Aayush Gunjal, Sre Krishna Subrahamanian, Michael Savo

LAB 10

SERIAL COMMUNICATIONS LAB EXERCISE: PERFORMANCE ANALYSIS

Contents

1 Introduction

1.1 Lab overview

In this lab, you will use an oscilloscope or logic analyzer to see how serial communcations operate.

2 Learning Objectives

- Use a logic analyser or oscilloscope to measure the response timing delay in serial communication.

3 Requirements

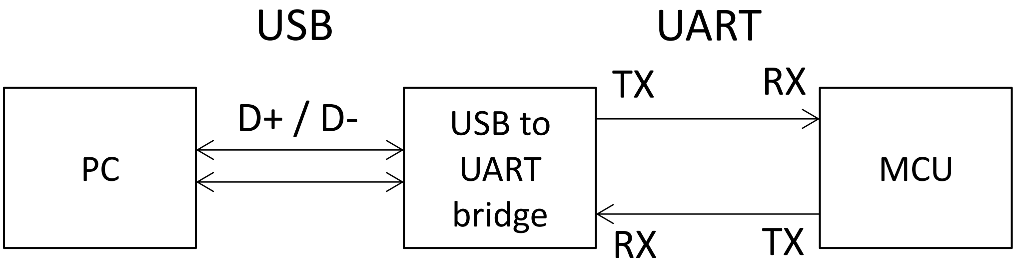
In this lab, we will be using the following hardware and software:

- **Arduino IDE (installation steps further down)**
- **Arduino Due board**
- **Logic Analyzer or Oscilloscope**

4 Details

4.1 Hardware

In theory we would use an Arduino as the bridge shown here:



In practice, we are just going to connect Arduino Uno TX or Due TX0 to the oscilloscope directly. Arduinos support communications at 57,600 Baud, 8 data bits, no parity, 1 stop bit via the Serial library defaults (8N1).

5 Procedure

Download Arduino IDE 2.3.5 at https://www.arduino.cc/en/software.

Then, connect the Arduino Uno via USB or Due via the programming port (look at the bottom of the board to verify which is the programming port in particular). Make sure in the Arduino environment that you select the board (up toward the top where the default text should be “Select Board”). Then, upload the supplied lab code for the appropriate type of board onto the board via the Upload button.

First a theoretical question for the STM 32 board:

1. Examine the previously provided pins documentation for your board to determine the answers to the following questions.
   a. Which pins on our usual “Zio” headers can be used to receive (RX) over UART?
      i. You can use PC_0, PA_3, PD_6, PD_2, PC_11, PB_11, PB_10, PD_9, PA_1, and PB_10.
   b. Which pins on our usual “Zio" headers can be used for I2C channel 1?
      i. PB_8, PB_9, and PB_6
   c. Which pins on our usual “Zio" headers can be used for SPI channel 1?
      i. PA_5, PA_6, PA_7, and PA_4

The rest of our tasks relate to using the Arduino and oscilloscope:

2. Attach a wire to the Arduino transmit line (this should say TX on the Arduino Uno or TX1 on the Arduino Due). Connect an oscilloscope probe to this wire (and as usual attach the corresponding clip to ground). Scale the oscilloscope appropriately – please practice this on your own as much as possible, you’ll need this skill for the lab practical! Of note, the voltage range is similar to those we’ve seen in past labs

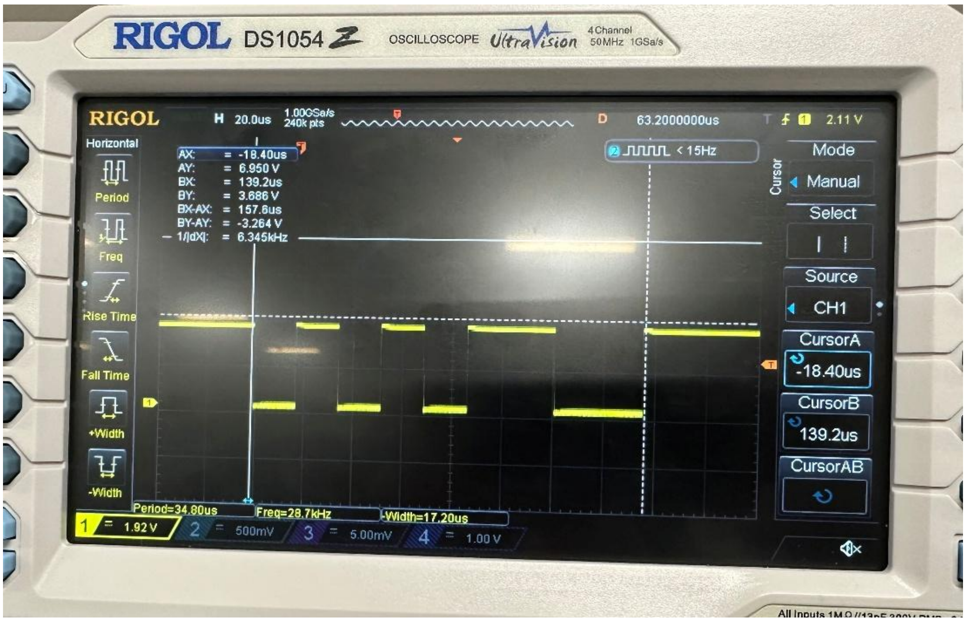
(the signal goes from roughly 0 to 5 V), but the time scale is quite small for sending one packet of information.

3. What is the minimum pulse duration to send one bit of information (whether that is our start or stop bit or the data itself)? Capture a screenshot of this measurement. How is it related mathematically to the transmission rate of 57,600 Baud?
    a. The minimum pulse duration we measured is 17.4us.
    b. If we divide 1 by the baud rate, we get 0.00001736 which is 17.36*10^-6 or roughly 17.4us, which is the minimum pulse duration.
4. How long does it take to transmit a one-digit number (including things like the stop and start bit) from the Arduino? Modify the code to test this (as it is currently sending a one-character letter technically). Capture a screenshot of that full length of time.
    a. We changed the Arduino code to transmit the number 9. We noticed the length to transmit the signal is around 138.8us.
5. For whatever one-digit number you chose, look that number up in the Ascii table (see https://www.techonthenet.com/ascii/chart.php). Identify the data bits in the data shown on the oscilloscope and isolate the bits of just the number you are sending. Review the slides again if you are not sure about the bits you are seeing on the oscilloscope. Verify that you are sending the correct Ascii information by taking a screenshot!
    a.

57	39	071	9	Digit nine

    b. The digit 9 would send the signal of 57 over according to the ascii table.
    c.

6. Does the length of time change if we instead try a five-digit number? What is that new length of time?
    a. We used the number ‘12345’ and it took 157.6us.
7. Capture a screenshot that shows this full length of time for the five-digit number.



a.

8. Modify the code to communicate at a higher Baud rate (e.g. 115,200 Baud, 234,000 Baud). Are the results still consistent with the earlier information you obtained? Capture a screenshot showing your measurement of the new higher Baud rate.

   a. Using the same five digit number 12345, the time to transmit the signal is 38us. Since the baud rate is 4x higher than 57600, the time to transmit the signal is also four times faster.

b.