Software Embarcado

06 – USART

Francisco Sant'Anna
Sala 6020-B
francisco@ime.uerj.br

http://github.com/fsantanna-uerj/SE

Universal Synchronous Asynchronous Receiver Transceiver

- Comunicação full duplex
- Modo síncrono e assíncrono
- Váridos modos de dados/stop bits/paridade
- Detecção de overflow
- 3 fontes de interrupções
- Operação básica
  - U**BR**R, U**CS**R, U**D**R

Transmissão

```
#define BAUD (F_CPU/16/9600-1)

void setup (void) {
  /*Set baud rate */
  UBRR0H = BAUD >> 8;
  UBRR0L = BAUD;

  /* Enable transmitter */
  UCSR0B = 1 << TXEN0;   // TXEN="Transmitter Enable"

  /* Set frame format: 8data, 2stop bit */
  UCSR0C = (1<<USBS0)|(3<<UCSZ00);
}

char data = '0';
int  i    = 0;

void loop (void) {
  delay(1000);

  /* Wait for empty transmit buffer */
  while ( !( UCSR0A & (1<<UDRE0)) ) // UDRE="Data Register Empty"
    ;

  /* Put data into buffer, sends the data */
  UDR0 = data + i;                  // UDR="Data Register"
  i = (i + 1) % 10;
}
```

Recepção

```
#define BAUD (F_CPU/16/9600-1)

void setup (void) {
  /*Set baud rate */
  UBRR0H = BAUD >> 8;
  UBRR0L = BAUD;

  /* Receive transmitter */
  UCSR0B = 1 << RXEN0;   // RXEN="Receiver Enable"

  /* Set frame format: 8data, 2stop bit */
  UCSR0C = (1<<USBS0)|(3<<UCSZ00);

  pinMode(13, OUTPUT);
}

void loop (void) {
  /* Wait for data to be received */
  while ( !(UCSR0A & (1<<RXC0)) )
    ;

  unsigned char data = UDR0;
  digitalWrite(13, data%2);
}
```

UBRRL

24.12.5. USART Baud Rate 0 Register Low

Name: UBRR0L
Offset: 0xC4
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	UBRR0[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – UBRR0[7:0]: USART Baud Rate 0
This is a 12-bit register which contains the USART baud rate. The UBRR0H contains the four most significant bits and the UBRR0L contains the eight least significant bits of the USART baud rate. Ongoing transmissions by the Transmitter and Receiver will be corrupted if the baud rate is changed. Writing UBRR0L will trigger an immediate update of the baud rate prescaler.

24.12.6. USART Baud Rate 0 Register High

Name: UBRR0H
Offset: 0xC5
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
					UBRR0[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bits 3:0 – UBRR0[3:0]: USART Baud Rate 0 n [n = 11:8]
Refer to UBRR0L.

UCSRA

24.12.2. USART Control and Status Register 0 A

Name: UCSR0A
Offset: 0xC0
Reset: 0x20
Property: -

Bit	7	6	5	4	3	2	1	0
	RXC0	TXC0	UDRE0	FE0	DOR0	UPE0	U2X0	MPCM0
Access	R	R/W	R	R	R	R	R/W	R/W
Reset	0	0	1	0	0	0	0	0

Bit 7 – RXC0: USART Receive Complete
This flag bit is set when there are unread data in the receive buffer and cleared when the receive buffer is empty (i.e., does not contain any unread data). If the Receiver is disabled, the receive buffer will be flushed and consequently the RXC0 bit will become zero. The RXC0 Flag can be used to generate a Receive Complete interrupt (see description of the RXCIE0 bit).

Bit 6 – TXC0: USART Transmit Complete
This flag bit is set when the entire frame in the Transmit Shift Register has been shifted out and there are no new data currently present in the transmit buffer (UDR0). The TXC0 Flag bit is automatically cleared when a transmit complete interrupt is executed, or it can be cleared by writing a one to its bit location. The TXC0 Flag can generate a Transmit Complete interrupt (see description of the TXCIE0 bit).

Bit 5 – UDRE0: USART Data Register Empty
The UDRE0 Flag indicates if the transmit buffer (UDR0) is ready to receive new data. If UDRE0 is one, the buffer is empty, and therefore ready to be written. The UDRE0 Flag can generate a Data Register Empty interrupt (see description of the UDRIE0 bit). UDRE0 is set after a reset to indicate that the Transmitter is ready.

UCSRB

24.12.3. USART Control and Status Register 0 B

Name: UCSR0B
Offset: 0xC1
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	RXCIE0	TXCIE0	UDRIE0	RXEN0	TXEN0	UCSZ02	RXB80	TXB80
Access	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

Bit 7 – RXCIE0: RX Complete Interrupt Enable 0
Writing this bit to one enables interrupt on the RXC0 Flag. A USART Receive Complete interrupt will be generated only if the RXCIE0 bit is written to one, the Global Interrupt Flag in SREG is written to one and the RXC0 bit in UCSR0A is set.

Bit 6 – TXCIE0: TX Complete Interrupt Enable 0
Writing this bit to one enables interrupt on the TXC0 Flag. A USART Transmit Complete interrupt will be generated only if the TXCIE0 bit is written to one, the Global Interrupt Flag in SREG is written to one and the TXC0 bit in UCSR0A is set.

Bit 5 – UDRIE0: USART Data Register Empty Interrupt Enable 0
Writing this bit to one enables interrupt on the UDRE0 Flag. A Data Register Empty interrupt will be generated only if the UDRIE0 bit is written to one, the Global Interrupt Flag in SREG is written to one and the UDRE0 bit in UCSR0A is set.

Bit 4 – RXEN0: Receiver Enable 0
Writing this bit to one enables the USART Receiver. The Receiver will override normal port operation for the RxDn pin when enabled. Disabling the Receiver will flush the receive buffer invalidating the FE0, DOR0, and UPE0 Flags.

Bit 3 – TXEN0: Transmitter Enable 0
Writing this bit to one enables the USART Transmitter. The Transmitter will override normal port operation for the TxD0 pin when enabled. The disabling of the Transmitter (writing TXEN0 to zero) will not become effective until ongoing and pending transmissions are completed, i.e., when the Transmit Shift Register and Transmit Buffer Register do not contain data to be transmitted. When disabled, the Transmitter will no longer override the TxD0 port.

UCSRC

24.12.4. USART Control and Status Register 0 C

Name: UCSR0C
Offset: 0xC2
Reset: 0x06
Property: -

Bit	7	6	5	4	3	2	1	0
	UMSEL01	UMSEL00	UPM01	UPM00	USBS0	UCSZ01 / UDORD0	UCSZ00 / UCPHA0	UCPOL0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	1	0

Bits 7:6 – UMSEL0n: USART Mode Select 0 n [n = 1:0]
These bits select the mode of operation of the USART0

Table 24-8. USART Mode Selection

UMSEL0[1:0]	Mode
00	Asynchronous USART
01	Synchronous USART
10	Reserved
11	Master SPI (MSPIM)(1)

UDR

Name: UDR0
Offset: 0xC6
Reset: 0x00
Property: -

Bit	7	6	5	4	3	2	1	0
	TXB / RXB[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – TXB / RXB[7:0]: USART Transmit / Receive Data Buffer

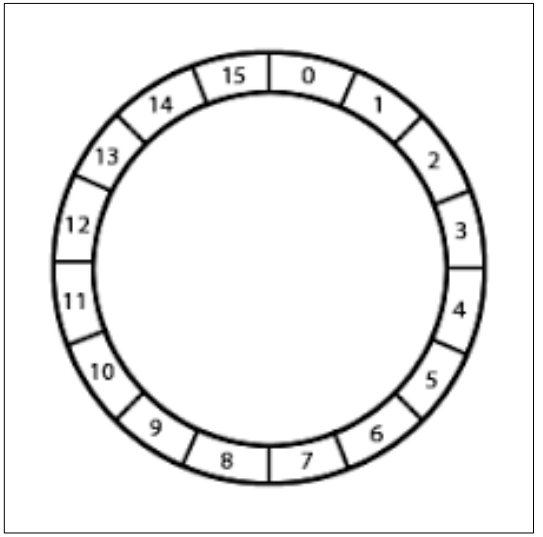
Transmissão e Recepção

```
#define BAUD (F_CPU/16/9600-1)

void setup (void) {
  UBRR0H = BAUD >> 8;
  UBRR0L = BAUD;

  UCSR0B = (1<<RXEN0)|(1<<TXEN0);
  UCSR0C = (1<<USBS0)|(3<<UCSZ00);
}

void loop (void) {
  while ( !(UCSR0A & (1<<RXC0)) )
    ;

  unsigned char data = UDR0;
  data = data + ('A' - 'a');
  //delay(2);

  while ( !( UCSR0A & (1<<UDRE0)) )
    ;

  UDR0 = data;
}
```

Buffers

Buffers

- Região de memória para dados temporários
- Mediador entre dispositivos com taxas de transmissão ou processamento incompatíveis
  - Transmissão e recebimento em “background”
- Tipicamente implementados como fila (FIFO)
  - Em particular como “ring buffer”



Interrupções

Transmissão e Recepção

```
#define BUF 32

unsigned char rx[BUF];
int rx_0 = 0;
int rx_n = 0;

unsigned char tx[BUF];
int tx_0 = 0;
int tx_n = 0;
int tx_ing = 0;

void setup (void) {
  ...
  UCSR0B = (1<<RXCIE0) |
           (1<<TXCIE0) |
           ...;
}
```

```
void loop (void) {
  unsigned char data;
  int todo = 0;
  cli();
    if (rx_0 != rx_n) {
      todo = 1;
      data = rx[rx_0];
      rx_0 = (rx_0 + 1) % BUF;
    }
  sei();

  if (todo) {
    //delay(100);
    data = data + ('A' - 'a');
    cli();
      if (!tx_ing) {
        UDR0 = data;
        tx_ing = 1;
      } else {
        tx[tx_n] = data;
        tx_n = (tx_n + 1) % BUF;
      }
    sei();
  }
}
```

```
ISR (USART0_RX_vect) {
  rx[rx_n] = UDR0;
  rx_n = (rx_n + 1) % BUF;
}

ISR (USART0_TX_vect) {
  if (tx_0 != tx_n) {
    UDR0 = tx[tx_0];
    tx_0 = (tx_0 + 1) %
BUF;
    tx_ing = 1;
  } else {
    tx_ing = 0;
  }
}
```