ECE 341 Lab 2

Zachary DeLuca

February 7th 2023

---

1 Introduction

In this lab we were focusing on the processes of delaying the while(1) loop. For this purpose we explored the variations of how to implement the delay function. The two types we tried in this lab were delays based on pure software, and one that relied on a hardware timer in conjunction with a software component.

2 Implementation

For this lab we used two separate functions, one that implements the hardware assisted delay and one that implements the pure software delay. The software delay is shown in listing 2.1 below

Table 1: Listing 2.1

```
void sw_msDelay (unsigned int mS){
    int i;
    while(mS –) // SW Stop breakpoint
        { for (i = 0; i¡ COUNTS_PER_MS; i++) // 1 ms delay loop
            {
            // do nothing
            }
            LATBINV = LEDA; // Toggle LEDA each ms for
    }
}
```

This section of the lab required that we calibrate the count per ms for the delay to be accurate, and after trial and error of running and testing with the oscilloscope, the count was found to be 8886 counts per ms to get the pulse to be 1 ms wide.

The next step was to implement the hardware section of the lab, which has its function shown

Table 2: Listing 2.2

```
void hw_msDelay(unsigned int mS){
    unsigned int tWait, tStart;
    tStart=ReadCoreTimer(); // Read core timer count - SW Start breakpoint
    tWait= (CORE_MS_TICK_RATE * mS); // Set time to wait
    while((ReadCoreTimer() - tStart) ¡ tWait); // Wait for the time to pass
    LATBINV = LEDA; // Toggle LED at end of delay period
}
```

above in the listing 2.2:

For this part of the lab we needed instead of changing the ticks per ms, we needed instead to change the core timer tick rate. The scaling that calibrated the specific board in the lab was to divide the rate by 1995.

3 Testing and Verification

To test how accurate each method is, we measures the real delay vs the target delay, as shown in the table below:

Target	SW	HW
1	1.00	1.00
10	10.00	10.00
100	100.00	100.00
1000	1000.00	1000.00

For this part of the lab, we found no measurable difference between the expected values and the measured values for our calibrated values. We were required to calibrate the values due to slight physical differences in each of the boards that would change the results from board to board.

The verification for this lab was using an oscilloscope to measure the true output and compare it to the expected value. There weren't variables to watch in this particular lab setup.

4 Post Lab Questions

–Under what circumstances might you use the software-only delay instead of the hardware-assisted delay? Conversely, when might you use the hardware-assisted delay over the software-only one?

You would use the hardware timer if you wanted to use a lower power option or if you wanted to free up the processor for another task and you would use the software timer when trying to increment much higher than the hardware timer could.

– What are the advantages and disadvantages with both types of delays?

The hardware timer uses embedded resources and allows the processor to work on other things whereas the software timer would be good for boards that don't have the dedicated hardware.

– Address the limitations of each delay: What is the smallest length of time you can delay with the two types of delays? What is the longest length of time you can delay for both types?

The smallest increment of the software timer is one clock cycle, which is set by the frequency of the processor, and the timer has a minimum time of 1 timer tick, which is slower than the software timer. The maximum for the software however is 49.71 days and the hardware maximum is 1.79 minutes.

– Describe potential method(s) that could be used to increase the maximum delay period for both the software-only and the hardware-assisted delay methods.

For the hardware method you could scale the timer using an internal scalar and the software method you could increase the bits by adding in additional incrementing variables.

5 Conclusion

In this lab we talked about the different kinds of delays and were able to implement two of those kinds. The lab was straightforward and made sense. The scaling of each of the delays was easy to do given the tasks at hand.