Arpan Choudhury arpanc2
Joseph Yang josephy2
Robert Conklin rmc2

Digital Logic Smart Breadboard Proposal
Educational Smart Breadboard Rework (Spring 2018)

1. Introduction

1.1 Objective

When it comes to engineering prototyping, breadboarding is the gold standard for early hardware testing. However, as the system being prototyped gets larger in size, for example a 4-bit calculator as implemented in ECE385 for Lab 3, debugging can become extremely challenging. This is due to the overlapping clusters of wires, along with limited insight into what is going on in each part of the circuit. This increase in complexity makes the difficulty in debugging the system significantly greater, and can lead to a strong sense of frustration. This experience could potentially drive away people interested in the field of electrical engineering.

The goal of this project is to make debugging easier on a breadboard. We intend to do so in two main ways through a smart breadboard. First, we will be able to read digital logic values in every row on the breadboard using IO expanders communicating with a microcontroller via I2C. Second, the breadboard will be able to write logic values to each of the rows on the breadboard using the same IO expanders. The state of the breadboard will be configured by and communicated back to the host PC. The user will then be able to adjust inputs to their logic circuit, and observe the output of their circuit.

1.2 Background

An important part of hardware design is testing and validation. This is especially important in digital logic circuit design, and can get extremely complicated to debug more complex logic circuits due to the increasing number of important signals to track. For students and novice circuit designers, the debugging process can be especially painful, as traditional approaches to debugging logic circuits with LEDs and probing of the breadboard offer a narrow window into the processes going on in a circuit. The goal of our project is to widen this window into the behavior of digital logic circuits, and provide a more comprehensive set of tools for not only debugging existing circuits, but also testing them to ensure proper functionality. This will be done by allowing the user to both read and write logic values to the rows of the breadboard using a host PC and a provided software library to handle communication with the breadboard.

A similar project was originally proposed by Chinemelum Chibuko, Minseong Kim, and Mostafa Elkabir in Spring 2018 that was called the Educational Smart Breadboard[1]. Their original design was a standalone device capable of reading voltages from the rows of a breadboard and displaying them on an integrated touchscreen display. Their design divided the full breadboard into 8 distinct sub-boards that were multiplexed into the microcontroller using a hierarchical tree structure. Our approach to tackling the problems with debugging complex breadboard circuits differs from the original design in a few key details.

The original design included a touchscreen interface to read and display information about the state of the breadboard. This interface would not lend itself for more complex debugging and automation of digital logic circuit testing. Our design would instead communicate over USB with a host PC to configure the breadboard and acquire data. By shifting away from the standalone device design and instead opting for a USB peripheral, our design allows for more straightforward and versatile use of the device while maximizing functionality. Furthermore, the hardware architecture of the original design also greatly limited its functionality in regards to data acquisition and logic testing. This is because the multiplexing of the sub-board rows prevented the microcontroller from interacting with multiple sub-boards at a single time. This meant logic values could not be written to different sub boards at the same time, which would be incompatible with any testing that would include configuration of inputs. Alongside this, the original design highlighted the ability to test any IC with a fixed truth table. While this is useful, it excludes many ICs from testing. This is why our hardware architecture utilizes IO expanders communicating with the central microcontroller over the I^2C protocol instead of multiplexers. It allows for greater ability to configure the state of the breadboard, while minimizing the cost and complexity of the design. As a whole, our design changes improve the versatility of the device, and allow for debugging of much more complex digital circuits.

1.3 High Level Requirements

- The device can read and write logic values to each row of the breadboard.
- The device can utilize logical high values at 3.3V or 5V.
- The device is capable of communicating with a host PC over USB 2.0.

2. Design

2.1 Block Diagram

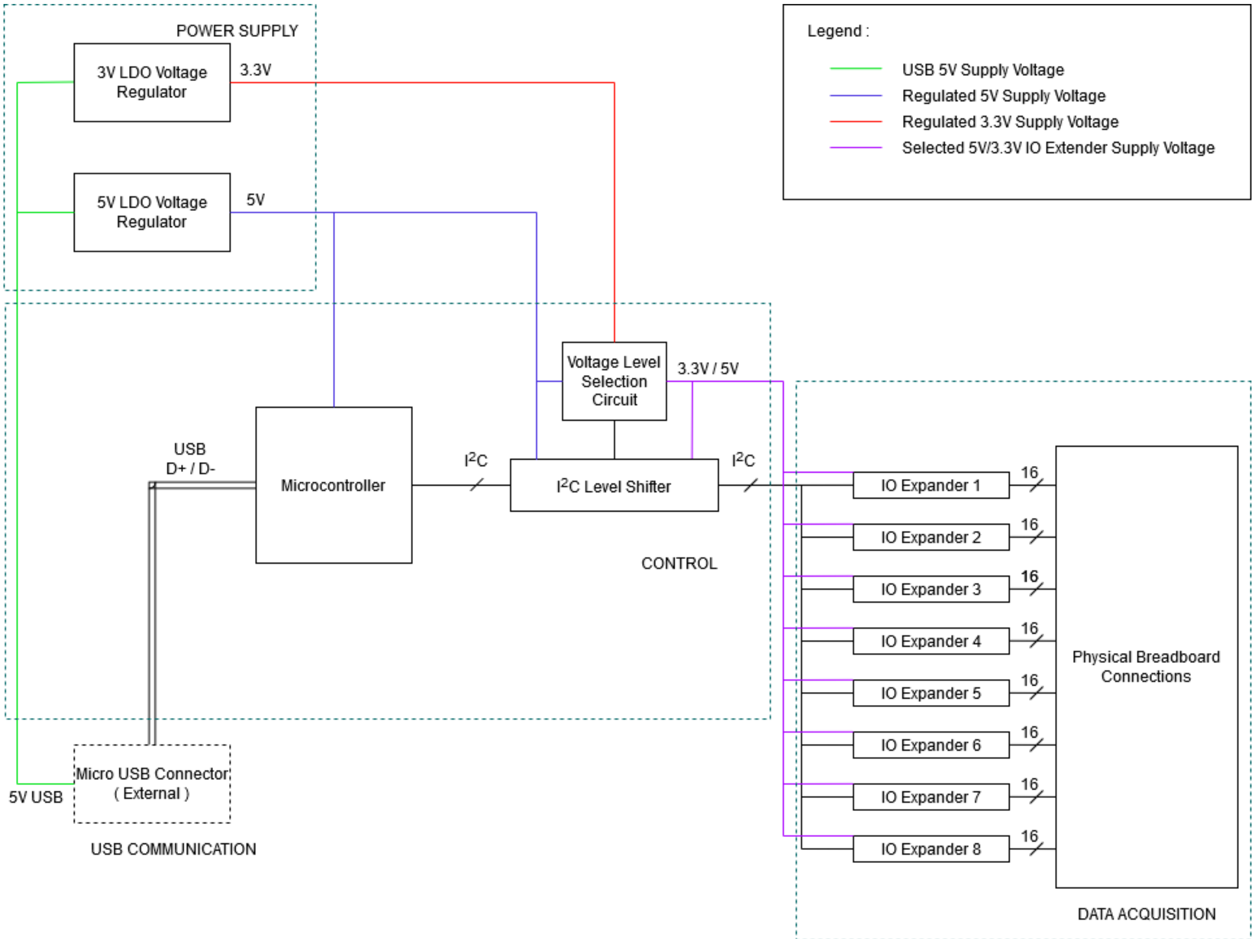

Figure 1. Block

2.2 Physical Design

The device will be mounted to the backs of two 400-tie modular breadboards [2], connecting the metal rails on the back of the breadboard to a PCB integrating all the blocks in the design. Each of these rails will connect to a line on one of the eight IO expanders in our design. This allows the device to interface with each signal in the logic circuit implemented on the board, as well as read and write logic values in each of these rows. Figure 1. shows a typical 400-tie breadboard, onto the back of which our PCB would be attached to access the signals in each row.

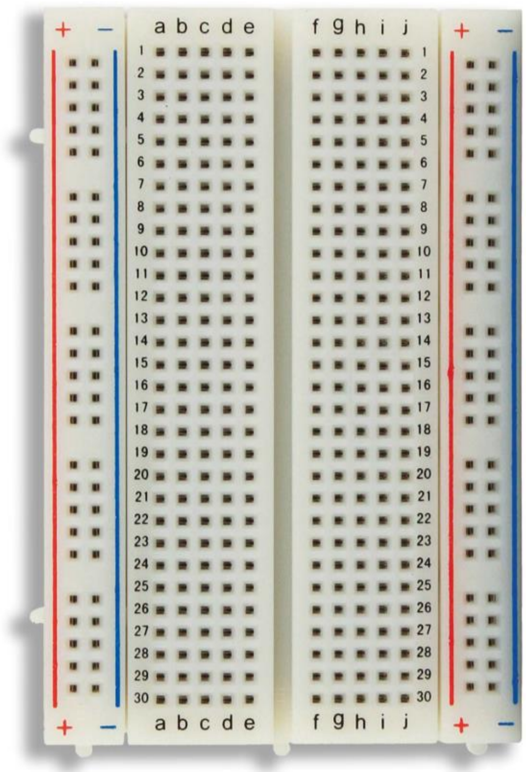

Figure 2. GS-400 Modular

2.3 Functional Overview

Our device can be split into four distinct blocks, each with specific functions in facilitating the reading and writing of signals on a breadboard. First, the USB Communication block externally connects to a host PC, through which users interact with our device. Second, the control block consists of two components: a microcontroller and I^2C Level Shifter. The microcontroller controls the level shifter which then manages the signals from the IO expanders in the Data Acquisition block via I^2C. The control block also features a voltage level selection circuit to distribute the needed voltage values to dedicated IO expanders. Third, the Data Acquisition block consists of several IO expanders that connect externally to our breadboard with the purpose of reading/writing voltage values. Finally, the power supply block provides necessary power to corresponding components.

2.4 Block Level Requirements

2.4.1 Control

Block Requirement: The control block should manage data acquisition, and configure peripherals for USB communication, voltage selection, and data acquisition.

The control block will consist of an ATmega32U4 microcontroller that will communicate with the host PC through USB and IO expanders through I^2C, and a voltage selection circuit to set the operating voltage of the IO expanders. The microcontroller will operate at 5V from the 5V regulated supply from the power system. Due to the required capability of operating at multiple voltages, a level shifter is required between the microcontroller and I^2C peripheral devices.

The control system will also process the instructions and information sent from the host PC, appropriately configure the required peripheral devices, acquire requested data, and transmit it back to the host PC. This process will be handled on the device, at the request of the host PC.

2.4.2 Power System

Block Requirement: The power block should provide 3.3V and 5V for the device, as well as be capable of providing enough current for the components.

The power system will consist of two power circuits in order to provide the board with two voltage levels: one at 5V and one at 3.3V. Each of these power circuits will consist of a voltage regulator and its needed components (capacitors and diodes). Our USB input can provide up to a maximum of 500 mA at 5V. This means the maximum power through our system is 2.5W.

The voltage regulators take in the input voltage and supply a different output voltage. One regulator will supply 5V, and the other will supply 3.3V. Because the output of the first regulators is the same as the input, the component we use must be a low dropout regulator. These components must have a large current tolerance as there is a relatively large amount of current going through the circuit.

2.4.3 Data Acquisition

Block Requirement: Must be capable of performing digital read and write operations at 3.3V and 5V for each row in the breadboard, as configured by the control block.

The data acquisition block consists of eight IO expander ICs, interfacing with the control block's microcontroller via I^2C. The IO expanders must operate at both 3.3.V and 5V, and communicate with the microcontroller at whichever operating voltage they are using at the time. This is done using a high-speed bidirectional level shifter, specifically used in I^2C applications. This allows the microcontroller to always operate at 5V, while allowing the peripheral devices to operate at a user-defined voltage.

2.4.4 USB Communication

Block Requirement: Able to transmit data to and receive data from the device on a host PC.

The block powers the device and communicates with the PC via a standard USB 2.0. In order to communicate effectively between the host PC and our breadboard, we are planning to use the open source library libusb [3] to manage the USB communication with the microcontroller. There are multiple benefits for choosing libusb. First, it is an active open source library which our users have the benefit of accessing online forums. Second, it is compatible with multiple platforms, including Windows, IOS, Linux, etc. Third, libusb is adaptable to multiple languages. It very well suits our needs because the educational breadboard could be used across a broad scope of operating systems and coding environments.

2.5 Risk Analysis

The USB hardware and communication will be the greatest challenge in our design. This is because our group has little to no prior knowledge and experience in dealing with implementation of USB communication with an embedded device, and the importance of this part of our device in relation to the other systems. Due to the design decision to solely configure the device via USB communication, this places the highest priority of functionality on this system to work as expected. A good deal of time must be spent researching hardware and software requirements to meet the USB 2.0 device standard to ensure our device can communicate with the host PC. Overall, this system poses the highest risk and has the highest importance out of all the other systems in our design.

3. Ethics and Safety

There are a number of safety issues that could arise depending on the user's decision in creating a circuit. Careless actions such as using too high a voltage or shorting a line somewhere could damage used chips or burn certain components. A regular breadboard is typically rated at 5W [4], so operating components at values above this could result in parts catching on fire or exploding, which endangers people in the surrounding area. The ACM Code of Ethics and Professional Conduct specifies avoiding harm, 'unless there is a compelling ethical reason to do otherwise.' [5] In accordance with these guidelines, we will shield our circuitry from the user such that it is not easily tampered with. The user will not be able to come in contact with anything that could cause harm. With a commercial product, we would also warn the user of potential consequences of poor circuitry design.

Our approach to the potential safety issues with our device fall in line with the first IEEE code of ethics: "to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment" [6]. The safety of the user will always be of utmost importance.

4. References

[1] Chinemelum Chibuko, Minseong Kim, and Mostafa Elkabir, “Educational Smart Breadboard,” *courses.engr.illinois.edu/ece445*, February 7, 2018. [Online] Available:
https://courses.engr.illinois.edu/ece445/pace/view-topic.asp?id=25937
[Accessed: March 22, 2020]

[2] Global Specialties, “GS-400: Solderless Breadboard, 400 Tie-Points, with Bus Strip” *globalspecialties.com,* 2017. [Online]. Available: https://www.globalspecialties.com/solderless-breadboards/breadboards-singles/37-gs-400.html
[Accessed: April 2, 2020]

[3] libusb, “libusb,” *libusb.info*, 2012. [Online]. Available:
https://libusb.info/
[Accessed: March 31, 2020]

[4] George Leger, “Common Breadboard Specifications,” *circuitspecialists.com*, March 29, 2014. [Online] Available:
https://www.circuitspecialists.com/blog/common-breadboard-specifications/

[5] ACM, ‘ACM Code of Ethics and Professional Conduct,’ *acm.org*, [Online]. Available:
https://www.acm.org/code-of-ethics
[Accessed: February 4, 2020]

[6] IEEE, ‘IEEE Code of Ethics,’ *ieee.org*, [Online]. Available:
https://www.ieee.org/about/corporate/governance/p7-8.html
[Accessed: February 4, 2020]