

CPTR-480 In-Class
Serial Data Transmission

Goals for today

- Create a simple tool to display text messages from an embedded system on a terminal
 - Create a function to print a string of characters passed to it
 - Create a function to move cursor to the beginning of the next line
- Use a USB-to-serial cable to transfer serial data from an embedded system and display it on a virtual terminal.

To Do

- 1) Create a new project for this assignment in a new folder. (you can copy over your .c files from the previous assignment to the new folder. After creating the new project in that folder import the existing .c files into the new project).
- 2) In the new project, add the `#define CLOCK_SETUP 1` statement to `system_MKL25Z4.h`
- 3) Write a function named `print_string` that will accept an array of type `char`, i.e. an array of characters, and send one character at a time to UART2 for transmission. Note that your function can be called like this: `print_string("Hello");` and that 5 characters should be sent as ASCII codes to a receiving terminal. The length of the string passed to `print_string` will be 6 bytes because in addition to 5 bytes with the desired characters there will be a 6th byte with a numeric value of zero (called a null character). You will have a loop in `print_string`. Exit the loop when the next character is a null character. Also, after sending a character to the UART transmit buffer wait for the TDRE flag (Transmit Data Register Empty Flag) to be set before sending the next character to the transmit buffer.
- 4) Write a function named `print_crlf` that will send ASCII carriage return and line feed characters via UART2.
- 5) In `main.c` you will call `init_UART2` once and then enter `while(1)` loop. In that loop do the following:
 - call `print_string("Q")`
 - delay for 2 millisecondsLook at the output from pin PTD3 with a scope. Trigger on falling edge.
- 6) Modify the `while(1)` loop to this:
 - call `print_string("Hello123")`
 - call `print_crlf`
 - delay 1000 ms
- 7) Connect embedded board to PC using a USB-to-serial cable.
 - connect black ground wire to ground
 - connect the yellow wire with the RXD connector to PTD3
 - plug USB connector into PC
 - use Device Manager to find out the COM port number assigned to the USB cable. It will be something like COM1 or COM2 etc.
 - start the Putty program. This is a terminal emulator program. In the Putty startup window there is a row of circles and the one labeled SSH likely has a black dot in it. Click on the Serial circle to put Putty into serial port mode. If needed, change the COM port number. Then click open and a new "terminal" window will open. If your program is running you should see characters displayed.

Make sure you update the comment header in `main.c`. Report the outcome in the header. Turn in `main.c` and then the complete project zipped up to the D2L dropbox.