University of British Columbia
Electrical and Computer Engineering
Electrical and Computer Engineering Laboratory II
EECE 281/282

Module 7 – Interrupts and Interfacing

Introduction

Interrupts are very important and useful for embedded systems. An interrupt is an event that will trigger the automatic execution of a given piece of code. The code used to handle interrupts is often referred as the Interrupt Service Routine (ISR). The normal or default flow of code in a microcontroller system can be interrupted by different interrupt events, for example: timing events, external events, critical errors, power failure, communication events, etc. In this module you will be using the timer 0 interrupt and you will be programming it using the C language. Additionally you'll learn how to control power loads with the microcontroller using optoisolators and MOSFETs.

References

SDCC user manual available at http://sdcc.sourceforge.net

Any book you have about programming in C. Also, information about the standard C runtime library functions could be handy both for this module and the final project.

P89LPC9351 microcontroller user manual.

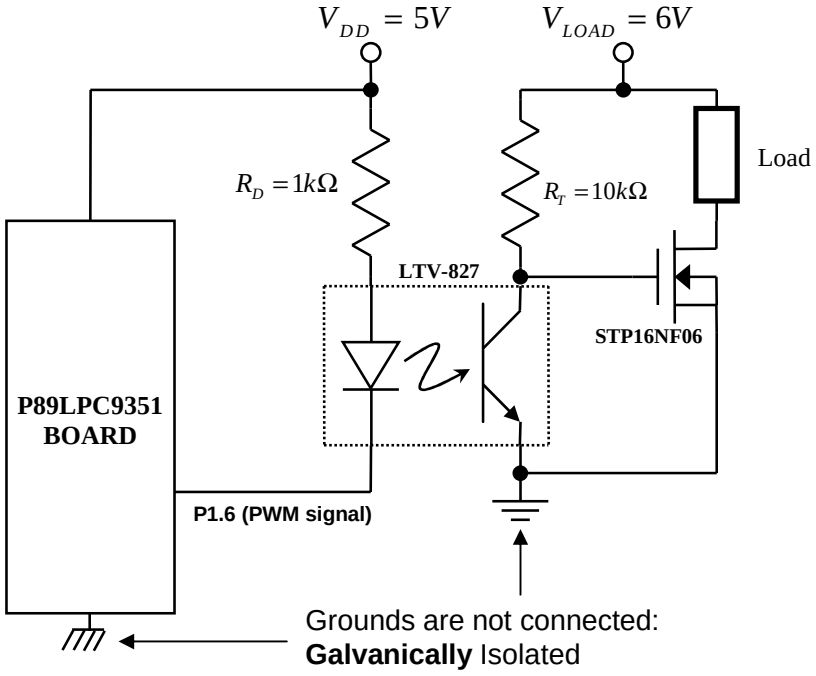
Pre-laboratory

1) Explain what Pulse Width Modulation (PWM) is. What is it used for?
2) Why 'volatile' variables are sometimes needed in C programs?
3) In your parts kit you have a four channel optoisolator: the LTV-827. The BJT in an optoisolator very rarely comes with a base pin connection. Explain how and why the BJT in an optoisolator is driven into saturation 'without' a base current. What is the definition of "current transfer ratio" (or CTR) in an optoisolator? What is the typical CTR of the LTV-827 optoisolator?
4) Find the pin out of the STP16NF06 MOSFET and LTV-827 optoisolator.

Laboratory

1) PWM signal generation. Using the example 'square.c' available in the WebCT page for the course as a template, create a new program called 'pwm.c'. The program should be capable of receiving two numbers between zero and one hundred from HyperTerminal, or any other terminal emulation program, and implement adjustable PWM signals for pins P1.6 and P1.7 at 100 Hz. You may want to use the standard C runtime library functions gets() and atoi(). Remember to make any variables shared between the main program and the ISR 'volatile'. Test the outputs using the oscilloscope at different PWM ratios. Draw in your notebook the output for the 0, 33, 66, and 100 PWM values. Also, record into your notebook the flowchart and listing of your working program.

2) Motor/Light-Intensity controller. PWM is often used for speed control of DC motors or to control the amount of power delivered to certain loads such as light bulbs or even LEDs. To achieve this, a suitable driver must be connected to the microcontroller. For this experiment you will be using an LTV-827 optoisolator and a STP16NF06 N-Channel MOSFET as shown in the figure below. Notice that the driver part of the circuit composed by the BJT, the MOSFET, and the 6V power supply is electrically isolated from the microcontroller part!



Demonstrate the working circuit to your TA/Instructor who will sign your notebook if everything works correctly. This circuit can better demonstrated using an small DC motor as a load, but if you don't have a motor handy, the load can be replaced with an LED in series with a 100Ω resistor. *Note: motors are part of the EECE28/EECE2821 final project kit, available for 45$ in the EECE stores (MCLD 112b). Only one kit is necessary for each group of three students!*