Contents

[**Introduction** 2](#_Toc94281544)

[**USART Baud Rate Generator (BRG)** 4](#_Toc94281545)

[**USART Synchronous Master Mode** 5](#_Toc94281546)

[**USART Synchronous Master Transmission** 5](#_Toc94281547)

[**Steps to follow when setting up a Synchronous Master Transmission** 5](#_Toc94281548)

[**USART Synchronous Master Reception** 6](#_Toc94281549)

[**USART Synchronous Slave Mode** 7](#_Toc94281550)

[**USART Synchronous Slave Transmit** 7](#_Toc94281551)

[**Steps to follow when setting up a Synchronous Slave Transmission** 7](#_Toc94281552)

[**USART Synchronous Slave Reception** 8](#_Toc94281553)

[**Code** 9](#_Toc94281554)

[**MASTER SYCHRONOUS USART MODE (TRANSMIT)** 9](#_Toc94281555)

[**SLAVE SYCHRONOUS USART MODE (RECIEVE):** 10](#_Toc94281556)

[Figure 1 Transmit status and control register datasheet 2](#_Toc94281479)

[Figure 2 receive status and control register datasheet 3](#_Toc94281480)

[Figure 3 Registers associated with baud rate generator 4](#_Toc94281481)

[Figure 4 Baud rate formula 4](#_Toc94281482)

[Figure 5 Baud rates for synchronous mode 4](#_Toc94281483)

[Figure 6 Synchronous master transmission registers 6](#_Toc94281484)

[Figure 7 Synchronous master reception registers 6](#_Toc94281485)

[Figure 8 Synchronous slave transmission registers 7](#_Toc94281486)

[Figure 9 synchronous slave reception registers 8](#_Toc94281487)

# **Introduction**

The Universal Synchronous Asynchronous Receiver Transmitter (USART) module is one of the two serial I/O modules. The USART is also known as a Serial Communications Interface or SCI. The USART can be configured as a full duplex asynchronous system that can communicate with peripheral devices such as CRT terminals and personal computers, or it can be configured as a half-duplex synchronous system that can communicate with peripheral devices such as A/D or D/A integrated circuits, Serial EEPROMs etc.

The USART can be configured in the following modes:

* Asynchronous (full duplex)
* Synchronous - Master (half duplex)
* Synchronous - Slave (half duplex)

The SPEN bit (RCSTA), and the TRIS bits, must be set to configure the TX/CK and RX/DT pins for the USART.

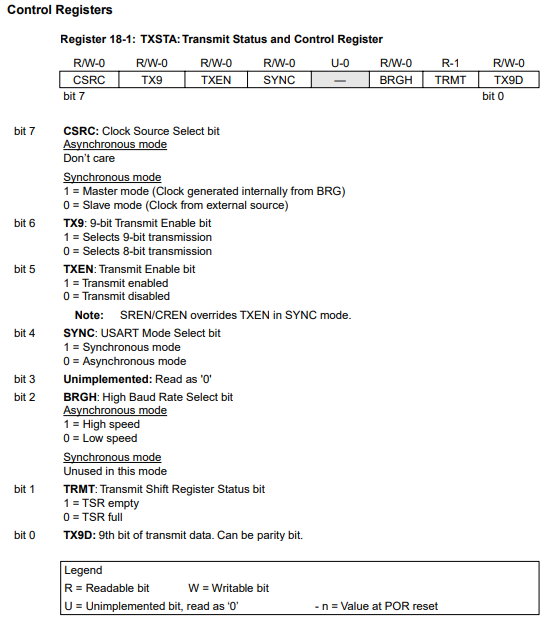


Figure 1 Transmit status and control register datasheet

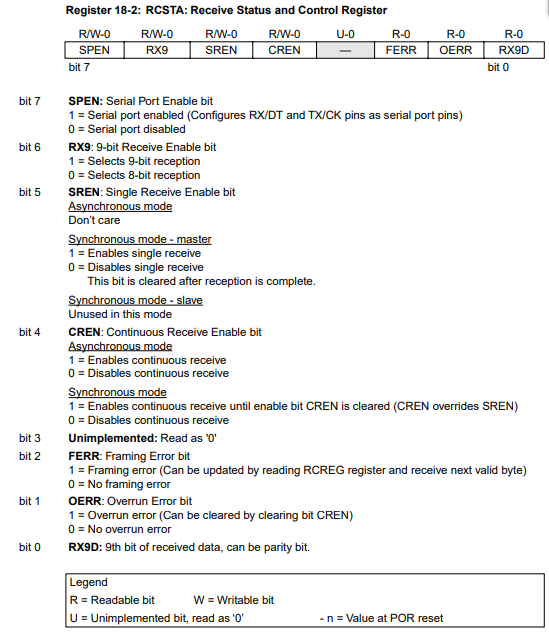


Figure 2 receive status and control register datasheet

# **USART Baud Rate Generator (BRG)**

The BRG supports both the Asynchronous and Synchronous modes of the USART. It is a dedicated 8-bit baud rate generator. The SPBRG register controls the period of a free running 8-bit timer. In synchronous mode bit BRGH is ignored. Table 18-1 shows the formula for computation of the baud rate for different USART modes which only apply in master mode (internal clock). Given the desired baud rate and frequency oscillation, the nearest integer value for the SPBRG register can be calculated using the formula in Table 18-1, where X equals the value in the SPBRG register (0 to 255). From this, the error in baud rate can be determined. Writing a new value to the SPBRG register causes the BRG timer to be reset (or cleared). This ensures the BRG does not wait for a timer overflow before outputting the new baud rate.

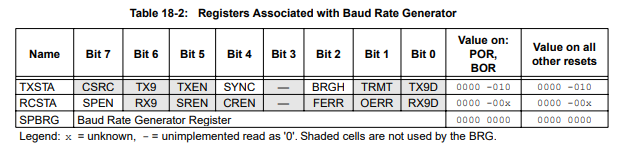


Figure 3 Registers associated with baud rate generator

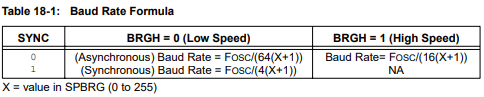


Figure 4 Baud rate formula

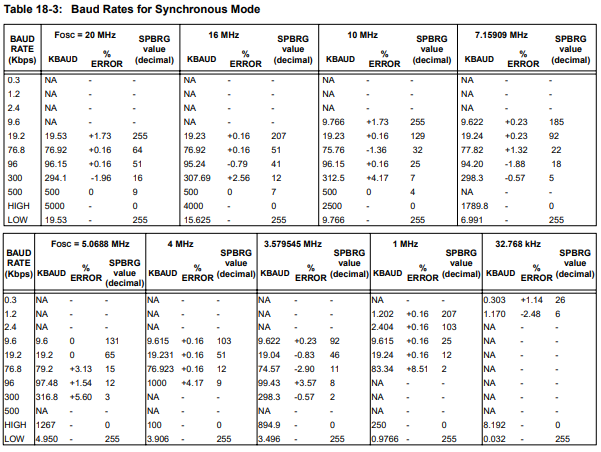


Figure 5 Baud rates for synchronous mode

# **USART Synchronous Master Mode**

In Synchronous Master mode, the data is transmitted in a half-duplex manner, i.e. transmission and reception do not occur at the same time. When transmitting data, the reception is inhibited and vice versa. Synchronous mode is entered by setting the SYNC bit (TXSTA). In addition, the SPEN enable bit (RCSTA) is set in order to configure the TX/CK and RX/DT I/O pins to CK (clock) and DT (data) lines respectively. The Master mode indicates that the processor transmits the master clock on the CK line. The Master mode is entered by setting the CSRC bit (TXSTA).

## **USART Synchronous Master Transmission**

The heart of the transmitter is the transmit (serial) shift register (TSR). The shift register obtains its data from the read/write transmit buffer register TXREG. The TXREG register is loaded with data in software. The TSR register is not loaded until the last bit has been transmitted from the previous load. As soon as the last bit is transmitted, the TSR is loaded with new data from the TXREG (if available). Once the TXREG register transfers the data to the TSR register (occurs in one period cycle), the TXREG is empty, and the TXIF interrupt flag bit is set. The interrupt can be enabled/disabled by setting/clearing enable the TXIE bit. The TXIF flag bit will be set regardless of the state of the TXIE enable bit and cannot be cleared in software. It will reset only when new data is loaded into the TXREG register. While the TXIF flag bit indicates the status of the TXREG register, the TRMT bit (TXSTA) shows the status of the TSR register. The TRMT bit is a read only bit which is set when the TSR is empty. No interrupt logic is tied to this bit, so the user has to poll this bit in order to determine if the TSR register is empty. The TSR is not mapped in data memory so it is not available to the user. Transmission is enabled by setting the TXEN bit (TXSTA). The actual transmission will not occur until the TXREG register has been loaded with data. The first data bit will be shifted out on the next available rising edge of the clock on the CK line. Data out is stable at the falling edge of the synchronous clock (Figure 18-10). The transmission can also be started by first loading the TXREG register and then setting the TXEN bit. This is advantageous when slow baud rates are selected, since the BRG is kept in reset when the TXEN, CREN, and SREN bits are clear. Setting the TXEN bit will start the BRG, creating a shift clock immediately. Normally when transmission is first started, the TSR register is empty, so a transfer to the TXREG register will result in an immediate transfer to TSR resulting in an empty TXREG. Back-to-back transfers are possible. Clearing the TXEN bit during a transmission will cause the transmission to be aborted and will reset the transmitter. The DT and CK pins will revert to hi-impedance. If either of the CREN or SREN bits are set during a transmission, the transmission is aborted, and the DT pin reverts to a hi-impedance state (for a reception). The CK pin will remain an output if the CSRC bit is set (internal clock). The transmitter logic is not reset although it is disconnected from the pins. To reset the transmitter, the user must clear the TXEN bit. If the SREN bit is set (to interrupt an on-going transmission and receive a single word), then after the single word is received, the SREN bit will be cleared, and the serial port will revert to transmitting since the TXEN bit is still set. The DT line will immediately switch from hi-impedance receive mode to transmit and start driving.

## **Steps to follow when setting up a Synchronous Master Transmission**

1. Initialize the SPBRG register for the appropriate baud rate

2. Enable the synchronous master serial port by setting the SYNC, SPEN, and CSRC bits.

3. If interrupts are desired, then set the TXIE bit.

4. If 9-bit transmission is desired, then set the TX9 bit.

5. Enable the transmission by setting the TXEN bit.

6. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.

7. Start transmission by loading data to the TXREG register

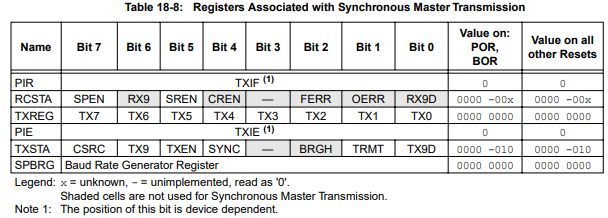


Figure 6 Synchronous master transmission registers

## **USART Synchronous Master Reception**

Once Synchronous mode is selected, reception is enabled by setting either of the SREN (RCSTA) or CREN (RCSTA) bits. Data is sampled on the RX/DT pin on the falling edge of the clock. If the SREN bit is set, then only a single word is received. If the CREN bit is set, the reception is continuous until the CREN bit is cleared. If both bits are set, then the CREN bit takes precedence. After clocking the last serial data bit, the received data in the Receive Shift Register (RSR) is transferred to the RCREG register (if it is empty). When the transfer is complete, the RCIF interrupt flag bit is set. The actual interrupt can be enabled/disabled by setting/clearing the RCIE enable bit. The RCIF flag bit is a read only bit which is cleared by the hardware. In this case it is cleared when the RCREG register has been read and is empty. The RCREG is a double buffered register, i.e. it is a two deep FIFO. It is possible for two bytes of data to be received and transferred to the RCREG FIFO and a third byte to begin shifting into the RSR register. On the clocking of the last bit of the third byte, if the RCREG register is still full then overrun error bit, OERR (RCSTA), is set and the word in the RSR is lost. The RCREG register can be read twice to retrieve the two bytes in the FIFO. The OERR bit must be cleared in software (by clearing the CREN bit). If the OERR bit is set, transfers from the RSR to the RCREG are inhibited, so it is essential to clear the OERR bit if it is set. The 9th receive bit is buffered the same way as the receive data. Reading the RCREG register will load the RX9D bit with a new value, therefore it is essential for the user to read the RCSTA register before reading RCREG in order not to lose the old (previous) information in the RX9D bit. Steps to follow when setting up a Synchronous Master Reception: 1. Initialize the SPBRG register for the appropriate baud rate. (Subsection 18.3 “USART Baud Rate Generator (BRG)” ) 2. Enable the synchronous master serial port by setting the SYNC, SPEN, and CSRC bits. 3. Ensure that the CREN and SREN bits are clear. 4. If interrupts are desired, then set the RCIE bit. 5. If 9-bit reception is desired, then set the RX9 bit. 6. If a single reception is required, set the SREN bit. For continuous reception set the CREN bit. 7. The RCIF bit will be set when reception is complete, and an interrupt will be generated if the RCIE bit was set. 8. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception. 9. Read the 8-bit received data by reading the RCREG register. 10. If any error occurred, clear the error by clearing the CREN bit.

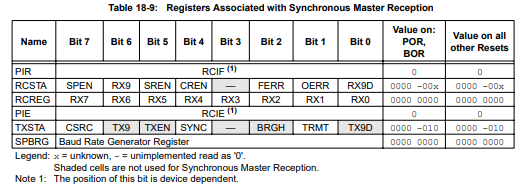


Figure 7 Synchronous master reception registers

# **USART Synchronous Slave Mode**

Synchronous slave mode differs from the Master mode in the fact that the shift clock is supplied externally at the TX/CK pin (instead of being supplied internally in master mode). This allows the device to transfer or receive data while in SLEEP mode. Slave mode is entered by clearing the CSRC bit (TXSTA).

## **USART Synchronous Slave Transmit**

The operation of the synchronous master and slave modes are identical except in the case of the SLEEP mode. If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

1. The first word will immediately transfer to the TSR register and transmit.
2. The second word will remain in TXREG register.
3. The TXIF flag bit will not be set.
4. When the first word has been shifted out of TSR, the TXREG register will transfer the second word to the TSR and the TXIF flag bit will now be set.
5. If the TXIE enable bit is set, the interrupt will wake the chip from SLEEP and if the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

**Steps to follow when setting up a Synchronous Slave Transmission**

1. Enable the synchronous slave serial port by setting the SYNC and SPEN bits and clearing the CSRC bit.
2. Clear the CREN and SREN bits.
3. If interrupts are desired, then set the TXIE enable bit.
4. If 9-bit transmission is desired, then set the TX9 bit.
5. Enable the transmission by setting the TXEN enable bit.
6. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D bit.
7. Start transmission by loading data to the TXREG register.

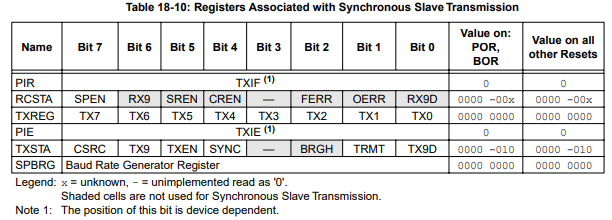


Figure 8 Synchronous slave transmission registers

## **USART Synchronous Slave Reception**

The operation of the synchronous master and slave modes is identical except in the case of the SLEEP mode. Also, bit SREN is a don't care in slave mode. If receive is enabled, by setting the CREN bit, prior to the SLEEP instruction, then a word may be received during SLEEP. On completely receiving the word, the RSR register will transfer the data to the RCREG register and if the RCIE enable bit is set, the interrupt generated will wake the chip from SLEEP. If the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Steps to follow when setting up a Synchronous Slave Reception:

* Enable the synchronous master serial port by setting the SYNC and SPEN bits and clearing the CSRC bit.
* If interrupts are desired, then set the RCIE enable bit.
* If 9-bit reception is desired, then set the RX9 bit.
* To enable reception, set the CREN enable bit.
* The RCIF bit will be set when reception is complete and an interrupt will be generated, if the RCIE bit was set.
* Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
* Read the 8-bit received data by reading the RCREG register.
* If any error occurred, clear the error by clearing the CREN bit.

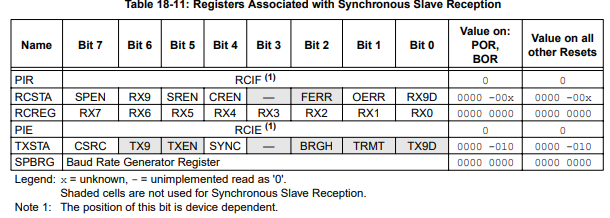


Figure 9 synchronous slave reception registers

# **Code**

## **MASTER SYCHRONOUS USART MODE (TRANSMIT)**

#INCLUDE "P16F877A.INC" ;

\_\_CONFIG \_CP\_OFF & \_WDT\_OFF & \_PWRTE\_OFF & \_BODEN\_OFF & \_LVP\_OFF & \_HS\_OSC ;

REG1 EQU 0X20 ;

REG2 EQU 0X21 ;

REG3 EQU 0X22 ;

TEMP EQU 0X23 ;

ORG 0x00

CONFI BSF STATUS,RP0

MOVLW H'FF'

MOVWF TRISB ;PORT B INPUT

MOVLW H'FF'

MOVWF TRISC ;PORT C USART MODE

MOVLW D'12'

MOVWF SPBRG

MOVLW H'B0' ; Synchronous Master,8-bit transmit

MOVWF TXSTA

BCF STATUS,RP0

MOVLW H'80' ; Synchronous Master,8-bit transmit

MOVWF RCSTA

MAIN BSF STATUS,RP0

BTFSS TXSTA,TRMT ; TEST IF TRMT IS EMPTY

GOTO MAIN ; KEEP ON TESTING IF FULL

BCF STATUS,RP0

MOVF PORTB,W ; ELSE START SENDING DATA

MOVWF TXREG

DELAY MOVLW D'1' ; DELAY 100 MS

MOVWF REG3

REFILL2 MOVLW D'98'

MOVWF REG2

REFILL1 MOVLW D'255'

MOVWF REG1

CORE DECFSZ REG1,F

GOTO CORE

DECFSZ REG2,F

GOTO REFILL1

DECFSZ REG3,F

GOTO REFILL2

GOTO MAIN

END

## **SLAVE SYCHRONOUS USART MODE (RECIEVE):**

#INCLUDE "P16F877A.INC" ;

\_\_CONFIG \_CP\_OFF & \_WDT\_OFF & \_PWRTE\_OFF & \_BODEN\_OFF & \_LVP\_OFF & \_HS\_OSC ;

REG1 EQU 0X20 ;

REG2 EQU 0X21 ;

REG3 EQU 0X22 ;

TEMP EQU 0X23 ;

ORG 0x00

CONFI BSF STATUS,RP0

MOVLW H'00'

MOVWF TRISD ;PORT D OUTPUT

MOVLW H'FF'

MOVWF TRISC ;PORT C USART MODE

MOVLW D'12'

MOVWF SPBRG

MOVLW H'30' ; Synchronous slave,8-bit recieve

MOVWF TXSTA

BCF STATUS,RP0

MOVLW H'90' ; Synchronous slave,8-bit recieve

MOVWF RCSTA

MAIN BTFSS PIR1,RCIF ; TEST IF RCIF IS READY

GOTO MAIN ; KEEP ON TESTING IF EMPTY

MOVF RCREG,W ; ELSE START RECIEVING DATA

MOVWF PORTD

END