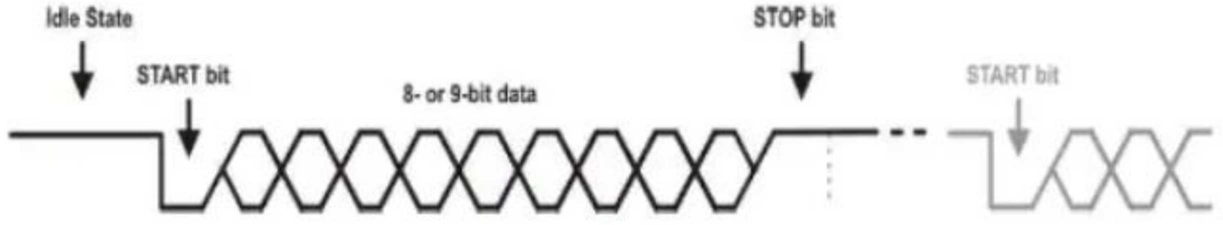
U n i v e r s a l
S y n c h r o n o u s
A s y n c h r o n o u s
R e c e i v e r
T r a n s m i t e r



USART

BY: Eng\ Rawan Adel

About

- Full Duplex, serial i/o communication peripheral
- Contains: shift registers , clock generation, Data buffer (for serial communication).
- Modes: Synchronous / Asynchronous
- It uses 2 i/o ports for transmiting and receiving of serial data (may occur at same time as it’s a full duplex).

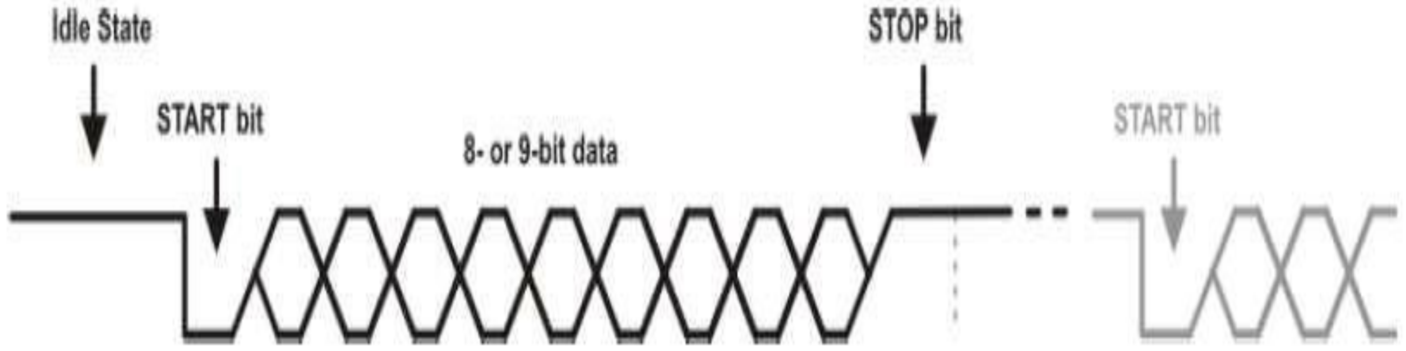
About

- To send a byte, the application writes the byte to the transmit buffer.
- The UART then sends the data, bit by bit in the requested format, adding Stop, Start, and parity bits as needed.
- In a similar way, the UART stores received bytes in a buffer.
- Then the UART can generate an interrupt to notify the application or software can poll the port to find out if data has arrived.

❖ **Asynchronous Mode:**

Data transfer happens in the following way:

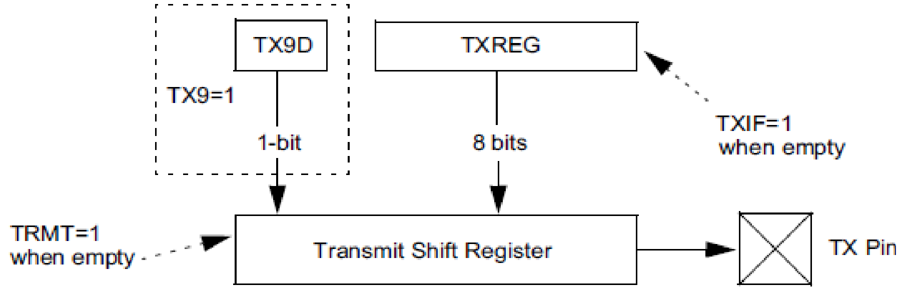
1. In idle state, data line has logic high (1).
2. Data transfer starts with a start bit, which is always a zero.
3. Data word is transferred (8 or 9 bit), LSB is sent first.
4. Each word ends with a stop bit, which is always high (1).
5. Another byte can be sent directly after, and will start also with astart bit befor data.

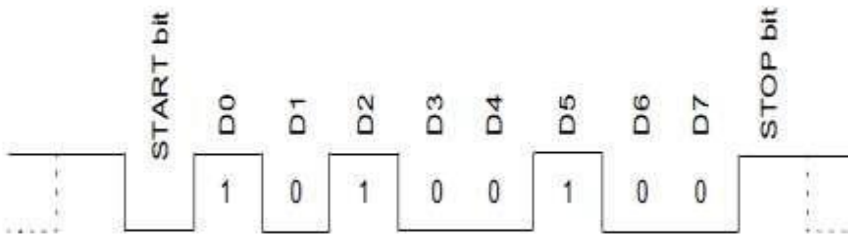


- The Nine-bit mode is useful when parity or an extra STOP bit is needed.
- Even if Parity is not supported by the hardware, but can be implemented in software.
- The ninth bit can also be used for the Addressable mode.
- The **TXSTA** and **RCSTA** registers are used to control transmission and reception but there are some overlapping functions and both registers are always used.
- The USART peripheral consists of three main parts:
  - ❑ Transmitter.
  - ❑ Receiver.
  - ❑ Baud generator.

❖ Transmitter

1. The module is enabled by setting the TXEN bit.
2. Data to be sent should be written into the TXREG register. When using 9- bit, TX9D must be written before writing TXREG.
3. Byte will be immediately transferred to the shift register TSR after the STOP bit from the pervious load is sent.
4. From there, data will be clocked out onto the TX pin preceded by a START bit and followed by a STOP bit





❖ Transmitter

4. Every bit is sent in the same amount of time, beginning with the LSB.
5. After data has been shifted (on the start of the STOP bit), the TRMT bit in the TXSTA register will be set.
6. A new byte can be sent in the same way beginning with a new start bit.

❖ Transmitter

TXIF bit : in the PIR1 register

- indicates when data can be written to TXREG (when data is moved from TXREG into the Transmit Shift Register,It cannot be cleared in software. It will reset only when new data isloaded into the TXREG register.it doesn’t indicate that the transmission has completed.)

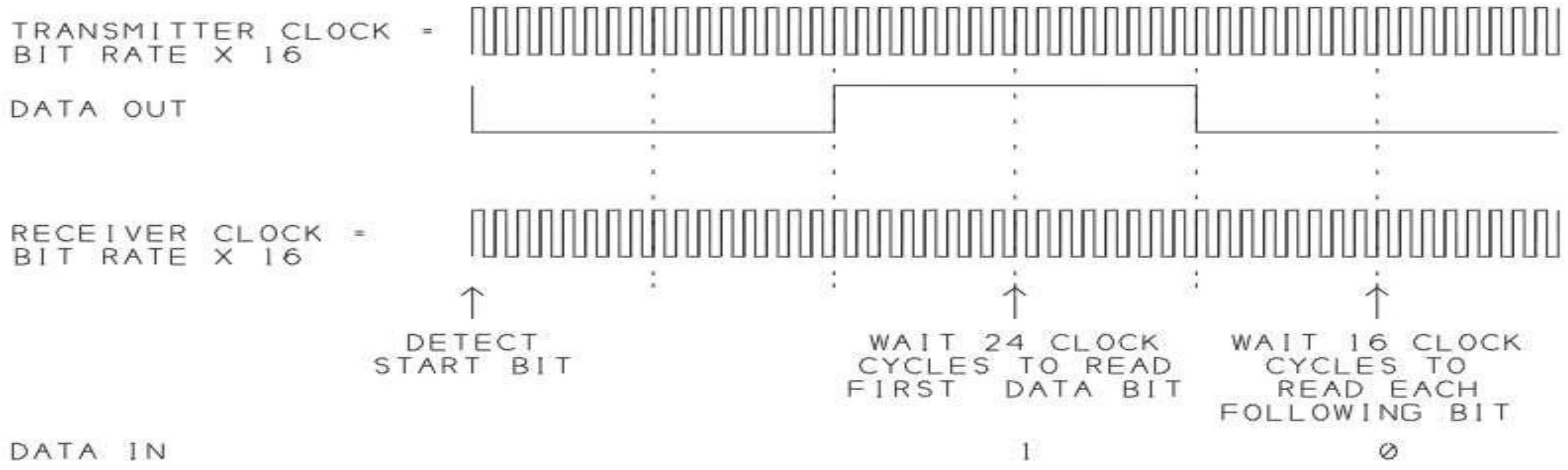
TRMT bit:

- Once the data in the TSR register has been clocked out on the TX pin (at the beginning of the STOP bit), the TRMT bit in the TXSTA register will be set,indicating that the transmission has been completed.

❖ Receiver

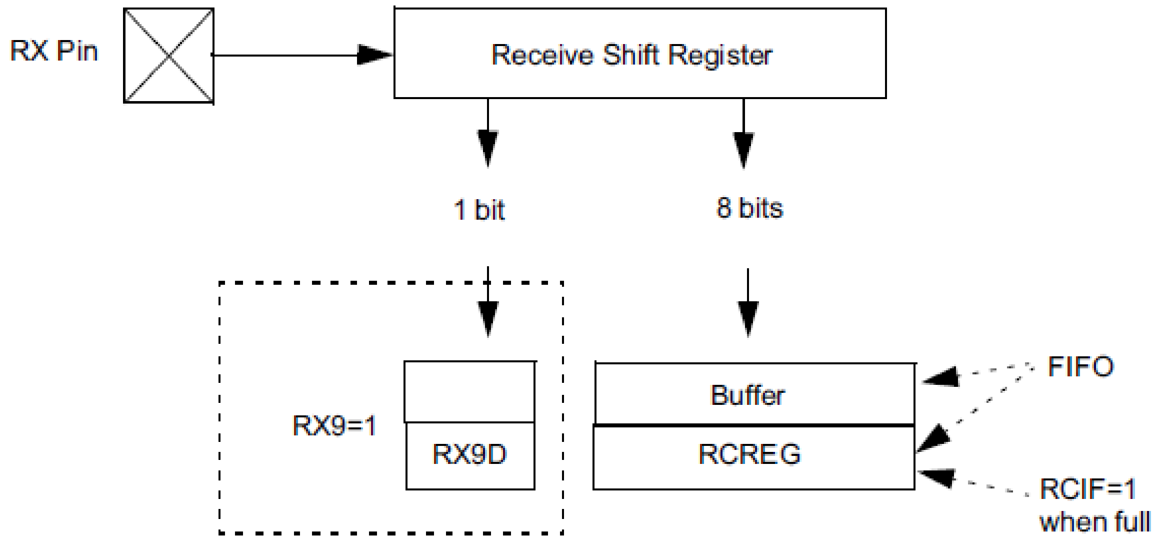
1- The clock of the receiver is a multiple of the bit rate, in PIC 16f877A,it's x16 or x64
So, each bit is transmitted/received in 16 clock cycle.

2- If the receiver detects a start bit for a period= bit period (16 clock cycles), then it waits for the period of half bit, and then sample the value on the RX pin and shift it in the receiving shift register.


TRANSMITTER CLOCK = BIT RATE X 16
DATA OUT
RECEIVER CLOCK = BIT RATE X 16
DETECT START BIT
WAIT 24 CLOCK CYCLES TO READ FIRST DATA BIT
WAIT 16 CLOCK CYCLES TO READ EACH FOLLOWING BIT
DATA IN
1
0

❖ Receiver

3. Every received bit is sampled at the middle of the bit's time period.
4. The USART can be configured to receive eight or nine bits by the RX9 bit in the RCSTA register.
5. After the detection of a START bit, eight or nine bits of serial data are shifted from the RX pin into the Receive Shift Register, one bit at a time.
6. After the last bit has been shifted in, the STOP bit is checked and the data is moved into the FIFO buffer.



❖ Receiver

7. RCREG is the output of the two element FIFO buffer. A next start
   - bit can be sent immediately after the stop bit

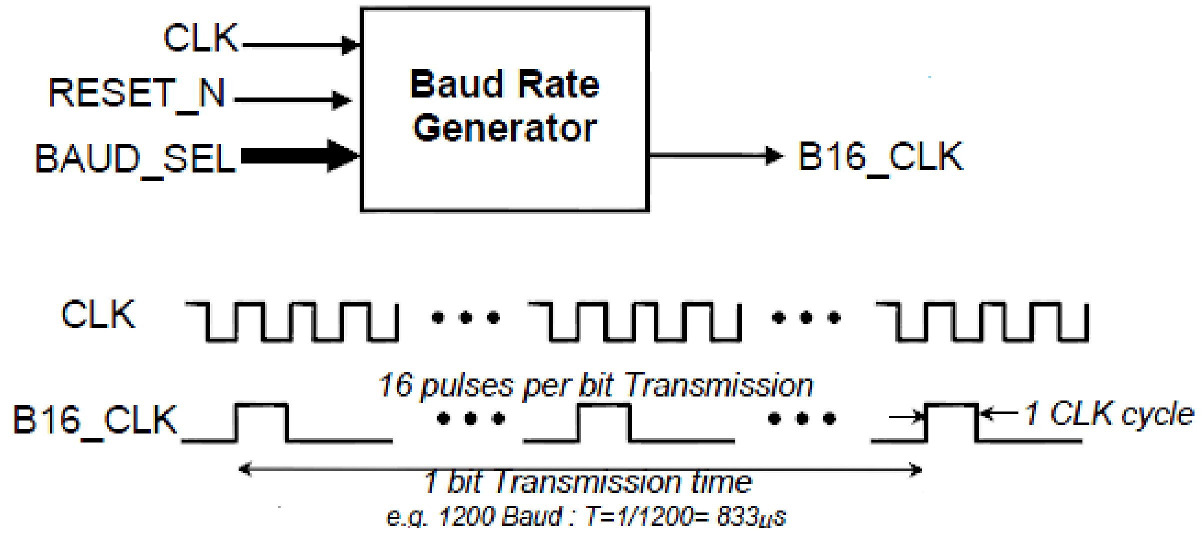
❖ **<u>RCIF:</u>** indicates when data is available in the RCREG.

- It is read only bit, cleared by hardware when the RCREG register has been read and is empty.
- If two bytes have been received, the RCIF bit will remain set until all the data has been read from RCREG.

❖ Baud Rate Generator

As we said, each bit is received/transmitted using a clock that is a multiple of the bit rate (x16 or x64).
So, we have to get this clock frequency from the oscillator frequency (Fosc) using the baud rate generator.
The BRG supports both the Asynchronous and Synchronous modes.

It is a dedicated 8-bit baud rate generator.



The End