



ĐH Sư Phạm Kỹ Thuật TP. Hồ Chí Minh

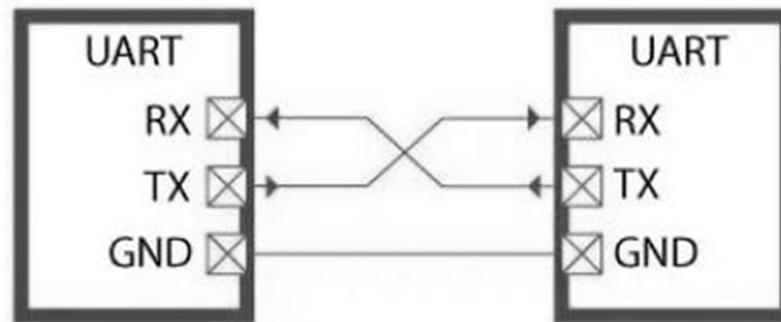
Môn học

Thực tập Vi Điều Khiển²

Tuần 11

Universal Synchronous/Asynchronous Receiver Transmitter(USART)

USART là bộ truyền nhận dữ liệu nối tiếp có thể đồng bộ hoặc không đồng bộ. Là một trong những giao thức truyền nhận dữ liệu giữa thiết bị - thiết bị phổ biến nhất



- ❖ Giao tiếp 2 chiều USART, MCU cần phải kết nối với thiết bị bên ngoài ít nhất 2 chân là chân TX (chân truyền dữ liệu ra bên ngoài) và chân RX (chân nhận dữ liệu từ bên ngoài).
- ❖ Chân TX của MCU sẽ được nối với chân RX của thiết bị ngoài và ngược lại chân RX của MCU sẽ được nối với chân TX của thiết bị ngoài.

Các bước thiết lập chế độ USART

Step 1: Thiết lập chân truyền dữ liệu (TX), và chân nhận dữ liệu (RX)

Step 2. Cấp xung clock cho bộ USARTx

Step 3: Thiết lập tốc độ truyền baud cho module USARTx

Step 4: Kích hoạt module USARTx

Step 5: Thiết lập Interrupt Transmitter/ Receiver (nếu sử dụng)

Step 2. Thiết lập chân TX, RX

Khai báo chế độ hoạt động cho vi điều khiển tương ứng với module USART. Cần chú ý là một ngoại vi nên chủ yếu khai báo các chân ở chế độ Alternate Function (nếu có).

Xác định module USARTx cần sử dụng để khai báo chân

	USART1	USART2	USART3
TX pin	PA9	PA2	PB10
RX pin	PA10	PA3	PB11

❖ F1 Series:

Chân TX: Output AF mode

Chân RX: Input Floating mode

❖ F2,3,4 Series:

Chân TX: AF mode

Chân RX: AF mode

Sử dụng thanh ghi AFRL/AFRH để định hình USARTx – AFy trong bảng tương quan

Cấu hình USART

UT

Step 3. Cấp xung clock cho bộ USART sử dụng

Được quản lý bởi module RCC (Reset and Clock Control). Để USART hoạt động được thì cần cấp xung cho Timer tương ứng tại các thanh ghi quản lý bus APB1ENR hoặc APB2ENR

❖ F1 Series : RCC_APB1ENR

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		DAC EN	PWR EN	BKP EN	Res.	CAN EN	Res.	USB EN	I2C2 EN	I2C1 EN	UART5 EN	UART4 EN	USART3 EN	USART2 EN	Res.
		rw	rw	rw		rw		rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 EN	SPI2 EN	Reserved		WWD GEN	Reserved		TIM14 EN	TIM13 EN	TIM12 EN	TIM7 EN	TIM6 EN	TIM5 EN	TIM4 EN	TIM3 EN	TIM2 EN
rw	rw			rw			rw	rw	rw	rw	rw	rw	rw	rw	

Ví dụ: **RCC** -> **APB1ENR** |= **(1<<17)** ; // Clock for USART2

Step 4. Thiết lập tốc độ truyền baud

Tốc độ truyền của module USART được quyết định bởi thanh ghi USARTx_BRR

USARTx_BRR: *Thanh ghi cho phép thiết lập baud rate cho module USART tương ứng*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIV_Mantissa[11:0]												DIV_Fraction[3:0]			
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Công thức tính baudrate. Chế độ 8 bit (OVER 8) chỉ có trên F2,3,4 series

❖ *Chế độ 16 bit (mặc định)*

$$\text{Baudrate} = \frac{\text{Clock Frequency}}{(8 * 2)(\text{USART Div})}$$

If OVER8 = 0.

❖ *Chế độ 8 bit*

$$\text{Baudrate} = \frac{\text{Clock Frequency}}{(8)(\text{USART Div})}$$

If OVER8 = 1

Step 4. Thiết lập tốc độ truyền baud

Example: Tốc độ truyền 9600 bit/s, Clock Freq = 16Mhz.

$$\text{Baud Rate} = \frac{\text{Clock Frequency}}{(8 * 2)(\text{USART Div})}$$

=> USART Div = 104.17

❖ Mantissa = 104

❖ Fraction = $0.17 * 16 = 2.72 \Rightarrow \text{Fraction} = 3$

Ví dụ: **USART2** -> **BRR** = **(104 << 4) | (3 << 0)** ; // baudrate 9600

Step 5. Kích hoạt USART

Sau khi thiết lập các thông số cần thiết, tiến hành khai kích hoạt USART tương ứng bằng thanh ghi:

USARTx_CR1

USARTx_CR1: Thanh ghi cho phép thiết lập các chế độ hoạt động của module USART tương ứng

❖ *F2,3,4 Series:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	Reserved	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	RWU	SBK
rw	Res.	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Ví dụ:

USART2 -> **CR1** |= (1<<2) | (1<<3) ; // Enable TX, RX

USART2 -> **CR1** |= (1<<13) ; // Enable USART2

Step 5. Thiết lập ngắt USART (option)

Chế độ cho phép CPU xảy ra ngắt mỗi khi: có gói dữ liệu truyền đến vi điều khiển, hoặc vi điều khiển gửi đi 1 gói dữ liệu. Thiết lập các thông số ở thanh ghi USARTx_CR1

❖ *F2,3,4 Series:*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	Reserved	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	RWU	SBK
rw	Res.	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Ví dụ:

USART2 -> **CR1** |= **(1<<7)** ; // Enable TX interrupt

USART2 -> **CR1** |= **(1<<5)** ; // Enable RX interrupt

NVIC -> **ISER[1]** |= **(1<<6)** ; // Enable USART2 interrupt

Step 5. Thiết lập ngắt USART (option)

Sau khi chế độ ngắt tương ứng được thiết lập. Cấu trúc chương trình ngắt của USART

Vector ngắt của USART.

```
/// USARTx interrupt vector
void USARTx_IRQHandler (void)    // x = 1, 2, 3,....
{
    //clear USARTx interrupt flag
    // flag is cleared by read or clear register USARTx_DR

    //placed code begin
    //placed code end
}
```

Theo dõi các trạng thái của USART (option register)

Sau khi chế độ truyền nhận. Truyền nhận dữ liệu thông qua module USART được giám sát thông qua thanh ghi USARTx_SR

USARTx_SR: Thanh ghi cho phép theo dõi các trạng thái hoạt động của truyền nhận dữ liệu

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CTS	LBD	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
						rc_w0	rc_w0	r	rc_w0	rc_w0	r	r	r	r	r

- ❖ **TC bit:** (Transmission complete): Bit cho phép kiểm tra dữ liệu có được gửi đi thành công hay không ? Khi dữ liệu được gửi đi thành công qua cổng USART thì bit này được set lên 1.
- ❖ **RXNE bit:** (Read data register not empty): Bit cho phép kiểm tra thanh ghi USART_DR có nhận dữ liệu hay không ?. Nếu nhận dc dữ liệu mới thì bit này được set lên 1.

Bài tập 1. **Viết chương trình truyền nhận UART của vi điều khiển với máy tính, điều khiển hoạt động của led theo yêu cầu sau :**

Vi điều khiển nhận một số từ máy tính và điều khiển hai led RE0 và RE1 với 4 trạng thái

- ❖ Nhận số 1: Hai led tắt, gửi lên máy tính chuỗi "Number 1".
- ❖ Nhận số 2: Hai led sáng, gửi lên máy tính chuỗi "Number 2".
- ❖ Nhận số 3: Led RE0 sáng, RE1 tắt, gửi lại máy tính chuỗi "Number 3".
- ❖ Nhận số 4: Led RE1 tắt, RE1 sáng, gửi lại máy tính chuỗi "Number 4".
- ❖ Số khác : Hai led cùng chớp tắt với chu kỳ _delay 1s gửi lại máy tính chuỗi "Other numbers".

Bài tập 2. **Viết chương trình truyền nhận UART của vi điều khiển với máy tính, điều khiển hoạt động theo yêu cầu sau :**

- ❖ Gửi số lần nhấn nút lên máy tính.

Bài tập 3. **Viết chương trình truyền nhận USART của vi điều khiển với máy tính, điều khiển hoạt động của băng tải yêu cầu sau :**

- ❖ Gửi chuỗi “Start”: Băng tải chạy
- ❖ Băng tải chỉ chạy khi cảm biến có vật, và băng tải dừng khi vật ở cuối hành trình. Cứ mỗi lần có vật là số sản phẩm tăng lên 1 và gửi lên máy tính số vật đếm được.
- ❖ Nhấn nút Stop (đỏ) thì băng tải dừng lại và trả số lần đếm vật về 0, và gửi lên máy tính chuỗi “Stop”.
- ❖ Nhấn nút Pause (màu vàng): thì gửi lên máy tính chuỗi “Nhấn nút vàng còn lại”. Nếu trong 10s không nhấn nút vàng còn lại thì băng tải chạy lại nếu trên băng tải đang có vật hoặc cảm biến phát hiện có vật.

Bài tập 4. Viết chương trình thực hiện trên mô hình LED_BUTTON theo yêu cầu sau.

Hiện số lần nhấn nút lên LED 7 đoạn. Giá trị đếm từ 0 đến 99. Yêu cầu: Mỗi lần nhấn nút chỉ tăng 1 đơn vị.

- ❖ Thực hiện đếm lên và đếm xuống.
- ❖ Cứ mỗi lần nhấn nút, giá trị của đếm được gửi lên Terminal máy tính.
- ❖ Từ Terminal máy tính chuỗi số “XX” : với XX là giá trị bất kỳ: giá trị đếm trên LED 7 đoạn được thiết lập về số XX và nhấn nút đếm lên/xuống từ giá trị này.