Exercise in Embedded Computing: Interrupt Controllers

Sivan Toledo, Tel-Aviv University

October 31, 2010

The goal of this exercise is to learn how to use an advanced interrupt controller. Use the makefile and support files from the UART exercise and `vic.c` from `exercises/vic`.

1. Modify your LPC2148 switch-debouncing program or your bike blinker to to use IRQ interrupts and the VIC rather than FIQ interrupts.

2. Modify your LPC2148 UART driver to use an IRQ interrupt and the VIC rather than a FIQ interrupt. The main program should instruct the driver which interrupt priority to use using a constant.

3. Optional: write a program that demonstrates that the FIQ interrupt-service routine indeed suspends a running IRQ interrupt-service routine. The program should use two interrupt sources. One should always invoke an IRQ ISR. This ISR should not return until a volatile variable changes state. If there are no other interrupts, this ISR remains in an infinite loop checking this variable. The other interrupt source should drive either a FIQ or an IRQ ISR that changes the state of the variable. Write the program so that you know whether the first ISR is running or not.