CMPE 310
Systems Design and Programming
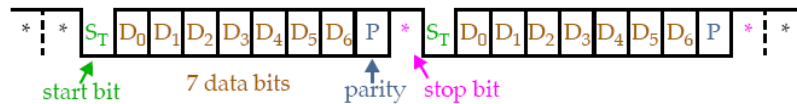
L13: Chapter 11 – BASIC I/O Interface

UMBC
AN HONORS UNIVERSITY IN MARYLAND

L13 Objectives

* Describe the function of each pin of the 16550
  * Diagram how the 16550 is connected to the x86/88 PC
  * Define the terms simplex, half duplex & full duplex
  * Program the 16550 (Initialization & Operation)

Programmable Communications Interface: 16550

* A universal asynchronous receiver/transmitter (UART).
* Operation speed: 0-1.5M Baud (Baud is number of bits transmitted/sec, including start, stop, data and parity).
* Includes:
    * A programmable Baud rate generator.
    * Separate FIFO buffers for input and output data (16 bytes each).
* Asynchronous serial data:
   * Transmitted and received without a clock or timing signal.

| * | * | S_T | D_0 | D_1 | D_2 | D_3 | D_4 | D_5 | D_6 | P | * | S_T | D_0 | D_1 | D_2 | D_3 | D_4 | D_5 | D_6 | P | * | * |

start bit   7 data bits   parity   stop bit

   * Two 10-bit frames of asynchronous data.
   * 7- or 8- bit ASCII can be used, e.g. w or w/o parity.

Pinout of the 16550

* A_0, A_1 and A_2: Select an internal register for programming and data transfer.

A_2	A_1	A_0	Register
0	0	0	Receiver buffer (read) and transmitter holding (write)
0	0	1	Interrupt enable
0	1	0	Interrupt identification (read) and FIFO control (write)
0	1	1	Line control
1	0	0	Modem control
1	0	1	Line status
1	1	0	Modem status
1	1	1	Scratch
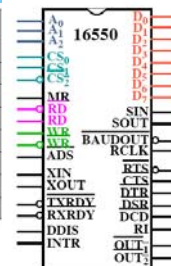
* $\overline{ADS}$: Address strobe used to latch address and chip select lines. Not needed on Intel systems - connected to ground.
* $\overline{BAUDOUT}$: Clock signal from Baud rate generator in transmitter.
* CS_0, CS_1, $\overline{CS_2}$: Chip selects
* CTS: Clear to send -- indicates that the modem or data set is ready to exchange information.

Pinout of the 16550

* D_7-D_0: The data bus pins are connected to the microprocessor data bus.
* DCD: The data carrier detect -- used by the modem to signal the 16550 that a carrier is present.
* DDIS: Disable driver output -- set to 0 to indicate that the microprocessor is reading data from the UART. Used to change direction of data flow through a buffer.
* $\overline{DSR}$: Data set ready is an input to 16550 -- indicates that the modem (data set) is ready to operate.
* DTR: Data terminal ready is an output -- indicates that the data terminal (16550) is ready to function.
* INTR: Interrupt request is an output to the microprocessor -- used to request an interrupt.
    * Receiver error
    * Data received
    * Transmit buffer empty
* MR: Master reset -- connect to system RESET

Pinout of the 16550

* $\overline{OUT1}$, $\overline{OUT2}$: User defined output pins for modem or other device.
* RCLK: Receiver clock -- clock input to the receiver section of the UART.
    * Always 16 times the desired receiver baud rate.
* RD, $\overline{RD}$: Read inputs (either can be used) -- cause data to be read from the register given by the address inputs to the UART.
* $\overline{RI}$: Ring indicator input -- set to 0 by modem to indicate telephone is ringing.
* $\overline{RTS}$: Request-to-send -- signal to modem, indicating UART wishes to send data.
* SIN, SOUT: Serial data pins, in and out.
* $\overline{RXRDY}$: Receiver ready -- used to transfer received data via DMA techniques.
* $\overline{TXRDY}$: Transmitter ready -- used to transfer transmitter data via DMA.
* WR, $\overline{WR}$: Write (either can be used) -- connects to microprocessor write signal to transfer commands and data to 16550.
* XIN, XOUT: Main clock connections -- a crystal oscillator can be used.

Example of 16550
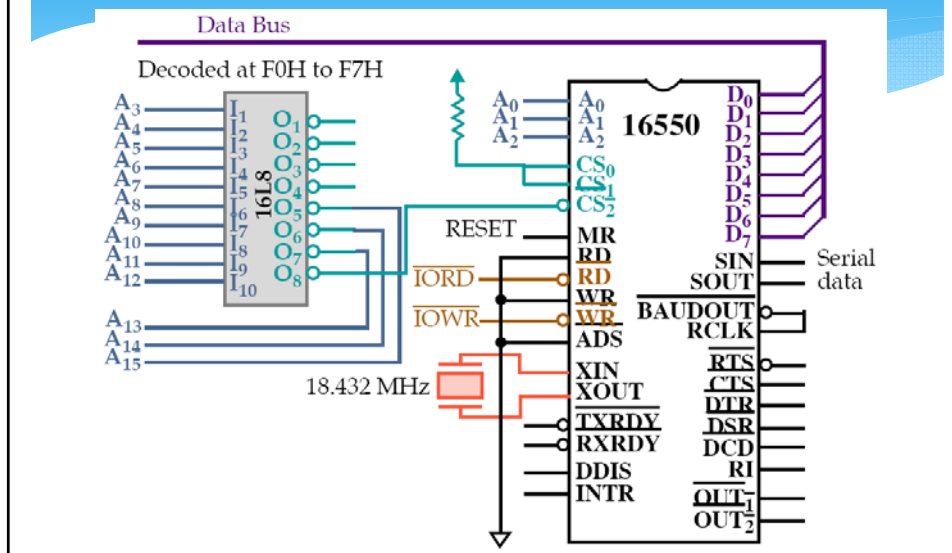
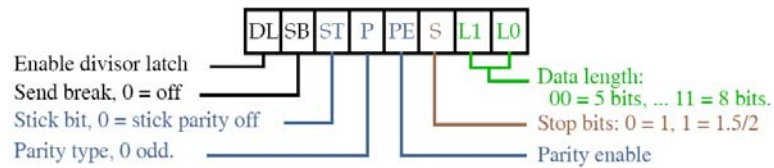## Programmable Communications Interface: 16550

* Two separate sections are responsible for data communications:
* ***Receiver & Transmitter***
* Can function in:
  * ***simplex:*** transmit only (unidirectional communication line)

  * ***half-duplex:*** transmit and receive but not simultaneously (bidirectional communication line) (CTS to turn the line around).

  * ***full-duplex:*** transmit and receive simultaneously (two communication lines)
    * Higher performance communication

* The 16550 can control a modem through $\overline{DSR}$, $\overline{DTR}$, $\overline{CTS}$, $\overline{RTS}$, $\overline{RI}$ and $\overline{DCD}$.
* In this context, the modem is called the ***data set*** while the 16550 is called the ***data terminal.***

Programming the 16550

* Two phases: Initialization, operation.
* Initialization:
    * After RESET, the *line control register* and *baud rate generator* need to be programmed.
    * Line control register selects the number of data bits, number of stop bits and the parity.
    * Addressed at location 011.

DL	SB	ST	P	PE	S	L1	L0

Enable divisor latch
Send break, 0 = off
Stick bit, 0 = stick parity off
Parity type, 0 odd.

Data length:
00 = 5 bits, ... 11 = 8 bits.
Stop bits: 0 = 1, 1 = 1.5/2
Parity enable

* Stop bits: S = 1, 1.5 stop bits used for 5 data bits, 2 used for 6, 7 or 8.

Programming the 16550

* Initialization (cont.)
    * ST, P and PE used to send even or odd parity, to send no parity or to send a 1 or a 0 in the parity bit position for all data.
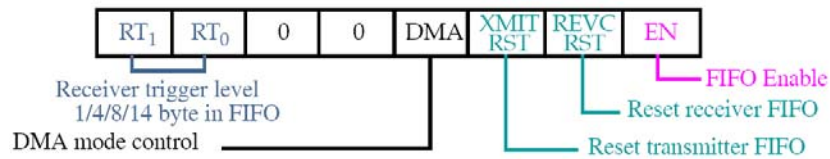
ST	P	PE	Function
0	0	0	No parity
0	0	1	Odd parity
0	1	0	No parity
0	1	1	Even parity
1	0	0	Undefined
1	0	1	Send/receive 1
1	1	0	Undefined
1	1	1	Send/receive 0

    * No parity, both ST and PE are 0 -- used for internet connections.
* SB = 1 causes a break to be transmitted on SOUT.
    * A break is at least two frame of logic 0 data.
* DL = 1 enables programming of the baud rate divisor.
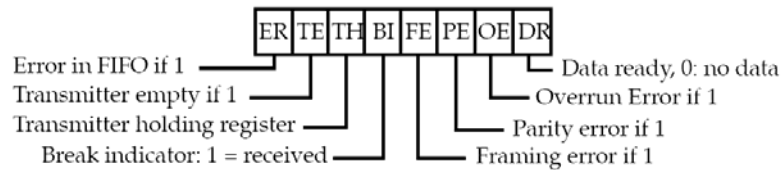
Programming the 16550

* Initialization (cont.)
  * Baud rate generator is programmed with a divisor that sets baud rate of transmitter.
  * Baud rate generator is programmed at 000 and 001.
    * Port 000 used to hold least significant byte, 001 most significant byte
  * Value used depends on external clock/crystal frequency.
    * For 18.432MHz crystal, 10,473 gives 110 baud rate, 30 gives 38,400 baud.
    * Note, number programmed generates a clock 16 times the desired baud rate.
  * Last, the FIFO control register must be programmed at 010.

RT_1	RT_0	0	0	DMA	XMIT RST	REVC RST	EN

Receiver trigger level
1/4/8/14 byte in FIFO

DMA mode control

FIFO Enable

Reset receiver FIFO

Reset transmitter FIFO

Programming the 16550

* Operation:
  * Status line register gives information about error conditions and state of the transmitter and receiver.

ER	TE	TH	BI	FE	PE	OE	DR

Error in FIFO if 1

Transmitter empty if 1

Transmitter holding register

Break indicator: 1 = received

Data ready, 0: no data

Overrun Error if 1

Parity error if 1

Framing error if 1

  * This register needs to be tested in software routines designed to use the 16550 to transmit/ receive data.
  * Suppose a program wants to send data out SOUT.
    * It needs to poll the **TH** bit to determine if transmitter is ready to send data.
  * To receive information, the **DR** bit is tested.

Programming the 16550

* Operation:
    * It is also a good idea to check for errors.
    * **Parity error:** Received data has wrong error -- transmission bit flip due to noise.
    * **Framing error:** Start and stop bits not in their proper places.
      This usually results if the receiver is receiving data at the incorrect baud rate.
    * **Overrun error:** Data has overrun the internal receiver FIFO buffer.
      Software is failing to read the data from UART before the receiver FIFO is full.
    * **Break indicator bit:** Software should check for break condition as well i.e. two consecutive frames of os on SIN pin

Next time

❑ Interrupts – Hardware Interrupts

STOP