AE6705 Lab 4
Code Composer Studio

Yunqing (JJ) Jia

September 26, 2024

1. Suppose you are using a baud rate of 38400. How far off can the clock rate of your two devices be before you start reading data incorrectly? Do not use the 5% approximation, but rather compute the answer exactly. How does it compare to the 5% approximation?

   Time to transmit one bit using baud rate of 38400:

   $$T = \frac{1}{38400} = 26.04\mu s$$

   We can tolerate a time offset that's less than or equal to half of the bit time ($13.02\mu s$) for one byte, meaning that the two rates can be differ by at most

   $$T_e = 13.02/9 \approx 1.45\mu s/\text{bit}$$

   $$B_l = \frac{1}{T + T_e} = 36377$$
   $$\frac{B - B_l}{B} = \frac{38400 - 36377}{38400} = 5.26\%$$
   $$B_u = \frac{1}{T - T_e} = 40667$$
   $$\frac{B_u - B}{B} = \frac{40667 - 38400}{38400} = 5.90\%$$

   The upper bound for the baud rate would be 40667, meaning one of the clocks can be at most 5.90% faster, assuming that the other device is transmitting at a baud rate of 38400
   The lower bound for the baud rate would be 36377, meaning that one of the clocks can be at most 5.26% slower, assuming that the other device is transmitting at a baud rate of 38400

2. Suppose your clock signals differ by more than the amount in Question 1. What error flag would you expect to see, in what register? Why would this error flag get triggered in this situation?
   If the clock signals differ by more than the tolerance amount, it framing error flag would be triggered in the receive data register (`UART0->RXDATA`). It's set when the stop bit is incorrect, indicating a mismatch in baud rate.

3. How could you solve the above problem (i.e., by changing the baud rate)? What is the inherent performance penalty associated with this solution?
   We can lower the baud rate for better accuracy, but this solution comes with the penalty of not being able to transmit as much data in a given time.