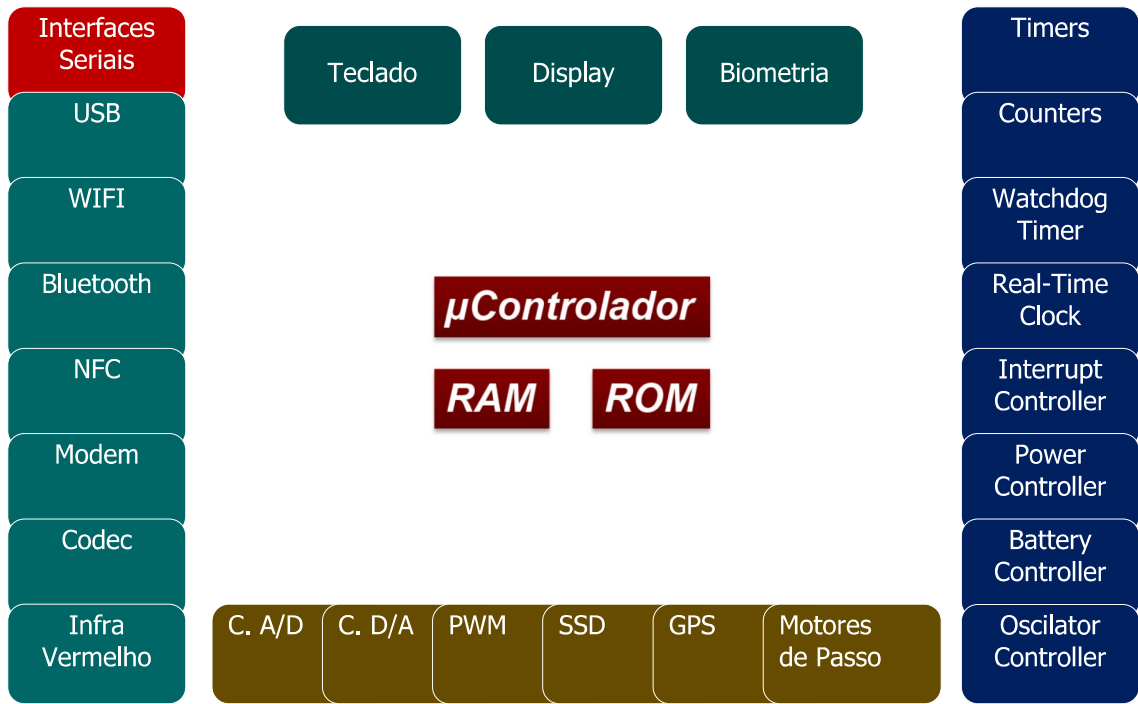


# SE::Visão Geral:: Arquitetura Geral de Hardware



## Experimento 4: Uso da Interface Serial

---

Construa um programa que envie e receba strings pela interface serial.

- Programe a interface serial no Modo 1. Utilize o registrador SCON para isso.
- Ajuste a velocidade de comunicação (baudrate) o mais próximo de 1200bps que puder. Para isso programe o Timer 1 no Modo 2 e defina os valores de TH1 e SMOD (PCON.7) para ajustar este baudrate. Qual a velocidade real que você conseguiu?
- Utilize a interrupção da serial para receber e transmitir bytes (o vetor de interrupção da serial é o 4).

## Experimento 4: Uso da Interface Serial

---

Para testar seu projeto usando apenas o simulador (1):

- Construa no *main()* uma rotina que consuma cada caracter recebido pela Rotina de Interrupção Serial, some 1 a este valor e transmita pela Interface Serial.
- Use a janela "UART #1" do simulador da Keil. Acesse a aba "View" do ambiente da Keil no modo "Debug", selecione "Serial Windows" e depois "UART #1".
  - Os caracteres que você digitar nesta janela são interpretados pelo simulador como sendo enviados de algum transmissor externo para o receptor da Interface Serial do seu 8051.
  - Os caracteres que você visualizar nesta janela, são os que o seu 8051 enviou pelo transmissor da sua Interface Serial.
- Com o Ambiente Keil no modo Debug, acesse a aba "Peripherals" e escolha "Serial". O valor do baudrate deve estar 0 bps. Deixe o seu programa executar a inicialização da Serial e verifique se o valor do BaudRate mudou para próximo de 1200 bps.

## Experimento 4: Uso da Interface Serial

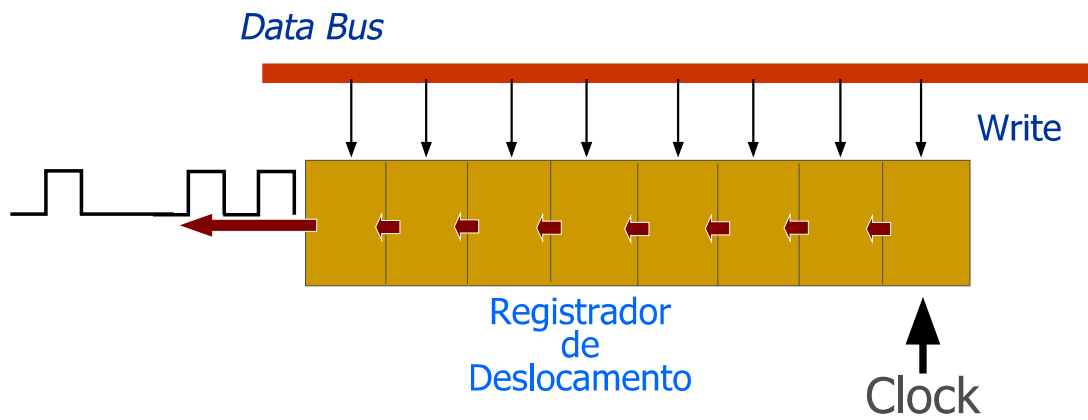
---

Para testar seu projeto usando dois PCs (2):

- "Conecte" a interface serial virtual do simulador da Keil com a interface serial do PC. Utilize os comandos "mode" e "assign" do debugger.
- Crie um arquivo de inicialização para fazer a conexão acima automaticamente sempre que você ligar o debugger para este projeto.
- Teste seu programa conectando dois PCs através de um cabo serial, sendo que um PC executa o programa que você desenvolveu e o outro roda o comtest.exe (está na homepage do curso)

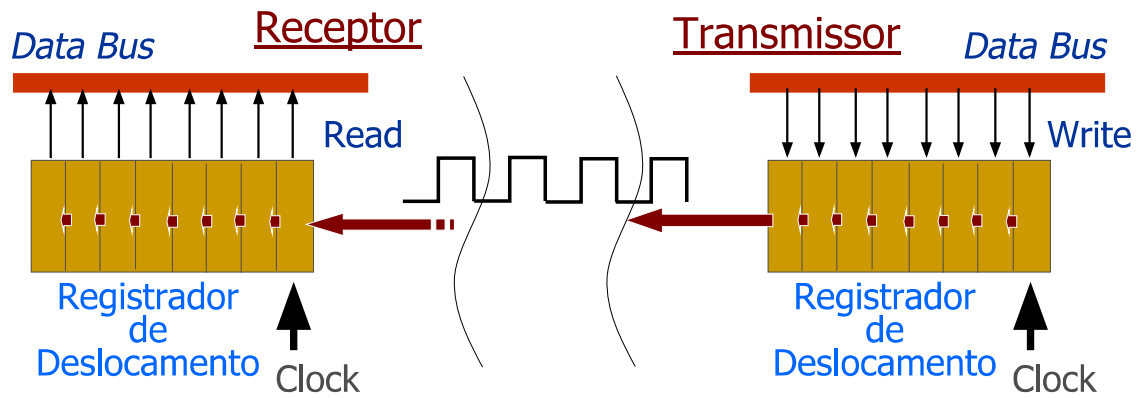
# Interface Serial

- Comunicação em que cada bit é enviado em sequência por um único fio



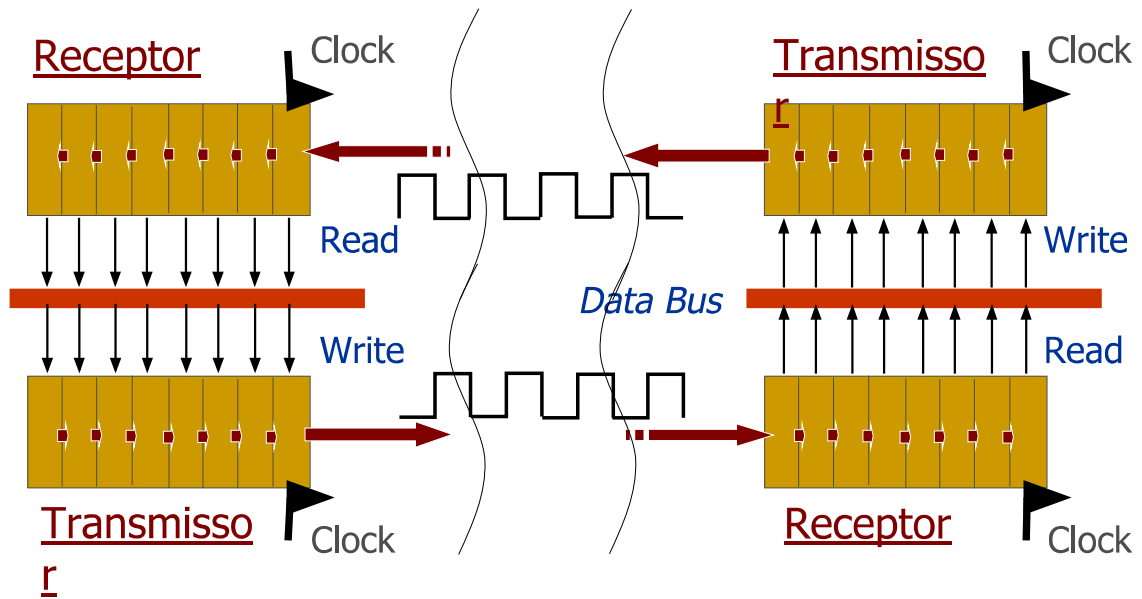
# Interface Serial

- Exemplo de Comunicação



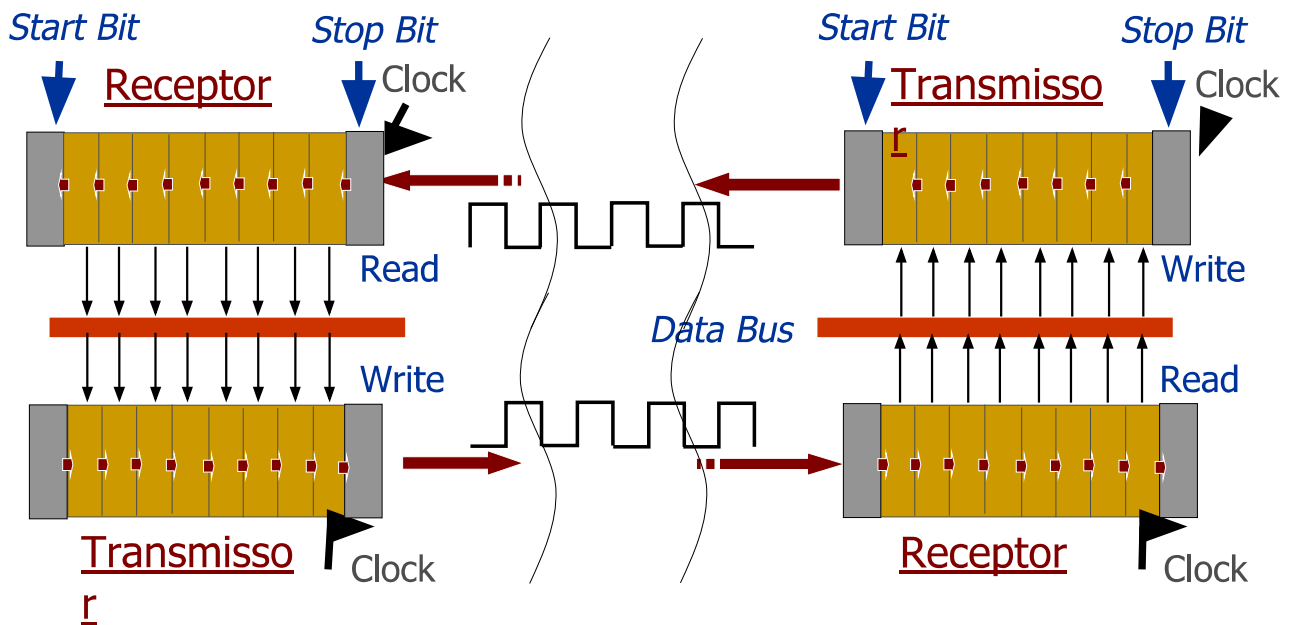
# Interface Serial

## Comunicação Síncrona



# Interface Serial

## Comunicação Assíncrona





# Interface Serial

---

- Vários modos de operação
  - Modo 0
    - Transmissão síncrona, 8 bits
    - Baudrate = Frequência do Clock / 12
  - Modo 1
    - Transmissão assíncrona, 10 bits sendo 1 start bit, 8 bits de dados e 1 stop bit
    - Na recepção, o stop bit é armazenado em RB8i (i indica qual a interface serial usada, caso haja mais de uma).
    - Baudrate variável

# Interface Serial

---

- Vários modos de operação
  - Modo 2
    - Transmissão assíncrona, 11 bits sendo 1 start bit, 8 bits de dados, 1 bit especial (valor definido por TB8) e 1 stop bit.
    - Na recepção, o bit especial é armazenado em RB8
    - Baudrate =  $1/32$  ou  $1/64$  Freq.Clock
  - Modo 3
    - Transmissão igual ao Modo 2
    - Baudrate variável

# Interface Serial

---

- Transmissão
  - É realizada assim que um byte é armazenado em SBUF
  - Ao transmitir faz com que o bit TI seja 1
- Recepção
  - Ao receber faz com que o bit RI seja 1.
  - Modo 0 (comunicação síncrona):
    - É realizada quando RI =0 e REN =1
  - Modo 1, 2 e 3 (comunicação assíncrona):
    - É realizada quando REN =1 e ocorre a chegada de um start bit.

# Interface Serial

---

- Interrupção
  - Pode ou não ser habilitada.
  - Quando habilitada, é gerada se  $TI = 1$  ou  $RI = 1$ .
  - O mesmo vetor de interrupção é usado tanto para transmissão quanto para recepção. O valor de  $TI$  e  $RI$  deve ser testado para verificar se houve transmissão e/ou recepção, respectivamente.
  - Para habilitar a interrupção da serial faça  $ES = 1$ .

# Interface Serial

SM0	SM1	SM2	REN	TB8	RB8	Ti	RI	SCON
-----	-----	-----	-----	-----	-----	----	----	------

*SM0, SM1*: Se 00 ⇒ Modo 0; 01 ⇒ Modo 1; 10 ⇒ Modo 2; 11 ⇒ Modo 3

*SM2* ⇒ Se Modo = 0, *SM2* deve ser 0.

Se Modo = 1, *SM2* deve ser igual a 0 para que stop bit funcione.

Se Modo = 2 ou 3, define se o bit especial deve ser igual a 1 para que *RI* = 1. (Comun. em rede)

*REN* ⇒ Habilita a recepção.

*TB8* ⇒ Nos modos 0 e 1, guarda cópia do bit 8 de transmissão

Nos modos 2 e 3, guarda o valor do bit especial.

*RB8* ⇒ No modo 0 não é usado.

No modo 1, se *SM2* = 0, guarda o stop bit recebido.

Nos modos 2 e 3, guarda o valor do bit especial recebido.

*TI* ⇒ Flag de fim de transmissão. *TI* = 1 quando o último bit começa a ser transmitido.

*RI* ⇒ Flag de fim de recepção. *RI* = 1 quando o último bit é recebido.

# Interface Serial

---

- Comunicação em Rede ou Multiprocessador (**Modos 2 e 3**)
  - Quando modo = 2 ou 3, e se SM2 = 1, a interrupção de recepção só é gerada se o bit especial = 1.
  - Esta característica pode ser usada em redes mestre-escravo da seguinte forma:
    - 1) Se o mestre deseja se comunicar com um dos escravos, primeiro ele envia o endereço do escravo.
    - 2) Endereços diferem de dados porque endereços tem o bit especial = 1 e os dados tem este bit = 0.
    - 3) Se SM2 = 1, nenhum escravo será interrompido por bytes de dados. Mas todos serão interrompidos por bytes de endereços.
    - 4) Ao receber seu endereço, o escravo faz com que SM2 = 0 de modo a permitir a recepção de bytes de dados.

# Interface Serial

---

- Taxas de Comunicação (Baud Rates)
  - A interface serial requer um clock 16 vezes maior que a taxa de comunicação desejada.
- Baudrate Modo 0 =  $\text{Freq. Clock} / 12$
- Baudrate Modo 2
  - SMOD é o bit 7 (mais significativo) do registrador PCON
  - $\text{baudrate} = (2^{\text{SMOD}}/64) \times \text{Freq. Clock}$ , ou  
 $\text{Freq. Clock} / 64$  se  $\text{SMOD} = 0$   
 $\text{Freq. Clock} / 32$  se  $\text{SMOD} = 1$

# Interface Serial

- Baudrate Modos 1 e 3
  - Pode ser fornecido pelo Timer1 (padrão)

## 1. Para usar o Timer 1

- Ponha o Timer 1 no Modo 2 (auto-reload)
- Baudrate =  $(2^{\text{SMOD}} / 32) \times \text{Freq.Timer1}$

$$\Rightarrow \text{Baudrate} = \frac{2^{\text{SMOD}}}{32} \times \frac{\text{Freq.Clock}}{12 \times (256 - \text{TH1})}$$

Obs: **SMOD** é o bit 7 (mais significativo) do registrador PCON



# Padrão Serial RS232

---

- RS232: *Recommended Standard 232* da EIA
  - Comunicações ponto a ponto
  - Usado por modems, mouses e algumas impressoras
  - Baixa imunidade a ruído
  - Mencionado no padrão:
    - limite de transmissão 20kbps
    - distância máxima entre dispositivos: 15 metros
  - Na prática:
    - pode-se transmitir até cerca de 200kbps
    - atinge 100m.

# Padrão Serial RS422 e RS485

---

- Adequadas para condições adversas de operação
- comunicação por pares de fios trançados  
([https://pt.wikipedia.org/wiki/Cabo\\_de\\_par\\_trançado](https://pt.wikipedia.org/wiki/Cabo_de_par_trançado))
- velocidades superiores a 100Mbps
- distâncias de vários quilômetros
- conexão de vários dispositivos na mesma linha

# Padrão Serial Universal Serial Bus - USB 1.0

---

- Taxas de até 12Mbps
- Até 127 periféricos por linha
- Conexão com sistema ligado (*hot plugging*)
- Reconhecimento automático do dispositivo (*Plug-and-Play*)
- Microcontroladores dotados de interfaces USB