

رابط سریال USART

فهرست مطالب

1. مباحث مقدماتی
2. آشنایی با واحد سریال USART
3. انواع روش های انتقال سریال

انتقال موازی و سریال

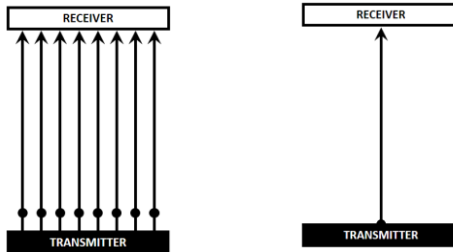
انتقال اطلاعات به دو روش انجام می پذیرد :

□ روش موازی : در این روش n بیت اطلاعات توسط n خط، انتقال می یابد.

□ روش سریال: در این روش در هر لحظه فقط یک بیت ارسال می شود.

در روش انتقال موازی سرعت بیشتر است، ولی برای فواصل طولانی به علت ازدیاد سیم های ارتباطی، هزینه ی این روش بالا می باشد.

در فواصل طولانی به کار بردن روش سریال مناسب تر است.



ارتباط بین فرستنده و گیرنده

به طور کلی سه روش ارتباط بین فرستنده و گیرنده قابل بررسی می باشد:

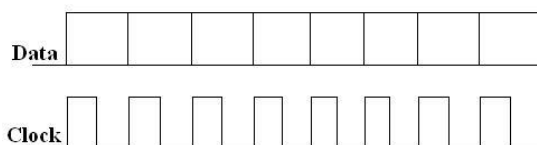
1. روش یک طرفه یا ساده (Simplex): اطلاعات فقط در یک جهت انتقال می یابد. (مثل رادیو و تلویزیون)
2. روش نیم دوطرفه (Half Duplex): اطلاعات میتواند در هر دو جهت انتقال یابد. اما در هر لحظه فقط یک جهت امکان پذیر است. (مثل Walkie-Talkie)
3. روش دوطرفه (Full Duplex): اطلاعات در یک لحظه در هر دو جهت انتقال می یابد. (مثل موبایل)

در میکرو کنترلرهای AVR از روش دو طرفه استفاده شده است.

روش های مختلف ارسال سریال

1. انتقال سریال به روش سنکرون یا همگام: فرستنده به همراه اطلاعات پالس ساعت را نیز ارسال می کند (به ازای هر پالس ساعت یک بیت ارسال می شود) تا گیرنده به کمک آن خود را تنظیم و همگام کند و اطلاعات را به درستی از روی خط داده بردارد.

- این نوع ارتباط سریال به دلیل اینکه دارای پالس های همگام سازی می باشد در سرعت های بالا قابل استفاده است.



حالت های مختلف ارتباط سریال همگام

میکرو کنترلرهای AVR دارای دو حالت ارتباط سریال همگام می باشند:

- ❖ ارتباط سریال همگام Master: در این حالت پالس های همگام سازی توسط خود میکروکنترلر روی پایه ی XCK تولید و به عنوان پالس ساعت همگام با اطلاعات به گیرنده ارسال می شود، تا گیرنده خودش را تنظیم کند.
- ❖ ارتباط سریال همگام Slave: در این حالت میکروکنترلر نقش گیرنده دارد و پالس های همگام سازی را که از فرستنده ارسال شده است، از طریق پایه ی XCK دریافت می کند.

انتقال سریال ناهمگام

2. **انتقال سریال به روش آسنکرون یا ناهمگام:** یک کاراکتر همراه با یک بیت توازن، یک بیت پایانی و یک بیت شروع در یک قاب یا فریم فرستاده می شود. در واقع در این روش هیچ پالس ساعتی فرستاده نمی شود. در این حالت گیرنده خود را با جست و جو برای یک بیت شروع همگام می کند. بیت توازن نشان می دهد که آیا اطلاعات به طور صحیح دریافت شده است یا نه.



قالب بندی ارسال سریال به روش ناهمگام

- از رشته اطلاعات نمونه برداری می شود تا یک منطق صفر به دست آید.
- بعد از آن نیمی از زمان یک بیت سپری شده تا مجدداً از آن نمونه برداری صورت گیرد.
- بعد از نمونه برداری در وسط بیت شروع به **تعداد باقیمانده بیتها** به فواصل زمانی یک بیت، از اطلاعات نمونه برداری می شود
- اطلاعات نمونه برداری شده در یک **رجیستر انتقالی** جمع شده تا به صورت یک **بایت اطلاعات موازی** در آید.
- سپس با بررسی بیت توازن، بررسی خطاها و بیت پایانی آزمایش می گردد.
- اگر بیت پایانی موجود باشد، اطلاعات قابل قبول است و در غیر این صورت **خطای قالب بندی** نشان داده خواهد شد.
- خطای قالب بندی معمولاً در صورتی اتفاق می افتد که اطلاعات با نرخ انتقال (Baud Rate) اشتباه دریافت گردند.

آشنایی با واحد سریال USART

در میکروکنترلرهای AVR

USART در AVR

USART : Universal Synchronous/Asynchronous Receiver Transmitter

ارسال کننده و دریافت کننده ی همزمان و غیر همزمان در میکروکنترلرهای AVR دارای ویژگی های زیر بوده است:

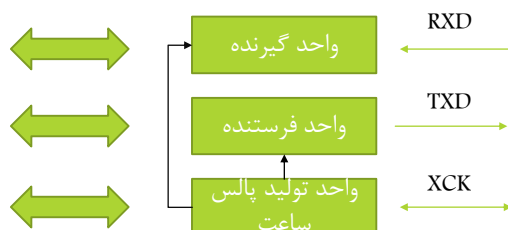
- ▶ عملکرد کاملاً دوطرفه (Full Duplex)
- ▶ عملکرد سریال به صورت سنکرون و آسنکرون
- ▶ عملکرد سنکرون در دو مد Master و Slave
- ▶ پشتیبانی از قالب های داده 5 الی 9 بیتی و نیز یک یا دو بیت توقف که به صورت اختیاری انتخاب می گردد.
- ▶ مولد بیت توازن زوج یا فرد و بررسی بیت توازن و تست آن توسط سخت افزار

ادامه

- ▶ آشکار سازی خطای قالب بندی
- ▶ حالت ارتباطی چند پردازنده ای
- ▶ سه وقفه مجزا برای کامل شدن ارسال، کامل شدن دریافت و خالی بودن رجیستر ارسال
- ❖ حداقل طول فریم در اینجا، 7 و حداکثر آن 13 می باشد. چرا؟

دیاگرام بلوکی ساده USART

گذرگاه داده داخلی
میکروکنترلر



PDIP			
(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RESET	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5 (TDI)
(TXD) PD1	15	26	PC4 (TDO)
(INT0) PD2	16	25	PC3 (TMS)
(INT1) PD3	17	24	PC2 (TCK)
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP) PD6	20	21	PD7 (OC2)

واحد تولید کننده ی پالس ساعت

USART چهار حالت عملکرد را پشتیبانی می کند :

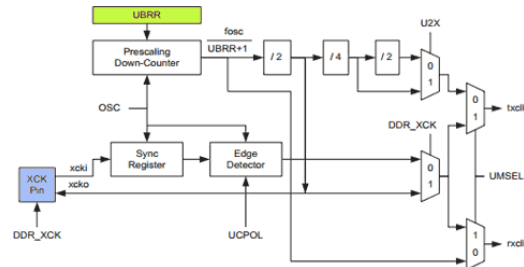
1. سنکرون Master
2. سنکرون Slave
3. آسنکرون معمولی
4. آسنکرون با سرعت دو برابر

برای هر یک از این حالات، به واحد تولید کننده پالس ساعت نیاز است.

واحد تولید کننده ی پالس ساعت (2)

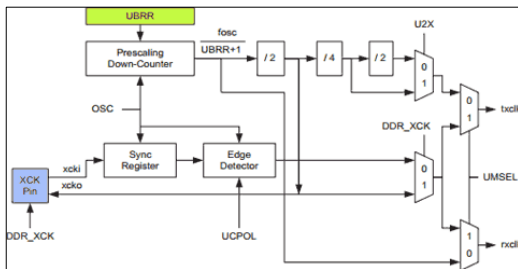
برای مد ارتباط سنکرون، بلوک تولید کننده پالس ساعت وظیفه ی دریافت و یا ارسال **پالس های همگام سازی** از طریق پین XCK را بر عهده دارد.

در حالت ارتباط آسنکرون این بلوک وظیفه ی **تولید و تنظیم نرخ بود (Baud Rate)** ارتباط سریال را برای دریافت و ارسال داده بر عهده دارد. این امر به کمک رجیسترهای UBRRH و UBRRL و بلوک تولید کننده ی نرخ انتقال صورت می گیرد.



واحد تولید کننده ی نرخ انتقال

- واحد ارتباط سریال، یک تولید کننده ی پالس ساعت داخلی را شامل می شود که از آن در حالت آسنکرون و حالت سنکرون در مد Master استفاده می کند.
- در این واحد یک شمارنده (Prescaling DownCounter) وجود دارد که همواره به صورت کاهشی می شمارد.
- ورودی پالس ساعت این شمارنده در واقع پالس ساعت کاری میکروکنترلر با همان فرکانس کاری میکروکنترلر است.



- محتوای شمارنده از رجیستر UBRR بارگذاری می شود
- هر بار که محتوای این شمارنده به صفر برسد، یک پالس ساعت برای USART تولید می شود.
- بسته به مد عملکرد ارتباط سریال پالس ساعت تولید شده بر 2، 8 یا 16 تقسیم می شود که قسمت های مختلف از آن استفاده کنند.

محاسبه ی مقدار Baud Rate

Table 52. Equations for Calculating Baud Rate Register Setting

Operating Mode	Equation for Calculating Baud Rate ⁽¹⁾	Equation for Calculating UBRR Value
Asynchronous Normal mode (U2X = 0)	$BAUD = \frac{f_{OSC}}{16(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = 1)	$BAUD = \frac{f_{OSC}}{8(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{OSC}}{2(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{2BAUD} - 1$

Note: 1. The baud rate is defined to be the transfer rate in bit per second (bps)

BAUD Baud rate (in bits per second, bps)

f_{OSC} System Oscillator clock frequency

UBRR Contents of the UBRRH and UBRRL Registers (0 - 4095)

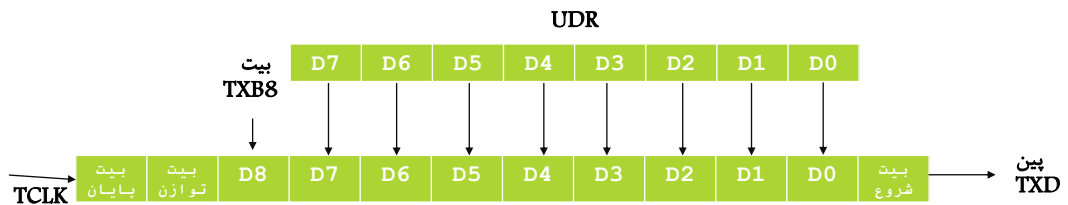
- می توان نرخ ارسال را با یک کردن بیت U2X در رجیستر UCSRA دو برابر نمود. تنظیم این بیت در مد ناهمگام تاثیر خواهد داشت. با یک کردن این بیت نرخ ارسال به جای 16 بر 8 تقسیم می گردد، بنابراین نرخ ارسال را دو برابر می کند.

واحد فرستنده سریال

این قسمت دارای یک رجیستر به نام UDR (USART Data Register) است که با قرار دادن یک بایت در آن، میکروکنترلر اقدام به ارسال آن خواهد نمود.

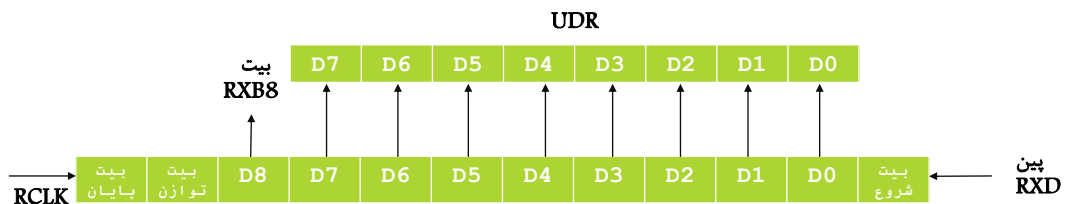
اطلاعات از این رجیستر وارد یک **شیفت رجیستر PISO (Parallel Input Serial Output)** شده و با اعمال پالس ساعت به شیفت رجیستر، اطلاعات به صورت سریال خارج می شود.

در حالت آسنکرون، واحد فرستنده سریال، وظیفه ی تولید بیت شروع و پایان و ایجاد یک قالب استاندارد را برعهده دارد.



واحد گیرنده ی سریال

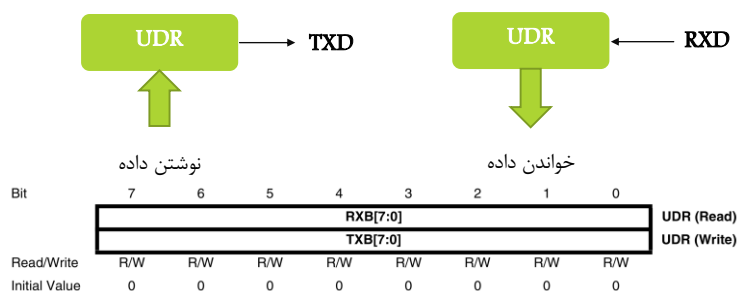
در این قسمت از USART، ابتدا شیفت رجیستر گیرنده با سرعتی که توسط واحد تولید Baud Rate تنظیم شده است (RCLK)، اطلاعات را بیت به بیت از پایه ی RXD دریافت می کند، پس از حذف بیت های اضافی مانند بیت های پایان و توازن، داده ها درون رجیستر UDR قرار می گیرند.



رجیستر UDR

□ رجیستر (USART Data Register) UDR

رجیستر هشت بیتی است که فقط برای ارتباط سریال در AVR استفاده می شود. بایستی که قرار است از طریق خط TXD ارسال شود، ابتدا در این رجیستر قرار می گیرد. همین طور این رجیستر یک بایت داده که به وسیله ی خط RXD دریافت شده است را در خود نگه می دارد.



رجیستر UCSRA

□ رجیستر (USART Control and State Register A) UCSRA

این رجیستر هشت بیتی برای برنامه ریزی و کنترل واحد سریال به کار می رود.

Bit	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

بیت های رجیستر UCSRA

1. RXC (USART Receive Complete)

اگر در بافر گیرنده (UDR) داده ای موجود بوده و توسط کاربر خوانده نشده باشد، این بیت یک خواهد بود. در صورت خالی بودن بافر گیرنده این بیت صفر خواهد شد.

این بیت برای تولید وقفه ی دریافت نیز به کار می رود.

2. TXC (USART Transmit Complete)

زمانی که تمامی داده سریال به بیرون شیفت رجیستر انتقال یابد و داده ی جدیدی برای ارسال وجود نداشته باشد، این بیت یک می شود. این در حالی است که داده ی جدیدی در بافر فرستنده (UDR) موجود نباشد.

این بیت می تواند یک وقفه ی کامل شدن ارسال را ایجاد کند.

7	6	5	4	3	2	1	0	
RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA

بیت های رجیستر UCSRA

3. UDRE (USART Data Register Empty)

این بیت زمانی فعال خواهد شد که بافر ارسال (UDR) خالی و آماده ی دریافت اطلاعات جدید باشد. همچنین این بیت می تواند برای فعال کردن وقفه ی خالی بودن UDR به کار رود. در واقع، زمانی این بیت یک می شود، که بایت مدنظر برای ارسال، از UDR به PISO منتقل شود. باید توجه داشت که این به معنی تکمیل ارسال این بایت نیست!

UDRE بعد از بازنشانی شدن میکرو کنترلر یک می شود.

4. FE (Framing Error)

این بیت زمانی که در قالب داده ی دریافتی، اشکالی وجود داشته باشد یک خواهد شد. به طور مثال اگر بیت Stop کاراکتر دریافتی صفر باشد، FE یک خواهد شد. این بیت تا زمانی معتبر است که بافر دریافت UDR خوانده نشده باشد.

هنگام نوشتن در رجیستر UCSRA باید این بیت را صفر کرد.

7	6	5	4	3	2	1	0	
RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA

بیت های رجیستر UCSRA

5. DOR (Data Overrun)

این بیت زمانی فعال خواهد شد که بافر دریافت پر شده باشد و کاراکتر جدیدی به شیفت رجیستر وارد یا بیت شروع جدید تشخیص داده شود. در این حالت اطلاعات جدید از بین می رود و یک خطا به نام Data Over Run به وجود می آید.

این بیت تا زمانی معتبر است که بافر دریافت UDR خوانده شود. باید در مقداردهی اولیه این بیت را صفر کنیم.

6. PE (Parity Error)

این بیت زمانی که خطای توازن در قالب دریافتی گیرنده ایجاد شود، یک خواهد شد و تا زمانی معتبر است که رجیستر دریافت UDR خوانده نشده باشد.

باید در مقداردهی اولیه این بیت را صفر کنیم.

7	6	5	4	3	2	1	0	
RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA

بیت های رجیستر UCSRA

7. U2X (Double the USART Transmission Speed)

این بیت دو برابر کننده ی سرعت بوده و فقط در زمان ارسال داده به حالت آسنکرون کاربرد دارد و در حالت سنکرون باید آن را صفر نمود.

8. MPCM (Multi Processor Communication Mode)

یک شدن این بیت، میکروکنترلر را به حالت ارتباط چند پردازنده ای می برد. در این حالت می توان چند میکروکنترلر را به صورت شبکه ای به یکدیگر متصل نمود.

توضیحات تکمیلی در اسلایدهای بعدی ارائه خواهد شد.

7	6	5	4	3	2	1	0	
RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA

رجیستر UCSRB

□ رجیستر (USART Control and State Register B) UCSRB

این رجیستر هشت بیتی برای برنامه ریزی و کنترل واحد سریال به کار میرود.

Bit	7	6	5	4	3	2	1	0	
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB
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	

بیت های رجیستر UCSRB

1. RXCIE (RX Complete Interrupt Enable)

با یک کردن این بیت، وقفه دریافت USART فعال خواهد شد. در صورتی که وقفه ی سراسری (I) فعال باشد با دریافت داده و فعال شدن پرچم RXC وقفه مورد نظر به اجرا در می آید.

2. TXCIE (TX Complete Interrupt Enable)

با یک کردن این بیت، وقفه ارسال USART فعال خواهد شد. در صورتی که وقفه ی سراسری (I) فعال باشد با ارسال داده و فعال شدن پرچم TXC وقفه مورد نظر اجرا می گردد.

7	6	5	4	3	2	1	0	
RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB

بیت های رجیستر UCSRB

3. UDRIE (USART Data Register Interrupt Enable)

با فعال شدن این بیت وقفه ی خالی بودن رجیستر داده (UDRE) فعال خواهد شد، به شرط آنکه وقفه ی سراسری فعال باشد. در نتیجه اینکه وقتی بافر ارسال خالی بود، یک وقفه توسط CPU اجرا می شود.

4. RXEN (Receiver Enable)

با یک کردن این بیت گیرنده ی USART فعال خواهد شد، زمانی که این بیت فعال می شود، عملکرد عادی پایه ی RXD قطع شده و به عنوان ورودی سریال عمل خواهد کرد. اگر این بیت فعال باشد و بعد آن را غیر فعال کنیم در این حالت بافر دریافت سریال (UDR) خالی می شود و بیت های DOR،FE و PE نامعتبر خواهند شد.

7	6	5	4	3	2	1	0	
RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB

بیت های رجیستر UCSRB

5. TXEN (Transmitter Enable)

با یک کردن این بیت فرستنده USART فعال خواهد شد، هنگامی که این بیت یک می شود، عملکرد عادی پایه ی TXD قطع شده و به عنوان خروجی سریال عمل خواهد نمود. صفر کردن این بیت تا هنگامی که اطلاعات به طور کامل ارسال نشود، فرستنده را غیر فعال نمی کند.

6. UCSZ2 (Character Size)

این بیت به همراه بیت های UCSZ1..0 در رجیستر UCSRC تعداد بیت های داده ی سریال (تعداد کاراکتر) را در یک قالب برای ارسال و دریافت معین می کند.

7	6	5	4	3	2	1	0	
RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB

بیت های رجیستر UCSRB

7. RXB8 (Receive Data Bit 8)

بیت نهم داده دریافت شده ی سریال می باشد، البته این بیت زمانی که از قالب دریافت و ارسال 9 بیتی استفاده شود معتبر است.

همچنین این بیت باید قبل از خوانده شدن رجیستر دریافت خوانده شود.

8. TXB8 (Transmit Data Bit 8)

وقتی از قالب ارسال سریال 9 بیتی استفاده می کنید، این بیت به عنوان بیت نهم ارسال می شود.

همچنین این بیت باید قبل از نوشته شدن هشت بیت در رجیستر UDR، نوشته شود.

7	6	5	4	3	2	1	0	
RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB

رجیستر UCSRC

□ رجیستر UCSRC (USART Control and State Register C)

این رجیستر از نظر آدرس حافظه با رجیستر UBRRH مشابه می باشد. به همین دلیل برای دسترسی به هر دو رجیستر یک بیت کنترلی در نظر گرفته شده است. (بیت URSEL)

Bit	7	6	5	4	3	2	1	0	
	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	0	0	0	0	1	1	0	

بیت های رجیستر UCSRC

1. URSEL (Register Select)

این بیت دسترسی به یکی از رجیسترهای UBRRH یا UCSRC را تعیین می کند.

2. UMSEL (USART Mode Select)

این بیت جهت انتخاب مد کاری USART به کار می رود. اگر این بیت صفر باشد مد کاری آسنکرون و اگر یک باشد مد کاری، سنکرون خواهد بود.

UMSEL	Mode
0	Asynchronous Operation
1	Synchronous Operation

7	6	5	4	3	2	1	0	
URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC

بیت های رجیستر UCSRC

3.4. UPM 1,0 (Parity Mode)

این دو بیت جهت تنظیم بیت توازن مورد استفاده قرار می گیرند.

UPM1	UPM0	Parity Mode
0	0	Disabled
0	1	Reserved
1	0	Enabled, Even Parity
1	1	Enabled, Odd Parity

5. USBS (Stop Bit Select)

در صورت فعال بودن USBS، دو بیت پایان و با صفر شدن این بیت نیز یک بیت پایان ارسال می شود.

دریافت کننده ی سریال با این بیت کاری ندارد و تاثیری بر عملکرد دریافت کننده نخواهد گذاشت.

USBS	Stop Bit(s)
0	1-bit
1	2-bit

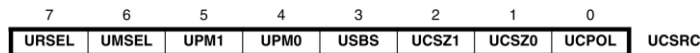
7	6	5	4	3	2	1	0	
URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC

بیت های رجیستر UCSRC

6.7. UCSZ1,0 (Character Size)

این دو بیت به همراه بیت UCSZ 2 در UCSRB تعداد بیت های یک داده سریال (تعداد کاراکتر) را انتخاب می کنند.

UCSZ2	UCSZ1	UCSZ0	Character Size
0	0	0	5-bit
0	0	1	6-bit
0	1	0	7-bit
0	1	1	8-bit
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Reserved
1	1	1	9-bit



بیت های رجیستر UCSRC

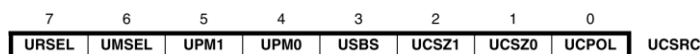
8. UCPOL (Clock Polarity)

این بیت تنها در حالت ارتباط سریال سنکرون استفاده می شود.

اگر این بیت یک شود : نمونه برداری از پایه ی RXD در لبه ی بالارونده و تغییر بیت داده روی پایه ی TXD در لبه ی پایین رونده ی پالس XCK صورت خواهد گرفت.

اگر این بیت صفر شود : نمونه برداری از پایه ی RXD در لبه ی پایین رونده و تغییر بیت داده روی پایه ی TXD در لبه ی بالا رونده ی پالس XCK صورت خواهد گرفت.

UCPOL	Transmitted Data Changed (Output of TxD Pin)	Received Data Sampled (Input on RxD Pin)
0	Rising XCK Edge	Falling XCK Edge
1	Falling XCK Edge	Rising XCK Edge



رجیستر UBRR

□ رجیستر (UBRR (USART Baud Rate Register

یک رجیستر شانزده بیتی است که از دو رجیستر UBRRH و UBRRL تشکیل شده است.

Bit	15	14	13	12	11	10	9	8	
	URSEL	—	—	—	UBRR[11:8]				UBRRH
	UBRR[7:0]								UBRRL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	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	

بیت های رجیستر UBRR

URSEL (Register Select)

این بیت با همین نام در رجیستر UCSRC نیز وجود دارد و با توجه به اینکه دو رجیستر از آدرس یکسانی در فضای I/O بهره می برند در نتیجه این بیت بین دو رجیستر به صورت مشترک عمل می کند و با صفر شدن آن می توان به رجیستر UBRRH دسترسی داشت.

BIT12..14 (Reserved Bits)

این بیت ها بلا استفاده بوده است و در هنگام خواندن و نوشتن صفر هستند.

BIT11..0 (USART Baud Rate Register)

این بیت ها برای تعیین Baud Rate رابط USART مورد استفاده قرار می گیرند، در واقع به عنوان یک رجیستر دوازده بیتی برای تعیین Baud Rate در اختیار کاربر است.

برنامه ریزی اولیه USART

قبل از هرگونه ارتباط سریال باید USART را برنامه ریزی نماییم.

تنظیماتی مثل تعیین Baud Rate، نوع ارتباط سریال، قالب بندی اطلاعات و

قبل از برنامه ریزی اولیه ی USART وقفه ی سراسری را باید غیر فعال کرد ولی بعد از آن، باید وقفه سراسری را فعال نمود (مثال) مقدار مناسب رجیستر های UCSRB, UCSRC و UBRR که در آن USART، به صورت قالب بندی هشت بیتی و یک بیت توقف و بدون بیت توازن و نرخ انتقال 19200 عمل نماید را مشخص کنید. (واحد USART به صورت فرستنده و با فرض استفاده از کریستال خارجی 11.0592 و بیت U2X=0)

	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8
UCSRB	0	0	0	0	1	0	0	0
	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL
UCSRC	1	0	0	0	0	1	1	0

ادامه ی حل مسئله

$$19200 = \frac{11.0592 \text{ MHz}}{16(UBRR+1)} \quad \longrightarrow \quad UBRR = 23H$$

UBRRH	UBRRL
00H	23

در صورتی که نتیجه محاسبه به صورت دقیق درنیامد، نزدیکترین مقدار را به عنوان نرخ انتقال در نظر می گیریم. در اینصورت، مقدار نرخ انتقال با خطا همراه خواهد بود.

محاسبه خطای Baud Rate

$$\text{Error}[\%] = \left(\frac{\text{BaudRate}_{\text{Closest Match}}}{\text{BaudRate}} - 1 \right) \cdot 100\%$$

ارسال و دریافت سریال

- ❑ ارسال سریال : فرستنده ی سریال با یک کردن بیت فعال ساز ارسال (TXEN) در رجیستر UCSRB فعال خواهد شد. در این حالت فرستنده آماده ی ارسال سریال از پایه ی TXD است.
- ❑ قبل از قرار دادن اطلاعات در رجیستر UDR باید از وضعیت خالی بودن بافر فرستنده اطلاع حاصل نمود. به این منظور بیت UDRE را بررسی کرده و در صورت یک بودن آن، می توان در UDR نوشت.
- ❑ بیت نهم (در صورت وجود) قبل از نوشتن اطلاعات در UDR، در بیت TXB8 نوشته می شود.

ارسال و دریافت سریال

- ❑ دریافت سریال : عملکرد دریافت سریال با نوشتن یک در بیت RXEN از رجیستر UCSRB فعال خواهد شد. دریافت سریال با آشکار سازی یک بیت شروع روی پایه ی RXD آغاز می گردد و با توجه به نرخ دریافت و یا پالس ساعت پایه ی XCK نمونه برداری شروع می شود و داده ها به شیفت رجیستر گیرنده وارد خواهد شد، سپس می توان با خواندن UDR اطلاعات را به دست آورد.
- ❑ بیت نهم (در صورت وجود) قبل از خواندن اطلاعات از UDR، از بیت RXB8 نوشته می شود.
- ❑ قبل از خواندن از رجیستر UDR باید بیت RXC بررسی شود.

استفاده از وقفه ارسال برای ارتباط سریال

در سمت فرستنده یعنی Master دو پرچم زیر می توانند برای تولید وقفه به کار روند :

- پرچم UDRE : نشانه ی خالی بودن بافر ارسال
- پرچم TXC : نشانه ی تکمیل ارسال

ابتدا بیت فعال ساز وقفه ی خالی بودن رجیستر (UDRIE) را یک کرده و سپس با فعال شدن پرچم UDRE، به شرط اینکه وقفه ی سراسری فعال باشد، وقفه ی مربوطه فعال خواهد شد. با **نوشتن در رجیستر UDR** پرچم UDRE پاک می شود.

بیت TXC هنگامی که **ارسال به طور کامل** انجام شود فعال خواهد شد. برای استفاده از آن وقفه باید بیت TXCIE از رجیستر UCSRB را یک کرده تا وقفه ی TXC فعال شود.

استفاده از وقفه دریافت برای ارتباط سریال

□ در سمت گیرنده یعنی Slave، با **دریافت کامل اطلاعات**، پرچم RXC فعال خواهد شد که نشان دهنده ی وجود اطلاعات خوانده نشده در رجیستر UDR است

□ این پرچم هنگامی که بافر گیرنده UDR خالی باشد صفر است.

□ برای استفاده از وقفه ی دریافت:

□ ابتدا باید بیت RXCIE از رجیستر UCSRB یک شود

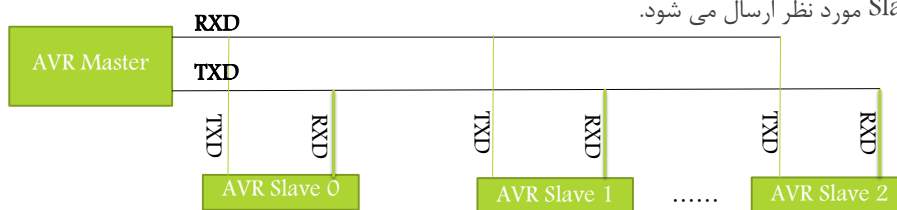
□ در صورت فعال بودن وقفه ی سراسری وقفه ی، وقتی RXC فعال شد، وقفه دریافت سرویس دهی شود.

□ سرویس دهی به این صورت است که، اطلاعات موجود در UDR خوانده شده و پرچم RXC پاک گردد.

مد ارتباطی چند پردازنده ای

این مد بیشتر برای شبکه بندی بین میکروکنترلرها استفاده می شود. با یک نمودن بیت MPCM از رجیستر UCSRA فعال خواهد شد.

در این حالت به طور مثال یک میکروکنترلر Master و تعدادی میکروکنترلر Slave وجود دارد. زمانی که Master قصد دارد به یکی از Slave ها دسترسی داشته باشد باید آن ها را آدرس دهی کند، سپس اطلاعات به Slave مورد نظر ارسال می شود.



مد ارتباطی چند پردازنده ای

می توان در Master از قالب داده 9 بیتی استفاده کرد. در این حالت:

بیت نهم (TXB8) هنگامی که اطلاعات ارسالی شامل آدرس باشد یک و در صورتی که اطلاعات ارسالی شامل داده باشد صفر می شود.

در ابتدا تمام Slave ها باید در مد چند پردازنده ای قرار بگیرند ($MPCM=1$).

در ابتدا Master یک فریم آدرس را به تمام Slave ها ارسال می کند.

آن ها با مقایسه فریم دریافتی با آدرس خود متوجه می شوند که انتخاب شده اند یا خیر

اگر انتخاب شده بودند بیت MPCM خود را پاک می کنند و منتظر دریافت داده می مانند و در غیر این صورت همچنان منتظر دریافت آدرس خواهند بود.

مد ارتباطی چند پردازشگر

- در سمت گیرنده، می توان فریم های دریافتی از نوع **آدرس** و **داده** را از هم جدا نمود.
- اگر در گیرنده $MPCM=1$ باشد (یعنی تا الان این **Slave** **انتخاب نشده است**) فریم هایی که شامل آدرس هستند (یعنی آنهایی که $TXB8 = 1$) در بافر UDR قرار می گیرند تا بررسی شود که آیا این **Slave** برای ارتباط بعدی انتخاب شده است یا خیر.
- اگر در گیرنده $MPCM=0$ باشد، (یعنی پیش از این، این **Slave** انتخاب شده بود)، هر دو فریم آدرس و داده در بافر UDR قرار می گیرد.
- این روش باعث کاهش تعداد فریم های دریافتی و درگیر شدن CPU در سمت **Slave**هایی می شود که برای ارتباط انتخاب نمی شوند. این **Slave**ها تنها زمانی فریم روی باس را دریافت و پردازش می کنند که یک فریم آدرس باشد.