

## Lab 3 Report

Samuel Zhang (shz96), Zi Zhou Wang (zw3948)

Saadallah

2/10/2018

## Objectives

### **Requirements document**

#### 1. Overview

##### 1.1. Objectives: Why are we doing this project? What is the purpose?

The objectives of this project are to design, build and test an alarm clock. Educationally, students are learning how to design and test modular software and how to perform switch/keypad input in the background.

##### 1.2. Process: How will the project be developed?

The project will be developed using the TM4C123 board. There will be switches or a keypad. The system will be built on a solderless breadboard and run on the usual USB power. The system may use the on board switches and/or the on board LEDs. Alternatively, the system may include external switches. The speaker will be external. There will be at least four hardware/software modules: switch/keypad input, time management, LCD graphics, and sound output. The process will be to design and test each module independently from the other modules. After each module is tested, the system will be built and tested.

##### 1.3. Roles and Responsibilities: Who will do what? Who are the clients?

EE445L students are the engineers and the TA is the client. Students are expected to modify this document to clarify exactly what they plan to build. Students are allowed to divide responsibilities of the project however they wish, but, at the time of demonstration, both students are expected to understand all aspects of the design.

##### 1.4. Interactions with Existing Systems: How will it fit in?

The system will use the TM4C123 board, a ST7735 color LCD, a solderless breadboard, and be powered using the USB cable.

##### 1.5. Terminology: Define terms used in the document.

- Power budget: Maximum power that the power supply can drive. This limits the number of components that you can connect to a supply
- Device driver: Interface between a device and a software module. An example would be the functions defined in ST7735.c
- Critical section: Multiple threads access shared memory locations/variables. A higher priority thread has to interrupt another thread and write to a variable that the lower priority thread was reading.
- Latency: The time between a service request and the process actually being serviced
- Time jitter: The difference in the maximum latency and the minimum latency that occurs. Some systems have a strict minimum time jitter requirement.
- Modular programming: Break up your software into modules with distinct tasks. Each of the modules should be kept separate so that you can easily debug or change specific features of your system.

##### 1.6. Security: How will intellectual property be managed?

The system may include software from Tivaware and from the book. No software written for this project may be transmitted, viewed, or communicated with any other EE445L student past, present, or future (other than the lab partner of course). It is the responsibility of the team to keep its EE445L lab solutions secure.

## 2. Function Description

### 2.1. Functionality: What will the system do precisely?

1. It will display hours and minutes in both graphical and numeric forms on the LCD. The graphical output will include the 12 numbers around a circle, the hour hand, and the minute hand. The numerical output will be easy to read.
2. It will allow the operator to set the current time using switches or a keypad.
3. It will allow the operator to set the alarm time including enabling/disabling alarms.
4. It will make a sound at the alarm time.
5. It will allow the operator to stop the sound. An LED heartbeat will show when the system is running.
6. Support a secondary alarm
7. Supports a snooze alarm
8. Change clock backgrounds
9. Supports a stopwatch mode
10. Has a minute hand

### 2.2. Scope: List the phases and what will be delivered in each phase.

The lab is divided up into 3 parts: preparation, demo, and lab report. In the preparation, we are expected to have software written for each module as well as a main program to test the alarm. For the demo, we will show the TA our system with the alarm-selector as an extra feature. We will then answer any questions that the TA may have for our team. For the lab report, we will include this requirements document, a software diagram, hardware specifications, and answers to the questions shown in the lab manual.

### 2.3. Prototypes: How will intermediate progress be demonstrated?

A prototype system running on the TM4C123 board, ST7735 color LCD, and solderless breadboard will be demonstrated. Progress will be judged by the preparation, demonstration and lab report.

### 2.4. Performance: Define the measures and describe how they will be determined.

The system will be judged by three qualitative measures. First, the software modules must be easy to understand and well-organized. Second, the clock display should be beautiful and effective in telling time. Third, the operation of setting the time and alarm should be simple and intuitive. The system should not have critical sections. All shared global variables must be identified with documentation that a critical section does not exist. Backward jumps in the ISR should be avoided if possible. The interrupt service routine used to maintain time must complete in as short a time as possible. This means all LCD I/O occurs in the main program. The average current on the +5V power will be measured with and without the alarm sounding.

### 2.5. Usability: Describe the interfaces. Be quantitative if possible.

There will be two to four switch inputs. In the main menu, the switches can be used to activate 1) set time; 2) set alarm; 3) turn on/off alarm; and 4) display mode. The user should be able to set the time (hours, minutes) and be able to set the alarm (hour, minute). Exactly how the user interface works is up to you. After some amount of inactivity the system reverts to the main menu. The user should be able to control some aspects of the display configuring the look and feel of the device. The switches MUST be debounced, so only one action occurs when the operator touches a switch once.

The LCD display shows the time using graphical display typical of a standard on the wall clock. The 12 numbers, the minute hand, and the hour hand are large and easy to see. The clock can also display the time in numeric mode using numbers.

The alarm sound can be a simple square wave. The sound amplitude will be just loud enough for the TA to hear when within 3 feet.

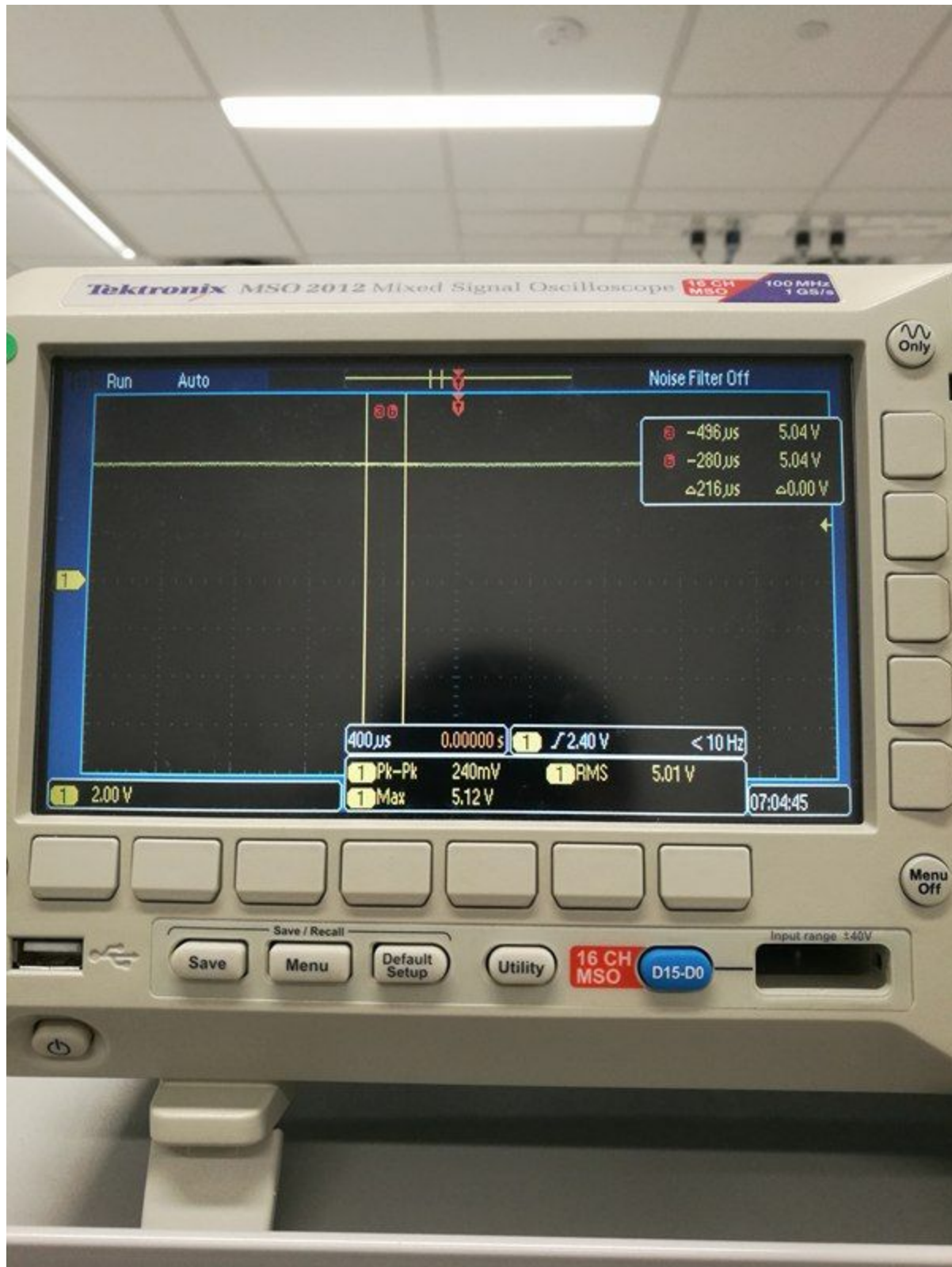
The alarm sound will be VERY quiet in order to respect other people in the room during testing. Connecting or disconnecting wires on the protoboard while power is applied may damage the board.

A lab report described below is due by 2/16/2018. The lab report will contain this requirements document, software and hardware designs, and answers to a few questions defined in the lab manual.

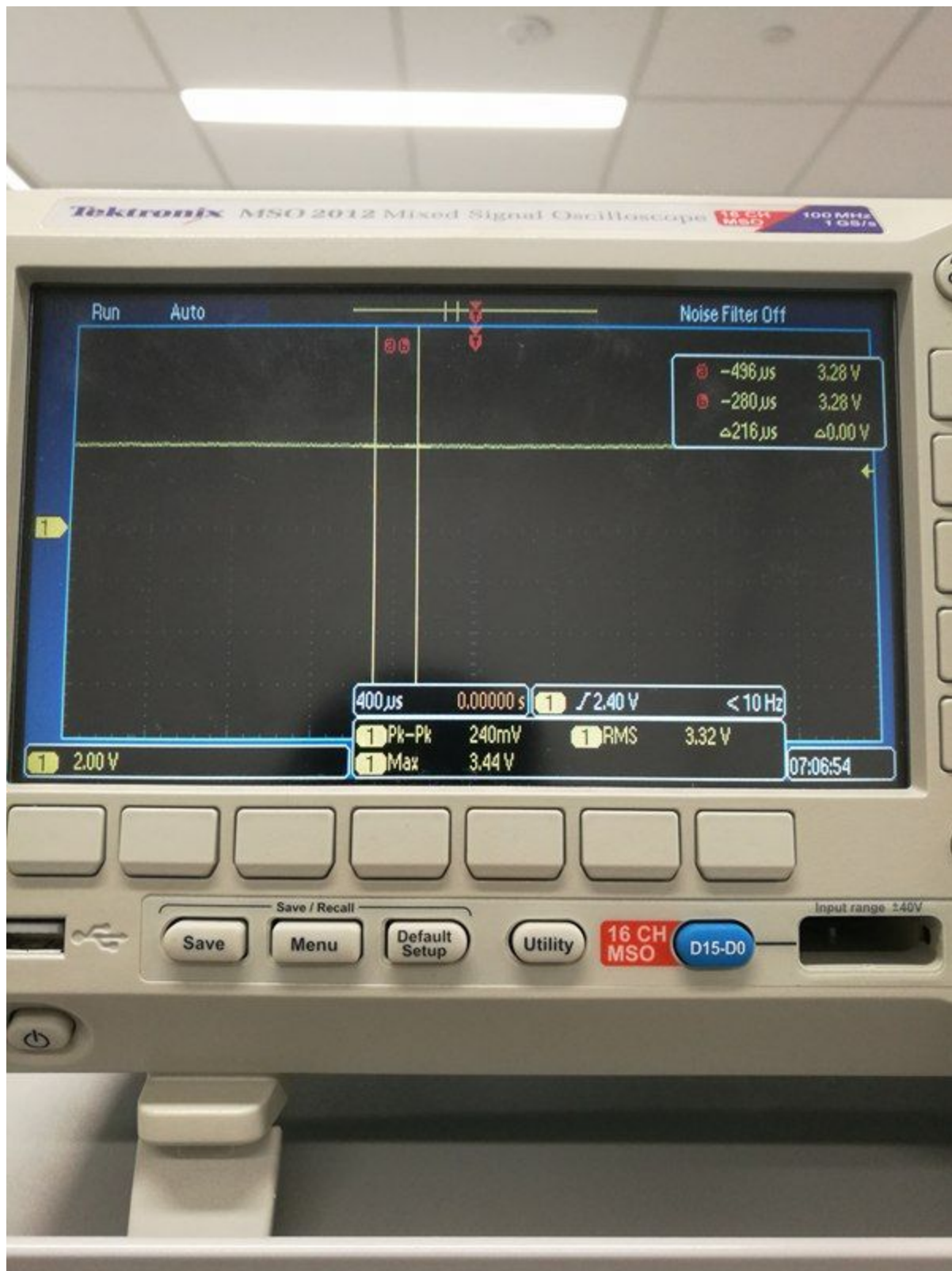
There are three deliverables: preparation, demonstration, and report.

## Measurement Data

+5 supply voltages versus time and the rms magnitude

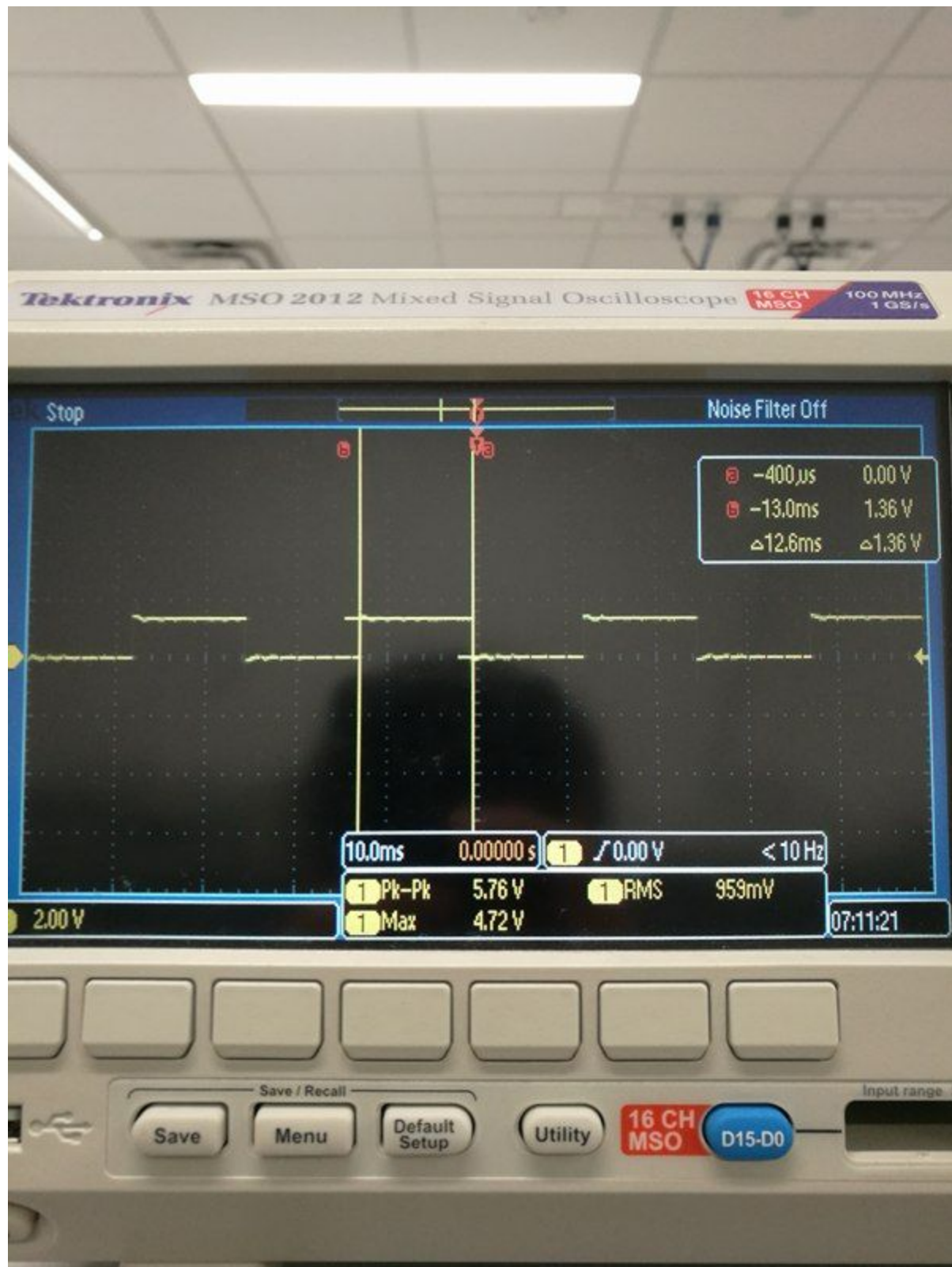


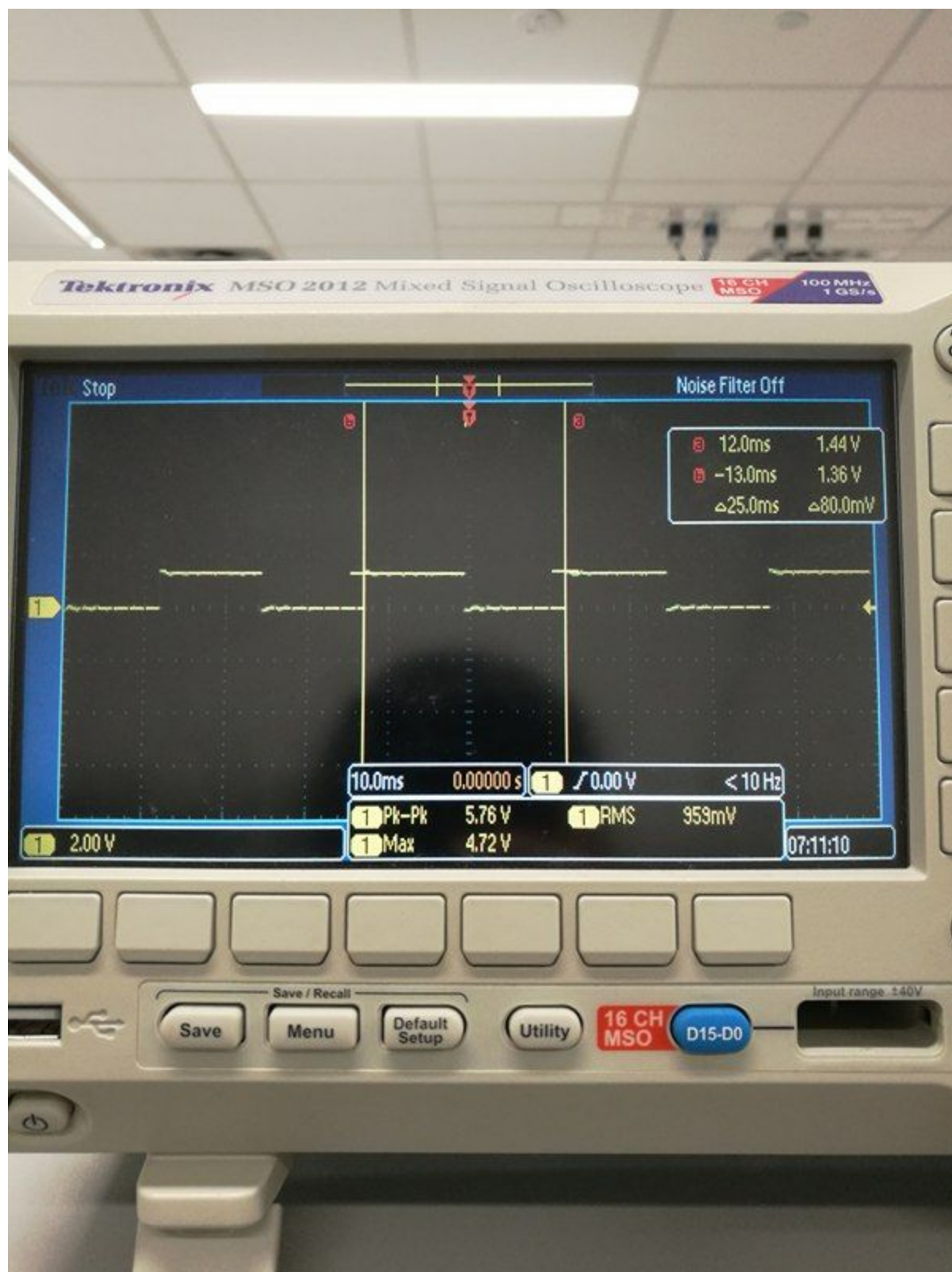
+3.3 supply voltages versus time and the rms magnitude





speaker voltage (or output voltage) versus time during an alarm sound





Measurements of current required to run the alarm clock, with and without the alarm







### Analysis and Discussion

1. Give two ways to remove a critical section.

- 1 - For each area of hardware that you access, make sure all accesses are made from the same thread.

- 2 - Disable interrupt

2. How long does it take to update the LCD with a new time?

0.02443638 seconds

3. What would be the disadvantage of updating the LCD in the background ISR?

Updating the LCD in the background ISR will make the interrupt more intrusive as it will take more time.

4. Did you redraw the entire clock for each output? If so, how could you have redesigned the LCD update to run much faster, and create a lot less flicker?

We redrew the entire clock every second, as we would need to redraw the entire clock image whenever the second hand changes. We could have improved this by shortening the watch hands so that they did not overlap the clock face background (numbers and circle). Then, when updating the analog clock face, we could have updated just the clock hands and not the background.

5. Assuming the system were battery powered, list three ways you could have saved power.

- 1 - We could have operated the speaker at a lower voltage so that it would have consumed less power.

- 2 - We could have optimized our clock to use fewer lit pixels (keep more pixels black) so that the total display brightness is lower.

- 3 - We could have turned off the display if the system hadn't been touched for a certain period of time.