Asynchronous Serial Interfaces

Exercise 1: Is the "Transmit Data" (TxD) signal an input or an output? How about "Receive Data" (RxD)? Is a computer a 'modem' (DCE) or a 'terminal' (DTE)?

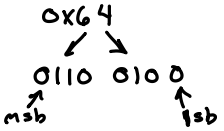
TxD could be either

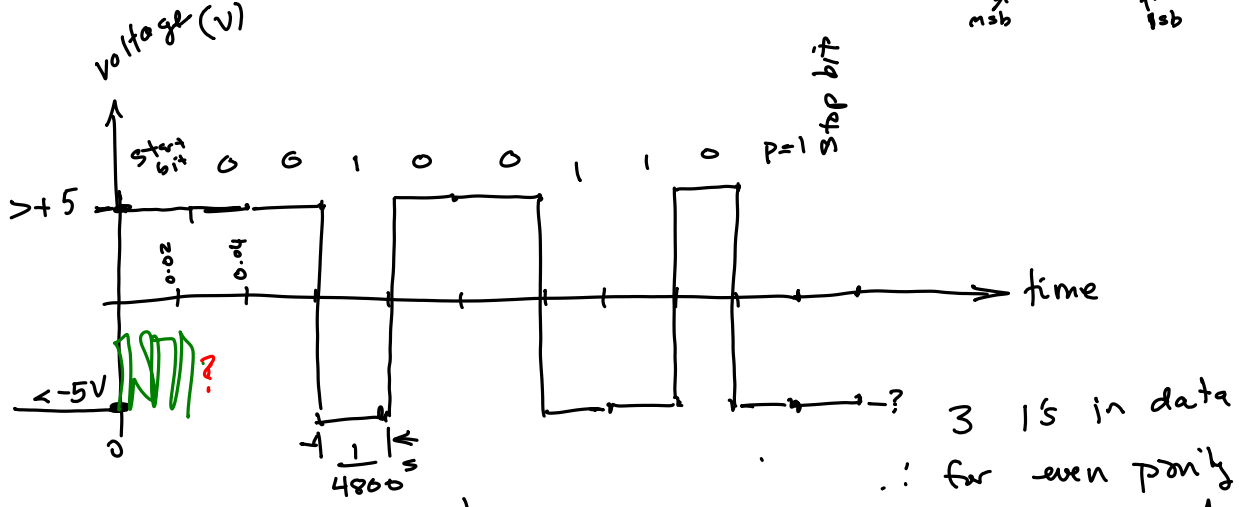
RxD as well

computer could be either:

— a PC typically DTE

Exercise 2: Draw the waveform used to send the ASCII character 'd' (hex 64) at 4800 bps with eight data bits and even parity.





4800 bps → bit rate or bit "frequency"

$\frac{1}{4800}$ s/bit → bit period.

≈ 20ms.

3 1's in data

∴ for even parity parity bit is 1

Exercise 3: Will the parity bit allow the receiver to detect all single-bit errors? All double-bit errors?

yes - any error changes number of 1's by 1
& makes even → odd or vice-versa.
0→1 +
1→0 -

No - two error adds / subtracts
2 to # of 1's & # even / odd
doesn't change

Exercise 4: What happens if the receiver's clock is running faster than the transmitter clock?

- depending on difference
  - might receive multiple characters
  - might see errors
  - nothing.

Exercise 5: What would happen if the receiver was expecting 8-bit characters and the transmitter was sending 7-bit characters? (1) What about the reverse case? (2)

(1) - receiver would see transmitted stop bit as the last (m.s.) bit. Stop bit is low (=1) so the m.s bit would be set on every received byte.

- receiver might see start bit (H) of next byte in the stop bit position (expected to be low) this would indicate a framing error.

(2) the m.s. transmitted bit would be received in the stop bit position. If this bit is '0'