Lab 8: UART

Nima Karimian

Preparation

- Understand the sample projects (Canvas->Labs->sample_projects)
    - switch_counter_interrupt_UART_TivaWare
    - UART0_terminal

References

- Getting Started with the Tiva TM4C123G LaunchPad Workshop Student Guide and Lab Manual (Chapter 12) (Canvas-> Reference Materials -> TM4C123G_LaunchPad_Workshop_Workbook.pdf)
- TivaWare Peripheral Driver Library User's Guide (Canvas-> Reference Materials -> SW-TM4C-DRL-UG-2.1.0.12573.pdf)
- Tiva TM4C123GH6PM Microcontroller Data Sheet (Canvas-> Reference Materials)

Purpose

The purpose of this lab is to learn how to implement UART communication on the TM4C microcontroller. Software skills you will learn include UART configurations, UART interrupt implementation, UART communication with terminal, and UART communication between two LaunchPads.

Procedure

1. Install the serial terminal TeraTerm

TeraTerm is a serial terminal program which allows us to easily see the output of the TM4C123G LaunchPad via UART0 serial communication. It can help greatly when debugging.

If you have already installed TeraTerm as instructed in the Software Installation Guideline at the beginning of the semester, you can skip this step.

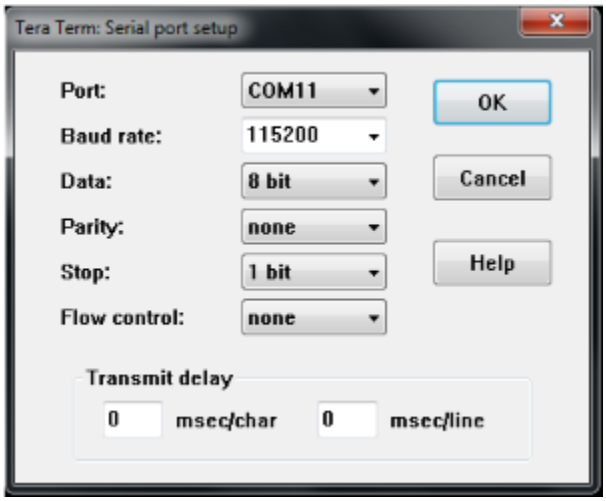
TeraTerm can be downloaded from SourceForge here. For this class, the standard installation will work fine.

2. Run the sample project UART0_terminal

Setup serial terminal

Open TeraTerm. In the 'New Connection' window, select the radio button for 'Serial', and in the dropdown menu, choose the COM port option for Stellaris Virtual Serial Port (the number of the COM port is assigned somewhat arbitrarily by windows, and is inconsequential).

For the sample project, the UART peripheral is configured for a baud rate of 115,200, with 8 data bits, 1 stop bit, and no parity bits, so we need to make sure that TeraTerm is configured with the same settings to ensure proper communication between the LaunchPad and the serial terminal. In the top menu of TeraTerm, choose 'Setup->Serial port...' and ensure that the settings match those shown below (except for the port, which may be different depending on what port number Windows assigned to your device):



Download and run the sample project

Download and open the **UART0_terminal** sample project in the Keil µVision IDE. Compile and download the project to the LaunchPad. Then you can run the project and see how the microcontroller communicates with TeraTerm.

Understand the sample project

Read Chapter 12 of the TM4C123G_LaunchPad_Workshop_Workbook to understand the **UART0_terminal** project. Try to understand how to configure the UART module and the UART interrupt. Please look up the used UART functions in the TivaWare Peripheral Driver Library User's Guide and understand how to use them. Try to understand the difference between

UARTCharPut()/UARTCharGet()

and

UARTCharPutNonBlocking()/UARTCharGetNonBlocking()

3. Run the sample project switch_counter_interrupt_UART_TivaWare

This sample project was built based on the switch_counter_interrupt_TivaWare project. The only difference is that it adds the UART function in the project in order to print the counter value in the TeraTerm terminal in real-time so that the user can use it as a more efficient way to debug the program. You can learn from this project on how to print out values of the variables you would like to watch without going to the debug mode.

4. Develop a new system

The system consists of two LaunchPads as shown in Figure 1. Device A and Device B are communicated via the UART1 modules. The UART1 Tx port PB1 on Device A is connected with the UART1 Rx port PB0 on Device B. The ground pins on both devices are also connected.

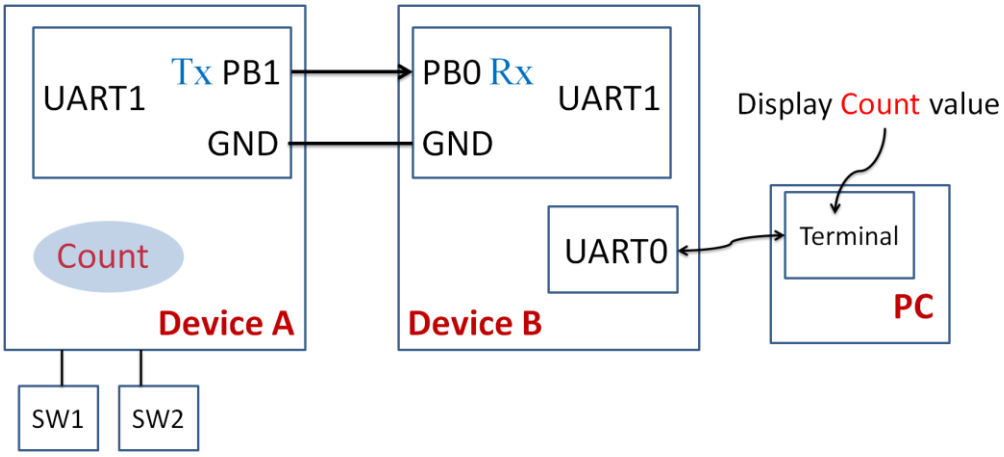


Figure 1

You need to develop two software projects in this lab, one for each device.

Software on Device A: The system is based on the switch_counter_interrupt_TivaWare project. The system has two input switches (SW1 (PF4) and SW2 (PF0)). When either SW1 or SW2 is pressed, a global variable 'count' is incremented by 1 using edge-triggered interrupt. Then the updated value of 'count' is transmitted to Device B via the UART1 module.

Software on Device B: The system receives serial inputs from Device A via the UART1 module. Whenever there is data available on the UART1 Rx pin, a UART1 RX interrupt is generated. In the interrupt service routine, the processor reads the received data from the UART1 module and then sends the data to the TeraTerm terminal via the UART0 module.

After you download the software projects onto Devices A and B respectively, turn off the power of both boards. Carefully connect two boards using two male-male wires (**make sure the power is off when you connect the boards**).
You can then turn on the power on both boards and verify your system functionality. You need to open TeraTerm on the PC that connects with Device B to see the output. If the system works correctly, you will see the PC connecting with Device B display the value of 'count' in real-time when you press SW1 or SW2 on Device A.

Demonstration and Submission

You will have one week to complete the lab. You can discuss with your group members and complete the lab work together. Every group will need to write and submit a lab report to Canvas->Labs->Lab8 report submission. The lab report should include

- The students' names, emails and IDs

- System specifications (based on the system requirement, which ports do you need to configure, how to configure them, any special actions needed? How do you configure the edge-triggered interrupt?)
- The flow chart of your design
- The C program listing with **detailed comments for each line of code**. You are allowed to use PinMux and TivaWare library defined functions in your implementation.
- The execution results of your program (how did you evaluate your implementation and verify your system correctness? your observations, any supplemental images).
- Discussion and suggestions: Through your lab experiments, what have you learned? Do you have any suggestions for future labs, lectures or improvement on the learning experiences?

If you finish the lab experiments during the lab time, please demonstrate your results to the instructor. The instructor may ask questions regarding your program. After the demonstration, you can leave. **The latest demonstration time will be the beginning of next lab. The lab report is due at 6pm on the day of your next lab.**

Again, you can work with your group members on all the lab activities, but make sure you understand all the materials.

Note: Don't forget to edit startup_rvmdk.s to change the names of the interrupt handlers.