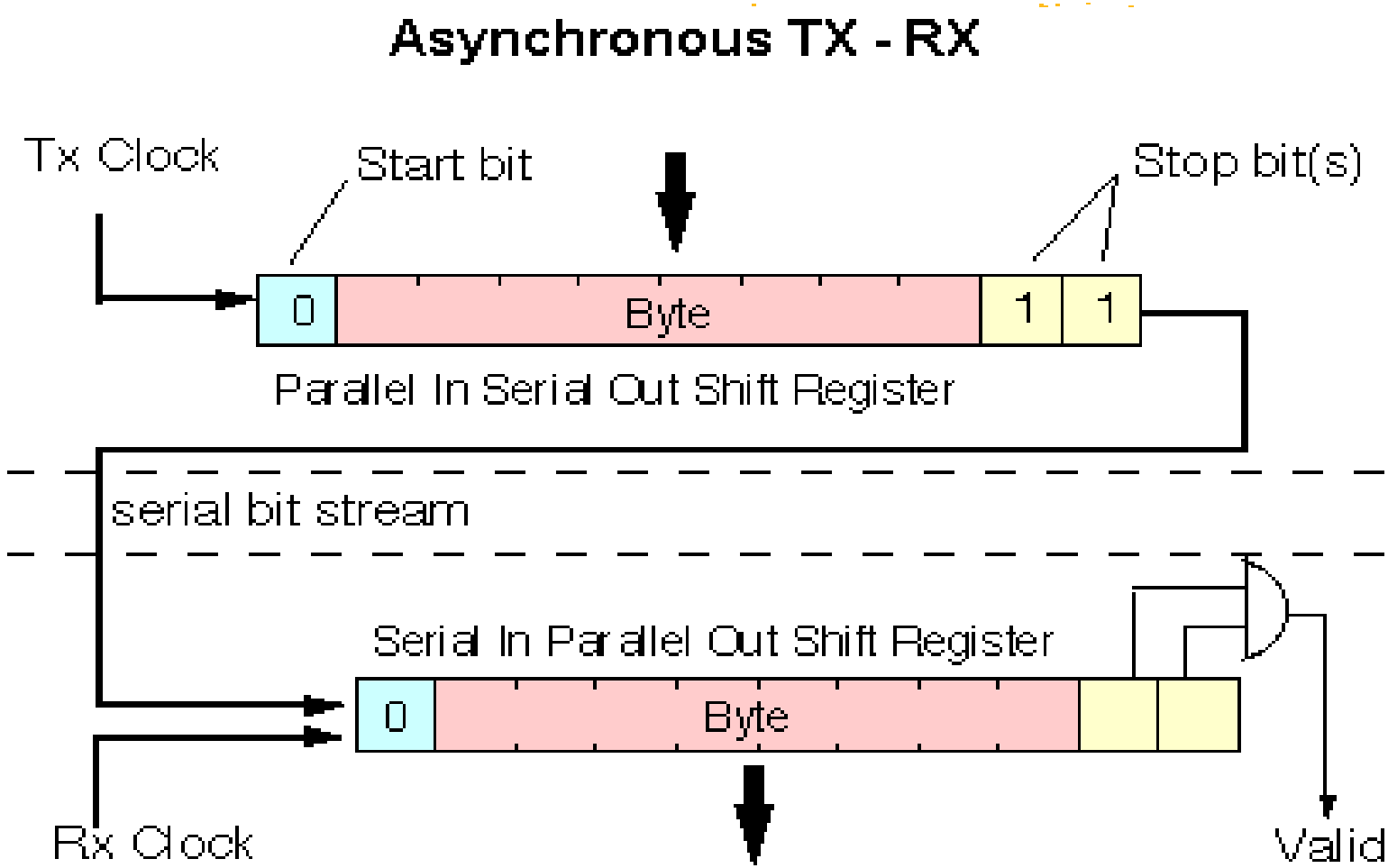
Serial Communication

- Serial Communication
  - Asynchronous Communication
    - Simple Interface
    - No clock sent
    - Requires Start and Stop bits
  - Synchronous Communication
    - More complex interface
    - Clock sent with data
    - Higher Rate

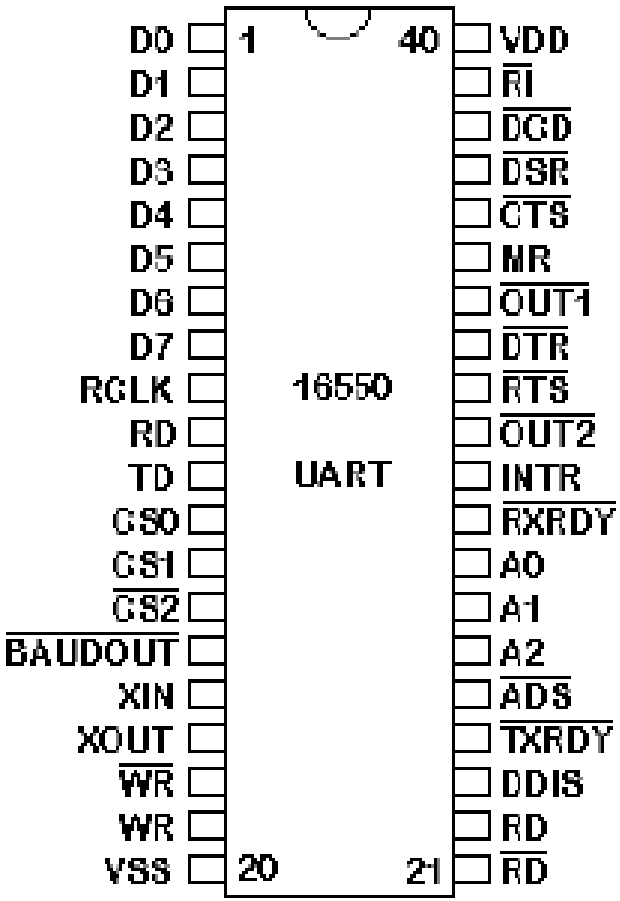
Asynchronous Communication

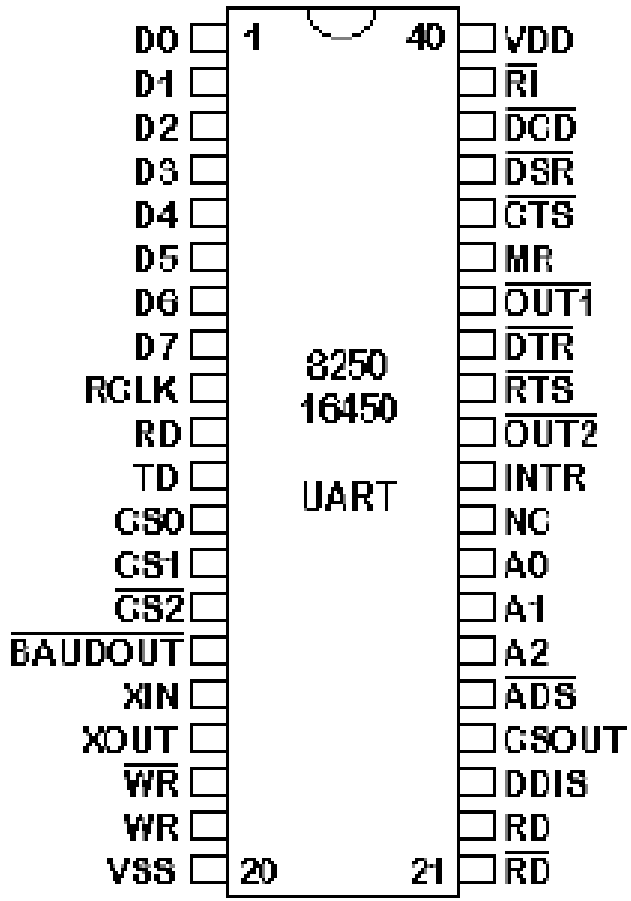


UART (Universal Asynchronous Receiver Transmitter)

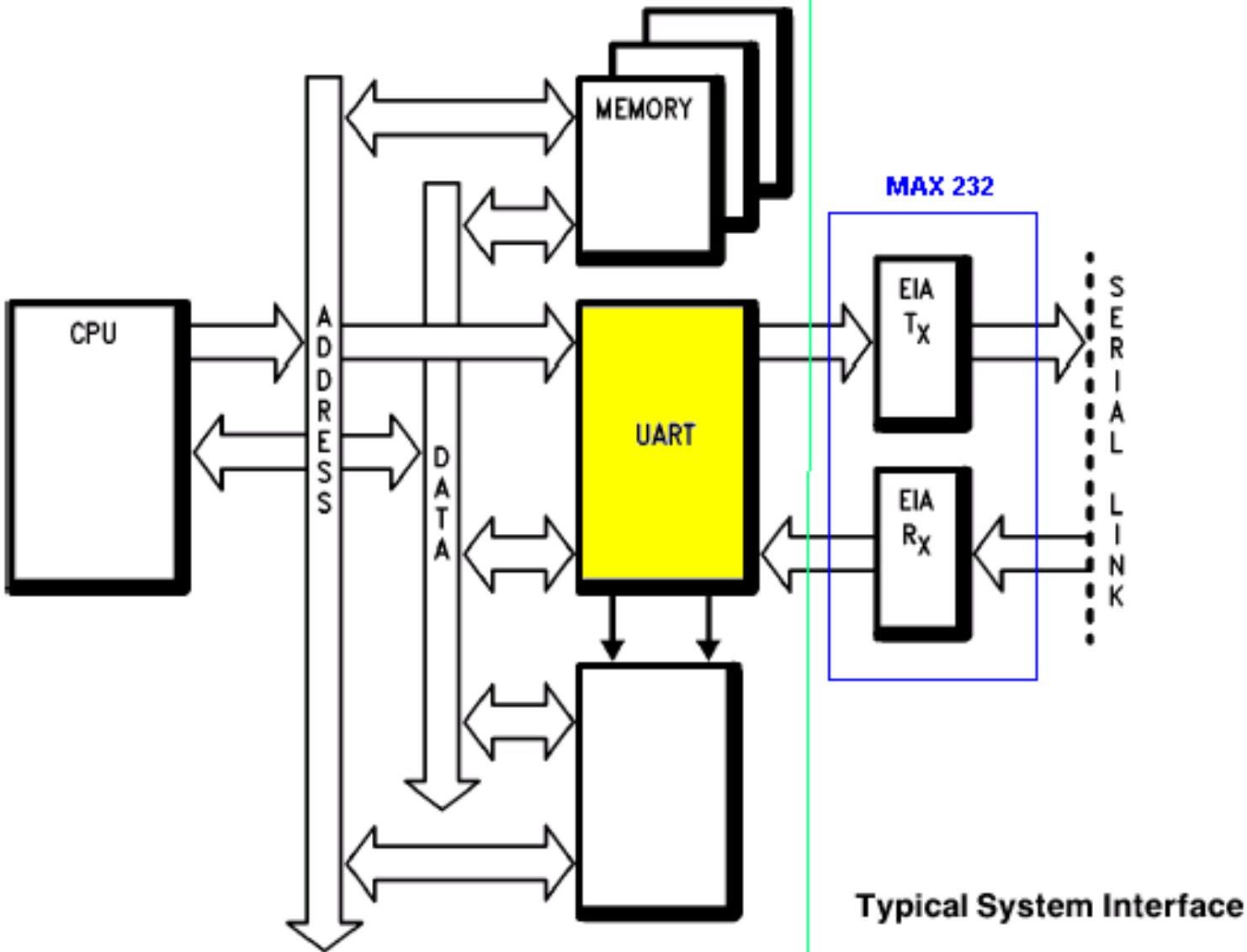
- Interface between CPU and Serial Port
- Converts the bytes it receives from **parallel** circuits into a single **serial bit stream** for outbound transmission
- On inbound transmission, converts the serial bit stream into the bytes that the computer handles
- Adds a **parity bit** (if it's been selected) on outbound transmissions and checks the parity of incoming bytes (if selected) and discards the parity bit
- Adds **start** and **stop** delineators on outbound and strips them from inbound transmissions
- Handles **interrupts** from the keyboard and mouse (which are serial devices with special ports)

Most Popular UART



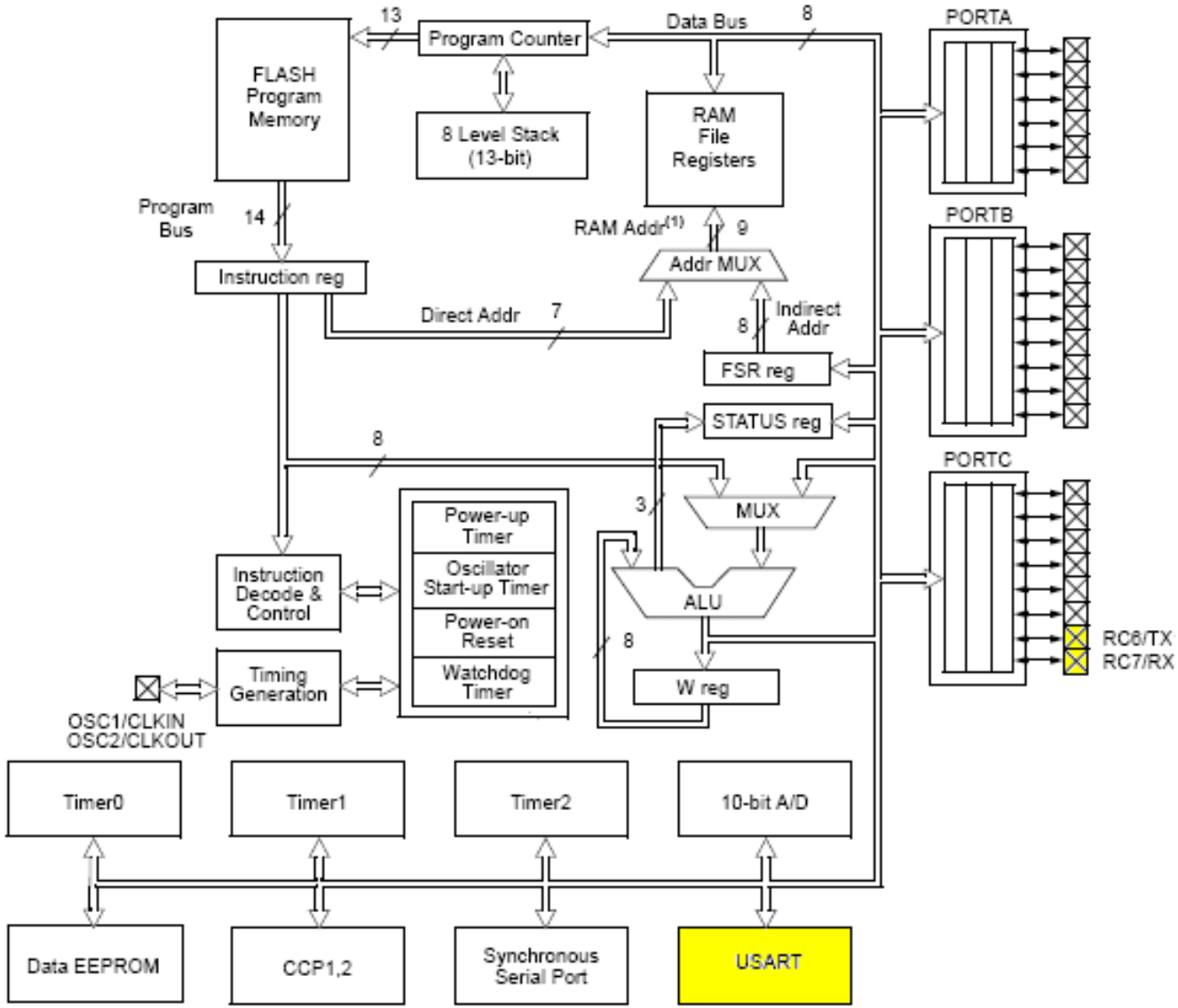


Typical Interface



Typical System Interface

USART Block of PIC



Registers for RX/TX

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit 7							bit 0

CSRC: Clock Source Select bit
Asynchronous mode
Don't care

TX9: 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
√ 0 = Selects 8-bit transmission

TXEN: Transmit Enable bit
1 = Transmit enabled
√ 0 = Transmit disabled

SYNC: USART Mode Select bit
1 = Synchronous mode
√ 0 = Asynchronous mode

Unimplemented: Read as '0'

BRGH: High Baud Rate Select bit
Asynchronous mode
1 = High speed
√ 0 = Low speed

TRMT: Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

TXSTA: 0 0 1 0 0 0 0 0

TX9D: 9th bit of transmit data.

Registers for TX/TX

RCSTA: Receive Status and Control Register

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R-0	R-0	R-0
SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D
bit 7							bit 0

SPEN: Serial Port Enable bit
1 = Serial port enabled
0 = Serial port disabled

RX9: 9-bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception

SREN: Single Receive Enable bit
Asynchronous mode
Don't care

CREN: Continuous Receive Enable bit
Asynchronous mode
1 = Enables continuous receive
0 = Disables continuous receive

Unimplemented: Read as '0'

FERR: Framing Error bit
1 = Framing error
0 = No framing error

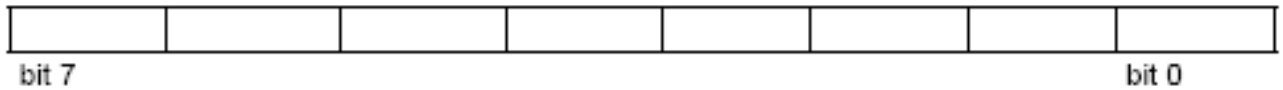
OERR: Overrun Error bit
1 = Overrun error
0 = No overrun error

RX9D: 9th bit of received data, can be parity bit.

RCSTA: 1 0 0 1 0 0 0 0

Registers for RX/TX

SPBRG: Baud Rate Generator



Baud Rate Formula

BRGH = 0 (Low Speed)	BRGH = 1 (High Speed)
(Asynchronous) Baud Rate = Fosc/(64(X+1))	Baud Rate= Fosc/(16(X+1))

X = value in SPBRG (0 to 255)

Baud Calculation and Error

1. Desired Baud Rate: 19200(B)

2. $X = \frac{F_{osc}}{64xB} - 1 = \frac{20{,}000{,}000}{(64)\ (19200)} - 1 = 15.27 \rightarrow X=15$

3. Calculated Baud rate with X: $\frac{20{,}000{,}000}{(64)\ (15+1)} = 19531$

4. Error = $\frac{19531 - 19200}{19200} = 1.7\ \%$

Registers for RX/TX

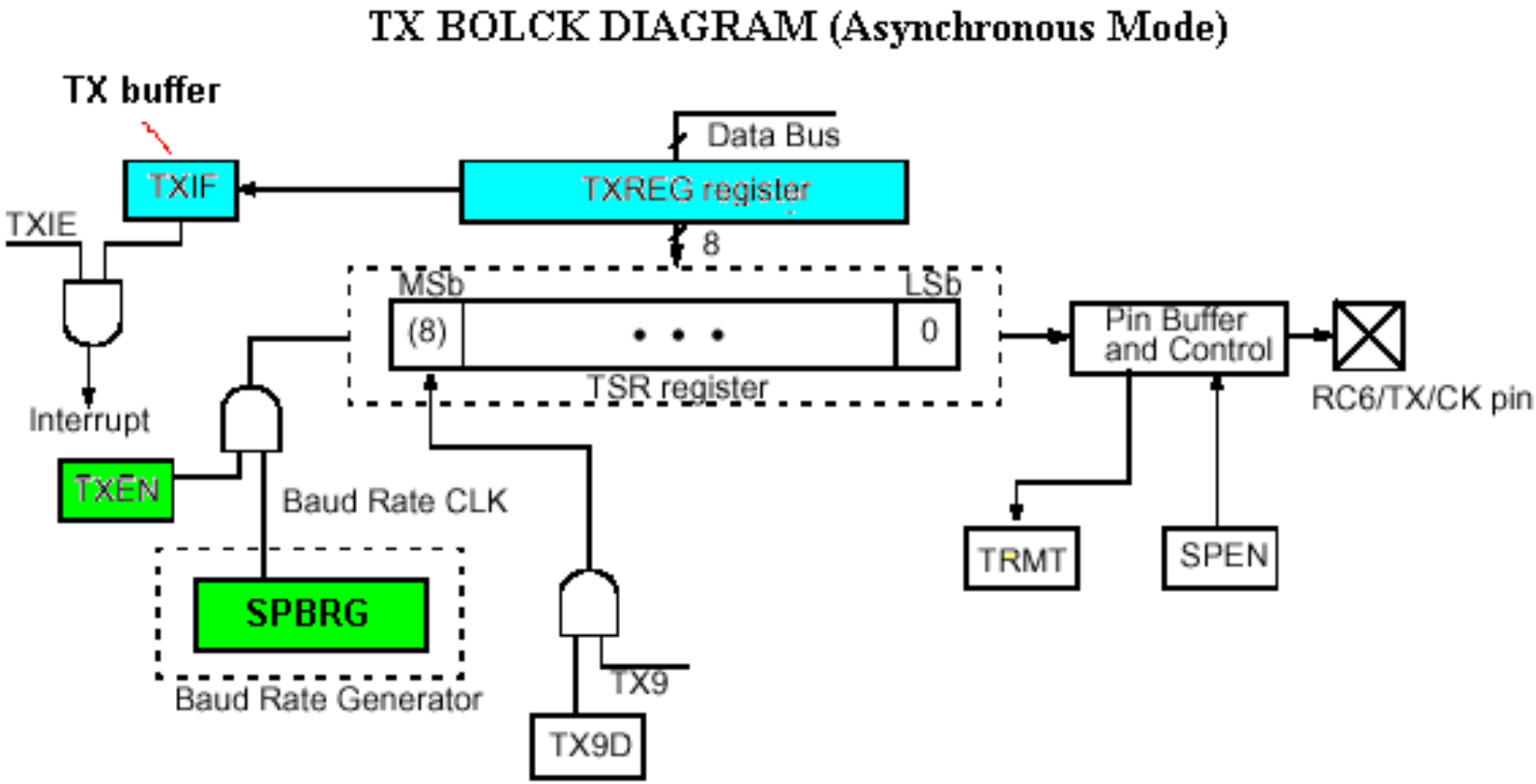
PIR1: Peripheral Interrupt Reg

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

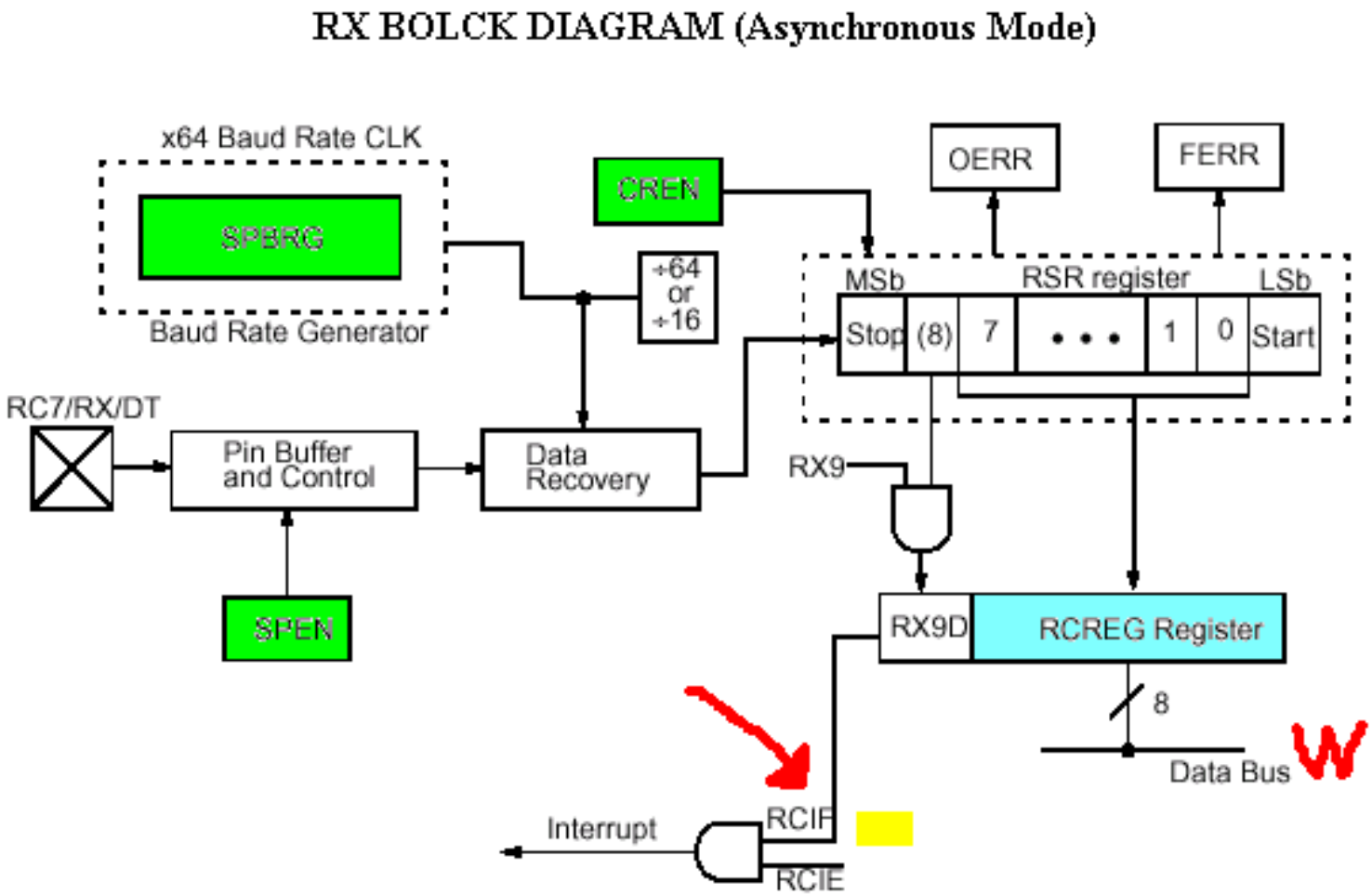
RCIF: USART Receive Interrupt Flag bit
1 = The USART receive buffer is full
0 = The USART receive buffer is empty

TXIF: USART Transmit Interrupt Flag bit
1 = The USART transmit buffer is empty
0 = The USART transmit buffer is full

PIC16F877 TX Block



PIC 16F877 RX Block



Registers

REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'.

Shaded cells are not used for asynchronous reception.

REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
19h	TXREG	USART Transmit Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'.

Shaded cells are not used for asynchronous transmission.

RX/TX PROGRAMMING

INITIALIZATION

BAUD RATE ----> SPBRG

TXMODE ---->TXSTA ← CREN & SPEN

RXMODE ---->RXSTA ← TXEN

RX POLLING (RXREG ---> W)

RCIF

RX BUFFER FULL? <--- PIR1<5>

TX POLLING (W ---> TXREG)

TXIF

TX BUFFER EMPTY? <--- PIR1<4>

```
call    ASYNC               ;Initialization

TX
        banksel PIR1
        btfss   PIR1, TXIF  ;TX buffer empty?
        goto    TX
        banksel TXREG
        movlw   '>'
        movwf   TXREG

RX
        banksel PIR1
        btfss   PIR1, RCIF  ;RX buffer full
        goto    RX
        banksel RXREG
        movf    RXREG,0     ;move the data to W
```

```
;Subroutine ASYNC

ASYNC
        banksel SPBRG
        movlw   H'0F'
        movwf   SPBRG
        banksel TXSTA
        movlw   B'00100000'
        movwf   TXSTA
        banksel RCSTA
        movlw   B'10010000'
        movwf   RCSTA
        return
```