EEENG490, Capstone:

Integrated Oscilloscope/Logic Analyzer/Signal Generator

Jeremy Munson Braeden Reed Bryant Clark

Eastern Washington University
Electrical and Computer Engineering Department

Student Report

Mar 21, 2024

Contents

List of Figures

1 Abstract

2 Proposal

2.1 Introduction

In engineering laboratories and industrial settings, electrical engineers make use of a wide variety of test equipment. The large number of devices becomes difficult to manage on a desktop, especially in a University setting where twenty or more stations need to fit into a single room. We will construct a product that is able to cover all of the common test-equipment and lab needs for electrical engineering education with a single device that is still capable enough to fit into industrial use.

2.2 Motivation

Often, a proper electrical engineering workstation would need many devices: Bench power supplies, digital multimeters, signal generators, an LCR meter, logic analyzers, and oscilloscopes are the most common pieces of equipment...

2.3 Constraints

- Time
- Money
- Part Availability

2.4 Schedule

Oh Fuck. We've used half our time.

2.4.1 Gantt Chart

2.5 Costs

It will cost Jeremy way more than he wants to admit.

2.6 Deliverables

It'll make you breakfast and run your feet.

2.7 Conclusion

This thing will kick ass.

3 Team Organization

3.1 Background

3.1.1 Jeremy Munson

PCB Design Experience

3.1.2 Bryant

Programming Internship

3.1.3 Braeden

3.2 Responsibilities

3.2.1 Jeremy Munson

- PCB Design
- Power Supplies
- FPGA Programming
- Assembly
- Part Ordering

3.2.2 Braeden

- FPGA Programming
- Breadboard Pin Function Selection
- Embedded Programming

3.2.3 Bryant

- EZ USB
- User Space Software
- Embedded programming

4 Initial Research

4.1 Project Feasibility

Wanting a product to exist is all well and good, but if we don't consider the many moving parts needed to build a working product then we could be left with nothing but a dream, and no substance.

4.1.1 Oscilloscope

The oscilloscope really is the center-piece of the design. Preliminary research shows than often, oscilloscope manufacturers resort to ASIC[1] silicon for the Analog-to-digital conversion components. Since we don't have the millions of dollars and years of engineer time to design an ASIC, we will need to find a readily available chip that can perform the ADC conversions at an acceptable rate for an oscilloscope.

Jeremy spent many hours scouring over part catalogs, looking for a suitable chip that is cost-effective. Fortunately, a line of chips ... HMCAD

4.2 Similar Products

There are several products on the market that attempt to address the same goal, but they each fall short in one or more ways.

4.2.1 NI Instruments Elvis III

- Costs over $3000
- Oscilloscope updates at 0.5hz
- 50-ohm signal generator, but incapable of driving 50-ohm terminated load
- Slow Interface
- 20 second DMM settling time

5 Project Overview

5.1 Design Criteria

yep.

5.2 Overview

5.3 Architecture Block Diagram

Figure (1) is referenced here, as an example.

[1] Application Specific Integrated Circuit

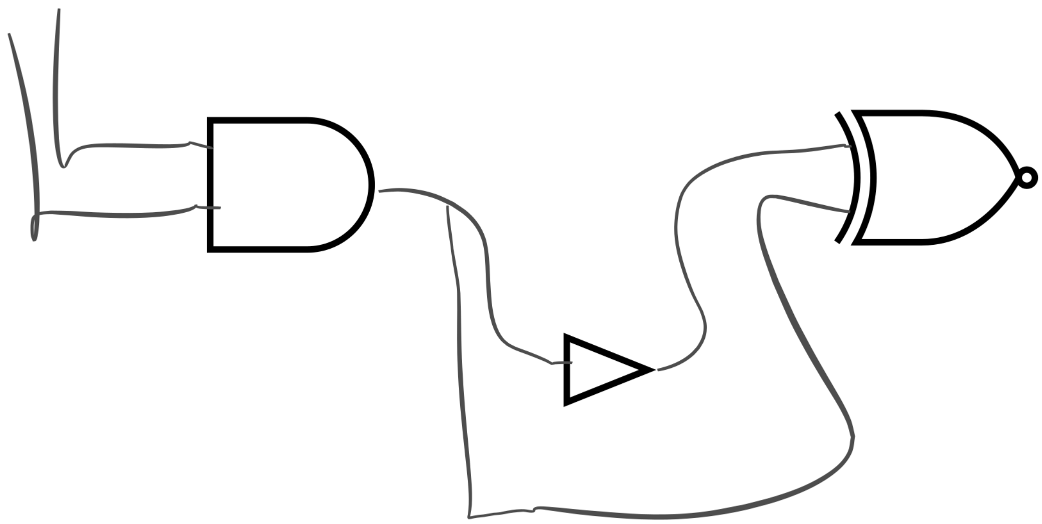

Figure 1: Block diagram showing high-level design

5.4 Interfaces

5.4.1 USB

5.4.2 Parallel

5.4.3 QSPI

6 Research of Proposed Solutions

6.1 Gsps ADC: HMCAD2011

The core of the oscilloscope design, the Analog To Digital converter takes the analog voltages that we want to measure.

6.2 Signal Generator

An unlikely source of capable signal generator chips: Display drivers! There are numberous triple-channel DAC chips that are intended to drive displays at a low(ish) cost. We can re-purpose one of these display driver chips as a three-channel signal generator, instead of a 3-color pixel control.

6.3 USB 3.2 Gen 1 Interface

The all-important interface that allows data to be delivered to a computer to be analyzed: the USB 3 interface. Our project is extremely data intensive so we had to find a solution that would have high bandwidth at a low cost. We decided to use Infineon's EZ-USB™ FX3. It is relatively affordable and it can transfer data at five gigabits per second. It can be configured for most common peripheral communication protocols such as I^2C, I^2S, UART, and SPI but we will be configuring it to run, effectively, our own communication protocol because that allows us to use the full 32-bit wide parallel bus. This will allow us to get the high data rates that we need to get all the data from the FPGA to the computer application.

6.4 Artix-7 FPGA

The glue that brings everything together. No other (feasible) device would be capable of aggregating the veritable firehose of data that we will be collecting. The FPGA forms the center of our design.

6.5 STM32, Providing ADC and DAC for logic analyzer

The STM32G474RE is a critical part of the design. It has several ADC's and DAC's at its disposal with a variety of useful configurations. Four 12-bit unbuffered internal DAC channels with 15 MSPS allows the board incredibly fast output, providing accurate data display for high speed processes. Five 12-bit ADC's provide the board with means to sample and gather data.

6.6 Power Supply

There are multiple aspects to the power supply. The standard power from the wall is 120v which must be stepped down to the desired voltages. There will be multiple voltage step downs, the first being accomplished by a switching power supply that will bring the voltage down to 24v. From there several voltage step down circuits will be placed in parallel, each will step the voltage down to a different amount. The range of voltages currently intended is 15v, 10v, 5v, and 3,3v. These voltages will have to be buffered to avoid causing any noise to the board as the board will feature an oscilloscope that will be sensitive to the noise.

6.7 Bread Board Unit

While test equipment often has BNC connectors, or test leads, or banana plug jacks to connect wires to your project, we would like to take a different approach. We want to have a breadboard that can be switched onto all of the possible connections at each pin in software. This lets up place components down on a breadboard, use jumper wires to form the connections, then apply a stimulus with the signal generator or power rail functionality using software. This allows exploring a wide variety of stimuli quickly. Since every pin will also be an analog logic analyzer channel we will be able to visualize the operation of the entire circuit at once.

6.8 NGScopeClient

Once the data is gathered from the hardware it needs to be presented in a readable manner. We originally planned to build our own front-end application. Bryant has experience with front-end development and we did not know of any open source front ends that we liked. However, during Jeremy's hardware research, he spoke extensively with an engineer in the field who happened to mention ngscopeclient.org to him. When we saw it, we were sold. It is an open-source platform that works across OSX, Windows, and Linux. It can be interacted with using a C/C++ backend and it has beautiful graphics as shown in figure 2. It also has many built-in features for recognizing communication and data types that are very useful for our logic analysis. These features fit our application perfectly.

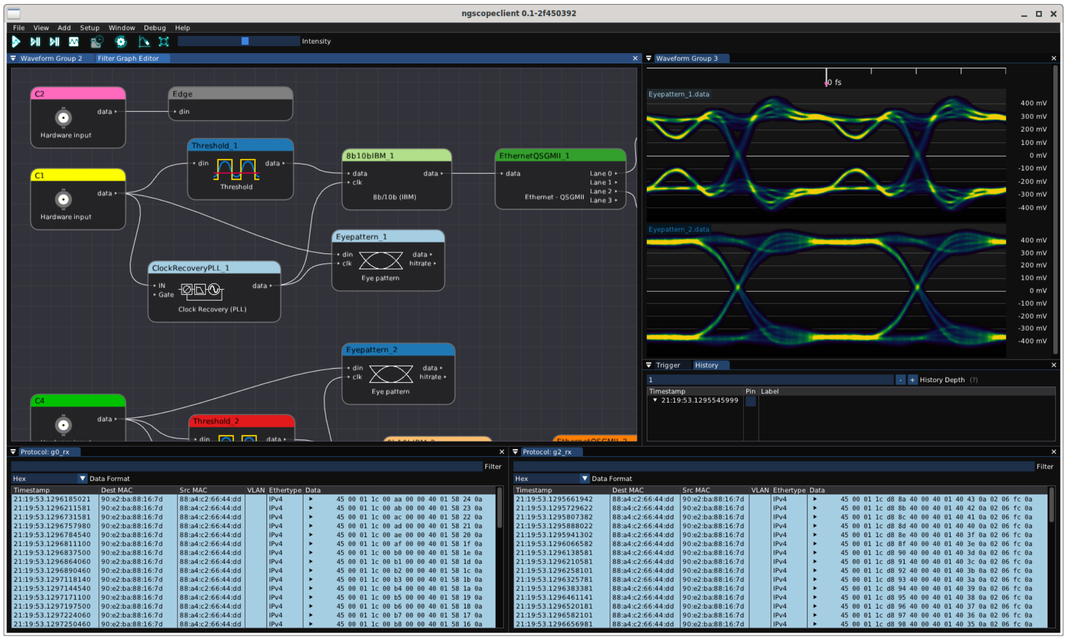

Figure 2: NGScopeClient Graphics [1]

7 References