ECE 3710 Lab 4 – Fall 2017

Due Date: Week of October 23 at the beginning of your lab section (20 points)

Objectives

The student should learn how to implement an interrupt service routine (ISR) in C. The student will also become familiar with the PS/2 (not Play Station 2) protocol for a keyboard. The student will learn more about interrupts and their configuration.

Overview

You have applied for a position with the NSA (No Such Agency). As part of your interview they have given you a PS/2 keyboard, breadboard, and the microcontroller that you are using in this class. They have explained to that they want you to create a hardware keylogger to be used to spy on a potential target. They do realize that this one won't be very discreet but a good proof of concept. They ask you to display the ASCII characters on a PC terminal via microcontroller UART0 and USB.

Preparation

1. You can find more information on hardware keyloggers at: https://en.wikipedia.org/wiki/Hardware_keylogger

2. Read over interrupts in your textbook and uC datasheet.
    a. *Introduction to ARM* textbook sections 9.5 and 9.7
    b. Data sheet section 10.2.2

3. Read over the web pages that give a brief description of the PS/2 protocol, such as:
    a. http://www.computer-engineering.org/ps2protocol/
    b. http://pcbheaven.com/wikipages/The_PS2_protocol/

4. Checkout a PS/2-to-USB adapter and purchase a PS/2 extension cable from the ECE store. (You will find PS/2 keyboards in the lab).

Hardware

Cut the PS/2 extension cable in half (**DO NOT CUT THE KEYBOARD CABLE!**). Now peel back the outer insulator exposing the 5 wires on the inside on both sides of the extender. Strip back the insulators on the inside wires exposing the wire. Try not strip them longer then need to make proper connection in a bread board. Next you want to splice the wire back together using the bread board. To make it easier to put the wires in the bread board a solder tinning of each wire can be used. This process won't let your wires fray. If you wish to do this there are many you tube videos instructing the process; solder and soldering irons are available in the ECE store.

Use the logic analyzer and multimeter to identify the wires. First find out which wire is power and which is ground using the multimeter. Next connect the other wires to the logic analyzer. The remaining wires will be n/c, Clock, and Data.

After finding the Clock and the Data wire you will need to determine how to connect these to your board. The keyboard utilizes 5V logic while the Tiva C uses 3.3V logic. **If connected improperly, this could fry your board.** Refer to the datasheet and report which pins are not 5V tolerant. Include this in your lab report.

The Clock and Data wires should be connected to an external interrupt pin and a digital input pin, respectively.

Software

Your microcontroller will be getting data from the keyboard. This data must be stored in in memory until a button is pressed that sends the characters to the computer via UART0 and USB (note that UART0 is connected to the DEBUG USB port on your TivaC Launchpad). This means that the key code for the key stokes must be converted to ASCII. This can be done by a look up table (we have provided an array that can be used for this purpose on the wiki).

Start a terminal emulator, such as HyperTerminal or Putty, on your computer and configure it for 9600 baud, 8 data bits, one stop bit, and no parity bit. **Make sure that you configure UART0 on your microcontroller with the same settings**. **Remember to configure the UART0 pins in their alternate function, rather than as general purpose inputs or outputs.**

Remember, the keyboard sends codes for each keystroke: once for a key down and another for a key up (see the lecture slides related to PS/2). The ISR must be called every time a bit is transmitted by the keyboard (i.e. each clock tick should trigger an interrupt). This ISR will be used to store the keystrokes somewhere in memory. You can ignore communication from the PC to the keyboard.

Requirements

Your program should accomplish the following:

1. The ISR must be triggered for each clock tick.
2. The keystrokes must be stored in memory.
3. You need to have a button that starts/stops the keylogger. Once the keylogger has been stopped, the captured keystrokes should be sent to the computer via UART0 and USB.
4. You must make use of interrupts to interface with the button. The button should have a higher priority than the clock interrupt.
5. The program only needs to be able to capture and display lowercase alphanumeric characters and space.

Connecting to the PC

The computers in the Lab don't have PS/2 ports on them. This is why we will need a PS/2 to USB adapter. Plug this into any of the USB ports. Next plug the spliced extension into the adapter then plug the keyboard into the extension.

Report Requirements

Your lab report should demonstrate that your program works as stated in the requirements by including:

1. A list of which Tiva C pins are not 5V tolerant.
2. Output from the logic analyzer that shows the clock signal of the keyboard with data from the keyboard.
3. Capture the waveform or logical values of a typed character. In Keil, grab a screenshot of the register that contains the value of the captured character.
4. A picture of the ASCII characters on Putty or some other serial watch program.
5. Circuit diagram of the spliced extension cable and how it wires into the microcontroller.