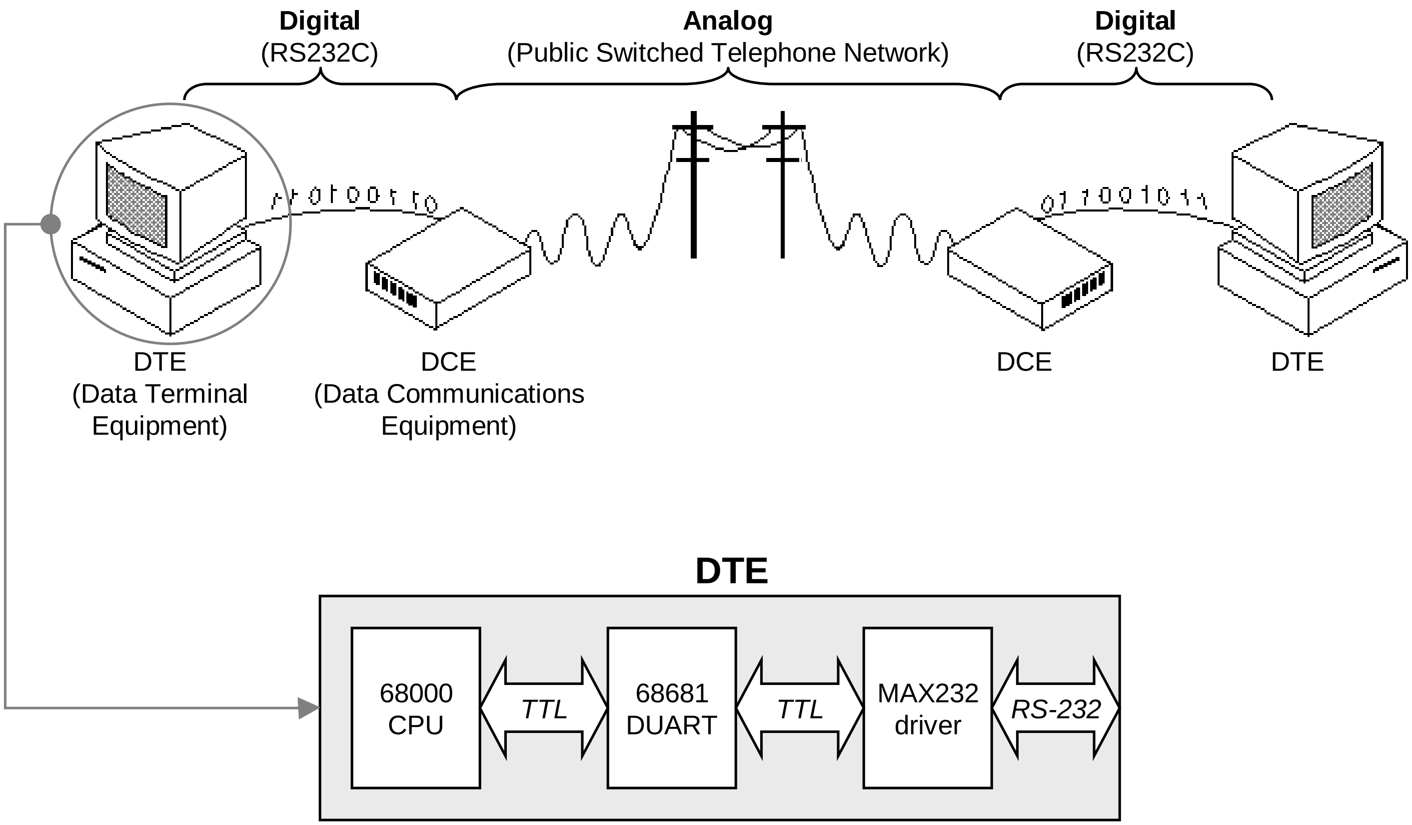
Lecture 13: DUART serial I/O, part I

- **The big picture of serial communications**
- **Analog communications**
  - Modems
  - Modulation-demodulation methods
  - Baud rate Vs. Bits Per Second
- **Digital serial communications**
  - Simplex, half-duplex and full-duplex
  - Synchronous serial transmission
  - Asynchronous serial transmission
- **The RS-232C standard**
  - Mechanical and electrical specs
  - Control lines
  - Handshaking
  - Null modem

The big picture of serial communications

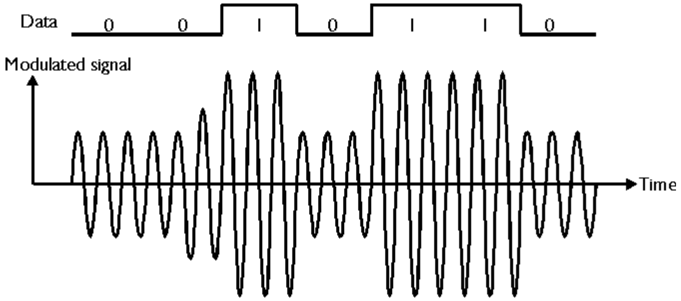


Introduction to modems

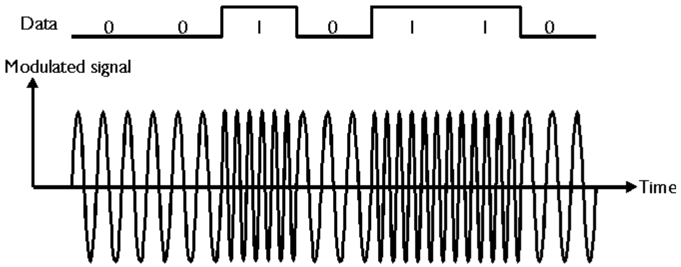
- **The POTS (Plain Old Telephone System) was designed to transmit voice-band analog signals in the range 300Hz to 3300Hz**
  - It is not appropriate for transmission of digital signals due to their high-frequency components (i.e.: sharp, edge transitions).
  - If a sequence of binary signals was presented to one end of a telephone network, the digital signals would be so severely distorted that they would be unrecognizable at the receiving end of the circuit
- **In order to transmit a digital signal, it must be converted to an analog signal within the bandwidth of the POTS**
  - A bit stream is converted into an analog signal at the sender by Modulation
  - The analog signal is converted back into a bit stream at the receiver by Demodulation
  - The D/A and A/D conversion is performed by a device called **modem** or <u>mod</u>ulator-<u>dem</u>odulator
- **There are three basic modulation methods:**
  - Amplitude modulation
  - Frequency modulation
  - Phase modulation

Amplitude, Frequency and Phase Modulation

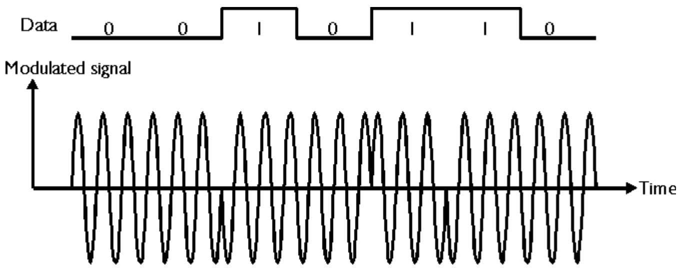
- **Amplitude Modulation (AM)**
  - A single carrier frequency is used to transmit data
  - Binary values are converted to different amplitudes of the carrier signal
  - Suitable only for low speed transmission



- **Frequency modulation (FM)**
  - Two carrier frequencies of the same amplitude are used
  - The higher frequency is associated with binary 1, the lower frequency with binary 0
  - Used for 1200 bps modems or slower speeds
  - FM is widely used because it has a better tolerance to noise than AM

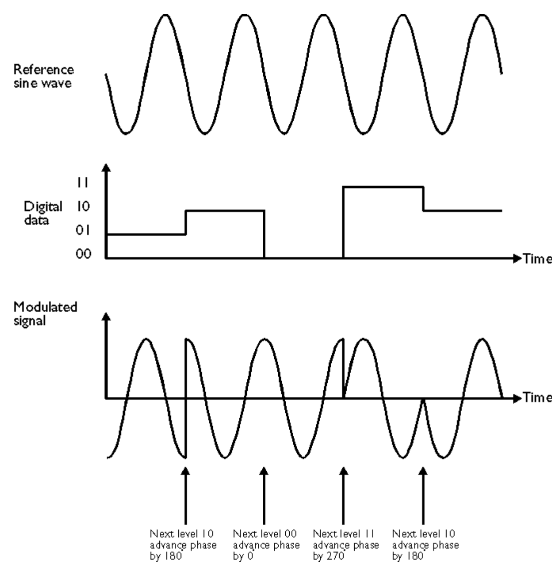


- **Phase modulation (PM)**
  - A single carrier frequency is used to transmit data
  - The phase of the signal is changed according to the binary value to be transmitted
  - PM is widely used and has fairly similar characteristics to FM



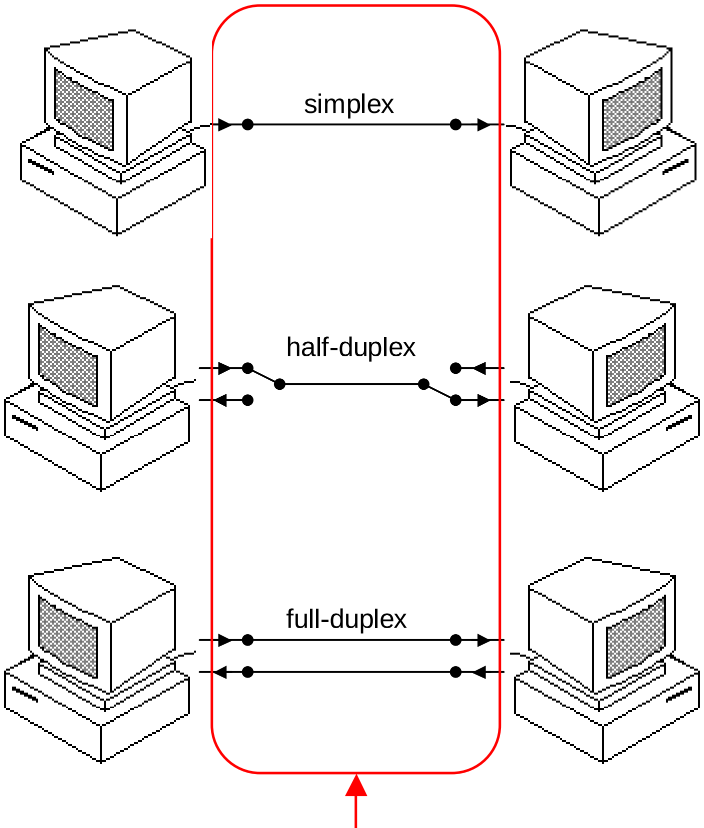
Units for speed of transmission

- **Baud rate**
  - Measures the switching speed of a signal (the number of times per second that a line changes state)
- **Bits per second (bps)**
  - Measures the rate at which information flows across a data link
- **Bauds and bps are not the same**
  - For a binary two-level signal, a data rate of 1 bps is equivalent to 1 Baud
  - But suppose a data transmission system that uses signals with 16 possible discrete levels. Each signal element can have one of 16=24 different values, that is, each discrete level encodes 4 bits
    - If the 16-level signals are transmitted at 1200 Baud, the data rate is 4x1200=4800 BPS
- **In general, baud rates over the POTS is limited to 2400 Bauds due to bandwidth limitations, so data rates higher than 2400 bps are achieved by sophisticated modulation techniques and data compression**
  - For example, using PM with phase shifts multiple of 90° we can double the data rate, as in differential phase modulation
  - High-speed modems operate by simultaneously changing the amplitude and phase of a signal. This modulation technique is called quadrature amplitude modulation (QAM)



Modes of channel operation

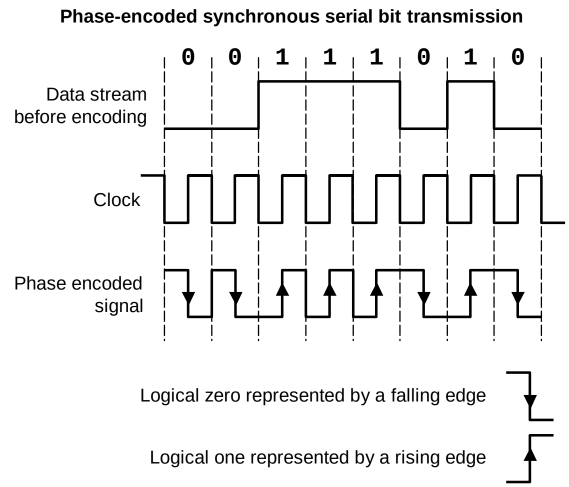
- **Simplex**
  - Data in a simplex channel is always one way Simplex channels are not often used because it is not possible to send back error or control signals to the transmit end. Its like a one way street
  - An example of simplex is television
- **Half Duplex**
  - A half duplex channel can send and receive, but not at the same time; only one end transmits at a time, the other end receives
  - Its like a one-lane bridge where two way traffic must give way in order to cross.
- **Full Duplex**
  - Data can travel in both directions simultaneously. There is no need to switch from transmit to receive mode like in half duplex
  - Its like a two lane bridge on a two-lane highway

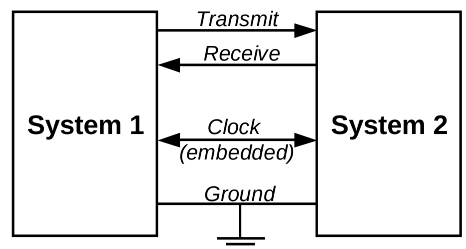


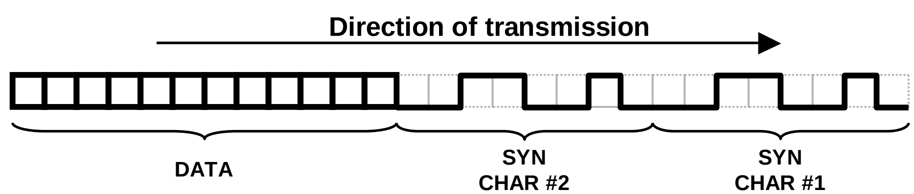
From now on we will ignore how the two computers are physically connected. They could be remote systems connected through a telephone line, or they could be local systems connected with a simple serial cable

Synchronous Transmission

- **In synchronous transmission, the clocks at the transmission and reception ends must be <u>permanently</u> synchronized by either**
  - Transmitting the clock signal on a separate wire (too expensive)
  - Embedding the clock signal in the data stream (preferred). An example of embedded clock is Phase Encoding
    - From the received phase-encoded signal it is possible to recover both the data stream and the clock signal
- **Procedure**
  - To initiate transmission, the transmitter first sends out synchronization characters to the receiver
  - The receiver reads the synchronization bit pattern and compares it with a known sync pattern. Once they are identified as bie4ng the same, the receiver begins to read character data off the communications line
  - Transfer of data continues until the complete block of data is received or synchronization is lost
  - If large blocks of data are being sent, the synchronization characters may be periodically resent to assure that synchronization is maintained
- **Synchronous transmission is typically used in high-speed applications**

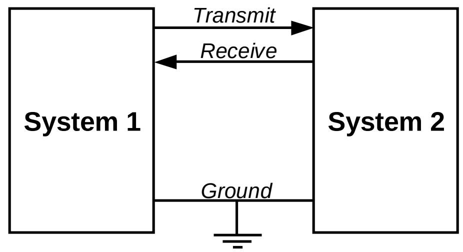




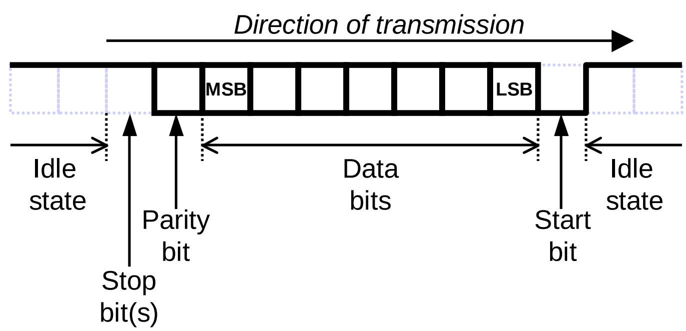


Asynchronous Transmission

- **In asynchronous transmission there is no need for the clocks to be synchronized**
  - Data to be transmitted is sent one character at a time
  - At the received end of the transmission synchronization is performed by examining synchronization bits that are included at the beginning and end of the character
- **The format of a typical asynchronous character is the following**
  - **Start bit (1 bit)**: a logical 0 (also called a *space*) that is used to indicate the beginning of a character
  - **Data bits (7-8 bits)**: they contain the information that is being transmitted. The data wordlength can be either 7 or 8 bits
  - **Parity bit (1-0 bits)**: optional, used for error checking. If odd parity is used, the parity bit is set to 1 or 0 such that the total number of 1-bits, including the data bits and the parity bit, is odd.
  - **Stop bit (1-2 bits)**: a logical 1 (also called a *mark*) that is used to indicate the end of the transmission

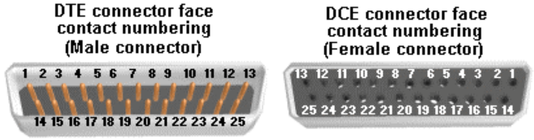


- **A total of 12 possible formats are possible since we can have**
  - 7 or 8 data bits
  - even, odd or no parity bits
  - 1 or 2 stop bits
- **Asynchronous data transmission is the most popular serial mode, but**
  - is highly inefficient: for 7 data bits and 2 stop bits, only 70% of the transmission contains information
  - is more suited for ASCII text. For 8-bit binary data it is quite difficult to embed control characters (i.e., tape start or stop) within the data stream



The RS232 standard

- **RS232C (Recommended Standard 232 version C) was published in 1969**
  - Originally designed for DTE (computer) to DCE (modem) communications
  - Currently being used for more general applications, such as DTE to DTE connections
- **RS232C specifies**
  - Mechanical interface (DB-25)
    - Female connector is connected to DCE and male connector to DTE. Short cables of less than 15 meters (50 feet) are recommended. The pin assignments detailed above must be used.
  - Electrical interface
    - All circuits carry bi-polar low-voltage signals, measured at the connector with respect to signal ground, and may not exceed ±25 volts. Signals are valid in the range ±3 volts to ±25 volts. Signals within the range -3 volts to +3 volts are considered invalid.
    - Different from TTL levels (0-5V)
      - Need drivers and receivers to convert voltages (between the ACIA/DUART and the RS232 line)
      - MAX232 by Maxim combines driver and receiver in a single package
  - Control Lines
    - The lines available (and their function) to connect DTE and DCE

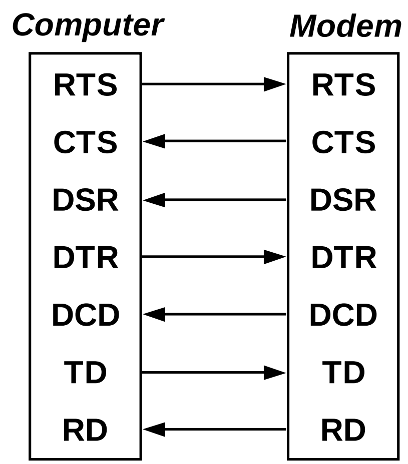


RS232C-to-TTL signal conversion

TTL		RS232C	
Logic	Voltage	Logic	Voltage
High(1)	*2.4 to 5 volts*	*MARK*	*-3 to -25 volts*
Low(0)	*0 to 0.8 volts*	*SPACE*	*+3 to +25 volts*

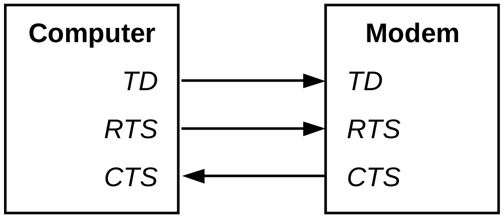
Basic RS232 Control Lines

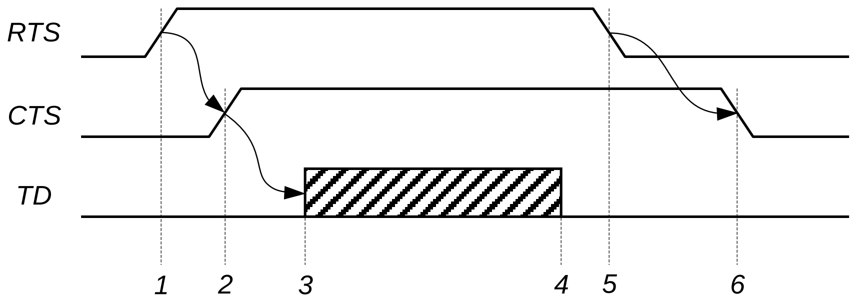
- **Request To Send (RTS), Pin 4**
  - This signal line is asserted by the computer (DTE) to inform the modem (DCE) that it wants to transmit data. If the modem decides this is okay, it will assert the CTS line. Typically, once the computer asserts RTS, it will wait for the modem to assert CTS. When CTS is asserted by the modem, the computer will begin to transmit data.
- **Clear To Send (CTS), Pin 5**
  - Asserted by the modem after receiving a RTS signal, indicating that the computer can now transmit.
- **Data Terminal Ready (DTR), Pin 20**
  - This signal line is asserted by the computer, and informs the modem that the computer is ready to receive data.
- **Data Set Ready (DSR), Pin 6**
  - This signal line is asserted by the modem in response to a DTR signal from the computer. The computer will monitor the state of this line after asserting DTR to detect if the modem is turned on.
- **Data Carrier Detect (DCD), Pin 8**
  - This control line is asserted by the modem, informing the computer that it has established a physical connection to another modem. It would be pointless a computer transmitting information to a modem if this signal line was not asserted. If the physical connection is broken, this signal line will change state.
- **Transmit Data (TD), Pin2**
  - The line where the data is transmitted from the computer to the modem.
- **Receive Data (RD), Pin 3**
  - The line where data is received (transmitted from the modem to the computer.)



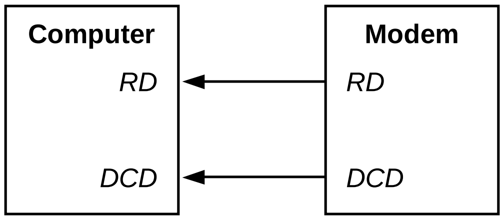
Examples of handshaking

- **When the computer wants to transmit data to the modem**
  - RTS is asserted by the DTE (1)
    - to indicate that the DTE wishes to send data
  - CTS is asserted by the DCE (2)
    - to indicate that the DCE is ready to receive data
  - DTE transmits data through TD (3)
  - DTE completes data transmission (4)
  - RTS is negated by the DTE (5)
    - to indicate that the DTE has finished transmitting data
  - CTS is negated by DCE (6)





- **When the modem wants to transmit data to the computer**
  - The DCE asserts DCD
    - To indicate that a data carrier is present and to transmit data
  - The DTE must be prepared to receive data any time DCD is asserted



Null modem

- **The RS-232C interface was designed to connect computers to modems**
  - However, many of today's RS-232C applications are for connecting a computer (DTE) to a peripheral (such a printer, also considered a DTE), rather than to a modem (DCE)
- **In order to connect a DTE to a DTE we must cross some of the control signals**
  - the DTE signals from one computer are swapped over as inputs to supply the DCE expected signals on the other DTE
- **This DTE-DTE interface is accomplished with a special connector cable called a *null modem***
  - A null modem simulates a DTE-DCE-DCE-DTE circuit
    - A null modem has a female DB-25 connector on both ends
    - The control lines are wired as indicated in the figure