Laboratory 8 Programmable Interval Timer

Last Revised: July 2000

1 Objectives

With the completion of this lab, you should be able to:

- Interface and configure an 8254 PIT for correct operation.
- Use the 8254's count feature to check elapsed time.

2 Introduction

In this lab, you will design and build an interface between an 8254 Programmable Interval Timer chip and the MUN-88 single board computer. Carefully read the data sheets of the 8254 before coming to the lab.

The timer chip is useful for generating periodic signals, including square waves and rate waves. On most PCs, the internal speaker is connected to an IC functionally equivalent to the 8254; by setting an appropriate count, a program can generate different audio frequencies. Other features include ways to generate a consistently timed interrupt request, and to generate clock signals for other system devices (such as for serial communications).

3 Programmable Interval Timer

The MUN-88 microcomputer already has its own 8254 PIT chip; in this lab, you will add a new 8254 that can be selected via an IOSEL* line.

3-1 Circuit Diagram

Sketch a circuit diagram for interfacing the 8254 timer chip to MUN-88; show all the necessary connections. You may I/O map the timer device using any of the available select lines. There are three counters in this device. Pull up all the ***gate*** inputs of these counters, thus making all of them ready for use.

3-2 Clock Configuration

In this section, be sure to report all values, and to show relevant calculations. Report on your success.

Connect **PCLK** output line of the MUN-88 to **CLK 0** input of the first counter. Using monitor commands configure **Counter 0** in Mode 3 (square wave output) with an output frequency of 100 Hz. Test this before proceeding.

Connect the **OUT 0** output of the first counter to the **CLK 1** input of **Counter 1**. Configure this counter in **Mode 3** with an output frequency of **2 Hz** using monitor commands and test the operation.

Connect the **OUT 1** output of the second counter to the **CLK 2** input of **Counter 2**. Through a driver, connect **OUT 2** to an LED. Configure this counter in **Mode 2** (rate

generator) with an output frequency of **0.25 Hz**. Verify correct operation by observing the LED.

3-3 Strobe

Using monitor commands configure **Counter 2** in **Mode 4** (software triggered strobe) with a pulse duration of **5 seconds**. You may modify the other counters to get more accurate results. Verify correct operation.

3-4 Delay Program

Write a delay routine that approximately waits for 200 milliseconds. Calculate the estimated delay by counting the execution times of the various instructions in the routine, and by using the clock frequency of MUN-88. Use the **COUNT** feature in the counter to verify this estimate.

3-5 Return of the Rat

In laboratory 3, you wrote a game called *Rat-bashing*. The easiest way to generate a sequence of rats was to pre-program them in an array. Unfortunately, such a scheme means that the sequence is the same every time the program is run.

One way to create a seemingly random sequence would be to use a counter on an 8254 IC to cycle through a sequence of values; the program could then read the count value to generate a rat in a pseudo-random location.

- Explain in more detail how the method described above would work. Briefly describe the changes that your *Rat-bashing* program would require to incorporate such a feature.
- **Optional Bonus (2 marks)** – Assume that the only 8254 PIT available for use is the one on the MUN-88 board. ***Using only Counter 0 of the built-in 8254***, modify your original *Rat-bashing* program to show random rats. Note that some of the details in your above procedure might have to change. Demonstrate your program to a TA. There is one key problem with the *randomness* of the counter 0 – what is it?

4 Submission

At the end of the lab submit print-outs of your ***list files***. Include circuit diagrams, answers to all questions in the lab, and test results.

Before leaving the lab, deconstruct your circuit and return your components to the provided kits.