

Lab 7: Periodic Interrupts

Nima Karimian

Preparation

- Understand the sample projects (Canvas->Labs->sample_projects)
 - switch_delay_interrupt/switch_delay_interrupt_TivaWare
 - switch_counter_interrupt/switch_counter_interrupt_TivaWare
 - switch_counter_interrupt_debounce
 - toggle_timer_interrupt/toggle_timer_interrupt_TivaWare

References

- Getting Started with the Tiva TM4C123G LaunchPad Workshop Student Guide and Lab Manual (Chapter 4) (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 edge-triggered and periodic interrupts on the TM4C microcontroller. Software skills you will learn include edge-triggered and periodic interrupt configurations, shared interrupt vector, switch debounce, interrupt handler implementation, and configuration of general purpose timers.

System Description

Implement a 3-bit rotary counter on the TM4C microcontroller. The counter is controlled by two types of interrupts simultaneously. 1) Edge-triggered interrupt – the counter is incremented by 1 when SW1 is pressed and decremented by 1 when SW2 is pressed. 2) Periodic interrupt – the counter is incremented by 1 periodically every 1 second. The periodic interrupt has a higher priority than the edge-triggered interrupts. Three on-board LEDs are used to display the counter.

System Requirements

The system has two input switches (SW1 (PF4) and SW2 (PF0)) and three output LEDs (red (PF1), blue (PF2), and green (PF3) LEDs). Overall functionality of this system is described in the following rules.

- 1) The red, blue, and green LEDs are used to display bit 0, 1, and 2 of the counter, respectively.
- 2) The system starts with the counter equal to 0.

3) The counter increments by 1 every one second. The general purpose timer TIMER0 is used to generate this periodic one second delay and periodic interrupt. The counter is updated by the interrupt. The priority of the periodic interrupt is 0.

4) If SW1 is pressed, the counter is incremented by 1; If SW2 is pressed, the counter is decremented by 1. Both SW1 and SW2 generate edge-triggered interrupts to update the counter. The priority of the edge-triggered interrupt is 2.

Demonstration and Submission

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->Lab7 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.