

Desenvolvimento de Sistemas Embarcados em Tempo Real

Prof. Hermano Cabral

Departamento de Eletrônica e Sistemas — UFPE

24 de agosto de 2018

Tema central

- Arquitetura AVR — USART

Tema central

- Arquitetura AVR — USART

Objetivos

- Conhecer as características da interface de comunicação serial USART do ATmega328p
- Usar esta interface para comunicação com outros dispositivos

Introdução

- O microcontrolador ATmega328p possui uma interface USART para comunicação serial altamente flexível.

Introdução

- O microcontrolador ATmega328p possui uma interface USART para comunicação serial altamente flexível.
- O nome USART vem de Universal Synchronous and Asynchronous Receiver and Transmitter.

Introdução

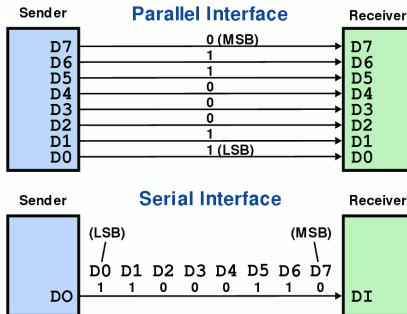
- O microcontrolador ATMega328p possui uma interface USART para comunicação serial altamente flexível.
- O nome USART vem de Universal Synchronous and Asynchronous Receiver and Transmitter.
- As placas arduino baseadas no ATMega328p direcionam os pinos de recepção e transmissão deste dispositivo para a porta USB.

Introdução

- O microcontrolador ATMega328p possui uma interface USART para comunicação serial altamente flexível.
- O nome USART vem de Universal Synchronous and Asynchronous Receiver and Transmitter.
- As placas arduino baseadas no ATMega328p direcionam os pinos de recepção e transmissão deste dispositivo para a porta USB.
- Isto permite que nos comuniquemos com o computador através desta interface.

Dispositivo de comunicação USART

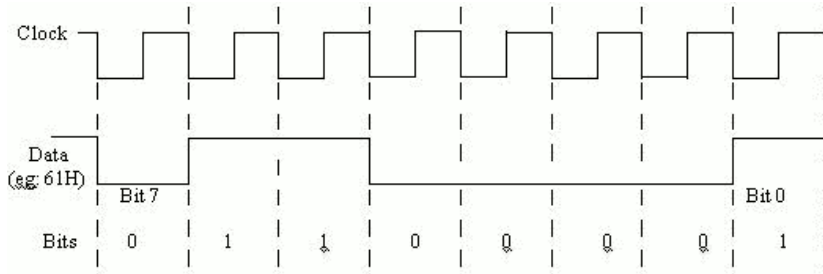
Example transfers of 01100011



Comunicação paralela versus serial

- Comunicação serial usa apenas 1 fio para transmitir dados.

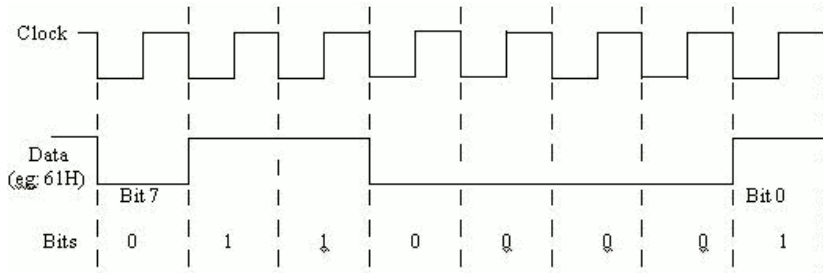
Dispositivo de comunicação USART



Comunicação serial síncrona

- A comunicação serial pode ser síncrona ou assíncrona.

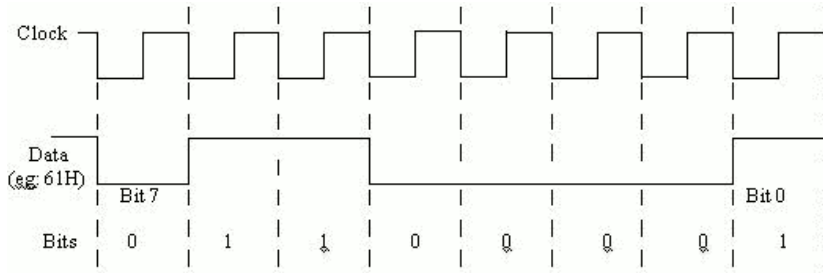
Dispositivo de comunicação USART



Comunicação serial síncrona

- A comunicação serial pode ser síncrona ou assíncrona.
- Na comunicação síncrona os bits dos bytes são transmitidos na cadência de um sinal de clock.

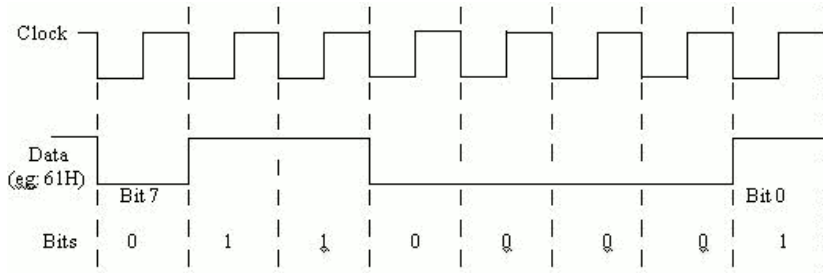
Dispositivo de comunicação USART



Comunicação serial síncrona

- A comunicação serial pode ser síncrona ou assíncrona.
- Na comunicação síncrona os bits dos bytes são transmitidos na cadência de um sinal de clock.
- Isto em geral implica no uso de um outro fio para transmitir o sinal de clock.

Dispositivo de comunicação USART

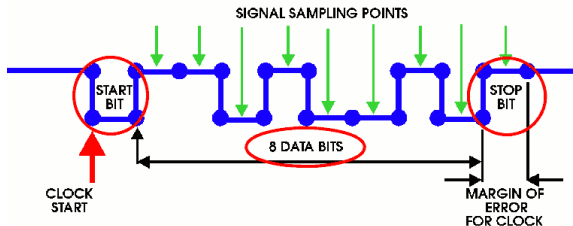


Comunicação serial síncrona

- A comunicação serial pode ser síncrona ou assíncrona.
- Na comunicação síncrona os bits dos bytes são transmitidos na cadência de um sinal de clock.
- Isto em geral implica no uso de um outro fio para transmitir o sinal de clock.
- A vantagem é o menor overhead com a sincronização.

Dispositivo de comunicação USART

ASYNCHRONOUS CHARACTER: 8 DATA BITS, ONE STOP BIT

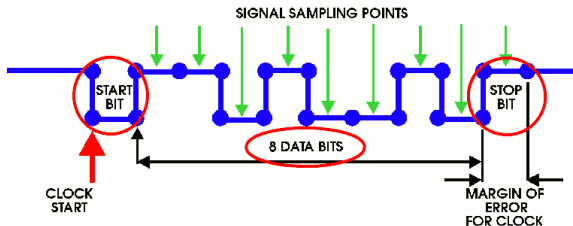


Comunicação serial assíncrona

- Na comunicação serial assíncrona o sincronismo ocorre para todo byte.

Dispositivo de comunicação USART

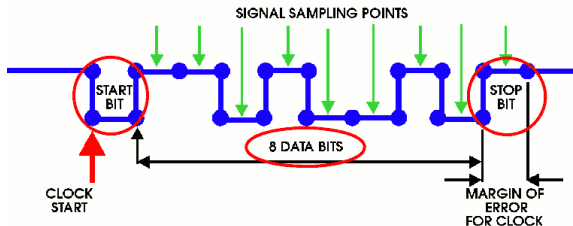
ASYNCHRONOUS CHARACTER: 8 DATA BITS, ONE STOP BIT



Comunicação serial assíncrona

- Na comunicação serial assíncrona o sincronismo ocorre para todo byte.
- O overhead é maior mas é mais resistente a ruído, além de só necessitar de 1 fio.

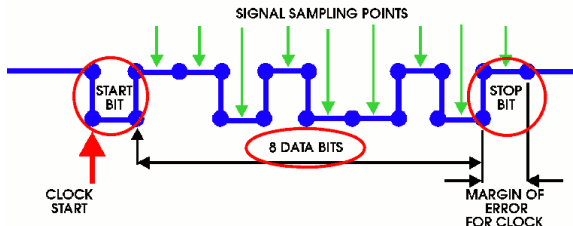
ASYNCHRONOUS CHARACTER: 8 DATA BITS, ONE STOP BIT



Comunicação serial assíncrona

- Na comunicação serial assíncrona o sincronismo ocorre para todo byte.
- O overhead é maior mas é mais resistente a ruído, além de só necessitar de 1 fio.
- Devido à sua flexibilidade, simplicidade e robustez, iremos nos concentrar na comunicação assíncrona.

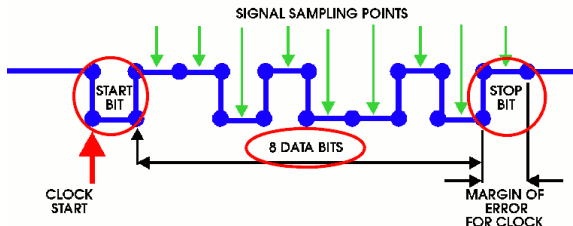
ASYNCHRONOUS CHARACTER: 8 DATA BITS, ONE STOP BIT



Comunicação serial assíncrona

- Para o sincronismo, usamos 1 bit de start e 1 ou mais bits de stop.

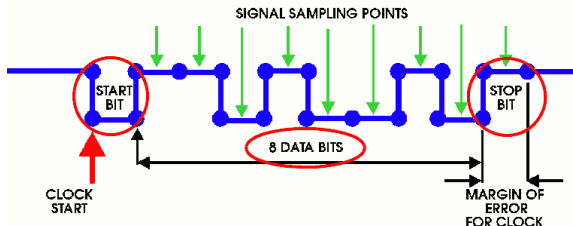
ASYNCHRONOUS CHARACTER: 8 DATA BITS, ONE STOP BIT



Comunicação serial assíncrona

- Para o sincronismo, usamos 1 bit de start e 1 ou mais bits de stop.
- A quantidade de bits de dados também pode variar.

ASYNCHRONOUS CHARACTER: 8 DATA BITS, ONE STOP BIT



Comunicação serial assíncrona

- Para o sincronismo, usamos 1 bit de start e 1 ou mais bits de stop.
- A quantidade de bits de dados também pode variar.
- Em alguns casos um bit extra de paridade também é usado.

Comunicação serial assíncrona

- Para 2 pontos se comunicarem entre si usando comunicação assíncrona, devemos definir os seguintes parâmetros:

Comunicação serial assíncrona

- Para 2 pontos se comunicarem entre si usando comunicação assíncrona, devemos definir os seguintes parâmetros:
 - Quantidade de bits: 8 é o mais comum

Comunicação serial assíncrona

- Para 2 pontos se comunicarem entre si usando comunicação assíncrona, devemos definir os seguintes parâmetros:
 - Quantidade de bits: 8 é o mais comum
 - Uso da paridade ou não: não usar é mais comum

Comunicação serial assíncrona

- Para 2 pontos se comunicarem entre si usando comunicação assíncrona, devemos definir os seguintes parâmetros:
 - Quantidade de bits: 8 é o mais comum
 - Uso da paridade ou não: não usar é mais comum
 - Quantidade de stop bits: 1 é o mais comum

Comunicação serial assíncrona

- Para 2 pontos se comunicarem entre si usando comunicação assíncrona, devemos definir os seguintes parâmetros:
 - Quantidade de bits: 8 é o mais comum
 - Uso da paridade ou não: não usar é mais comum
 - Quantidade de stop bits: 1 é o mais comum
- Essa combinação mais comum é conhecida como 8N1.

Comunicação serial assíncrona

- Além de definir estes parâmetros, devemos definir qual a velocidade de transmissão.

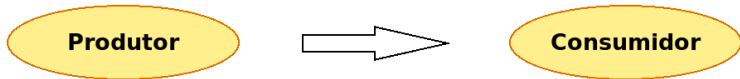
Comunicação serial assíncrona

- Além de definir estes parâmetros, devemos definir qual a velocidade de transmissão.
- Aqui não existe um valor que seja mais comumente usado.

Comunicação serial assíncrona

- Além de definir estes parâmetros, devemos definir qual a velocidade de transmissão.
- Aqui não existe um valor que seja mais comumente usado.
- Usaremos uma velocidade entre 57600 e 2 megabits/seg.

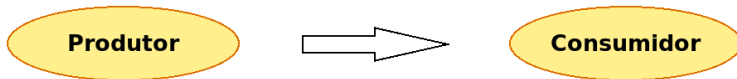
O problema do produtor-consumidor



Introdução

- O ato de 2 dispositivos se comunicarem pode ser visto como um exemplo do problema do produtor-consumidor.

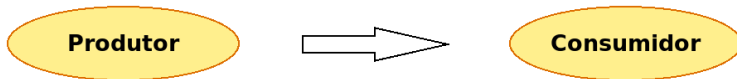
O problema do produtor-consumidor



Introdução

- O ato de 2 dispositivos se comunicarem pode ser visto como um exemplo do problema do produtor-consumidor.
- Temos um lado produzindo informação que é consumida pelo outro lado.

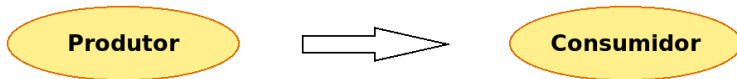
O problema do produtor-consumidor



Introdução

- O ato de 2 dispositivos se comunicarem pode ser visto como um exemplo do problema do produtor-consumidor.
- Temos um lado produzindo informação que é consumida pelo outro lado.
- O problema é sincronizar os 2 lados.

O problema do produtor-consumidor



Introdução

- O ato de 2 dispositivos se comunicarem pode ser visto como um exemplo do problema do produtor-consumidor.
- Temos um lado produzindo informação que é consumida pelo outro lado.
- O problema é sincronizar os 2 lados.
- Assumimos tacitamente que a taxa média de produção de informação é menor ou igual a de consumo.

O problema do produtor-consumidor



Desenvolvimento

- Muitas vezes é vantajoso bufferizar a comunicação entre eles.

O problema do produtor-consumidor



Desenvolvimento

- Muitas vezes é vantajoso bufferizar a comunicação entre eles.
- O tamanho do buffer vai depender da velocidade de transmissão e da variância na taxa de produção ou de consumo da informação.

O problema do produtor-consumidor



Desenvolvimento

- Muitas vezes é vantajoso bufferizar a comunicação entre eles.
- O tamanho do buffer vai depender da velocidade de transmissão e da variância na taxa de produção ou de consumo da informação.
- Se o buffer estiver cheio e o produtor quiser colocar mais informação, terá que esperar.

O problema do produtor-consumidor



Desenvolvimento

- Muitas vezes é vantajoso bufferizar a comunicação entre eles.
- O tamanho do buffer vai depender da velocidade de transmissão e da variância na taxa de produção ou de consumo da informação.
- Se o buffer estiver cheio e o produtor quiser colocar mais informação, terá que esperar.
- Se o buffer estiver vazio e o consumidor quiser mais informação, terá que esperar.

O problema do produtor-consumidor

Desenvolvimento

- O problema do que fazer enquanto espera é importante.

O problema do produtor-consumidor

Desenvolvimento

- O problema do que fazer enquanto espera é importante.
 - Pooling: ficamos testando constantemente se a operação pode ser feita.

Desenvolvimento

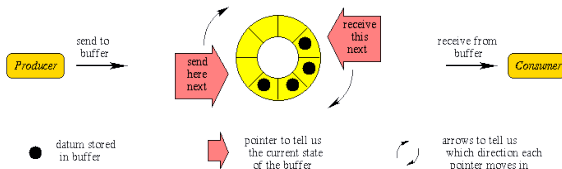
- O problema do que fazer enquanto espera é importante.
 - Pooling: ficamos testando constantemente se a operação pode ser feita.
 - Interrupção: pedimos para sermos avisado para repetir a operação.

O problema do produtor-consumidor

Desenvolvimento

- O problema do que fazer enquanto espera é importante.
 - Pooling: ficamos testando constantemente se a operação pode ser feita.
 - Interrupção: pedimos para sermos avisado para repetir a operação.
- Além disso, o produtor tem que decidir se continua a produzir informação ou não.

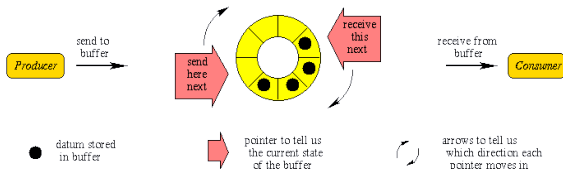
O problema do produtor-consumidor



Buffer circular

- Uma implementação tradicional do buffer é com uma lista circular.

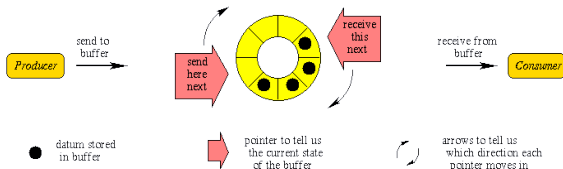
O problema do produtor-consumidor



Buffer circular

- Uma implementação tradicional do buffer é com uma lista circular.
- O produtor coloca informação na posição apontada por head.

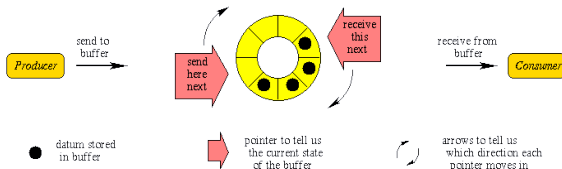
O problema do produtor-consumidor



Buffer circular

- Uma implementação tradicional do buffer é com uma lista circular.
- O produtor coloca informação na posição apontada por head.
- O consumidor retira informação da posição apontada por tail.

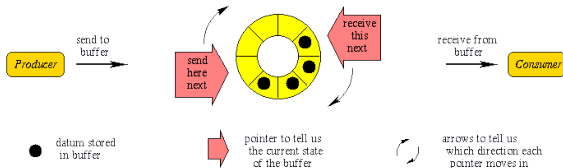
O problema do produtor-consumidor



Buffer circular

- Uma implementação tradicional do buffer é com uma lista circular.
- O produtor coloca informação na posição apontada por head.
- O consumidor retira informação da posição apontada por tail.
- Ambos os ponteiros são atualizados módulo o tamanho do buffer.

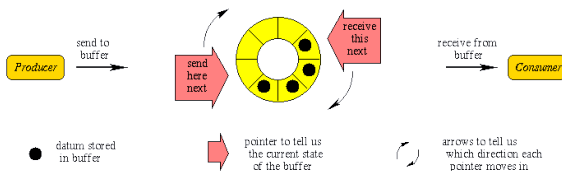
O problema do produtor-consumidor



Buffer circular

- Dois estados do buffer são importantes: buffer cheio e buffer vazio.

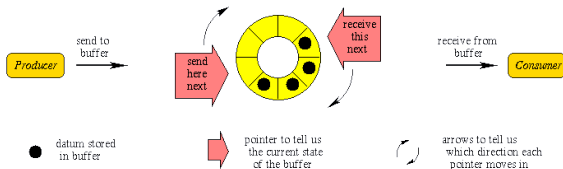
O problema do produtor-consumidor



Buffer circular

- Dois estados do buffer são importantes: buffer cheio e buffer vazio.
- Podemos detectar se o buffer estiver cheio se $\text{head} + 1 == \text{tail}$.

O problema do produtor-consumidor



Buffer circular

- Dois estados do buffer são importantes: buffer cheio e buffer vazio.
- Podemos detectar se o buffer estiver cheio se $head+1 == tail$.
- Podemos detectar se o buffer está vazio se $head == tail$.

Dispositivo de comunicação USART

UDRn – USART I/O Data Register n

| | | | | | | | | | |
|---------------|----------|-----|-----|-----|-----|-----|-----|-----|--------------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | RXB[7:0] | | | | | | | | UDRn (Read) |
| | TXB[7:0] | | | | | | | | UDRn (Write) |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Transmissão e recepção

- Tanto a transmissão como a recepção usam o registrador UDR0 como interface para os dados comunicados.

Dispositivo de comunicação USART

UDRn – USART I/O Data Register n

| | | | | | | | | | |
|---------------|----------|-----|-----|-----|-----|-----|-----|-----|--------------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | RXB[7:0] | | | | | | | | UDRn (Read) |
| | TXB[7:0] | | | | | | | | UDRn (Write) |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Transmissão e recepção

- Tanto a transmissão como a recepção usam o registrador UDR0 como interface para os dados comunicados.
- O registrador UDR0 é escrito para transmitir um byte e lido para receber um byte.

Dispositivo de comunicação USART

| Vector No | Program Address ⁽²⁾ | Source | Interrupts definition |
|-----------|--------------------------------|------------|---------------------------|
| 19 | 0x0024 | USART_RX | USART Rx Complete |
| 20 | 0x0026 | USART_UDRE | USART Data Register Empty |
| 21 | 0x0028 | USART_TX | USART Tx Complete |

Interrupções da USART

- A USART tem 3 interrupções, mostradas acima.

Configuração da USART

- A configuração da USART consiste em:
 - Escolher valores apropriados para o bit de stop, bit de paridade e quantidades de bits em uma transmissão.

Configuração da USART

- A configuração da USART consiste em:
 - Escolher valores apropriados para o bit de stop, bit de paridade e quantidades de bits em uma transmissão.
 - Configurar a velocidade de transmissão.

Configuração da USART

- A configuração da USART consiste em:
 - Escolher valores apropriados para o bit de stop, bit de paridade e quantidades de bits em uma transmissão.
 - Configurar a velocidade de transmissão.
 - Configurar as interrupções.

Configuração da USART

- A configuração da USART consiste em:
 - Escolher valores apropriados para o bit de stop, bit de paridade e quantidades de bits em uma transmissão.
 - Configurar a velocidade de transmissão.
 - Configurar as interrupções.
- Isto é feito através dos registradores UBRR0H, UBRR0L, UCSR0A, UCSR0B e UCSR0C.

Dispositivo de comunicação USART

UBRRnL and UBRRnH – USART Baud Rate Registers

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------------|-----|-----|-----|-------------|-----|-----|-----|--------|
| | – | – | – | – | UBRRn[11:8] | | | | UBRRnH |
| | UBRRn[7:0] | | | | | | | | UBRRnL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R/W | R/W | R/W | R/W | |
| | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Configuração da velocidade de transferência

- A velocidade de transferência é configurada pelos registradores UBRR0H e UBRR0L.

Dispositivo de comunicação USART

UBRRnL and UBRRnH – USART Baud Rate Registers

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
|---------------|------------|-----|-----|-----|-------------|-----|-----|-----|--------|
| | – | – | – | – | UBRRn[11:8] | | | | UBRRnH |
| | UBRRn[7:0] | | | | | | | | UBRRnL |
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Read/Write | R | R | R | R | R/W | R/W | R/W | R/W | |
| | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Configuração da velocidade de transferência

- A velocidade de transferência é configurada pelos registradores UBRR0H e UBRR0L.
- Eles se combinam em um valor de 12 bits que deve ser escolhido de acordo com

$$UBRR = \frac{f_{osc}}{8 \times \text{baudrate}} - 1$$

Configuração da USART

- A fórmula anterior é válida de estivermos usando o modo de recepção rápido, onde o bit é amostrado 8 vezes.

Configuração da USART

- A fórmula anterior é válida de estivermos usando o modo de recepção rápido, onde o bit é amostrado 8 vezes.
- Caso contrário, onde o bit é amostrado 16 vezes, a fórmula é

$$\text{UBRR} = \frac{f_{\text{osc}}}{16 \times \text{baudrate}} - 1$$

Dispositivo de comunicação USART

UCSRnA – USART Control and Status Register n A

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|-------------|-------------|--------------|------------|-------------|-------------|-------------|--------------|--------|
| | RXCn | TXCn | UDREN | FEn | DORn | UPEn | U2Xn | MPCMn | UCSRnA |
| Read/Write | R | R/W | R | R | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |

Configuração da USART

- O registrador UCSR0A contém o bit U2X0 de configuração do modo rápido.

Dispositivo de comunicação USART

UCSRnA – USART Control and Status Register n A

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|-------------|-------------|--------------|------------|-------------|-------------|-------------|--------------|--------|
| | RXCn | TXCn | UDREN | FEN | DORn | UPEn | U2Xn | MPCMn | UCSRnA |
| Read/Write | R | R/W | R | R | R | R | R/W | R/W | |
| Initial Value | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |

Configuração da USART

- O registrador UCSR0A contém o bit U2X0 de configuração do modo rápido.
- Além disso, este registrador tem 6 flags: 3 de operação normal e 3 de erro na transmissão.

Flags da USART

- As 3 flags de operação normal são associadas às 3 interrupções da USART.

Flags da USART

- As 3 flags de operação normal são associadas às 3 interrupções da USART.
- As flags são:
 - RXC0: recepção completa, setada quando um byte foi recebido.

Flags da USART

- As 3 flags de operação normal são associadas às 3 interrupções da USART.
- As flags são:
 - RXC0: recepção completa, setada quando um byte foi recebido.
 - TXC0: transmissão completa, setada quando um byte foi transmitido e o registrador UDR0 está vazio.

Flags da USART

- As 3 flags de operação normal são associadas às 3 interrupções da USART.
- As flags são:
 - RXC0: recepção completa, setada quando um byte foi recebido.
 - TXC0: transmissão completa, setada quando um byte foi transmitido e o registrador UDR0 está vazio.
 - UDRE0: o registrador UDR0 está vazio e pronto para receber um novo byte para ser transmitido.

Flags da USART

- A flag RXC0 é resetada quando o registrador UDR0 é lido ou quando a recepção é desabilitada.

Flags da USART

- A flag RXC0 é resetada quando o registrador UDR0 é lido ou quando a recepção é desabilitada.
- A flag TXC0 é resetada quando a interrupção associada é executada ou quando se escreve 1 na flag.

Flags da USART

- A flag RXC0 é resetada quando o registrador UDR0 é lido ou quando a recepção é desabilitada.
- A flag TXC0 é resetada quando a interrupção associada é executada ou quando se escreve 1 na flag.
- A flag UDRE0 é resetada escrevendo no registrador UDR0.

Flags da USART

- A flag RXC0 é resetada quando o registrador UDR0 é lido ou quando a recepção é desabilitada.
- A flag TXC0 é resetada quando a interrupção associada é executada ou quando se escreve 1 na flag.
- A flag UDRE0 é resetada escrevendo no registrador UDR0.
- Esta flag é a única que inicia com o valor 1.

Flags de erro da USART

- As flags de error são:
 - FE0: setada se houve um erro de recepção no 1º bit de stop.

Flags de erro da USART

- As flags de error são:
 - FE0: setada se houve um erro de recepção no 1º bit de stop.
 - DOR0: setada se o buffer de recepção está cheio e um byte a mais acabou de ser recebido.

Flags de erro da USART

- As flags de error são:
 - FE0: setada se houve um erro de recepção no 1º bit de stop.
 - DOR0: setada se o buffer de recepção está cheio e um byte a mais acabou de ser recebido.
 - UPE0: setada se houve erro de paridade.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| | | | | | | | | | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|---------------------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSR _n B |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Configuração da USART

- As outras configurações são feitas através dos registradores UCSR0B e UCSR0C.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| | | | | | | | | | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|---------------------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSR _n B |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Configuração da USART

- As outras configurações são feitas através dos registradores UCSR0B e UCSR0C.
- As interrupções da USART são habilitadas setando em 1 os bits RXCIE0, TXCIE0 e UDRIE0.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|---------------------|
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB_{8n} | TXB_{8n} | UCSR _n B |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Configuração da USART

- As outras configurações são feitas através dos registradores UCSR0B e UCSR0C.
- As interrupções da USART são habilitadas setando em 1 os bits RXCIE0, TXCIE0 e UDRIE0.
- A transmissão e recepção são habilitadas através dos bits RXEN0 e TXEN0.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|--------|
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSRnB |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

UCSRnC – USART Control and Status Register n C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------------------------|---------------------------|-------------------------|-------------------------|-------------------------|--------------------------|--------------------------|--------------------------|--------|
| | UMSEL_{n1} | UMSEL_{n0} | UPM_{n1} | UPM_{n0} | USBS_n | UCSZ_{n1} | UCSZ_{n0} | UCPOL_n | UCSRnC |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | |

Configuração da USART

- As configurações de paridade, número de stop bits e tamanho do byte são configurados como:
 - Os bits UMSEL01 e UMSEL00 escolhem entre comunicação assíncrona e síncrona.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|---------------------|
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSR _n B |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

UCSRnC – USART Control and Status Register n C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------------------------|---------------------------|-------------------------|-------------------------|-------------------------|--------------------------|--------------------------|--------------------------|---------------------|
| | UMSEL_{n1} | UMSEL_{n0} | UPM_{n1} | UPM_{n0} | USBS_n | UCSZ_{n1} | UCSZ_{n0} | UCPOL_n | UCSR _n C |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | |

Configuração da USART

- As configurações de paridade, número de stop bits e tamanho do byte são configurados como:
 - Os bits UMSEL01 e UMSEL00 escolhem entre comunicação assíncrona e síncrona.
 - Os bits UPM01 e UPM00 selecionam a paridade.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|---------------------|
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSR _n B |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

UCSRnC – USART Control and Status Register n C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------------------------|---------------------------|-------------------------|-------------------------|-------------------------|--------------------------|--------------------------|--------------------------|---------------------|
| | UMSEL_{n1} | UMSEL_{n0} | UPM_{n1} | UPM_{n0} | USBS_n | UCSZ_{n1} | UCSZ_{n0} | UCPOL_n | UCSR _n C |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | |

Configuração da USART

- As configurações de paridade, número de stop bits e tamanho do byte são configurados como:
 - Os bits UMSEL01 e UMSEL00 escolhem entre comunicação assíncrona e síncrona.
 - Os bits UPM01 e UPM00 selecionam a paridade.
 - O bit USBS0 seleciona se haverá 1 ou 2 bits de stop.

Dispositivo de comunicação USART

UCSRnB – USART Control and Status Register n B

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|--------------------------|--------------------------|--------------------------|-------------------------|-------------------------|--------------------------|-------------------------|-------------------------|---------------------|
| | RXCIE_n | TXCIE_n | UDRIE_n | RXEN_n | TXEN_n | UCSZ_{n2} | RXB8_n | TXB8_n | UCSR _n B |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

UCSRnC – USART Control and Status Register n C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|---------------------------|---------------------------|-------------------------|-------------------------|-------------------------|--------------------------|--------------------------|--------------------------|---------------------|
| | UMSEL_{n1} | UMSEL_{n0} | UPM_{n1} | UPM_{n0} | USBS_n | UCSZ_{n1} | UCSZ_{n0} | UCPOL_n | UCSR _n C |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | |

Configuração da USART

- As configurações de paridade, número de stop bits e tamanho do byte são configurados como:
 - Os bits UMSEL01 e UMSEL00 escolhem entre comunicação assíncrona e síncrona.
 - Os bits UPM01 e UPM00 selecionam a paridade.
 - O bit USBS0 seleciona se haverá 1 ou 2 bits de stop.
 - Os bits UCSZ02, UCSZ01 e UCSZ00 selecionam a quantidade de bits no byte.

Configuração da USART

- Para o caso comum da comunicação assíncrona 8N1, todos os bits são nulos exceto UCSZ1 e UCSZ0, que são setados em 1.

Configuração da USART

- Para o caso comum da comunicação assíncrona 8N1, todos os bits são nulos exceto UCSZ1 e UCSZ0, que são setados em 1.
- As outras configurações podem ser vistas no capítulo sobre USART da datasheet.