Development of an Integrated Platform for Modular Prototyping: the ‘Mod Duino’

Undergraduate Honors Thesis

Presented in Partial Fulfillment of the Requirements for Graduation with Honors Research Distinction, Bachelor of Science at The Ohio State University

By

Patrick Daniel Flannery

Undergraduate Program in Mechanical Engineering

The Ohio State University

2017

Defense Committee:

Sandra A. Metzler, D.Sc., P.E., Advisor

Blaine Lilly, PhD

ABSTRACT

One crucial portion of a design project involves the design and build of a prototype. Prototypes are advantageous because they help to reduce a project's time and cost, identify and address problems early on in the design process, and increase user involvement by confirming a common vision among users and developers. These factors all contribute to delivering a better quality final product. Embedded electromechanical prototypes use programmed electronics and a mechanical device to perform a specific function. Microcontrollers are small computers on integrated circuits that are often used to control the electronics of these embedded prototypes. One set of problems with these prototypes is that the breadboard circuitry can easily become messy and the connections between sensors/actuators and the microcontroller are not reliable. These factors make it difficult to troubleshoot the prototype when a wire accidentally gets disconnected and also makes it difficult to create a portable prototype. There are not many existing products that focus heavily on the integrity of the connections between the main prototyping platform, such as an Arduino microcontroller, and the sensors and actuators used to create the operational portion of the prototype. Previous research indicates that one line of products, the Tibbo Project Systems, focus on robust connections. However, the Tibbo products are complicated to use and are not open source. The widespread popularity and ease of use of the Arduino microcontroller for these simple electromechanical applications make it a perfect candidate for use in a comparable platform. The objective of this research project is to create a robust, easy-to-connect prototyping system for embedded products that can be used with a variety of sensors and the Arduino Uno microcontroller board. The product is expected to form a rigid and clean looking platform that provides a reliable connection between the Arduino and any number of additional modules needed to complete an embedded electromechanical prototype.

ACKNOWLEDGEMENTS

This research was supported by the College of Engineering Undergraduate Research Scholarship Fund. I would like to thank my advisor Drs. Sandra Metzler and Blaine Lilly, who sat on my defense committee. I would also like to thank Associate Professor Robert Siston, Akul Kakumani and the other research students in the Human Centric Research and Design Group, and my parents (especially Dr. Daniel J. Flannery for spending hours editing this document) for providing me with assistance and encouragement to conduct and carry out my undergraduate research.

Vita

May 2013……………………………………………………………………St. Edward High School

August 2013 – December 2017……………………………………….B.S. Mechanical Engineering,

The Ohio State University

Field of Study

Major Field: Mechanical Engineering

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Purpose of Research

There has recently been an uptick in activity among the DIY, Maker, and Hacker communities. These inventors, engineers, and designers work to create products that will benefit themselves or others by collecting information, displaying that information, and sometimes acting on the information. Constructing a prototype is a critical part of the design process used to create a new product. Prototyping is important because it helps to reduce time and cost, identify and address problems early on, confirm a common vision among users and developers, and help the designer deliver a better quality system [1]. Currently, once a member of these communities takes his/her ideas to the prototyping phase, they often times start wiring sensors and actuators to a microcontroller.

Unfortunately, many of these prototypes are unorganized and are not easily replicated. It is often difficult to house a microcontroller, breadboard, and an assortment of sensors and actuators in a way that is both aesthetically pleasing and functional (Figure 1). As a result, members of

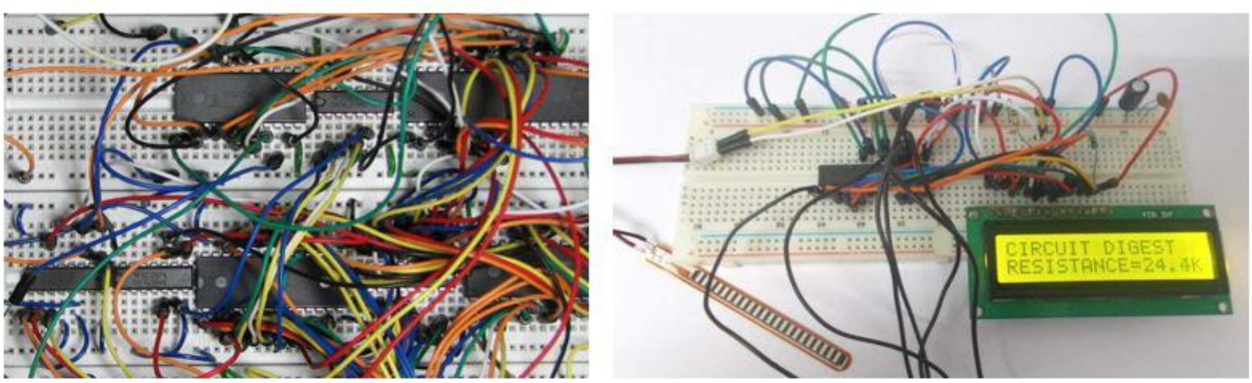


Figure 1: Examples of Typical Breadboards used in Prototyping [2 and 3]

these communities have difficulty creating prototypes that are structurally stable enough to be transported. These complexities create a gap in the market for a modular prototyping platform, which would allow for easy and reliable connection between the microcontroller and sensors that are needed for a particular prototype. The goal is to create an integrated style product, deemed the 'Mod Duino,' that will bridge this gap by securing modules (which hold sensors) to a main board that houses the microcontroller.

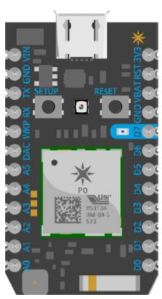
1.2 Related Work

Currently, there are several other products and services that are also attempting to create an easier path to prototyping. Defining what is already available to the consumer will narrow down the final design that builds upon the advantages of these existing products while clarifying which disadvantages need to be improved upon. These products can be placed in a variety of categories including developmental/ educational systems, mechanical prototyping systems, mechatronic prototyping systems, hardware enclosures, and embedded system platforms.

1.2.1 Developmental/ Educational Systems

Developmental and educational systems are the most widely used by young engineers and those new to prototyping, the target audience of this research. Some examples of developmental/ educational systems are Arduino, Raspberry Pi, micro:bit, and Particle (shown below in Table 1). Each of these platforms have their own websites, where you can purchase microcontrollers and learn how to code and prototype through examples and tutorials. One major advantage of these platforms is their low cost. An individual interested in learning the prototyping process can typically do so with one of these platforms for under forty US dollars. Another advantage is that these platforms are all open source, meaning the code is accessible by the public. These platforms also have communities that consist of people who are willing to help solve any problems a beginner might encounter.

Table 1: Developmental/ Education Systems

Photo	System	Product	Cost
 [4]	Arduino	Arduino Uno Rev3	$22.00
 [5]	Raspberry Pi	Raspberry Pi Model B	$35.00
 [6]	Particles	Photon	$19.00
 [7]	micro:bit	BBC micro:bit	$14.95

Arduino, Raspberry Pi, and Particle each offer their own forms of shields. Shields are "boards that can be plugged on top of [the main microcontroller] extending its capabilities. The different shields… are easy to mount, and cheap to produce" [8]. Although these shields are often beneficial for prototyping, they do not encourage a beginner to understand all the components they include. Furthermore, the shields do not provide an overall enclosure for the entire prototype, and do not assist in creating a reliable connection for any additional sensors that might be necessary for a particular prototype.

1.2.2 Mechanical Prototyping System

This type of system offers an assortment of metal and plastic parts that can be joined together in a variety of ways. Actobotics® is a "ball-bearing based, precision building system" that can be found and purchased on the ServoCity website. Some of the available Actobotics® components can be seen below in Figure 2. Although this type of system can be very useful for

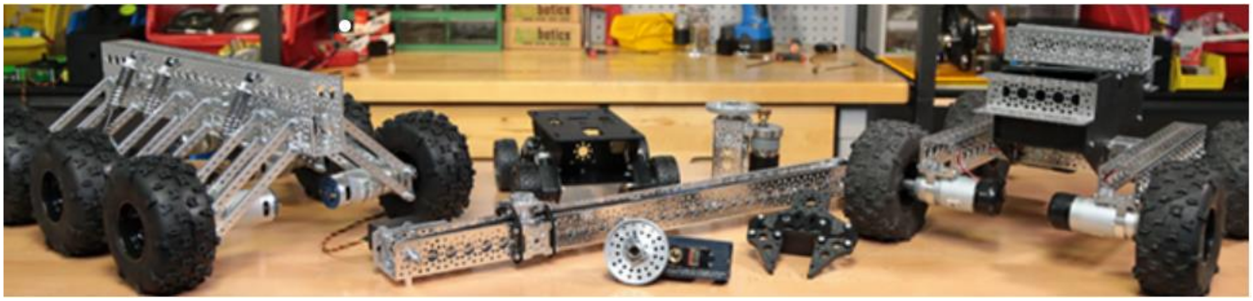


Figure 2: Actobotics® System Components [9]

completing a mechanical prototype, especially one that needs to move, most of the electronic components that can be purchased through the company are motors. This system will not be beneficial to someone trying to gain experience in a wide range of sensors. Actobotics® does not offer any type of bare board system that would allow for the addition of sensors. Even if they did, the mobile applications that this system offers are not always wanted or needed in basic prototyping projects. Additionally, some of the high end Actobotics® kits can cost up to two hundred and eighty US dollars [10].

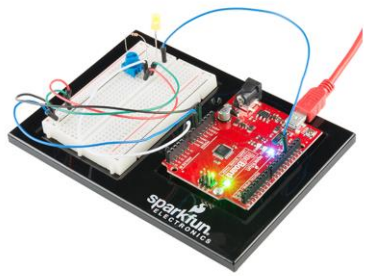
1.2.3 Mechatronic Prototyping System

Mechatronic prototyping systems include a set combination of hardware and software that can be used for a variety of prototyping projects. One approach, called the 20-SIM system, is a modeling and simulation tool that claims to help lower the cost of prototyping mechatronic systems. The program "enables modelling of physical systems such that the simulation model has a close resemblance to the physical reality and with parameters directly related to the physical phenomena" [11]. However, the 20-SIM system does not provide any software or hardware that can be directly used to more quickly put a prototype together. Instead, it focuses on making the prototyping process faster by more easily validating whether certain requirements of a physical prototype are met by simulating a model of that system. A mechatronic system like this may be useful for someone experienced in prototyping, but may be too complicated for a novice trying to get exposed to prototyping.

1.2.4 Hardware Enclosures

In search for something that focused heavily on holding the microcontroller to create a portable prototyping platform, a group of hardware enclosures were found on electronic retailer websites like SparkFun. Two examples are the Arduino Project Enclosure and the Arduino and Breadboard Holder shown below in Table 2. These products do a good job of allowing the

Table 2: Hardware Enclosures

Photo	Product	Cost
 [12]	Arduino Project Enclosure	$11.95
 [13]	Arduino and Breadboard Holder	$3.95

prototypes to be transported from place to place, and they are both relatively inexpensive. One limitation of the Arduino Project Enclosure is that it does not leave much space for any additional electronic components to be added to the microcontroller before the box is closed. Even if a couple of sensors fit, they would not be held securely within the enclosure. The Arduino Breadboard Holder does not solve the issue of wires becoming loose or detached all together. All of the wires used to connect the sensors to the microcontroller are completely exposed and vulnerable.

1.2.5 Embedded Platforms

Embedded platforms aim to ease the prototyping process by providing a computer system that can function at low power in a small area. All of these platforms also offer sensors that can be connected to the particular microcontroller in that small area. Liquidware's BeagleBoard, the iProtoXi Aistin, the Voxel8, and the Tibbo are all examples of embedded platforms.

Liquidware is a company that now offers three modules for the BeagleBoard embedded platform. The three rapid prototyping modules shown in Figure 3 are the BeagleTouch, the BeagleJuice, and the BeagleMod. Primarily, the BeagleTouch offers a touchscreen LED display, the BeagleJuice includes pre-loaded Linux and allows the system to run on an Android operating system, and the BeagleMod offers a connection to a set of sensors such as an accelerometer, GPS, a light sensor, and a heartbeat sensor. With this product line, Liquidware used an approach

Figure 3: The BeagleTouch, BeagleJuice, and BeagleMod [14]

called architectural innovation, which means that "modules are designed purposefully to connect to each other functionally, dramatically reducing the 'barrier effort' getting each module to talk to one another" [14]. Although Liquidware's product line offers a display screen, Android capabilities, and connectivity to a range of sensors, they are limited to the few sensing and actuating capabilities included in the three modules and do not offer any space for involving other sensors.

The iProtoXi Aistin modules are compact, mobile, and implement Arduino software. The modules each hold sensors that connect the user with the Internet of Things using Aistin Bus24, which is the "next generation solution for IoT applications where low power and small size are required" [15].Shown below in Figure 4 is an example of one of these modules called the Aistin

Figure 4: Aistin Set 868 MHz [16]

Set 868 MHz, which includes an accelerometer, barometer, and humidity sensor. Although Aistin offers a range of modules, they are really more of a concluding consumer product and do not allow for the possibility of personal prototyping.

Voxel8, shown in Figure 5, is a company that provides a 3D printer kit that allows the consumer to directly incorporate electrical circuitry into physical objects. Creating three dimensional objects that integrate electronics into mechanical objects allows the consumer to take on an extraordinary range of projects. Although this is cutting edge technology, and the

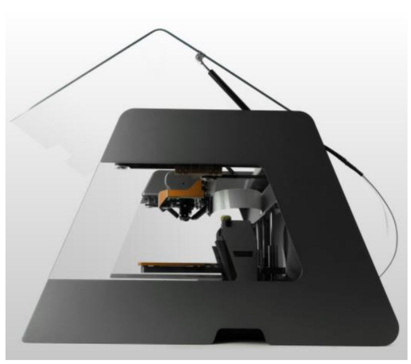


Figure 5: Voxel8 Multi-Material 3D Printer [17]

main focus of Voxel8 is prototyping, the product does not necessarily make the prototyping process easier for the average engineer or technically efficient hobbyist. It is a costly product, with a starting price in the range of nine thousand dollars, which is not realistic or affordable for the average consumer interested in prototyping.

The Tibbo Project System shown below in Figure 6 can be used to quickly move through the prototyping process to produce a wide range of presentable projects. The system "features an almost bare board powered by Texas Instruments Sitara processor, and a large area for Tibbit blocks to add features as needed, as well as an enclosure" [18]. Tibbo does a great job of

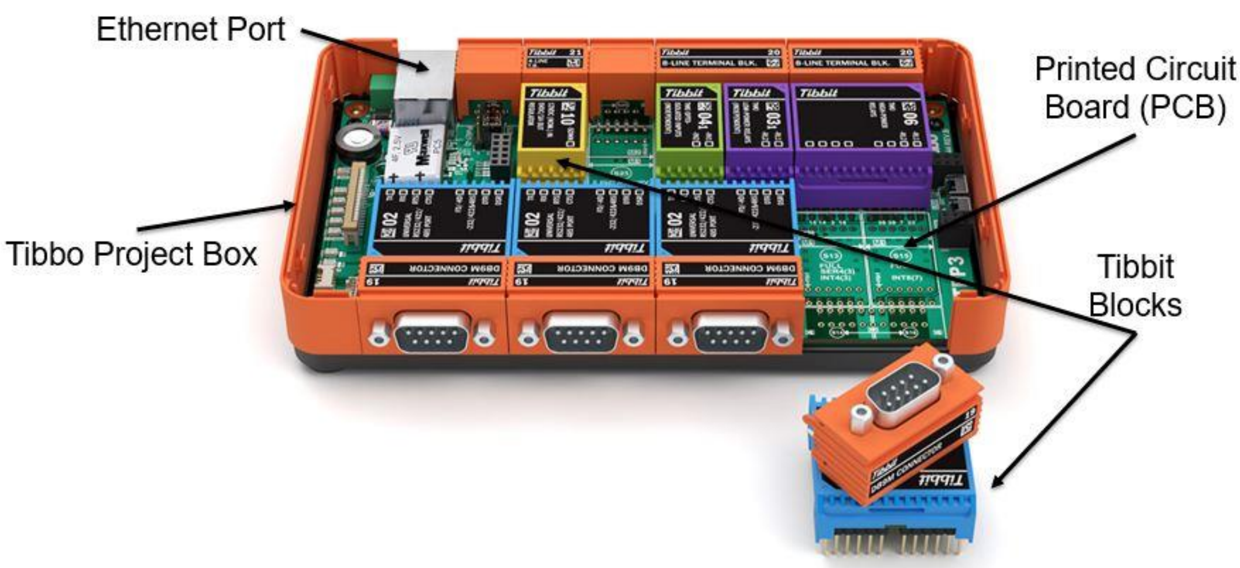


Figure 6: Tibbo Project System [19]

providing customers with the ability to choose from a wide variety of Tibbit blocks that include components such as sensors, actuators, power supplies, and USB ports. These blocks can be rigidly attached to the bare board to quickly put together a sturdy and reliable prototype that looks good enough to present. The company also offers bare boards of different sizes, gives you the option of creating a custom board before purchasing, and gives the option of enclosing the entire platform in a box for additional protection.

1.3 Deciding which Platform has the Greatest Advantages

The Tibbo Project System is the existing product that has advantages that most closely align with the platform being created in this research. The advantages of the system should be implemented in the 'Mod Duino.' As described in the previous section, the Tibbo creates reliable connections between modules and the microprocessor and provides a clean and professional final presentation, which are some of the primary goals of this research. However, one of the major disadvantages of the Tibbo Project Systems is that they run off TiOS, which not many people are familiar with. Arduino has its own open source coding program, which many more (even inexperienced) engineers have been exposed to.

The products Tibbo offers are also relatively expensive, in the range of a couple hundred USD. Tibbo also offers a wide range of modules that include an assortment of components, which would typically be a benefit. The problem is that implementing these modules can easily be confusing for a hobbyist. The modules provided by this company are so advanced that it would take a well-trained electrical engineer to understand and use them effectively. The Tibbo products may work better on the industrial level, but hobbyist and young engineers are much more likely to have an understanding of a basic microcontroller like an Arduino. Creating a system that incorporates an Arduino board would be much more useful for a beginner looking to gain experience with prototyping.

1.4 Deciding which Microcontroller to Implement

Arduino offers a range of microcontrollers that are in the thirty to fifty dollar range and are much more common in the United States when compared to the Tibbo Project System. The Arduino family of microcontrollers is also considered to be basic, even for beginners, which means individuals of all experience levels can enjoy the 'Mod Duino.' Figure 7 shows that out of

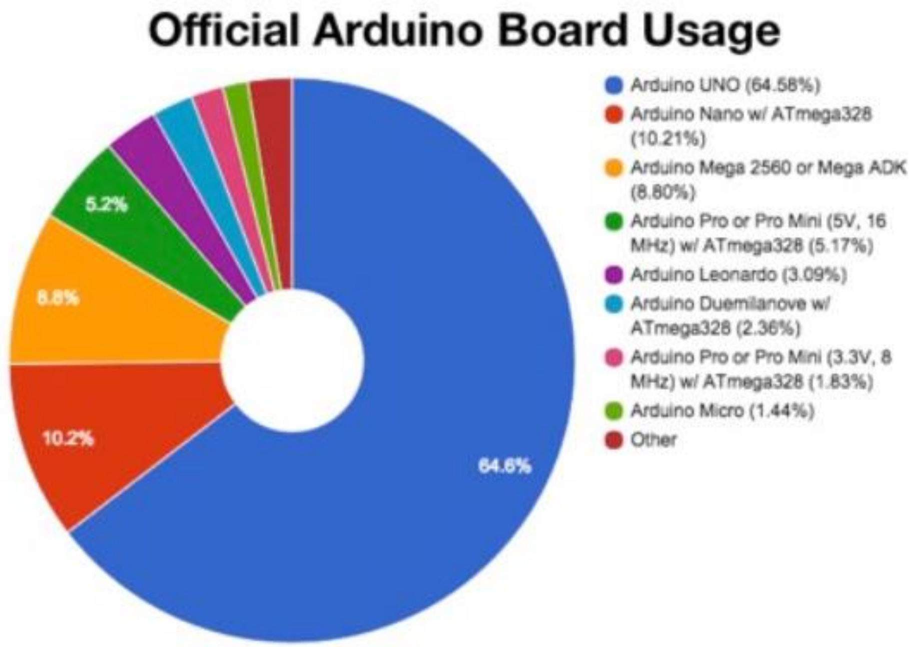


Figure 7: Arduino Board Usage [20]

the Arduino family of microcontrollers, the Uno is decidedly the most popular. Due to this fact, the ‘Mod Duino’ will be made to fit this specific microcontroller so that it can meet the needs of the largest number of people. Families of microcontrollers like Arduino and Raspberry Pi were previously considered to be developmental systems in themselves, but that was mostly due to the vast majority of tutorials and open source information available through their strong communities. The ‘Mod Duino’ will expand the microcontroller’s prototyping capabilities by adding some of the connectivity features that are so successful in the Tibbo Project System.

A wide range of prototyping projects are able to be completed with the ‘Mod Duino,’ because the user should be able to use multiple modules at one time and write unique code to control these modules. Connecting more than one module to the base platform can be beneficial, especially in larger projects that need to include an array of sensors and actuators. The main goal of the product is simplicity through organization on the hardware side of prototyping. The sleek, finished look of the modules and the easy and smooth connection between the modules and the base delivers this simplicity. This solves a common problem in the current prototyping area by making the physical connection between a microcontroller and its sensors/actuators easy, reliable, and aesthetically pleasing. All the components and wiring will neatly be enclosed in the casing of the main board.

1.5 Significance of Research

The purpose of this research is to create a system that can ensure a clean and reliable connection between an Arduino Uno microcontroller and its compatible sensors that might be used in a prototype. This is significant because often times, the wiring used in "low-end" prototyping projects become disconnected, causing them not to work properly. Ensuring these connections will allow the less experienced engineer to spend more time gaining knowledge of different types of sensors, instead of unnecessarily spending time trouble shooting their device. A rigid system would also lead to a better looking prototype, because there will not be any loose or tangled wires. This is important because an aesthetically appealing prototype can more easily be presented and explained than a messy one. The system will also make the prototype portable, which is a challenge for current prototypes that have floating breadboards and microcontrollers that are solely connected by unsoldered wires.

1.6 Outline of Thesis

The system will be created through the implementation of the engineering design process. First, this paper will define that process. Then there will be a detailed description of how this process will be applied specifically to the system being created. All of the hardware, software, and machines used to complete the prototypes made through the process will be explained. Next, the resulting prototype of the system will be described in full, along with an explanation of the tests applied to the system and a discussion of how the prototype works.

Lastly, the paper will describe what future work can be done to further increase the success of the system that has been created.

Perhaps the most important aspect of the design process is that it is iterative. For the current project, it is important to keep in mind that many of the steps in the design process will be revisited and reworked. This is a critical characteristic in any successful design, because as the research progresses and a deeper understanding of the system in achieved, the objectives and requirements should be altered and refined to ultimately create a system that better fits the overall goals of the research.

CHAPTER 2: METHODS

2.1 Engineering Design Process Overview

Any project that involves the design of an electromechanical system should follow the engineering design process. This process is visually summarized in Figure 8 below. The process

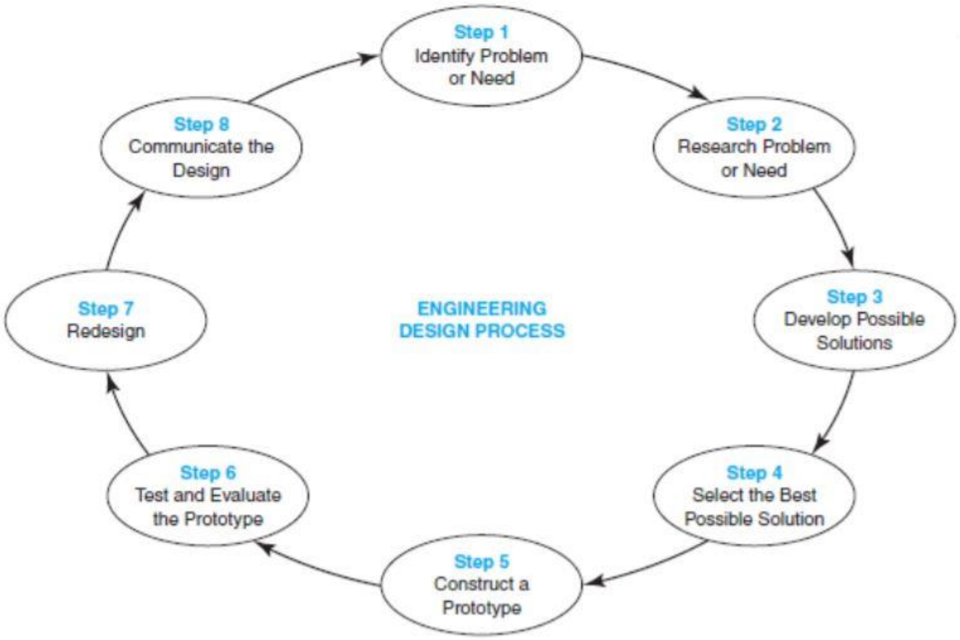


Figure 8: Engineering Design Process [21]

includes the identification of a problem, background research on the problem, brainstorming of ideas that may solve the problem, a strategy for deciding which idea should be pursued, the construction of a prototype, the testing of that prototype, often times a redesign of prototype, and finally a mode for communicating the prototype's design to others.

2.1.1 Applying the Design Process

The problem being solved and the background research of currently available platforms that apply to this problem have already been outlined and described above. This led to a description of the major goals that need to be achieved by the system, which will later be clarified through the objectives. These objectives will then be ranked through the use of a pairwise comparison chart. There will also be a list of constraints placed on the system that should not be exceeded throughout the design process.

To develop possible solutions to the problem, first the functions of the system will be outlined. Then functional requirements will be defined and scores will be assigned to these requirements. Once the functional requirements are determined, preliminary design sketches will be made to illustrate the possible solutions to the problem. Through decision matrices, the functional requirements will be applied to the preliminary designs to choose which design(s) would best meet the requirements put on the system.

Once the preliminary design(s) with the best combination of all the requirements of the system is identified, the prototyping stage will begin. The paper will define all the hardware, software, and machines used to create these prototypes. First, a more detailed drawing of the preliminary design will be made. Then, computer aided design software will be used to create three dimensional parts, and then assemblies, of the drawings. These three dimensional parts can then be machined into physical parts with the help of a 3D printer.

Although Figure 8 uses arrows to show this process is cyclical, which is true, it is important to note that not all of the steps necessarily need to be reworked when one step in the process is reevaluated. It may be more appropriate to add arrows to the figure that connect, perhaps, the “Construction of a Prototype” oval back to the “Select the Best Possible Solution” oval. This is because the steps six through three may not need to be completed in order to get back to the “Select the Best Possible Solution” oval. For example, the requirements or objectives may need to be to reviewed and redefined after a prototype or two have been made, but this can be done without testing, evaluating, or doing additional background research. In this way, the process is very much iterative.

2.2 Initial Objectives

The objectives are used to more clearly define the major goals of the system. The list of objectives may actually seem like broad topics, which is why a metric is placed on each of the objectives. All of these objectives and metrics are listed below in Table 3. The initial objectives

Table 3: Objectives and Metrics of the Prototype

Objective	Metric
Final Cost	The average cost of the main Arduino platform and module connections is below $50.
Appearance	The final product is aesthetically pleasing so that it can be presented to the end user/business professionals.
Size	The product is smaller than 6in x 6in x 6in.
Weight	The product weighs less than 2 lbs.
Ease of use	The product is easily understood and operated mainly through intuition (with little guidance) if user is familiar with Arduino programming.
Reliability	The system should not fail when assembled properly.
Strength/Durability	The materials used have a high enough strength to not break during typical use.
Maintainability	The system can be repaired or replaced within 30 days of request.
Portability	The embedded prototyping system can be moved from place to place without any connections being lost.
Life Cycle	The product should have a life cycle no shorter than ten years.
Safety	The product should be safe to use under normal operating conditions. No hazardous materials should be used when creating the product.

are to have a system that is affordable, aesthetically appealing, small, lite weight, easy to use, reliable, durable, easily maintained, portable, long lasting, and safe. The pairwise comparison chart (PCC) in Table 4 was used to rank the objectives defined for this system in order of importance, by directly comparing each objective to all of the others. The results of this PCC

Table 4: Objectives Pairwise Comparison Chart

Objectives	Final Cost	Appearance	Size	Weight	Ease of use	Reliability	Strength	Maintainabili.	Portability	Life Cycle	Safety	Total
Final Cost	X	1	1	1	0	1	1	1	1	1	1	9
Appearance	0	X	1	1	0	0	1	1	1	1	1	7
Size	0	0	X	1	0	0	0	1	0	1	0	3
Weight	0	0	0	X	0	0	0	0	0	0	0	0
Ease of use	1	1	1	1	X	1	1	1	1	1	1	10
Reliability	0	1	1	1	0	X	1	1	0	1	0	6
Strength	0	0	1	1	0	0	X	1	0	1	0	4
Maintainability	0	0	0	1	0	0	0	X	0	0	0	1
Portability	0	0	1	1	0	1	1	1	X	1	1	7
Life Cycle	0	0	0	1	0	0	0	1	0	X	0	2
Safety	0	0	1	1	0	1	1	1	0	1	X	6

are that the ease of use is the most important objective. This is because the system is meant to be used by someone with little to no prior electromechanical prototyping experience. A low final cost of under 50 USD ranks second highest for a similar reason. Beginners in this space are generally more willing to purchase a product that is below 50 USD to get exposure to the prototyping process, whereas someone with more experience that needs prototyping equipment for production or industrial applications may want to spend more money. Appearance, reliability, and portability of the system are also near the top of the list because they are the characteristics that will help meet the overall goals of the research.

2.3 Initial Constraints

Constraints are defined to place limitations on a systems design. The constraints placed on this system are listed in Table 5. These limitations are implemented so that the cost of the

Table 5: Constraints on the Prototype

The average cost of the product should not exceed $150.
No machines should be required to use the product.
Use of the product should not require any additional hardware on the side of the buyer.
Use of the product should not require any software other than the open source Arduino software.
The material(s) used to create the module enclosures and the main platform should not conduct electricity.

product does not get too high, there is no additional software or hardware (other than the Arduino Uno microcontroller) that needs to be purchased to operate the product effectively (which also helps to minimize cost), and the materials used do not cause harm to the user.

2.4 Initial Functions

The functions of a system define the tasks that the software and/or hardware need to perform. These functions are paired with functional requirements, which more specifically define the task that needs to be completed. The functions and paired requirements for this system are described in detail directly in Table 6 on the next page.

Table 6: Functions and Requirements for Prototype

	Functions	Requirements
1	Create modules that reliably encase sensors and actuators.	The modules are rigid enough to pick up and install/mount onto platform while retaining their shape.
2	The platform should be capable of holding each module type.	The modules are made in standard sizes in a fashion that allows multiple smaller modules to fit in the same exact volume as a larger module.
3	Implement sensors and actuators that are compatible with the required software.	The sensors and actuators must cooperate with the input/output capabilities of the ATmega328P microcontroller (operate at 5 volts and 20 mA; supports PWM, I2C, and SPI communication).
4	Create modules that can be easily connected to the platform.	The modules can be connected to the main platform with normal hand force and no additional tooling.
5	Create modules that can be easy disconnected to the platform.	The modules can be disconnected to the main platform with normal hand force and no additional tooling.
6	Create modules that can be easy reconnected to the platform.	The modules can be reconnected to the main platform with normal hand force and no additional tooling.
7	Create a reliable connection between the modules and platform.	The entire device can be rotated, moved, and dropped from a height of one foot without any parts becoming disconnected.
8	The platform should be open to housing multiple modules at once.	The platform can support at least five modules at one time.
9	Connect to power and ground.	Each module needs to be able to be connected to a power source and ground on the Arduino platform without lose wiring.
10	The device needs to include a microcontroller that is compatible with concurrent software.	The platform needs to include all major parts of the Arduino Uno (not limited to the microcontroller).
11	Data must be transmitted between the device and a PC (equipped with Arduino software).	A USB plug must be incorporated within the device.
12	The new systems will run using existing software.	The new system will use Arduino software for coding.
13	The new system should come with Arduino tutorials.	At least five tutorials/examples should be made for new users, including necessary Arduino code and hardware connections.
14	Any enclosure should not interfere with any connections.	Any enclosure should provide clearance or openings for all necessary connections.
	Optional:	
15	There should be an enclosure for the system.	The microcontroller and all other hardware should be globally encased.
16	There should be a form of communication between the user and platform.	The device features a reset button and an LED that lights up on the platform when a code is successfully uploaded.

These functional requirements will be used to determine which preliminary design will be pursued. They will also be used to test the effectiveness of the final prototype of the system. This is why possible scores need to be assigned to each requirement. As shown in Table 7, each function is given possible scores ranging from zero to two. A score of zero indicates that the functional requirement was barely met, while a score of two means that the functional requirement was met to its fullest capacity.

Table 7: Function Scoring Details

	Functions	Score (0-2)
1	Create modules that reliably encase sensors and actuators.	0: The modules cannot retain their shape when picked up. 1: The modules retain their shape when being held, but deform when force is applied to install them onto the platform. 2: The modules retain their shape when being handled and installed/removed from the platform.
2	The platform should be capable of holding each module type.	0: Each module is encased in a module with a unique (nonstandard) size. 1: Every module is the same size. 2: There are two standard sized modules so that space can be optimized.
3	Implement sensors and actuators that are compatible with the required software.	0: The sensors and actuators used to make the modules are not compatible with the ATmega328P. 1: Some of the components used to make the modules are compatible with the ATmega328P. 2: Every component used to make the modules is compatible with the ATmega328P.
4	Create modules that can be easily connected to the platform.	0: The modules cannot be connected to the main platform. 1: The modules can be connected to the main platform with additional tooling. 2: The modules can be connected to the main platform with normal hand force.
5	Create modules that can be easy disconnected to the platform.	0: The modules cannot be disconnected to the main platform. 1: The modules can be disconnected to the main platform with additional tooling. 2: The modules can be disconnected to the main platform with normal hand force.
6	Create modules that can be easy reconnected to the platform.	0: The modules cannot be reconnected to the main platform. 1: The modules can be reconnected to the main platform with additional tooling. 2: The modules can be reconnected to the main platform with normal hand force.
7	Create a reliable connection between the modules and platform.	0: The modules become disconnected from the platform when the device is moved or rotated. 1: The modules stay attached to the platform when the device moves horizontally, but disconnect when the device it rotated or dropped. 2: The modules stay attached to the platform when the device is moved, rotated, and dropped.

Continued

Table 7 Continued

8	The platform should be open to housing multiple modules at once.	0: The platform can support one module at a time. 1: The platform can support two through four modules at a time. 2: The platform can support at least five modules at a time.
9	Connect to power and ground.	0: None of the modules can be connected to power or ground. 1: Each module can be connected to power and ground through wires (that could potentially disconnect). 2: Each module can be connected to power and ground in a discrete fashion that does not involve any lose wiring.
10	The device needs to include a microcontroller that is compatible with concurrent software.	0: The hardware used for the platform does not work with Arduino software. 1: The ATmega328P microcontroller is used for the platform, but some of the necessary Arduino Uno components are not used. 2. All necessary components of the Arduino Uno are included in the main platform of the device.
11	Data must be transmitted between the device and a PC (equipped with Arduino software).	0: The device cannot communicate with a PC. 1: The device can communicate with a PC through an Ethernet connection. 2: The device includes a USB port that allows it to communicate with a PC.
12	The new systems will run using existing software.	0: No code can successfully be uploaded to the device. 1: Code from an open source software that is not Arduino can be successfully uploaded to the device. 2: Arduino software can be successfully uploaded to the device.
13	The new system should come with Arduino tutorials.	0: No tutorials are included with the product. 1: One-four tutorials are included with the product. 2: Five or more tutorials are included with the product.
14	Any enclosure should not interfere with any connections.	0: The enclosure prohibits all necessary connections. 1: The enclosure provides enough of an opening for all connections to be made, if two or less modules are in use. 2: The enclosure provides enough of an opening for all necessary connections to be made, independent of how many modules are in use.
Optional:		
15	There should be an enclosure for the system.	0: All components of the platform are exposed. 1: Most of the platforms components are hidden within a case. 2: All of the platforms components are hidden within a case.
16	There should be a form of communication between the user and platform.	0: There is no form of communication between the device and user. 1: The device features a reset button or an LED indicator. 2: The device features both a reset button and an LED indicator.

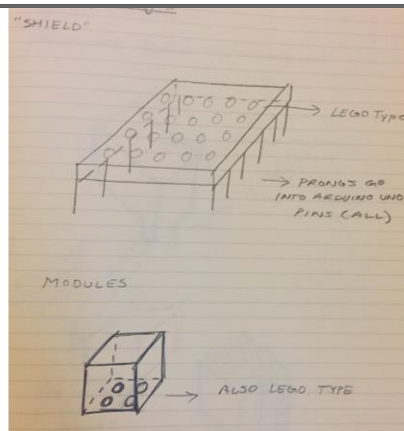
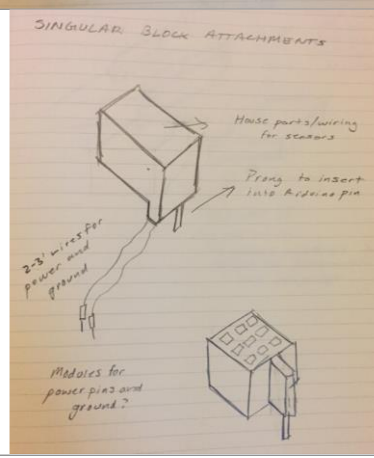
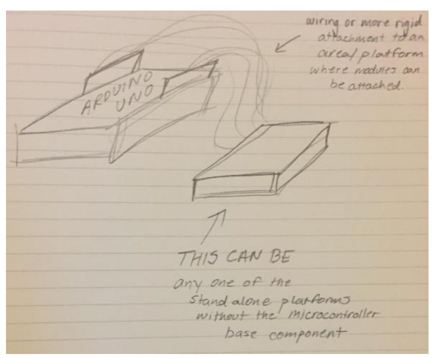
2.5 Preliminary Design Sketches

The initial plan was to separate the designs into two different categories. The first category is comprised of systems that could be retrofitted onto an existing Arduino Uno microcontroller. The second set of designs were meant to include the necessary components of the microcontroller so that the device could stand (work) on its own. Almost all of the preliminary designs include a module. The module is meant to hold an individual sensor. The difference in the designs all come from how the modules would be attached to either the pre-existing Arduino Uno microcontroller or the embedded board (depending on the category).

2.5.1 Retrofit Existing Arduino Uno Platform Sketches

Listed on the following page in Table 8 are the names and sketches for these preliminary designs. There is also a short description of each design, which explain how they would be attached to the Arduino microcontroller.

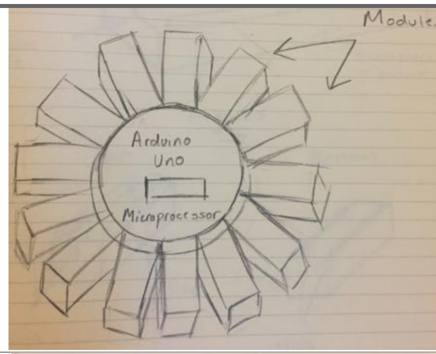
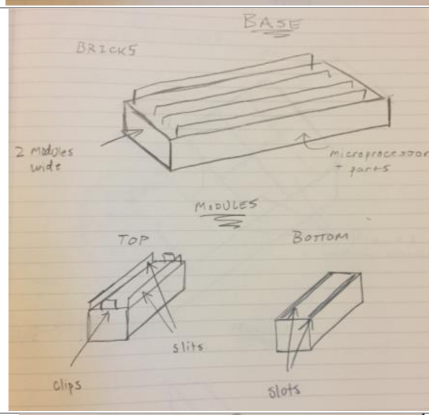
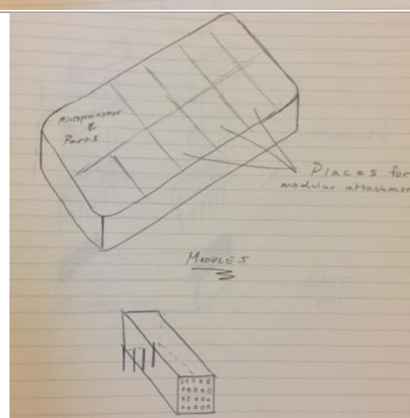
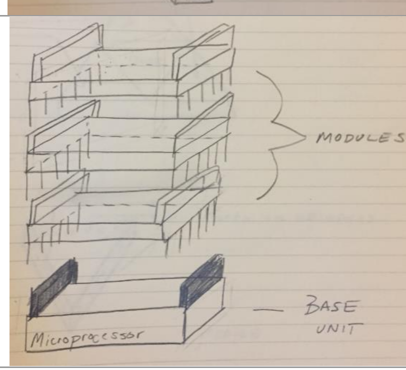
Table 8: Preliminary Designs that Retrofit Existing Arduino Uno Microcontroller

<table>
<tr><th>Name</th><th>Sketch</th><th>Description</th></tr>
<tr><td>Shield</td><td>
</td><td>This design would use header pins to connect a board to the Arduino. The top of the board would have extruded holes that the modules would snap into.</td></tr>
<tr><td>Singular Blocks</td><td>
</td><td>This design only involves individual modules. The modules would have a header pin that would connect the sensor to the appropriate I/O pin. Two additional wires would come from the module to connect to ground and the powere supply.</td></tr>
<tr><td>Platform Attachment</td><td>
</td><td>This design is similar to the Shield, but the attachment would not have extruded holes for modules to be attached. Instead, it would be a bare broadboard. The advantage from current prototyping would be that the breadboard has pins on the underside to connect direction onto the top of the microcontroller, instead of being attached via wire.</td></tr>
</table>

2.5.2 Stand Alone Platform Sketches

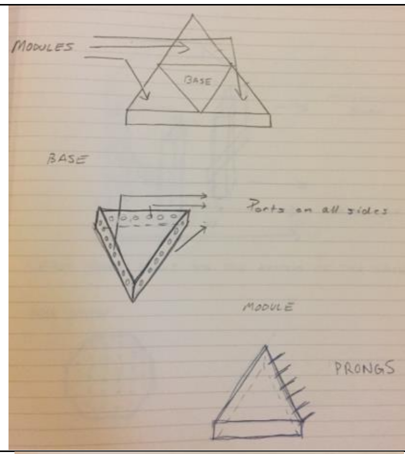
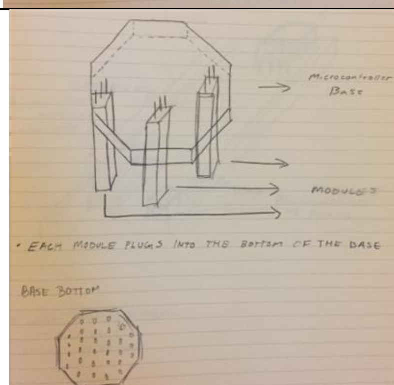
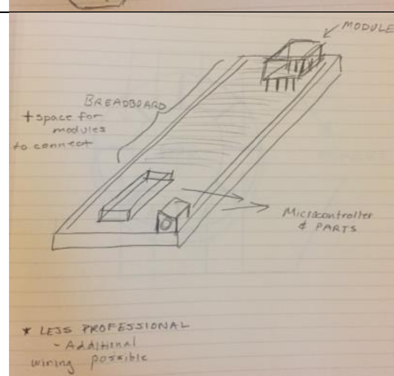
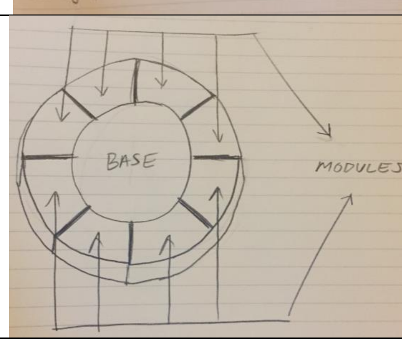
Listed on the following page in Table 9 are the names and sketches for these preliminary designs. Again, there is also a short description of each design, which explains how they would be attached to the Arduino microcontroller.

Table 9: Preliminary Designs for Stand Alone Platform

Name	Sketch	Description
Flower		This design would have the microcontroller components contained in a circular housing. The modules with be attached along the outer diameter of the circle, similar to a USB drive.
Bricks		This design would house the microcontroller parts in a rectangular box. The box would have linear slits where the modules would be able to slide into and connect.
Tibbo Style		This design would also house the microcontroller components in a rectangular box, but the modules would connect on the same level as the microcontroller components.
Tower		This design holds all the microcontroller components in a square shape box (on the bottom of the picture to the left). The modules would have female pins on the top and male header pins on the bottom that would allow them to be stacked on top of one another.

Continued

Table 9 Continued

Triangular		This design holds the microcontroller components in a triangular shaped housing compartment. The modules (up to three) would attach to any side of the housing.
Slim Plugins		This design holds the microcontroller components in an octagon shaped housing. A number of modules would be able to be plugged into the housing from the bottom.
Basic Bread Board		This design includes a breadboard that holds all of the components of the microcontroller. The modules would each have pins that could be used to mount them directly onto the breadboard.
Circular		This design would also hold the microcontroller parts in a circular housing (like the flower). The difference would come from the modules, which would form around the diameter of the circle and snap together for additional strength.

2.6 Initial Decision Matrices

A decision matrix was used to analytically determine which preliminary design best meets the functional requirements. The function numbers mirror those of Table 6 from Section 2.4. Each of the requirements was given a weight between one and four (four carrying the most weight) based on that function's importance in relation to the others. Then, the score that each design gets for a particular function is multiplied by that functions weight. The weighted scores are summed up for all of the requirements as a total score at the bottom of the matrix. The decision matrix used for the designs that retrofit an existing Arduino Uno microcontroller is shown here in Table 10. The result of the matrix is that the Shield design should be pursued.

Table 10: Decision Matrix for Retrofitted Platform

		Shield		**Singular Blocks**		**Platform Attachment**	
Functions	**Weight (1-4)**	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score
1	3	2	6	1	3	2	6
2	2	2	4	1	2	2	4
3	4	2	8	2	8	2	8
4	3	1	3	2	6	2	6
5	3	1	3	2	6	2	6
6	3	1	3	2	6	2	6
7	4	2	8	1	4	1	4
8	2	2	4	1	2	2	4
9	4	2	8	1	4	1	4
11	4	2	8	1	4	2	8
12	4	2	8	2	8	2	8
13	2	2	4	2	4	2	4
14	3	2	6	1	3	1	3
Optional:							
15	1	2	2	1	1	1	1
16	1	2	2	2	2	2	2
Total			**77**		63		74

A similar decision matrix was applied to the stand alone preliminary designs, which included the necessary components of the Arduino Uno microcontroller inside a main housing unit. This table was split into two parts, and can be found in Appendix A. The results of this matrix was to pursue the Tibbo Style design. A clearer depiction of the two preliminary designs that were chosen from the results of the decision matrices are shown below in Figure 9.

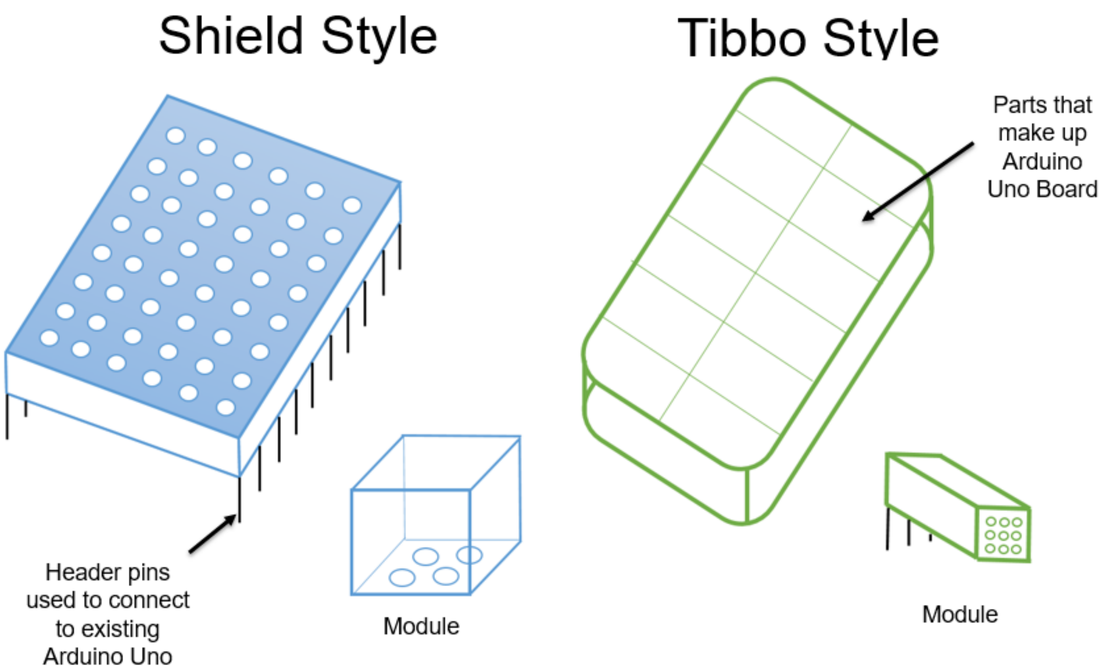


Figure 9: Preliminary Designs chosen by the Decision Matrices

2.7 Equipment Used in Prototyping

In the following three subsections, all of the equipment used throughout the prototyping stage of the engineering design process will be reported and described. These subsections include software equipment, electronic components, and machines, which were all critical for creating the prototypes for this system.

2.7.1 Software Equipment Used

Before creating a physical prototype, the design drawings need to be recreated in a computer aided design (CAD) program. The general CAD process is outlined here in Figure 10.

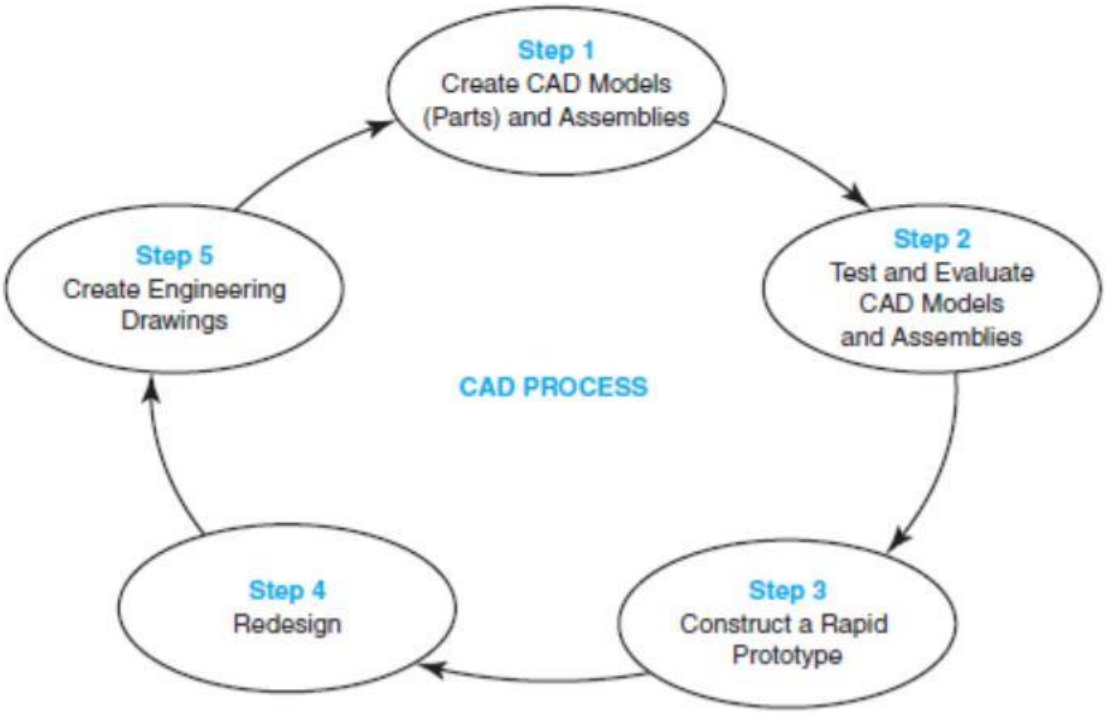


Figure 10: Computer Aided Design Process [22]

The three dimensional (3D) CAD program used in this research is SolidWorks 2016 x64 Edition. Step 2 in Figure 10 was not implemented in this research, all the testing and evaluation was done on the physical prototype, not within SolidWorks simulations. However, all of the prototypes were created using SolidWorks, before they were 3D printed. SolidWorks was also used to create the drawings for the parts used in the final prototype.

2.7.2 Electronic Components Used

The most important electronic device used in this research is the Arduino Uno. The technical specifications [4] given by the manufacturer for this device can be found in Appendix B. The detailed datasheet for the ATmega328P, the microcontroller used on this board, can be found from the link provided as a reference for the board's specifications. Here, in Figure 11, the major components of the board are labeled. The most important components concerning this

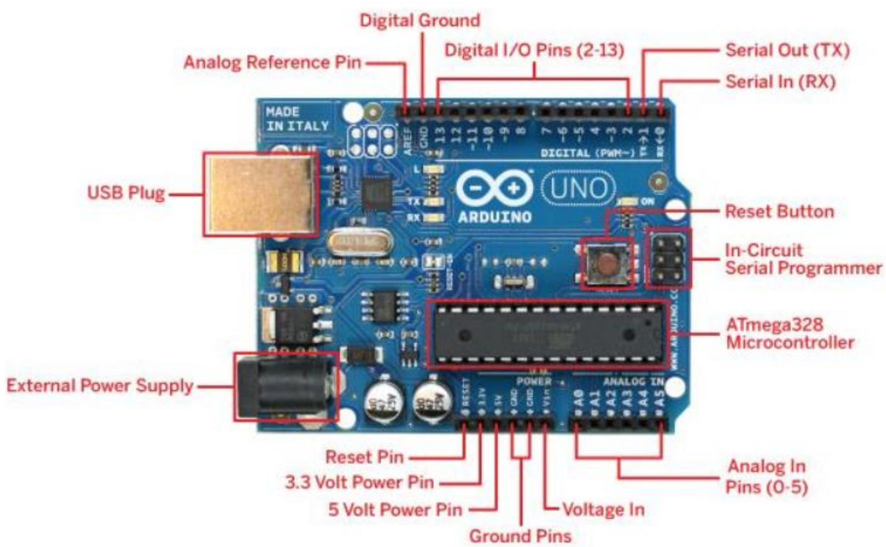


Figure 11: Arduino Uno Major Components Labeled [23]

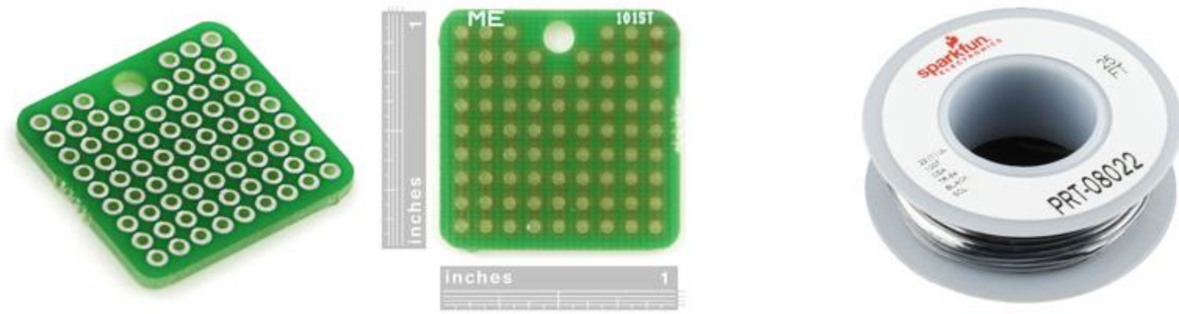
research project are the microcontroller, the USB plug, and the female header pins which include (but are not limited to) the power, ground, analog in, and digital in and out pins. The microcontroller stores and executes any functioning program that can be uploaded to board through the USB plug. The ground connection, one of the voltage in connections, and one analog in or digital in connection will always be used, regardless of the sensor being connected to the board.

For proof of concept, sensors were inserted into the modules and connected to the Arduino Uno board, which will be discussed later in more detail. The four sensors used for this purpose were a microphone amplifier, an infrared receiver, a temperature sensor, and an LED (Figure 12). The first page of the data sheets for these electronic components can be found in Appendix C. These sensors were chosen to be implemented in the proof of concept because

[24] [25] [26] [27]

Figure 12: Microphone Amplifier, Infrared Sensor, Temperature Sensor, and LED

they are relatively simple to connect to the Arduino board. Each of these sensors only has three male header pins, one leading to ground, another leading to power, and a third leading to an analog or digital pin. The small breadboard and hook up wire shown in Figure 13 were also used in the construction of the prototypes. The one square inch ProtoBoards are single sided perf



[28] [29]

Figure 13: Square 1 Inch ProtoBoard (Left) and 22 American Wire Gauge (AWG) Hook-up Wire (Right)

boards with “standard 0.1” spacing with a 4-40 (3mm) hole for mounting. Every three holes are connected with a 20mil trace” [28]. These boards were mounted inside of the modules so that the sensors could be electronically attached to them, while the 22 AWG wire was used to connect the sensors inside the modules to the Arduino Uno board.

The final system will also include a custom breadboard. This breadboard will be used as space to place any electronic components needed for a prototyping project that are not already included within the modules. Ideally, a custom breadboard would be created using the configuration of pins in Figure 14. Unfortunately, a platform (or software) for creating custom breadboards could not be found. Instead, Fritzing, an open source software used to create and manufacture custom electronic designs, was used to create this printed circuit board (PCB). The board is created strictly out of female header boards, to simulate a breadboard. In the configuration of Figure 14, the top three rows are each connected electrically in series and would be used as ground or voltage buses. The rest of the vertical buses (with either three or four header pins) can be used for things like light emitting diodes (LEDs), resistors, and integrated circuits (ICs).

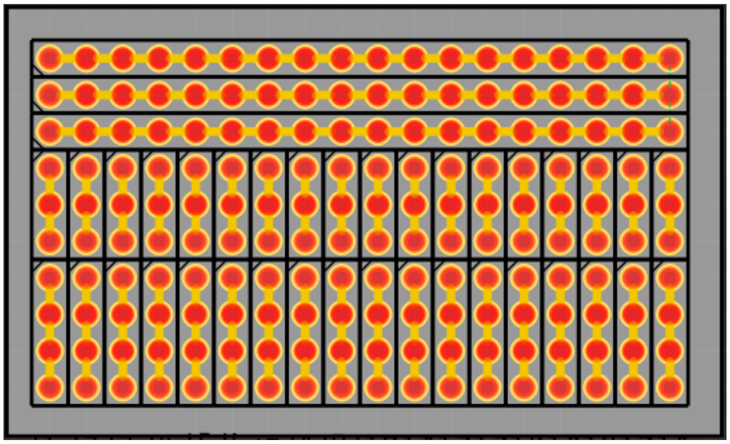


Figure 14: Custom PCB of Female Headers from Fritzing

The problem with this approach is that PCBs, unlike traditional breadboard, require wires to be soldered to the board for the circuit to connect properly. In other words, PCBs do not have sockets so wires cannot easily be removed and moved around. Without a currently available platform for creating a custom breadboard, the Mini Modular Breadboard from Sparkfun will be used for this project (Figure 15). These breadboards are 47x35x10 mm (~1.85x1.38x.40 inches),

Figure 15: Mini Modular Breadboard [30]

which makes them ideal for the system being created in this project. These mini breadboards are separated into two halves. Each half has seventeen rows that are made of buses that contain five sockets. The board can be made to fit into an even smaller space, by grinding off the two flanges on the sides of the board. These flanges are used to connect more than one Mini Modular Breadboard together, but are not needed for this project, because only one of the boards will be used. The boards also come with a peel and stick adhesive backing which can be used to secure the board to the platform that will be created. This Mini Modular Breadboard will be able to accommodate the same components (LEDS, resistors, ICs) as the custom PCB in Figure 14.

2.7.3 Machines Used

Two machines were used to create the physical prototypes of the system. The first machine used was the Mojo 3D printer produced by Stratasys (Figure 16 on the next page). The Mojo is a fused deposition modeling (FDM) 3D printer, which means that it "is accomplished by extruding thin layers of molten thermoplastic layer by layer until a part is produced" [31].The layer thickness of this 3D printer is 0.17 mm (0.007 in). Due to the nature of FDM, special considerations need to go into designing any part that will be printed. These considerations apply to shrinkage, warping, holes, columns, pins, wall thickness, threads, undercuts, fillets, draft angles, size and orientation, assemblies, sectioning parts, living hinges, fastening hardware, bosses and ribs, and text [31]. Due to the basic geometry of the main platform and modules (described later in more detail) that will be created in this project, the fact that the original designs already had features that fit the recommendations for these considerations, and the support material of the Mojo, many of these aspects did not need to be considered. The aspects that were taken into account for this project were shrinkage and enlargement of features due to thermal expansion and contraction that goes on during FDM. To account for this, a tolerance was added to the platform so that the modules would fit within the width of the main platform.

Figure 16: Mojo 3D Printer [32]

After the models were created in SolidWorks, they were converted into a stereolithography (STL) file format. This file "format describes only the surface geometry of a three-dimensional object… and today is widely used for rapid prototyping, 3D printing, and computer-aided manufacturing" [33]. The STL files are opened in the 'Mojo Print Wizard' program, which calculates the layers and path that the printer needs to follow to create the part.

Once the layers and path are determined, the parts are printed using the 'Mojo Control Panel.' The printer uses two different materials to construct the parts. The piece is made from ABSplus model material (P430) and the material used to combat gravity to maintain the desired geometry of the modeled piece is Soluble Support Material (P400-SR). To remove the support material after the pieces have been printed, they were placed in the Ultra 2000 cleaner made by UltraSonic LLC (Figure 17).

Figure 17: Ultra 2000 - 5.2 Gallon from UltraSonic LLC [34]

The Ultra 2000 is filled with Soluble Concentrate and water. Typically, three wash cycles of thirty minutes were used to clean each of the pieces. Three cycles were needed to ensure the internal temperature of the machine reached sixty degrees Celsius. An Exacto knife and filing tools were then used to eliminate any unwanted material from the pieces that was not removed in the ultrasonic washer. The ABSplus model material, soluble support material, and soluble concentrate used in this project were all purchased through Advanced Technologies Consultants, Inc. Example order forms for these materials, along with the specifications for each of the machines, can be found in Appendix D.

2.8 Prototype Iterations 1 and 2

After choosing the Tibbo and Shield Style designs from the results of the decision matrices, prototypes of these systems needed to be made. It was decided that the Tibbo Style prototype was to be created first, because a matrix score of 90 (Shield Style received a score of 77) meant that it might better fit the functional requirements of the system. This section will detail the attempts that were made to create this prototype, and the challenges that were encountered while doing so. The two major portions of the prototype are the base platform and the modules.

Before creating the modules, the electronic sensors needed to be soldered to the one square inch ProtoBoards. Below, in Figure 18, is the setup used to solder these parts onto the

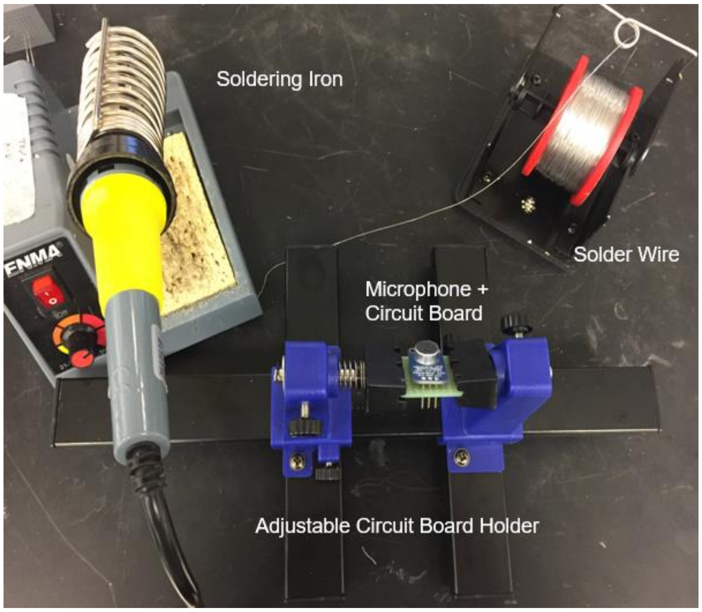


Figure 18: Soldering Setup

boards. This typical setup includes an adjustable circuit board holder, the soldering iron and wire, and the two components that need to be permanently joined together. The resulting configurations for each of the four sensors can be seen next in Figure 19.

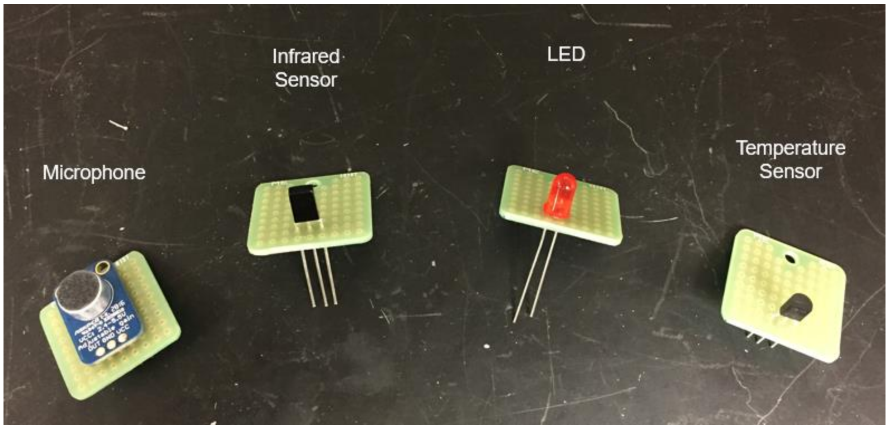


Figure 19: Sensors Soldered to Individual ProtoBoards

Once the sensors were securely attached to the boards, a 3D model of the modules was created in SolidWorks (Figure 20). The walls of the modules were made to be 0.1225 inches, for stability. The inner pocket of the modules were dimensioned to the boards, including the corner radii (0.078 inches), with an additional five thousands of an inch tolerance added in all directions. Extruded holes were also added to match the position and radius of the holes in the

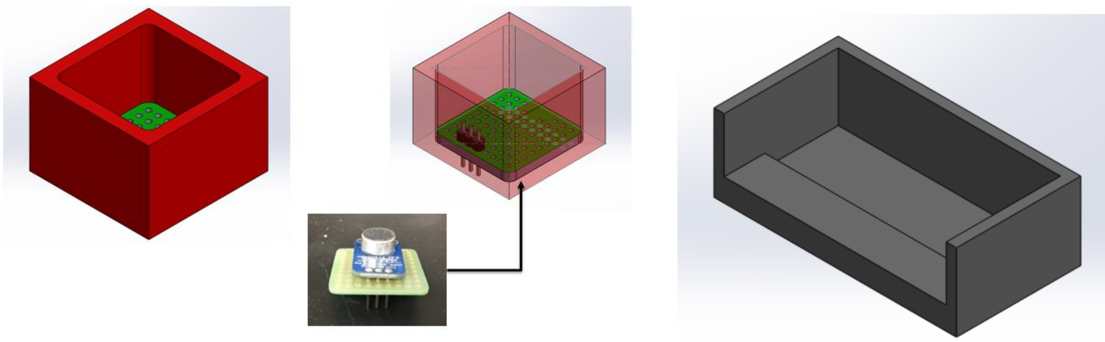


Figure 20: First Iteration SolidWorks Models: Module showing how the ProtoBoards would be placed (Left) and Main Platform (Right)

ProtoBoards. This feature was added so that the leads on the sensors would be able to be inserted through the bottom of the modules. The walls of the main platform were made to be 0.25 inches all around, and provides a wide enough front opening for four modules to be attached. The depth of the main housing is great enough to fit a larger breadboard in the rear as well as the four modules on the closer side. Figure 21 shows the assembly of the main platform, modules, ProtoBoards, larger breadboard, and male header pins. The larger breadboard is meant to give space for the components needed to create your own Arduino to be placed. The larger

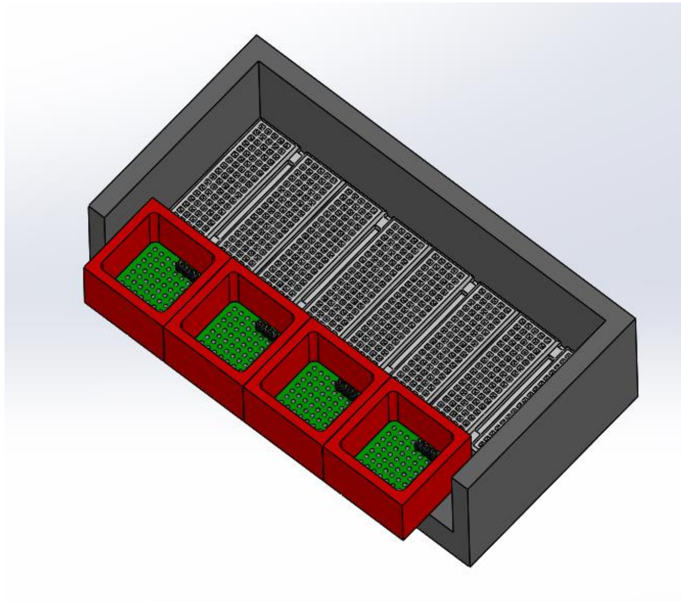


Figure 21: First Iteration SolidWorks Assembly

breadboard is also used to connect the modules to the mock Arduino board. This assembly does not include the sensors or the components needed to make the Arduino. After the model was completed, the Mojo 3D printer was used to create the physical part (Figure 22). After

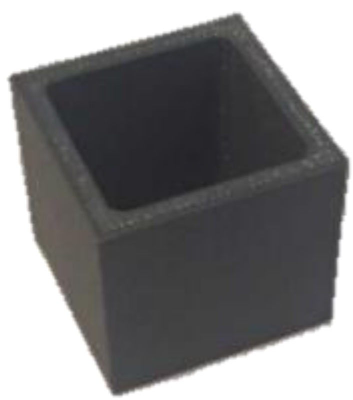


Figure 22: First Iteration of 3D Printed Module

visually evaluating the module, it was decided that the solid walls of the inner pocket may cause the sensors to overheat. To combat this, a second iteration of the modules were created. Although there are two different parts (shown below in Figure 23), it was decided to consider them to be in

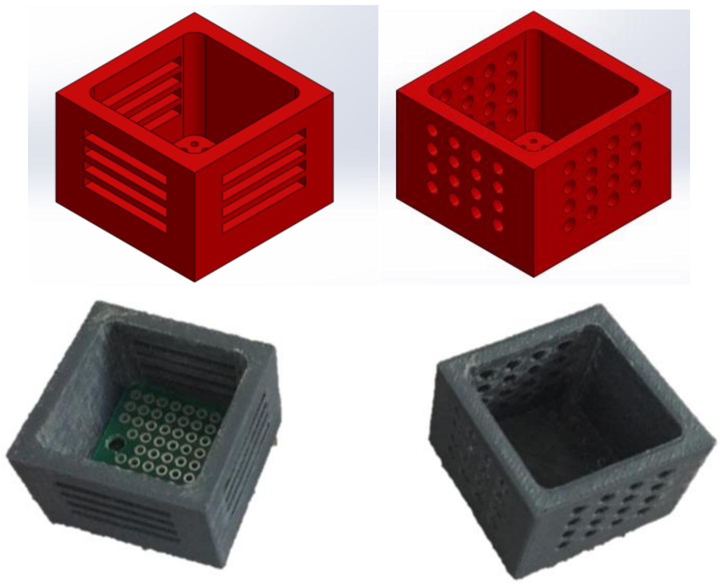


Figure 23: Iteration 2 SolidWorks Models (Top) and Physical Prototypes (Bottom) of Module with Slots (Left) and Holes (Right)

the same iteration because they shared a common goal. Both of the updated modules increased air circulation inside the inner pocket, one through the addition of slots, the other through holes. The main platform did not need to be altered from the one used in iteration 1.

The next portion of the prototype to be made was the mock Arduino Uno board. A tutorial provided on Arduino's website [35] was used to attempt the recreation (Figure 24).

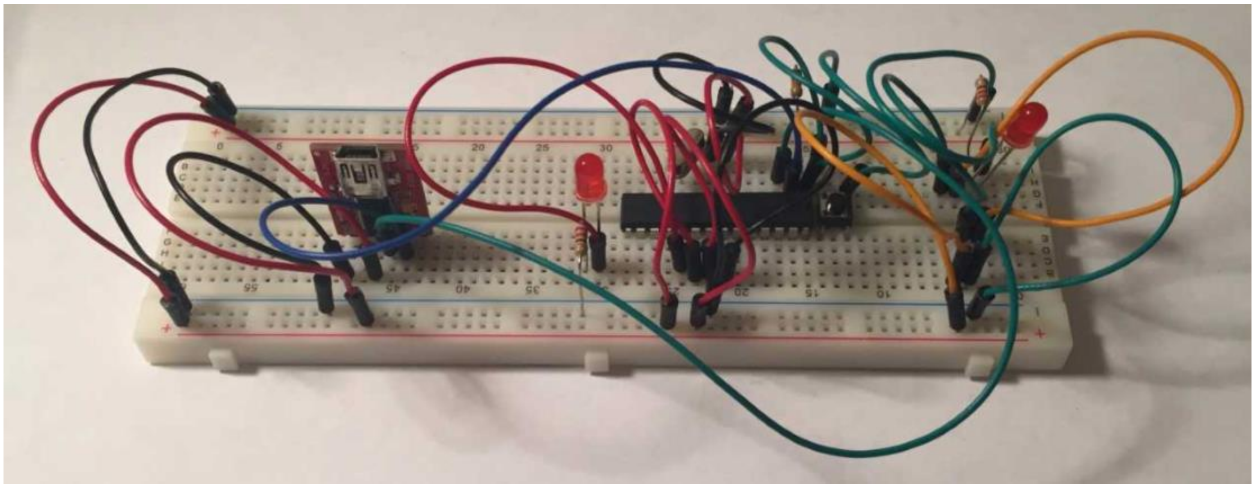


Figure 24: Attempt of the Mock Arduino Uno Board

Unfortunately, the mock Arduino board was never able to be completed successfully. This was mostly due to a lack of understanding of the FTDI Basic Breakout board and Pocket AVR Programmer. Additionally, even after the second iteration of the modules was completed, there was one major flaw. The header pins, or leads, of the modules were not long enough to be inserted through the Protoboard and bottom of the module and still have enough length to be securely mounted into the larger breadboard that was secured in the main platform.

2.8.1 Change in Strategy

These problems, along with time constraints placed on the research project, led to the decision to make one prototype instead of two (Tibbo style and Shield style). This new design would incorporate the look of the Tibbo style design, with a main platform accompanied by smaller modules that would fit inside it. However, instead of creating a mock Arduino board, the main platform would provide space for an actual Arduino Uno to be mounted securely. This decision was made because of the fact that many Arduino Uno's are already in use, which make this a better option for existing prototypers. The new design would also include the main "LEGO" attachments of the Shield style design. This change in strategy allowed time for a good, functional prototype to be created in the time allowed.

2.9 Reworking Objectives, Constraints, and Functions

Apart from the reasons described in the previous section, these specifications have also changed because the overall goals of the research project have been revised. The idea that it would be difficult to mass produce and sell the 'Mod Duino' due to the high cost of 3D printing, which is typically used for prototyping and not mass production, caused this shift. Instead of mass production for consumer purchase, the system will be made open source by providing the downloadable STL files on a public website. The change in strategy affected how the objectives, constraints, and functions of the prototype were altered. These reworked design specifications, along with a description of how they have been changed, are laid out in this section.

Table 11: Reworked Objectives and Metrics of the Prototype

Objective	Metric
Final Cost	The STL files to print the main platform and modules should be free to the public.
*** Appearance**	The final product is aesthetically pleasing so that it can be presented to the end user/business professionals.
Size	The product is smaller than 5in x 4in x 3in.
*** Weight**	The product weighs less than 2 lbs.
*** Ease of use**	The product is easily understood and operated mainly through intuition (with little guidance) if user is familiar with Arduino programming.
*** Reliability**	The system should not fail when assembled properly.
*** Strength/Durability**	The materials used have a high enough strength to not break during typical use.
*** Portability**	The embedded prototyping system can be moved from place to place without any connections being lost.
Life Cycle	The product should have a life cycle no shorter than five years.
*** Safety**	The product should be safe to use under normal operating conditions. No hazardous materials should be used when creating the product.

Above, Table 11 details the new design objectives and metrics. Any row marked with an asterisk (*) has not been modified from the original design specification. The original objective about maintainability was retracted because now that this system would be made open source, there is no concern for having to replace a damaged product. The size objective was modified to more closely align with the prototypes that have already been made. The life cycle objective was also slightly modified by changing the minimum life cycle of five years, which is more realistic for plastic 3D printed parts. The most important change in the objectives is perhaps to the final cost, which has been reduced to zero USD due to the open source STL files. Keep in mind that this final cost only includes the two major proponents (main platform and modules) of the system being created through this research.

The newly defined constraints are listed in Table 12. Again, the rows marked (*) have not been changed. The maximum cost for a consumer to implement this system for prototyping remained at 150 USD. However, now this cost would go towards an Arduino Uno microcontroller, whatever costs would go into actually 3D printing the main platform and modules, and some low cost hardware (like sensors, ProtoBoards, electronic wire, screws). A

Table 12: Reworked Constraints on the Prototype

Maximum use of the system should not require more than $150 to the user (this would include an Arduino Uno microcontroller, 3D printer material, and some low cost hardware).
Use of the system should not require any machines other than a 3D printer and an accompanying ultrasonic washer.
No tools other than a soldering iron and screw driver should be needed to use the system to its fullest capacity.
The main platform of the system cannot exceed 5in. X 5in. X 5in. due to the constraints of the Mojo 3D printer.
* Use of the system should not require any software other than the open source Arduino software.
* The material(s) used to create the module enclosures and the main platform should not conduct electricity.

significant change from the original constraints is that now a 3D printer and ultrasonic washer would not be excluded from the machines needed to use the system. Similarly, the use of the system will now require the use of some tools, which have been constrained to a soldering iron and screw driver, which are considered to be commonly owned by even hobbyists who are learning how to prototype. The constraint on size needed to be refined to a maximum of five cubic inches due to the constraints of the 3D printer being used throughout this research.

The last category of specifications that needed to be altered, with the changing scope of the research project, are the functions and their requirements. A complete list of these newly defined functions can be found in Appendix E. Some of these functions that no longer fit the overall goals of the research were removed. For brevity, only the function that was added is listed below in Table 13. This function was created to ensure that the system would be provided to the public in an open source format. The function is critical for meeting the newly defined goals and objectives of the research.

Table 13: Reworked Functions, Requirements, and Scores for Prototype (not all listed here)

Function	Requirement	Score (0-2)
Provide a format for the system to be used.	The system, along with instructions, should be open source through a website.	0: A website with a description of the system is created, but the STL files of the system are not openly available. 1: A website with a description of the system is created that has the STL files for the system openly available. 2: A website that has a description, the STL files of the system, a link to additional hardware needed, and instructions on how to assemble the system is created.

The reworked objectives and constraints better define what is going to be provided by the new system and what is expected out of the user. The system's design specifications will now be free, but will require the use of a 3D printer, ultrasonic washer, a couple of tools, and a monetary allowance for necessary hardware components. The newly defined specifications fit in line with the idea of creating a website that would make the STL files for the system open source. This will allow any student or hobbyist with access to a 3D printer to recreate the system for themselves. The website would also include a description of the system, directions for assembly, and links to the all the additional hardware used to complete the system. This type of format would allow individuals to purchase and implement whatever sensors they want to inside the modules, and would also encourage them to learn and practice their soldering skills.

CHAPTER 3: FINAL PROTOTYPE OF SYSTEM (3RD ITERATION)

3.1 CAD of Final Prototype of System (3rd Iteration)

Now that the scope and the specifications for the research have been redefined, it is time to step back into the realm of prototyping. A third iteration of the modules was designed and modeled in SolidWorks, shown below in Figure 25. These modules can be split up into three sections which can be described from bottom to top as the connector, pin, and sensor sections. These modules are taller than their iteration two equivalent to allow room for these three levels.

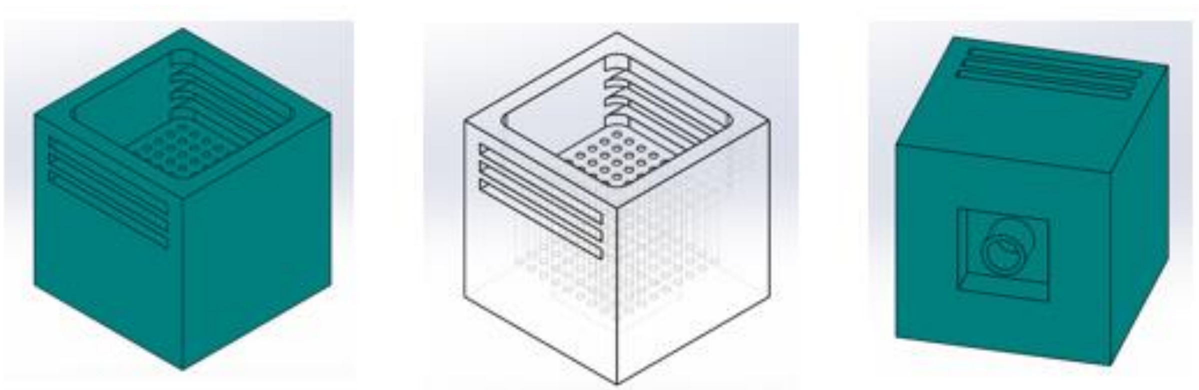


Figure 25: Third Iteration SolidWorks Models of Modules: Shaded with Edges Isometric View (Left), Hidden Lines Isometric View (Middle), Shaded with Edges Bottom View (Right)

The most obvious addition to these new pieces are the connector extrusions added to the bottom of the modules. These connectors will be used to attach the modules to the newly designed main platform that will be discussed later. The middle section, which can most clearly be seen in the hidden lines isometric view of Figure 25, includes the cylinders (nine by nine) that coincide with the holes of the ProtoBoards. The top section is again used as space for the sensors to be held. The slots on the side of the modules will be used as an access point for wires to be connected from the sensors ProtoBoard to the Arduino microcontroller.

After the modules were done being modeled, the main platform of the system was designed and created in SolidWorks. The top, isometric, bottom, and left side views of this platform are shown here in Figure 26. The top view shows the three areas that modules can be

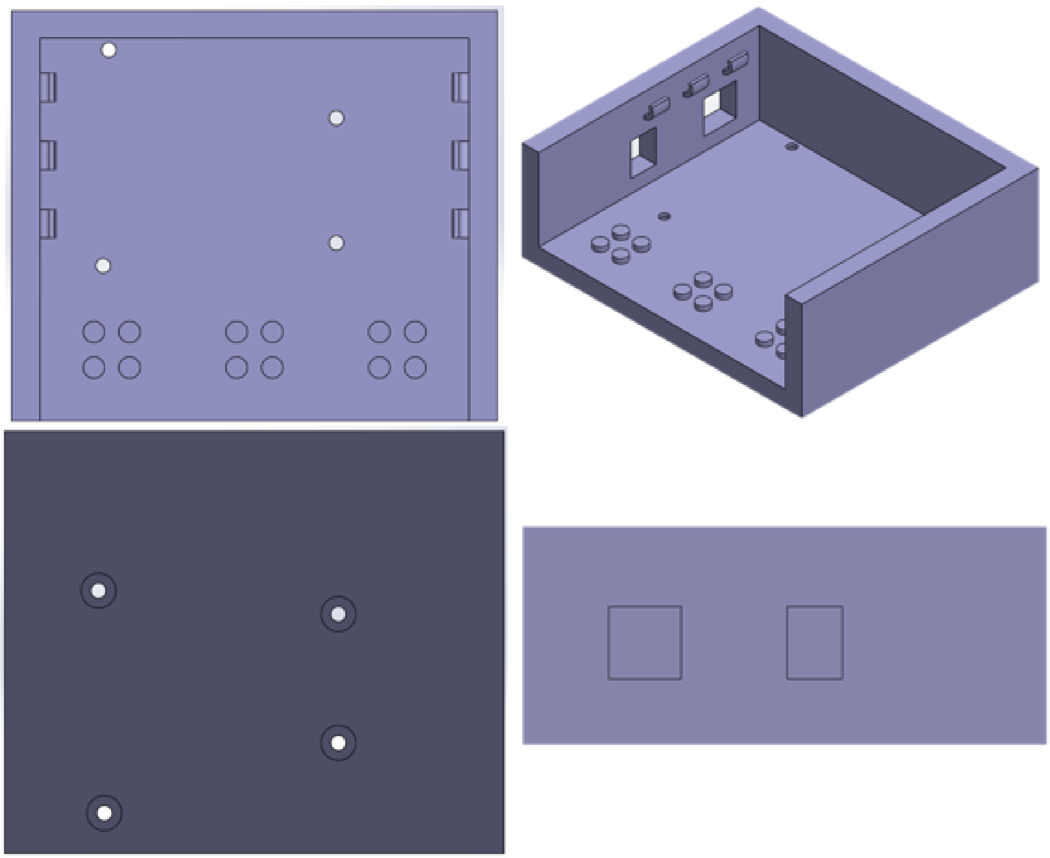


Figure 26: Third Iteration SolidWorks Model of Main Platform: Bottom View (Bottom Left), Top View (Top Left), Left Side View (Bottom Right), and Isometric View (Top Right)

connected to the platform. These sections each have four extruded cylinders. These extrusions fit around the extruded cylinder and within the square connector space of the modules. The top view also shows the circular cutouts used to mount the Arduino Uno microcontroller and the hooks on the walls of the platform that are used to hold wire from the modules on the platform that are not currently being used. The bottom view shows the counter bores that allow nuts to be placed on the end of the screws used to mount the Arduino board without causing the platform to become uneven. The left side view clearly shows the cutouts used for access to the Arduino boards USB port and barrel jack, which is necessary for code to be uploaded to the microcontroller. The assembly of the SolidWorks modeled parts is shown below in Figure 27. This assembly does not include the breadboard, the sensors, or the electrical wires. The Solidworks models of the Arduino Uno microcontroller board [36] and the header pins [37] were taken from GrabCAD. The engineering drawings for the main platform and modules can be found in Appendix F.

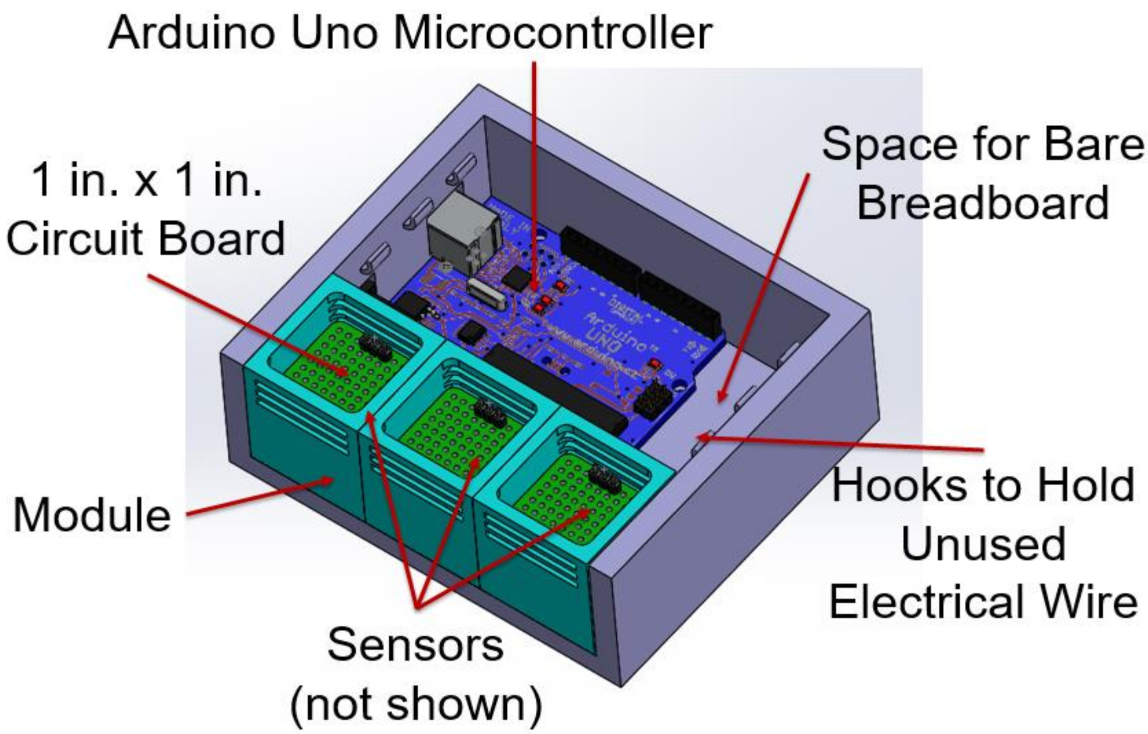
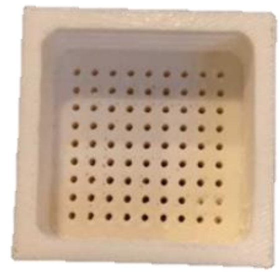


Figure 27: Third Iteration SolidWorks Model Assembly

3.2 3D Printed Parts for Final Prototype of System (3rd Iteration)

The Mojo 3D printer allows the user to choose which orientation a part is to be printed in. Choosing different printing orientations may affect the outcome of the printed part. Within this iteration of modules, three different orientations were tested. The results of these printed parts are shown on the next page in Table 14.

Table 14: Results of 3D Printing Modules in Different Orientations

	Upright	**Sideways**	**Upside Down**
Orientation			
Result (Top)			
Result (Bottom)			

As expected, the orientation affected the results of the print job. Printing the module in the upright position resulted in nicely formed cylinders, but made it difficult for the support material in the connector section to be removed. Removing this material affected the cylinder, which resulted in a looser connection with the main platform. Printing the module in a sideways orientation meant that there would be less support material to clear from the connector section of the part. Although this did result in better connectors, the cylinders did not print cleanly. This meant that the ProtoBoards could not be placed inside the modules because the pins from the sensors and wires could not be inserted into the holes. The solution came from orienting the part in the upside down position. Similar to the sideways orientation, very little support material needed to be used because the connector section was now at the top of the piece. The cylinders also came out cleanly because they were oriented vertically, as with the upright position. The modules used in the final prototype of the system were printed in the upside down orientation, because of the advantages associated with these results.

The main platform was oriented in the upright position (bottom flat on modeling base) while being printed. Even in this orientation, there was no problems with clearing the structure material in the counter bored holes. The top, isometric, bottom, and left side views of this 3D printed platform are shown here in Figure 28. Figures of the ‘Mojo Control Panel’s’ for the modules and main platform that show the amount of materials used and print time for each part can be found in Appendix D with the 3D printer specifications.

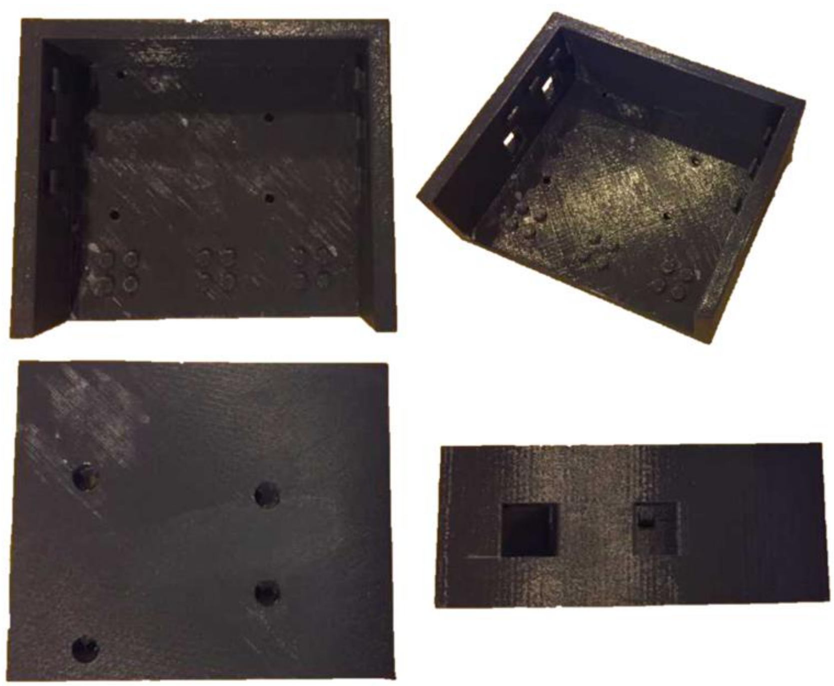


Figure 28: Third Iteration 3D Printed Main Platform: Bottom View (Bottom Left), Top View (Top Left), Left Side View (Bottom Right), and Isometric View (Top Right)

3.3 Physical Assembly of Final Prototype (3rd Iteration)

Once the base platform and the three modules were printed, the 'Mod Duino' could be assembled. The first step of the assembly is to solder the necessary wires onto the ProtoBoards from Figure 19 that already have the sensors soldered to them. Once the soldering is complete, the ProtoBoards are inserted into the top section of the modules. The completed modules are shown below in Figure 29. The wires that are soldered to the boards are inserted through one of the slots on the side of the module.

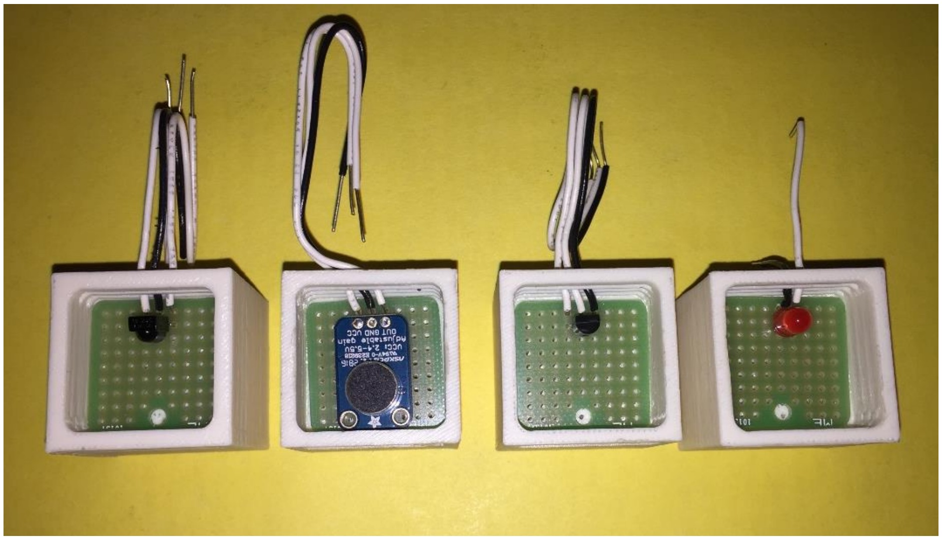


Figure 29: Final Assembly of Modules

After the modules have been assembled, the Arduino Uno is mounted to the main platform of the 'Mod Duino' with screws and nuts smaller than 1/8 of an inch, which is the diameter of the mounting holes on the Arduino board. Two strips of Velcro with sticky backsides were also added to the main platform and the bottom of the bare breadboard. The Velcro was used, instead of permanently mounting the breadboard onto the main platform, so that the breadboard could easily be detached and reattached to the platform. This will make it easier for components like resistors and LEDs to be placed onto the breadboard during the prototyping process. The breadboard used for the 'Mod Duino' is similar to the one from Figure 15, but was

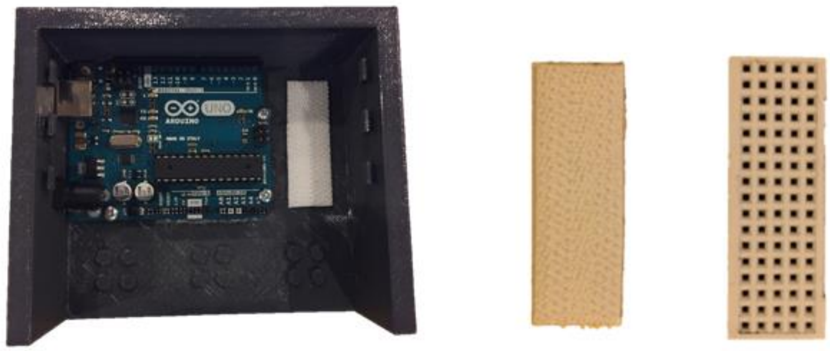


Figure 30: Main Platform with mounted Arduino Uno and Velcro used for mounting the Bare Mini Breadboard

sawed in half using a vertical band saw. The mounted Arduino board and the Velcro used to connect the bare breadboard to the main platform are shown in Figure 30. The breadboard was cut in half to fit inside the allowable space without affecting the buses that create an electrical connection within each of the seventeen columns (rows in the configuration of Figure 30). After the cut breadboard was attached to the main platform, three wires were used to connect the breadboard to the 3.3 volt, 5 volt, and ground pins of the Arduino Uno board (Figure 31).

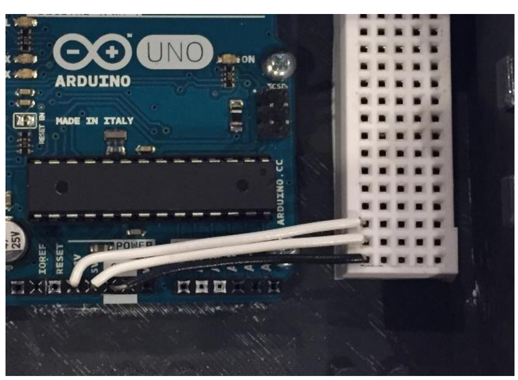


Figure 31: Creating the 3.3 Volt, 5 Volt, and Ground Buses on the Bare Breadboard

These connections were made to create two power buses (3.3 V and 5 V) and one ground bus on the bare platform. These buses give more space for power and ground connections while prototyping.

Now that the assembly of the main platform and modules is complete, the two components of the ‘Mod Duino’ can come together. The modules are mounted onto the main platform in the configuration shown in Figure 32. This figure gives four different views of the ‘Mod Duino’ final assembly. In this configuration, only the LED module is connected to the breadboard. The module housing the microphone and temperature sensor are not connected to the breadboard, so the wires connected to these modules are held to the side by the hooks on the walls of the base platform. This was done in an effort to keep things within the ‘Mod Duino’ organized.

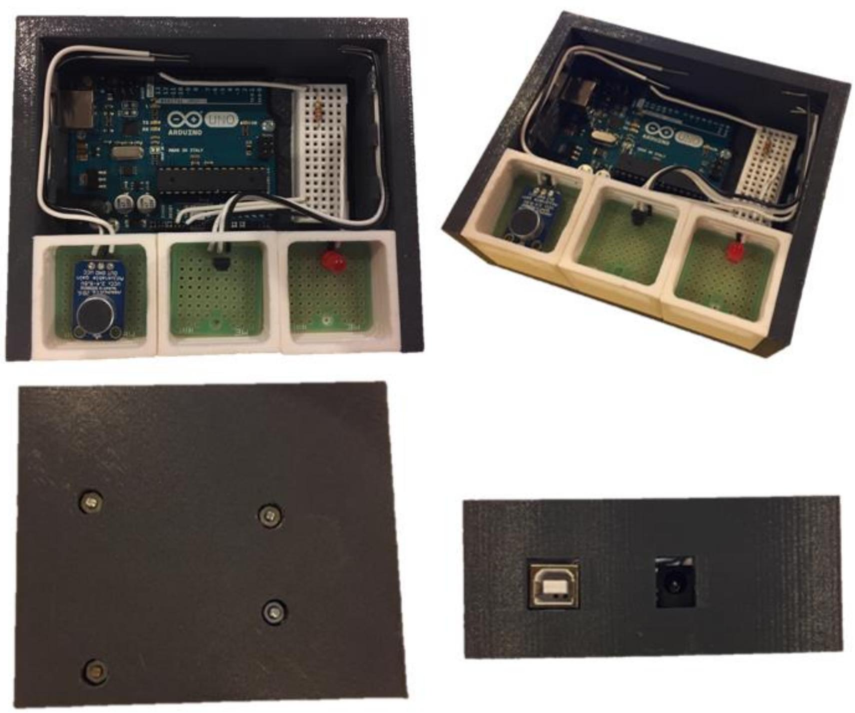


Figure 32: Final Assembly of 'Mod Duino': Bottom View (Bottom Left), Top View (Top Left), Left Side View (Bottom Right), and Isometric View (Top Right)

Once the assembly mimics that of Figure 32, the ‘Mod Duino” is ready for use. As discussed in previous sections, any type of sensor(s) that is needed for a prototype can be used within the module(s). Additionally, more than one sensor module can be used (via connection to the breadboard) for a given prototyping project. Shown below in Table 15 is the bill of materials for the complete assembly shown in Figure 32.

Table 15: Bill of Materials for Final Assembly of 'Mod Duino'

Description	**Quantity**	**Unit**	**Total Cost (Approximate)**
ABS (Part Material)	7.15	in^3	$57.20
S40 (Support Material)	1.38	in^3	$11.04
Arduino Uno Micrcontroller Board	1	N/A	$22.00
LED	1	Pack	$2.95
Temperature Sensor	1	N/A	$3.00
Electric Microphone Amplifier (MAX4466 with Adjustable Gain)	1	N/A	$6.95
ProtoBoard	3	N/A	$4.50
Resistor (220 ohm)	1	Pack	$0.95
Hook-up Wire (Black 22 AWG)	1	Spool	$2.50
Hook-up Wire (White 22 AWG)	1	Spool	$2.50
Mini Modular Breadboard	1	N/A	$3.95
Screw (4-40 UNC, ~1/4 inch long)	4	N/A	$1.00
Nut (4-40 UNC)	4	N/A	$1.00
Total Cost			$119.54

CHAPTER 4: RESULTS AND DISCUSSION

4.1 Testing and Evaluation of Final Prototype

After the final assembly of the 'Mod Duino' was completed, testing began by verifying that one of the modules could be used to successfully run a program. The example code "Blink" provided by Arduino was used for this verification. The code for this program can be found on the Arduino programming software by navigating through File/Example/0.1Basics/Blink. The entirety of the "Blink" code can be found in Appendix G. The circuit configuration and schematic used to run this program is shown in Figure 33. In order for the code to succesfully be

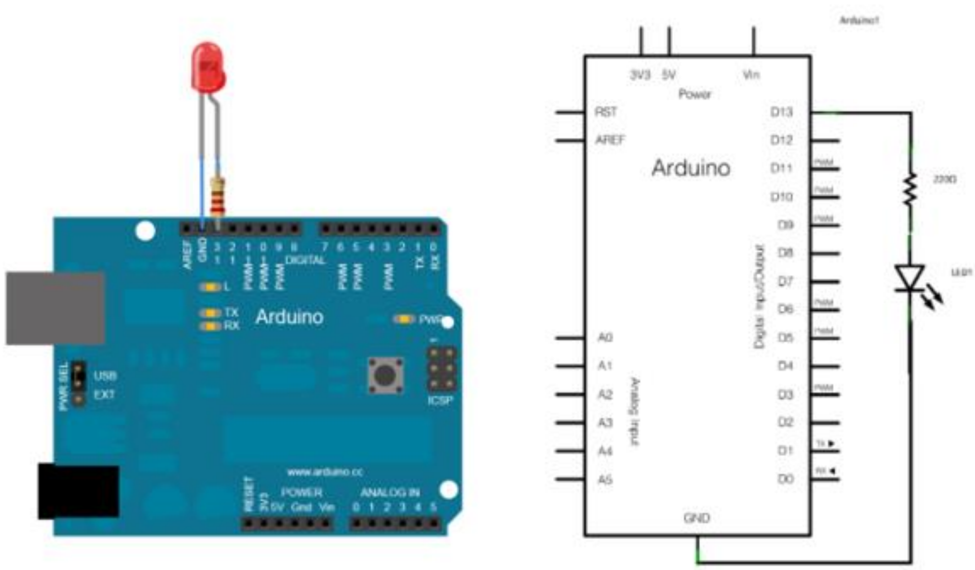


Figure 33: Circuit Configuration and Schematic for Arduino Code "Blink" [38]

uploaded to the Arduino microcontroller, it is important to check that the correct board and port are selected under the "Tools" tab. The result of running the "Blink" program using the 'Mod Duino' is shown in Figure 34. The red light emitting from the sensors in the illustration indicates

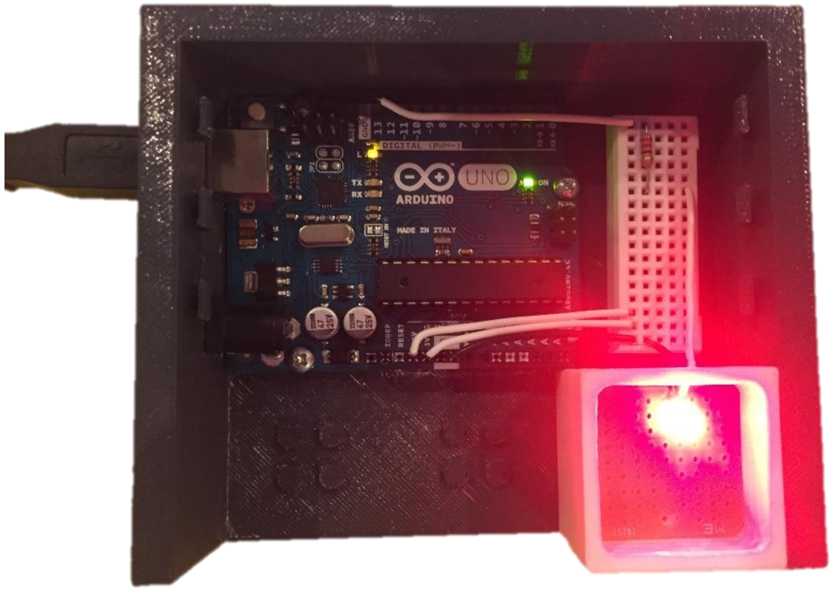


Figure 34: 'Mod Duino' Running the Arduino Example Code “Blink”

that the program was able to be run successfully. Uploading and running this code was used as a test case, and its success resulted in the assumption that prototypes involving all the other sensor modules would work as well.

In addition to verifying that the ‘Mod Duino’ is a functional prototype, the success of the system was measured by testing it against the research objectives and functional requirements. The results of this testing is shown in Table 16. All of the predefined objectives were met, other than the life cycle of the system. It could not be determined if this objective was met because of the time frame of this research, but an educated assessment would indicate that the ‘Mod Duino’ would remain working after five years due to the strength and resiliency of ABS.

Table 16: Test 'Mod Duino' Against Research Objectives

Objective	Metric	Objective Met?
Final Cost	The STL files to print the main platform and modules should be free to the public.	✓
Appearance	The final product is aesthetically pleasing so that it can be presented to the end user/business professionals.	✓
Size	The product is smaller than 5in x 4in x 3in.	✓
Weight	The product weighs less than 2 lbs.	✓
Ease of use	The product is easily understood and operated mainly through intuition (with little guidance) if user is familiar with Arduino programming.	✓
Reliability	The system should not fail when assembled properly.	✓
Strength/Durability	The materials used have a high enough strength to not break during typical use.	✓
Portability	The embedded prototyping system can be moved from place to place without any connections being lost.	✓
Life Cycle	The product should have a life cycle no shorter than five years.	Unknown
Safety	The product should be safe to use under normal operating conditions. No hazardous materials should be used when creating the product.	✓

The results of the 'Mod Duino' testing against functional requirements are indicated by the awarded scores in Table 17, which is presented at the end of this section. The possible scores range from zero to two, two being the most successful. The details of each of the possible score can be found in Appendix E. Including the optional function, there is a total possible score of twenty four. After evaluating the functional requirements, the 'Mod Duino' received a score of twenty two. The first requirement about providing the platform as open source will be discussed in detail in the following section. All of the requirements were met to the highest standard other than the number of modules the platform can hold and the total encasing of the platform. The 'Mod Duino' received a score of one for these requirements because it can only accommodate three modules at a time and does not completely encase the modules and Arduino Uno board. Although they did not meet the highest standard, these functions are considered to be successfully achieved, and the enclosure function was originally recognized as being optional.

Table 17: Testing 'Mod Duino' Against Functional Requirements

Functions	Requirements	Score (0 Min - 2 Max)
Provide a format for the system to be used.	The system, along with instructions, should be open source through a website.	2: A website that has a description, the STL files of the system, a link to additional hardware needed, and instructions on how to assemble the system is created.
Create modules that reliably encase sensors and actuators.	The modules are rigid enough to pick up and install/mount onto platform while retaining their shape.	2: The modules retain their shape when being handled and installed/removed from the platform.
Implement sensors that are compatible with the required software.	The sensors and actuators must cooperate with the input/output capabilities of the ATmega328P microcontroller.	2: Every component used to make the modules is compatible with the ATmega328P.
Create modules that can be easily connected to the platform.	The modules can be connected to the main platform with normal hand force and no additional tooling.	2: The modules can be connected to the main platform with normal hand force.
Create modules that can be easy disconnected to the platform.	The modules can be disconnected to the main platform with normal hand force and no additional tooling.	2: The modules can be disconnected to the main platform with normal hand force.
Create modules that can be easy reconnected to the platform.	The modules can be reconnected to the main platform with normal hand force and no additional tooling.	2: The modules can be reconnected to the main platform with normal hand force.
Create a reliable connection between the modules and platform.	The entire device can be rotated, moved, and dropped from a height of one foot without any parts becoming disconnected.	2: The modules stay attached to the platform when the device is moved, rotated, and dropped.
The platform should be open to housing multiple modules at once.	The platform can support at least five modules at one time.	1: The platform can support two through four modules at a time.
Connect to power and ground.	Each module needs to be able to be connected to a power source and ground on the Arduino platform without lose wiring.	2: Each module can be connected to power and ground in a discrete fashion that does not involve any lose wiring.
The new systems will run using existing software.	The new system will use Arduino software for coding.	2: Arduino software can be successfully uploaded to the device.
Any enclosure should not interfere with any connections.	Any enclosure should provide clearance or openings for all necessary connections.	2: The enclosure provides enough of an opening for all necessary connections to be made, independent of how many modules are in use.
Optional:		
There should be an enclosure for the system.	The microcontroller and all other hardware should be globally encased.	1: Most of the platforms components are hidden within a case.
Total Score		**22/24**

4.2 Publishing 'Mod Duino' to be Open Source

To fully satisfy the updated functional requirement, a website that has a description, the STL files of the system, a link to additional hardware needed, and instructions on how to assemble the system must be created. Thingiverse is the website that was used to satisfy this requirement. Thingiverse is a platform for engineers, designers, and prototypers of all experience levels to view and download each other's digital files for use. The 'Mod Duino' design (www.thingiverse.com/thing:2627924) was added to Thingiverse under the Creative Commons Attribute license, which "lets others distribute, remix, tweak, and build upon your work, even commercially, as long as they credit you for the original creation. This is the most accommodating of licenses offered" [39].

The top of the design page, shown in Figure 35, showcases photos of the 'Mod Duino's' final prototype, 3D printed parts, and Solidworks screenshots, as well as downloadable STL files for the modules and main platform. The subsequent sections included on the design page are

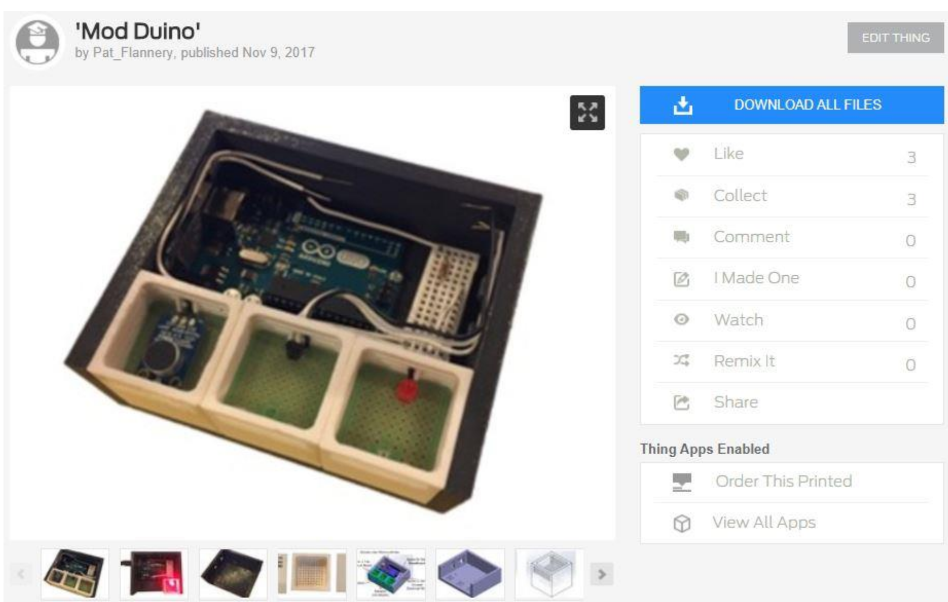


Figure 35: Top of 'Mod Duino' Design Page on Thingiverse

summary, print settings, post-printing, overview and background, materials needed, references, and miscellaneous figures. In general, the text and figures used for these sections were taken directly from this thesis. However, the "Summary" is laid out in a simple bullet point style that provides the project's most important pieces of information.

Additional notes giving recommendations on print orientations for both the modules and the main platform are provided under the "Print Settings" section. The recommendations are explained, and a figure showing the results of 3D printing the modules in three different orientations is provided at the bottom of the web page. Instructions for creating the final assembly of the 'Mod Duino' and the results of testing the Arduino "Blink" code are included in the "Post-Printing" section.

The abstract, significance, and information about the Tibbo Project System are included in the "Overview and Background" section. This section was placed closer to the bottom of the page because in this platform, information about printing and assembling the 'Mod Duino' are more important than the background research. The bill of materials and links to all of the components needed to assemble the 'Mod Duino' can be found under the "Materials Needed" and "References" sections. Tags were also added to the page so that anyone in the Thingiverse community interested in areas related to this research, such as 3D printing, prototyping, Arduino Uno, and modular design could more easily navigate to the 'Mod Duino' page.

4.3 Printing 'Mod Duino' with Different 3D Printers

Although there are many types of 3D printers such as stereolithography (SLA), digital light processing (DLP), selective laser sintering (SLS), selective laser melting (SLM), electronic beam melting (EBM), and laminated object manufacturing (LOM) [40], fused deposition modeling printers will be discusses in this section because that is the type of rapid prototyping machine used in this research. Here, the capabilities of a lower end 3D printers will be analyzed, due to the assumption that the target audience of this research is more likely to have access to such a machine. This analysis is done because the capabilities of a less expensive printer may have an effect on the creation of the final parts. In other words, the ability to create a main platform and modules that can be assembled and used from the open source STLs may be effected by the printer being used.

The Replicator Mini+ Compact 3D Printer offered by MakerBot in Figure 36 is the lower end printer that will be analyzed for comparison. A complete list of technical specifications for this printer can be found in Appendix D. At 1,299 USD on the company website, a person new to

Figure 36: Replicator Mini+ Compact 3D Printer [41]

3D printing in much more likely to have access to this caliber of printer. With an available build volume of 4x4x5 in, this printer has space to accommodate the components needed to create the 'Mod Duino.' The minimum later resolution is 0.0039 inches (0.1 mm) and the maximum layer resolution is 0.0157 inches (0.4 mm), which is comparable to the Mojo printer (0.007 in or 0.17 mm). Table 18 gives an overall summary of the differences between industrial and desktop FDM

3D printers [42]. The most important differences come from the part material, support material, and accuracy.

Table 18: Summary of Differences between Industrial and Desktop FDM 3D Printers

Property	Industrial FDM	Desktop FDM
Standard Accuracy	± 0.15% [lower limit ± 0.2 mm (0.008 in)]	± 1% [lower limit: ± 1.0 mm (0.04 in)]
Typical Layer Thickness	0.18 - 0.5 mm (0.007 – 0.02 in)	0.10 - 0.25 mm (0.004 – 0.01 in)
Minimum Wall Thickness	1 mm (0.04 in)	0.8 - 1 mm (0.03 – 0.04 in)
Maximum Build Envelope	Large (e.g. 900x600x900 mm or 35.4x23.6x35.4 in)	Medium (e.g. 200x200x200 mm or 7.9x7.9x7.9 in)
Common Materials	ABS, PC, ULTEM	PLA, ABS, PETG
Support Material	Water-soluble	Same as part (typically)
Production Capabilities (per machine)	Low/Medium	Low
Machine Cost	$50,000+	$500-$5000

4.4 Discussion on Creating 'Mod Duino' with Different Printers

The 3D Mojo printer was used for this project is generally considered to be one of the high end desktop printers. The Mojo is part of the Idea Series of printers offered by Stratasys, which produces industrial printers. In actuality, the specifications and characteristic of the Mojo printer align more closely with an industrial printer than a desktop printer (Table 18). It is assumed that any industrial FDM 3D printer has higher capabilities than the Mojo, and will be able to create a well-functioning 'Mod Duino' with ease.

As mentioned in the previous section, the layer thickness and build volume of the Mojo and Replicator Mini+ are comparable. The main differences between the machines are the material, supports, and accuracy. The material used in the Mojo is ABS while the Replicator

Mini+ uses PLA. In general, ABS is stronger and more resilient material while PLA "will give you more precise prints and better aesthetic quality" [43]. Overall, the strength lost from using PLA instead of ABS should not be enough to disapprove the use of the Replicator to create the 'Mod Duino.'

Secondly, the Mojo uses a soluble material for supports while the Replicator uses the part material for supports. This means that the support material used in the Mojo can be completely removed by the Ultra 2000, but the support material of the Replicator may be more difficult to remove. This may become especially apparent while attempting to remove support material needed near and around the small holes in the modules. Thirdly, as shown in Table 18, the accuracy of the lower end machine will most likely be lower than the Mojo (as expected). Again, the reduction in accuracy will be most apparent within the small features of the modules.

Without having time to locate a Replicator Mini+ or similar low end machine to actually print the 'Mod Duino,' it can be concluded from the similarity in specifications that a functional main platform should be able to be created with such a machine. However, due to the supports and reduction in accuracy of a lower end desktop 3D printer, it is more unclear as to whether functional modules would be able to be printed. The accuracy of a printer is not in the specifications for the machine, so it is difficult to determine if a functioning 'Mod Duino' would be able to be created with a lower end machine without further testing.

CHAPTER 5: FUTURE WORK

Although this project resulted in a functional prototype of the 'Mod Duino,' there are several aspects from the original objectives that can be expanded upon through future research. To begin with, the STL files created through this research could actually be printed with a lower end desktop 3D printer such as the Replicator Mini+ from MakerBot. Testing with this printer, as well as other popular printers on the market, would result in a better understanding of the applications of the 'Mod Duino.' Testing on additional printers would also allow the system to be created using materials other than ABS, such as PLA and PETG. Providing information about which printers and materials would result in the creation of a functioning 'Mod Duino' would be useful information to the public.

There are also opportunities to expand upon the system that has been created. One way to do this would be to design a better way to hold wires from modules that are not being used within a prototype, so that they do not get in the way or make the platform disorganized. The hook up wires used within this project were stiff enough to hold themselves against the walls of the main platform, but more flexible wires may not be able to be contained within the hooks on the walls of the main platform. Additional hooks could also be added to the back wall of the platform so that longer wires that are not being used would not fall onto the Arduino Uno or breadboard during prototyping. The platform could also be improved upon by adding space for an external power supply, such as batteries or DC adapter, so that the code for a given prototype can be run without having the system connected to a computer. The bare breadboard used within the 'Mod Duino' could also be improved upon by finding a way to custom manufacture the design in Figure 14. It would also be useful to provide a bigger breadboard than the one used in this research to allow a larger space for additional components needed for a prototyping project.

The Arduino Uno was the only microcontroller board used in this project, but the value of the prototyping platform would increase if a wider range of boards could be used within the system. Further research could be done to make the platform modular to fit other boards within the Arduino family as well as other popular microcontroller boards like the Raspberry Pi. Originally, the goal was to create a product that could be sold to the general public. Due to the high cost of mass producing 3D printed parts, the scope of the project changed. Further research could be done to figure out how the 'Mod Duino' could be mass produced in a cost efficient way. This would involve researching different materials and manufacturing methods other than 3D printing (rapid prototyping).

References

[1] "Benefits of Prototyping." *SQA*. [Online]. Retrieved from http://www.sqa.org.uk/e-learning/IMAuthoring01CD/page_06.htm.

[2] Udanis, Alex. "The Best Wire for Breadboarding." *All About Circuits,* 08 Jan. 2016. [Online]. Retrieved from https://www.allaboutcircuits.com/news/the-best-wire-for-breadboarding/.

[3] Raja, Dilip. "Flex Sensor Interfacing with AVR Microcontroller." *Circuit Digest*. [Online]. Retrieved from https://circuitdigest.com/microcontroller-projects/flex-sensor-interfacing-with-atmega8.

[4] "ARDUINO UNO REV3." *Arduino*. [Online]. Retrieved from https://store.arduino.cc/usa/arduino-uno-rev3.

[5] "Raspberry Pi 3 – Model B – ARMv8 with 1G RAM." Adafruit. [Online]. Retrieved from https://www.adafruit.com/product/3055?src=raspberrypi.

[6] "Prototyping tools for the Internet of Things." *Particle*. [Online]. Retrieved from http://staging-www2.spark.io/prototype#arduino-like.

[7] "BBC micro:bit." *Adafruit*. [Online]. Retrieved from https://www.adafruit.com/product/3530.

[8] "Shields." *Arduino*. [Online]. Retrieved from https://www.arduino.cc/en/Main/ArduinoShields.

[9] "Actobotics." *Servo City*. [Online]. Retrieved from https://www.servocity.com/actobotics.

[10] "Actobotics Kit – Nomad 4WD Off-Road Chassis." *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/13141.

[11] Jovanovic, Dusko, Bojan Orlic, Jan Broenink, and Job Van Amerongen. *INEXPENSIVE PROTOTYPING FOR MECHATRONIC SYSTEMS *)* page 8. Web. 20 July 2016.

[12] “Arduino Project Enclosure.” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/10088.

[13] “Arduino and Breadboard Holder.” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/11235.

[14] "Liquidware Brings Modular, Rapid Prototyping to the BeagleBoard Embedded Platform." *Cision PRWeb*, 14 June 2010. [Online]. Retrieved from http://www.prweb.com/releases/2010/06/prweb4131444.htm.

[15] “iProtoXi Published Intelligent Open Interface Standard for Low Power Internet of Things.” *iProtoXi*. [Online]. Retrieved from https://www.iprotoxi.fi/index.php/aistin-bus24.

[16] Kallio, Janne. “iProtoXi Aistin: Multi-Modular Sensor Platform.” *Indiegogo*. [Online]. Retrieved from https://www.indiegogo.com/projects/iprotoxi-aistin-multi-modular-sensor-platform#/.

[17] “Developer’s Kit.” *Voxel8*. [Online]. Retrieved from https://support.voxel8.co/hc/en-us/articles/207469773-Developer-s-Kit-Platform-Overview.

[18] “Tibbo Project System is a Modular Linux IoT Prototyping Platform based on TI Sitara Cortex A8 Processor.” *CNXSoft*, 28 Apr. 2016. [Online]. Retrieved from https://www.cnx-software.com/2016/04/28/tibbo-project-system-is-a-modular-linux-iot-prototyping-platform-based-on-ti-sitara-cortex-a8-processor/.

[19] “Announcing LTTP3, our first Linux-based TPS offering!” *Tibbo*, 04 Apr. 2016. [Online]. Retrieved from http://tibbo.com/.

[20] The Destroyer, Chelsea. “The Arduino popularity contest.” *Sparkfun*, 01 Dec. 2015. [Online]. Retrieved from https://www.sparkfun.com/news/1982.

[21] Zeid, Ibrahim. Figure 1.1 “Engineering design process (EDP)” taken from *Mastering Solidworks The Design Approach* (Second Edition) page 4. Pearson Education, Inc., 2015.

[22] Zeid, Ibrahim. Figure 1.2 “CAD process” taken from *Mastering Solidworks The Design Approach* (Second Edition) page 5. Pearson Education, Inc., 2015.

[23] Arefin, Utsho. “Arduino Pin Diagram.” *SearchGate*, Mar. 2016. [Online]. Retrieved from https://www.researchgate.net/figure/298882952_fig6_Fig-34-Arduino-Pin-Diagram.

[24] “Electret Microphone Amplifier – MAX4466 with Adjustable Gain.” *Adafruit*. [Online]. Retrieved from https://www.adafruit.com/product/1063.

[25] “IR Receiver Diode – TSOP38238.” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/10266.

[26] “Temperature Sensor – TMP36.” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/10988.

[27] “Basic Red LED 3mm (10 pack).” *Banana Robotics*. [Online]. Retrieved from https://www.bananarobotics.com/shop/Basic-Red-LED-3mm-(10-pack).

[28] “ProtoBoard – Square 1” Single Sided.” *Sparkfun*. [Online]. Retried from https://www.sparkfun.com/products/8808.

[29] “Hook-up Wire – Black (22 AWG).” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/8022.

[30] “Breadboard – Mini Modular (Red).” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/products/12044.

[31] “Fused Deposition Modeling (FDM) Design Guideline.” *Stratasys*. [Online]. Retrieved from https://www.stratasysdirect.com/resources/fused-deposition-modeling/.

[32] “Mojo Magic at Your Desk.” *Stratasys*. [Online]. Retrieved from http://www.stratasys.com/3d-printers/mojo.

[33] Chakravorty, Dibya. “STL File Format for 3D Printing – Simply Explained.” *All3DP*, 13 June 2017. [Online]. Retrieved from https://all3dp.com/what-is-stl-file-format-extension-3d-printing/.

[34] “Ultra 2000 – 5.2 Gallon.” *Stratasys*. [Online]. Retrieved from http://store.ultrasonicllc.com/item/ultra-ultrasonic-cleaning-machine-gallon.

[35] “Building an Arduino on a Breadboard.” *Arduino*. [Online]. Retrieved from https://www.arduino.cc/en/Main/Standalone.

[36] Silva, Jorge Alberto. “Arduino UNO Reference Design.” *GrabCAD*, 02 Nov. 2010. [Online]. Retrieved from https://grabcad.com/library/arduino-uno-reference-design.

[37] Apti, Murat. “HEADER PIN – MALE HEADER PIN – HEADER STRIP PIN.” *GrabCAD*, 11 Jan. 2015. [Online]. https://grabcad.com/library/header-pin-male-header-pin-header-strip-pin-1.

[38] “Blink.” *Arduino*. [Online]. Retrieved from https://www.arduino.cc/en/Tutorial/Blink.

[39] “About The Licenses.” *Creative Commons*. [Online]. Retrieved from https://creativecommons.org/licenses/.

[40] “Types of 3D printers or 3D printing technologies overview.” *3D Printing from Scratch*. [Online]. Retrieved from http://3dprintingfromscratch.com/common/types-of-3d-printers-or-3d-printing-technologies-overview/.

[41] “Replicator Mini+ Compact 3D Printer.” *MakerBot*. [Online]. Retrieved from https://store.makerbot.com/printers/replicator-mini/.

[42] Varotsis, Alkaios Bournias. “Industrial FDM vs. Desktop FDM.” *3D HUBS*. [Online]. Retrieved from https://www.3dhubs.com/knowledge-base/industrial-fdm-vs-desktop-fdm.

[43] Hesse, Brendan. “ABS or PLA: Which 3D printing filament should you use?” *Digital Trends*, 26 Sept. 2015. [Online]. Retrieved from https://www.digitaltrends.com/cool-tech/abs-vs-pla-3d-printing-materials-comparison/.

[44] “MAXIM Low-Cost, Micropower, SC70/SOT23-8, Microphone Preamplifiers with Complete Shutdown.” *Adafruit*. [Online]. Retrieved from https://cdn-shop.adafruit.com/datasheets/MAX4465-MAX4469.pdf.

[45] “CUI INC PART NUMBER: CMA-4544PF-W.” *Adafruit*. [Online]. Retrieved from https://cdn-shop.adafruit.com/datasheets/CMA-4544PF-W.pdf.

[46] “TSOP382…, TSOP384.. Vishay Semiconductors IR Receiver Modules for Remote Control Systems.” *Sparkfun*. [Online]. Retrieved from https://www.sparkfun.com/datasheets/Sensors/Infrared/tsop382.pdf.

[47] “ANALOG DEVICES Low Voltage Temperature Sensors.” *Sparkfun*. [Online]. Retrieved from https://cdn.sparkfun.com/datasheets/Sensors/Temp/TMP35_36_37.pdf.

[48] “LED MIXED PACK.” *Sparkfun*. [Online]. Retrieved from https://cdn.sparkfun.com/datasheets/Components/LED/COM-12062.jpg.

[49] Kakumani, Akul. “Mojo Materials.” *BuckeyeBox*, 25 Aug. 2017.

[50] Kakumani, Akul. “Soluble concentrate.” *BuckeyeBox*, 25 Aug. 2017.

APPENDIX A: DECISION MATRICES

Decision Matrices for Stand Alone Devices									
		Flower		**Bricks**		**Tibbo Style**		**Tower**	
Functions	**Weight (1-4)**	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score
1	3	2	6	2	6	2	6	2	6
2	2	1	2	2	4	2	4	1	2
3	4	2	8	2	8	2	8	2	8
4	3	2	6	1	3	2	6	2	6
5	3	2	6	1	3	2	6	2	6
6	3	2	6	1	3	2	6	2	6
7	4	2	8	1	4	2	8	2	8
8	2	2	4	2	4	2	4	1	2
9	4	1	4	1	4	2	8	1	4
10	4	2	8	2	8	2	8	2	8
11	4	2	8	2	8	2	8	2	8
12	4	2	8	2	8	2	8	2	8
13	2	2	4	2	4	2	4	2	4
14	1	1	1	1	1	2	2	1	1
Optional:									
15	1	1	1	2	2	2	2	1	1
16	1	2	2	2	2	2	2	2	2
Total			82		72		90		80

Figure 37: Decision Matrices for Stand Alone Devices (First Four Designs)

Decision Matrices for Stand Alone Devices									
		Triangular		**Slim Plugins**		**Basic Bread Board**		**Circular**	
Functions	**Weight (1-4)**	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score	Score	Weighted Score
1	3	2	6	2	6	2	6	2	6
2	2	1	2	1	2	2	4	1	2
3	4	2	8	2	8	2	8	2	8
4	3	1	3	2	6	2	6	2	6
5	3	1	3	2	6	2	6	2	6
6	3	1	3	2	6	2	6	2	6
7	4	2	8	1	4	1	4	2	8
8	2	1	2	2	4	2	4	2	4
9	4	1	4	1	4	1	4	1	4
10	4	2	8	2	8	2	8	2	8
11	4	2	8	2	8	2	8	2	8
12	4	2	8	2	8	2	8	2	8
13	2	2	4	2	4	2	4	2	4
14	1	2	2	1	1	1	1	1	1
Optional:									
15	1	2	2	2	2	1	1	2	2
16	1	2	2	2	2	2	2	2	2
Total			73		79		80		83

Figure 38: Decision Matrices for Stand Alone Devices (Last Four Designs)

APPENDIX B: ARDUINO SPECIFICATIONS

Table 19: Arduino Uno Microcontroller Board Specifications [4]

Microcontroller	**ATmega328P**
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

APPENDIX C: SENSOR DATASHEETS

19-1950; Rev 1; 4/01

MAXIM

Low-Cost, Micropower, SC70/SOT23-8, Microphone Preamplifiers with Complete Shutdown

MAX4465–MAX4469

General Description

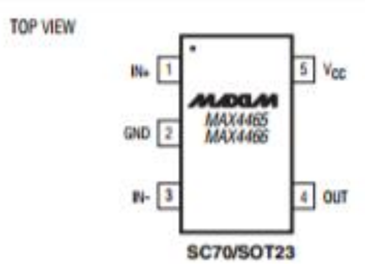
The MAX4465–MAX4469 are micropower op amps optimized for use as microphone preamplifiers. They provide the ideal combination of an optimized gain bandwidth product vs. supply current, and low voltage operation in ultra-small packages. The MAX4465/MAX4467/MAX4469 are unity-gain stable and deliver a 200kHz gain bandwidth from only 24µA of supply current. The MAX4466/MAX4468 are decompensated for a minimum stable gain of +5V/V and provide a 600kHz gain bandwidth product. In addition, these amplifiers feature Rail-to-Rail® outputs, high A_{VOL}, plus excellent power-supply rejection and common-mode rejection ratios for operation in noisy environments.

The MAX4467/MAX4468 include a complete shutdown mode. In shutdown, the amplifiers' supply current is reduced to 5nA and the bias current to the external microphone is cut off for ultimate power savings. The single MAX4465/MAX4466 are offered in the ultra-small 5-pin SC70 package, while the single with shutdown MAX4467/MAX4468 and dual MAX4469 are available in the space-saving 8-pin SOT23 package.

Applications

- Microphone Preamplifiers
- Hearing Aids
- Cellular Phones
- Voice-Recognition Systems
- Digital Dictation Devices
- Headsets
- Portable Computing

Pin Configurations



Pin Configurations continued at end of data sheet.

Rail-to-Rail is a registered trademark of Nippon Motorola, Ltd.

Features

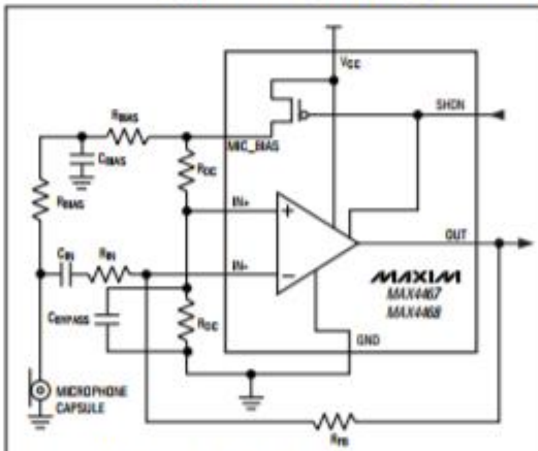
- +2.4V to +5.5V Supply Voltage Operation
- Versions with 5nA Complete Shutdown Available (MAX4467/MAX4468)
- Excellent Power-Supply Rejection Ratio: 112dB
- Excellent Common-Mode Rejection Ratio: 126dB
- High A_{VOL}: 125dB (R_L = 100kΩ)
- Rail-to-Rail Outputs
- Low 24µA Quiescent Supply Current
- Gain Bandwidth Product:
  200kHz (MAX4465/MAX4467/MAX4469)
  600kHz $A_V \geq 5$ (MAX4466/MAX4468)
- Available in Space-Saving Packages
  5-Pin SC70 (MAX4465/MAX4466)
  8-Pin SOT23 (MAX4467/MAX4468/MAX4469)

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX4465EXK-T	-40°C to +85°C	5 SC70-5
MAX4465EUK-T	-40°C to +85°C	5 SOT23-5
MAX4466EXK-T	-40°C to +85°C	5 SC70-5
MAX4466EUK-T	-40°C to +85°C	5 SOT23-5

Ordering Information continued at end of data sheet.

Typical Operating Circuit



MAX4467/MAX4468 TYPICAL OPERATING CIRCUIT WITH COMPLETE SHUTDOWN

MAXIM — *Maxim Integrated Products* 1

For pricing, delivery, and ordering information, please contact Maxim/Dallas Direct! at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

Figure 39: MAX4466 Low-Noise Microphone Amp Datasheet (Page 1) [44]

CUI INC

page 1 of 4
date 06/2008

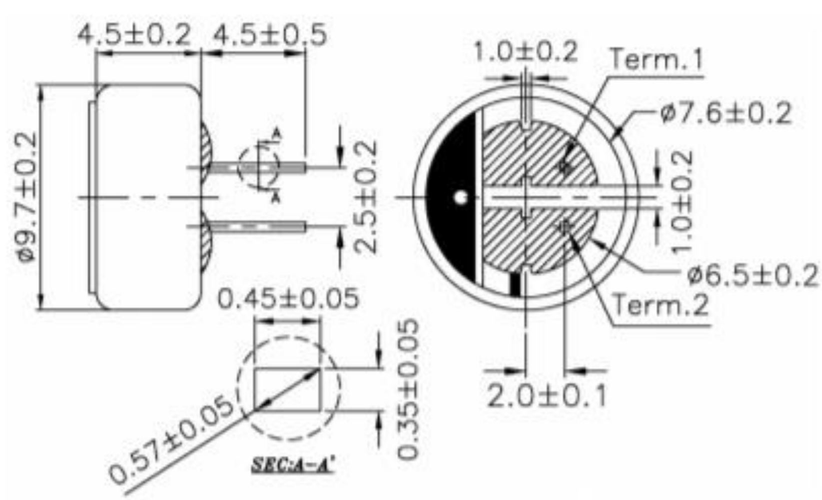
PART NUMBER: CMA-4544PF-W **DESCRIPTION:** electret condenser microphone

SPECIFICATIONS

directivity	omnidirectional	
sensitivity (S)	-44 ±2 dB	f = 1KHz, 1Pa 0dB = 1V/Pa
sensitivity reduction (ΔS-Vs)	-3 dB	f = 1KHz, 1Pa Vs = 3.0 ~ 2.0 V dc
operating voltage	3 V dc (standard), 10 V dc (max.)	
output impedance (Zout)	2.2 KΩ	f = 1KHz, 1Pa
operating frequency (f)	20 ~ 20,000 Hz	
current consumption (IDSS)	0.5 mA max.	Vs = 3.0 V dc RL = 2.2KΩ
signal to noise ratio (S/N)	60 dBA	f = 1KHz, 1Pa A-weighted
operating temperature	-20 ~ +70° C	
storage temperature	-20 ~ +70° C	
dimensions	ø9.7 x 4.5 mm	
weight	0.80 g max.	
material	Al	
terminal	pin type (hand soldering only)	
RoHS	yes	

note: We use the "Pascal (Pa)" indication of sensitivity as per the recomendation of I.E.C. (International Electrotechnical Commission). The sensitivity of "Pa" will increase 20dB compared to the "ubar" indication. Example: -60dB (0dB = 1V/ubar) = -40dB (1V/Pa)

APPEARANCE DRAWING



20050 SW 112th Ave. Tualatin, Oregon 97062 **phone** 503.612.2300 **fax** 503.612.2383 www.cui.com

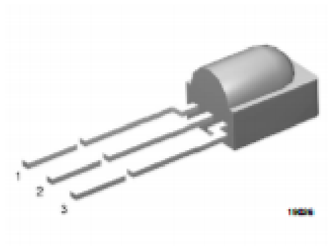
Figure 40: Electret Capsule Microphone Datasheet (Page 1) [45]

TSOP382.., TSOP384..

Vishay Semiconductors

VISHAY.

IR Receiver Modules for Remote Control Systems



FEATURES

- Very low supply current
- Photo detector and preamplifier in one package
- Internal filter for PCM frequency
- Improved shielding against EMI
- Supply voltage: 2.5 V to 5.5 V
- Improved immunity against ambient light
- Insensitive to supply voltage ripple and noise
- Component in accordance to RoHS 2002/95/EC and WEEE 2002/96/EC

Pb-free e3 RoHS COMPLIANT

MECHANICAL DATA

Pinning:

1 = OUT, 2 = GND, 3 = V_S

DESCRIPTION

The TSOP382.., TSOP384.. series are miniaturized receivers for infrared remote control systems. A PIN diode and a preamplifier are assembled on a lead frame, the epoxy package acts as an IR filter.

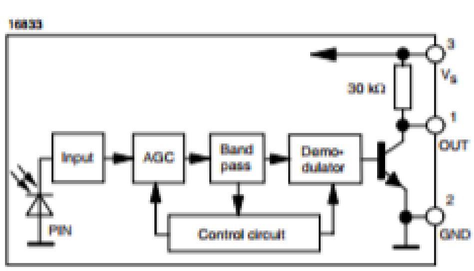
The demodulated output signal can be directly decoded by a microprocessor. The TSOP382.. is compatible with all common IR remote control data formats. The TSOP384.. is optimized to suppress almost all spurious pulses from energy saving fluorescent lamps but will also suppress some data signals.

This component has not been qualified according to automotive specifications.

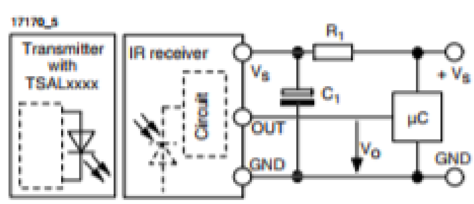
PARTS TABLE

CARRIER FREQUENCY	STANDARD APPLICATIONS (AGC2/AGC8)	VERY NOISY ENVIRONMENTS (AGC4)
30 kHz	TSOP38230	TSOP38430
33 kHz	TSOP38233	TSOP38433
36 kHz	TSOP38236	TSOP38436
38 kHz	TSOP38238	TSOP38438
40 kHz	TSOP38240	TSOP38440
56 kHz	TSOP38256	TSOP38456

BLOCK DIAGRAM



APPLICATION CIRCUIT



R_1 and C_1 are recommended for protection against EOS. Components should be in the range of 33 Ω < R_1 < 1 kΩ, C_1 > 0.1 µF.

www.vishay.com
140

Document Number: 81733
Rev. 1.3, 29-Jan-09

Figure 41: Infrared Sensor Datasheet (Page 1) [46]

ANALOG DEVICES

Low Voltage Temperature Sensors

TMP35/TMP36/TMP37

FEATURES

Low voltage operation (2.7 V to 5.5 V)
Calibrated directly in °C
10 mV/°C scale factor (20 mV/°C on TMP37)
±2°C accuracy over temperature (typ)
±0.5°C linearity (typ)
Stable with large capacitive loads
Specified −40°C to +125°C, operation to +150°C
Less than 50 µA quiescent current
Shutdown current 0.5 µA max
Low self-heating
Qualified for automotive applications

APPLICATIONS

Environmental control systems
Thermal protection
Industrial process control
Fire alarms
Power system monitors
CPU thermal management

GENERAL DESCRIPTION

The TMP35/TMP36/TMP37 are low voltage, precision centigrade temperature sensors. They provide a voltage output that is linearly proportional to the Celsius (centigrade) temperature. The TMP35/ TMP36/TMP37 do not require any external calibration to provide typical accuracies of ±1°C at +25°C and ±2°C over the −40°C to +125°C temperature range.

The low output impedance of the TMP35/TMP36/TMP37 and its linear output and precise calibration simplify interfacing to temperature control circuitry and ADCs. All three devices are intended for single-supply operation from 2.7 V to 5.5 V maximum. The supply current runs well below 50 µA, providing very low self-heating—less than 0.1°C in still air. In addition, a shutdown function is provided to cut the supply current to less than 0.5 µA.

The TMP35 is functionally compatible with the LM35/LM45 and provides a 250 mV output at 25°C. The TMP35 reads temperatures from 10°C to 125°C. The TMP36 is specified from −40°C to +125°C, provides a 750 mV output at 25°C, and operates to 125°C from a single 2.7 V supply. The TMP36 is functionally compatible with the LM50. Both the TMP35 and TMP36 have an output scale factor of 10 mV/°C.

FUNCTIONAL BLOCK DIAGRAM

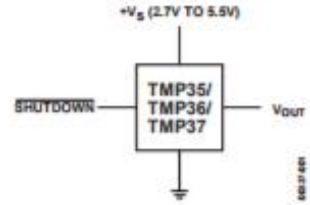


Figure 1.

PIN CONFIGURATIONS

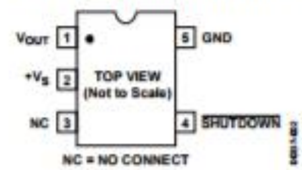


Figure 2. RJ-5 (SOT-23)

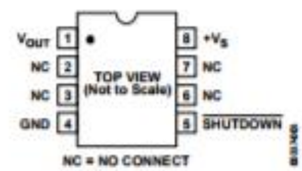


Figure 3. R-8 (SOIC_N)

Figure 4. T-3 (TO-92)

The TMP37 is intended for applications over the range of 5°C to 100°C and provides an output scale factor of 20 mV/°C. The TMP37 provides a 500 mV output at 25°C. Operation extends to 150°C with reduced accuracy for all devices when operating from a 5 V supply.

The TMP35/TMP36/TMP37 are available in low cost 3-lead TO-92, 8-lead SOIC_N, and 5-lead SOT-23 surface-mount packages.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 www.analog.com
Fax: 781.461.3113

Figure 42: Temperature Sensor – TMP36 Datasheet (Page 1) [47]

LED MIXED PACK sparkfun.com

COLOR	BRIGHTNESS	WAVELENGTH	VDROP
BASIC RED	150 - 200mcd	620 - 625nm	2.0 - 2.4 V

Figure 43: LED – Red Information Sheet [48]

APPENDIX D: 3D PRINTER SPECIFICATIONS AND DETAILS

Trust your designs to a world-class, professional 3D Printer.

PRODUCT SPECIFICATIONS

Model Material:
P430 ABSplus in ivory, white, blue, fluorescent yellow, black, red, nectarine, olive green, gray

Support Material:
SR-30 Soluble

Build Size:
12.7 x 12.7 x 12.7 cm (5 x 5 x 5 in)

Layer Thickness:
.17 mm (.007 in)

Workstation Compatibility:
Windows® 7 and 8

Size and Weight:
Mojo 3D Printer:
63 x 45 x 53 cm (25 x 18 x 21 in)
27 kg (60 lbs)
WaveWash 55 Support Cleaning System:
33.5 x 33.5 x 33.5 cm (13.2 x 13.2 x 13.2 in)
5 kg (12 lbs)
Maximum Size Part(s): 12.7 x 12.7 x 12.7 cm (5 x 5 x 5 in)
Volume Capacity: 3.78 L (1 gallon)

Power Requirements:
Mojo 3D Printer:
100-127 VAC, 6A, 60 Hz or 220-240 VAC, 2.5A, 50 Hz
WaveWash 55 Support Cleaning System:
100-120 VAC, 7A, 60 Hz or 220-240 VAC, 4A, 50 Hz

Regulatory Compliance:
Mojo 3D Printer: CE / TUV / KCC / RoHS / WEEE
WaveWash 55 Support Cleaning System: CE / TUV / RoHS / WEEE

Special Facility Requirements:
Mojo 3D Printer: None
WaveWash 55 Support Cleaning System: None

Mojo 3D Print Pack

Packaged to perfection. The Mojo 3D Print Pack gives you everything you need to start making quality, durable 3D models – right out of the box.

Includes:

- Mojo 3D Printer
- Print Wizard and Control Panel software
- WaveWash 55 Support Cleaning System
- Start-up supplies*

**Start-up supplies include one starter QuickPack Print Engine of ivory P430 ABSplus modeling material, one starter QuickPack Print Engine of SR-30 soluble support material, Ecoworks Tablets, two modeling bases and more.*

Mojo Makes 3D Printing As Easy As 1-2-3.

1. Create your idea. Design your idea in 3D CAD software and then just click print." Print Wizard software processes your CAD program's STL output and orients your 3D model with real-time auto-packing, auto-scaling and 3D thumbnail views.
2. Print your 3D model. The Mojo print head glides quickly and quietly above the build chamber, using FDM Technology to build your 3D model and its support material, layer by layer. The Mojo Control Panel provides print job status, including estimated printing time.
3. Remove support material. Once you've placed your printed model and Ecoworks Tablet into the compact WaveWash 55 Support Cleaning System, internal agitation within the covered carafe dissolves the soluble support material away quickly and quietly.

stratasys

HEADQUARTERS
7665 Commerce Way,
Eden Prairie, MN 55344
+1 800 801 6491 (US Toll Free)
+1 952 937-3000 (Intl)
+1 952 937-0070 (Fax)

2 Holtzman St., Science Park,
PO Box 2496
Rehovot 76124, Israel
+972 74 745 4000
+972 74 745 5000 (Fax)

E info@stratasys.com / STRATASYS.COM
ISO 9001:2008 Certified

Figure 44: Stratasys Mojo Printer Specifications [32]

Table 20: UltraSonic LLC Ultra 2000 – 5.2 Gal. [34]

Specifications	Ultra 2000
Machine Dimensions	21.5”L x 13”W x 12”H
Basket Frame	18.25”L x 10”W x 3.9” D
Working Depth	5.5”
Volume	5.2 Gal.
UltraSonic Power	400 W
Electrical	120 V
Heater	500 W

Advanced Technologies Consultants, Inc.

110 W. Main St., P.O. Box 905
Northville, MI 48167
Home Office Phone: (800) 348-8447
Home Office Fax: (248) 348-3040

Price Quotation
Date
MOJO Consumables

Prices valid for 60 days
Prices and packages are subject to change without notice

PREPARED FOR:

Consultant	Terms	Delivery	FOB Point
	Net 30 Days	7-15 Days ARO	Minnesota
	Education Materials Agreement Pricing		

Item#	Qty	Description	Part#	Unit Cost	Extended
		Mojo QuickPack Print Engines - Color			
1		P430 Model Ivory	350-80100	$200.00	
2		P430 Model Black	350-80101	$200.00	
3		P430 Model Steel Gray	350-80102	$200.00	
4		P430 Model Red	350-80103	$200.00	
5		P430 Model Blue	350-80104	$200.00	
6		P430 Model Nectarine	350-80105	$200.00	
7		P430 Model Yellow	350-80106	$200.00	
8		P430 Model Olive Green	350-80107	$200.00	
9		P430 Model White	350-80108	$200.00	
		Support Material			
10		Mojo QuickPack Print Engine - SR-30 Support (80 ci/1311 cc)	350-80200	$399.00	
		Bases			
11		Mojo Modeling Bases 5 x 5 in (12.7 x 12.7 cm) - case of 24	350-10000	$70.00	
		Material Packages			
12		Mojo Education Material Pkg.	350-10100	$669.00	
		includes:			
		1 ea Mojo Model QuickPack Print Engine (Ivory)			
		1 ea Mojo Support QuickPack Print Engine			
		3 ea Mojo Modeling Bases (case of 24)			

Sub-Total	**$0.00**
Shipping & Handling	
Grand Total	**0.00**

Figure 45: Example Order Form for Mojo Materials to Advanced Technologies Consultants, Inc. [49]

Advanced Technologies Consultants, Inc.

110 W. Main St., P.O. Box 905
Northville, MI 48167
Home Office Phone: (800) 348-8447
Home Office Fax: (248) 348-3040

Price Quotation
12/3/2015
SST 1200es Consumables

Prices valid for 60 days
Prices and packages are subject to change without notice

PREPARED FOR:
Dr. Sandra Metzler
Ohio State University
201 W. 19th Ave | W296N Scott Lab
Columbus, OH 43210

Consultant	Terms	Delivery	FOB Point
Chad Whited	Net 30 Days	7-15 Days ARO	Minnesota
	Education Materials Agreement Pricing		

Item#	Qty	Description	Part#	Unit Cost	Extended
		Model Material			
1		Ivory - P-430 ABSplus Model Material	340-21200	$130.00	
2		White - P-430 ABSplus Model Material	340-21201	$130.00	
3		Black - P-430 ABSplus Model Material	340-21202	$130.00	
4		Dark Grey - P-430 ABSplus Model Material	340-21203	$130.00	
5		Olive Green - P-430 ABSplus Model Material	340-21208	$130.00	
6		Florescent Yellow - P-430 ABSplus Model Material	340-21207	$130.00	
7		Blue P-430 ABSplus Model Material	340-21205	$130.00	
8		Red - P-430 ABSplus Model Material	340-21204	$130.00	
9		Nectarine - P-430 ABSplus Model Material	340-21206	$130.00	
		Soluble Support Material			
10		Soluble Support Material - P400-SR	340-30200	$250.00	
		Bases			
11		10" x 10" Modeling Bases (case of 24)	340-00300	$125.00	
		Soluable Concentrate			
12	1	Soluble Concentrate (case of 12)	300-00600	$149.00	$149.00

Sub-Total	**$149.00**
Shipping & Handling	**$20.00**
Grand Total	**$169.00**

Figure 46: Example Order Form for Soluble Concentrate to Advanced Technologies Consultants, Inc. [50]

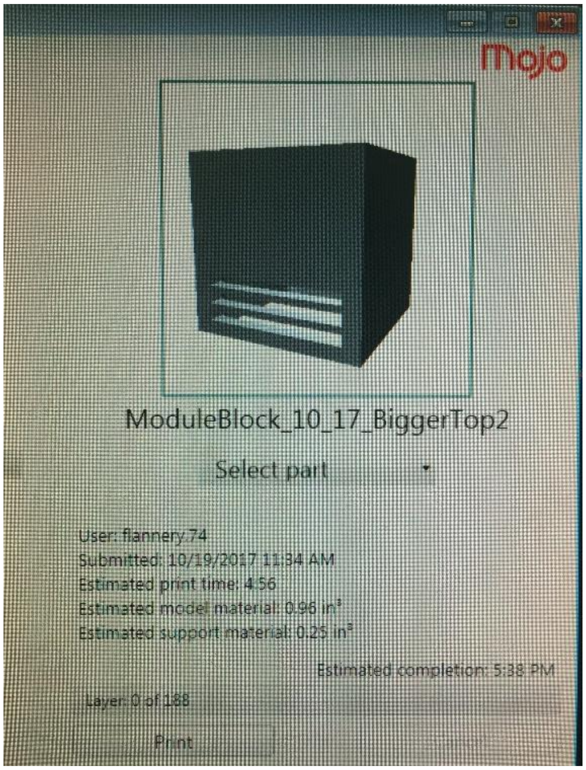


Figure 47: Mojo Control Panel' – Modules

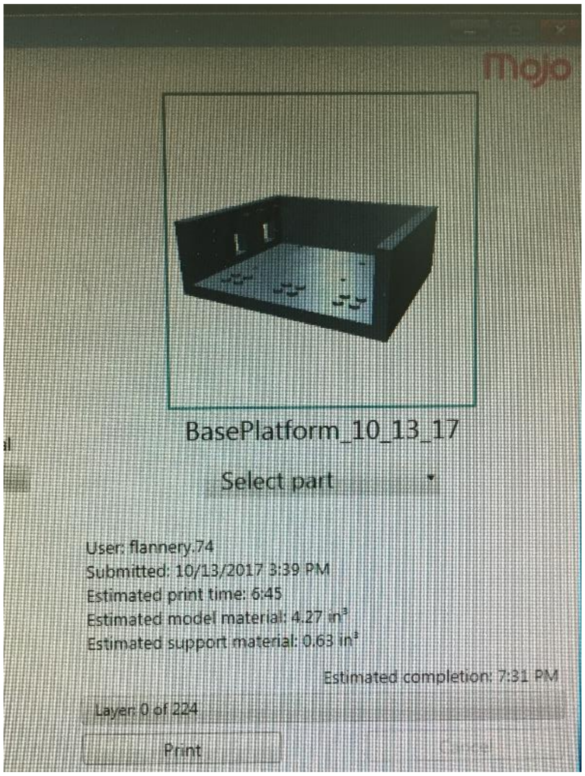


Figure 48: 'Mojo Control Panel' – Main Platform

PRINTING

Print Technology: Fused Deposition Modeling

Build Volume: 4 L x 5 W x 5 H in [10.1 L x 12.6 W x 12.6 H cm]; 98 in3 [1603 cm3]

Minimum Layer Resolution: 100 microns [0.0039 in]

Maximum Layer Resolution: 400 microns [0.0157 in]

Minimum Feature Size: 0.0315 in [0.8 mm]

Build Plate Leveling: Factory Leveled

Build Surface: Grip Surface

Filament Diameter: 0.069 in [1.75 mm]

Extruder Compatibility: Smart Extruder+

Filament Compatibility:

MakerBot PLA Filament Small Spool 0.2 kg [0.5 lb]

Nozzle Diameter: 0.4 mm [0.015 in]

Operating Sound Level: 58.8 dBA RMS

Print File Type: .Makerbot

CAMERA

Camera Resolution: 640 x 480

SOFTWARE

Software Bundle: MakerBot Print Software, MakerBot Mobile

Operating Systems for MakerBot Print: Windows (7,10), Mac Os X (10.9+), No Linux Support

Supported File Types:

Mac: MakerBot (.makerbot), STL (.stl)

Windows: MakerBot (.makerbot), STL (.stl), SolidWorks (.sldprt, .sldasm), InventorOBJ (.ipt, .iam), IGES (.iges, .igs), STEP AP203/214 (.step, .stp), CATIA (.CATPart, .CATProduct), Wavefront Object (.obj), Unigraphics/NX (.prt), Solid Edge (.par, .asm), ProE/Creo (.prt, .prt., .asm, .asm.), VRML(.wrl), Parasolid (.x_t, .x_b)

ELECTRICAL

Power Requirements: 100–240 VAC; 0.75-0.45 A; 50/60 Hz

Connectivity: Wi-Fi, USB

Figure 49: Replicator Mini Compact 3D Printer Specifications [41]

Continued

Figure 49 Continued

TEMPERATURE

Ambient Operating Temperature: 15 to 26°C [60 to 78°F]

Storage Temperature: 0 to 38°C [32 to 100°F]

SIZE & WEIGHT

Product Dimensions: 11.6 L X 13.8 W X 15.0 H in [29.5 L X 34.9 W X 38.1 H cm]

Product Weight: 20.5 lbs [9.3 kg]

Shipping Box: 15.8 L X 17.9 W X 20.6 H in [40.0 L X 45.4W X 52.2 H cm]

Shipping Weight (includes Accessory Kit): 26.8 lbs [12.2 kg]

APPENDIX E: REWORKED FUNCTIONAL REQUIREMENTS

Table 21: Reworked Functions, Requirements, and Scores for the Prototype (Entire Table)

# Assigned to Initial Function	Functions	Requirements	Score
N/A	Provide a format for the system to be used.	The system, along with instructions, should be open source through a website.	0: A website with a description of the system is created, but the STL files of the system are not openly available. 1: A website with a description of the system is created that has the STL files for the system openly available. 2: A website that has a description, the STL files of the system, a link to additional hardware needed, and instructions on how to assemble the system is created.
1	Create modules that reliably encase sensors and actuators.	The modules are rigid enough to pick up and install/mount onto platform while retaining their shape.	0: The modules cannot retain their shape when picked up. 1: The modules retain their shape when being held, but deform when force is applied to install them onto the platform. 2: The modules retain their shape when being handled and installed/removed from the platform.
3	Implement sensors that are compatible with the required software.	The sensors and actuators must cooperate with the input/output capabilities of the ATmega328P microcontroller (operate at 5 volts and 20 mA; supports PWM, I2C, and SPI communication).	0: The sensors used to make the modules are not compatible with the ATmega328P. 1: Some of the components used to make the modules are compatible with the ATmega328P. 2: Every component used to make the modules is compatible with the ATmega328P.
4	Create modules that can be easily connected to the platform.	The modules can be connected to the main platform with normal hand force and no additional tooling.	0: The modules cannot be connected to the main platform. 1: The modules can be connected to the main platform with additional tooling. 2: The modules can be connected to the main platform with normal hand force.
5	Create modules that can be easy disconnected to the platform.	The modules can be disconnected to the main platform with normal hand force and no additional tooling.	0: The modules cannot be disconnected to the main platform. 1: The modules can be disconnected to the main platform with additional tooling. 2: The modules can be disconnected to the main platform with normal hand force.
6	Create modules that can be easy reconnected to the platform.	The modules can be reconnected to the main platform with normal hand force and no additional tooling.	0: The modules cannot be reconnected to the main platform. 1: The modules can be reconnected to the main platform with additional tooling. 2: The modules can be reconnected to the main platform with normal hand force.

Continued

Figure 49 Continued

7	Create a reliable connection between the modules and platform.	The entire device can be rotated, moved, and dropped from a height of one foot without any parts becoming disconnected.	0: The modules become disconnected from the platform when the device is moved or rotated. 1: The modules stay attached to the platform when the device moves horizontally, but disconnect when the device it rotated or dropped. 2: The modules stay attached to the platform when the device is moved, rotated, and dropped.
8	The platform should be open to housing multiple modules at once.	The platform can support at least five modules at one time.	0: The platform can support one module at a time. 1: The platform can support two through four modules at a time. 2: The platform can support at least five modules at a time.
9	Connect to power and ground.	Each module needs to be able to be connected to a power source and ground on the Arduino platform without lose wiring.	0: None of the modules can be connected to power or ground. 1: Each module can be connected to power and ground through wires (that could potentially disconnect). 2: Each module can be connected to power and ground in a discrete fashion that does not involve any lose wiring.
12	The new systems will run using existing software.	The new system will use Arduino software for coding.	0: No code can successfully be uploaded to the device. 1: Code from an open source software that is not Arduino can be successfully uploaded to the device. 2: Arduino software can be successfully uploaded to the device.
14	Any enclosure should not interfere with any connections.	Any enclosure should provide clearance or openings for all necessary connections.	0: The enclosure prohibits all necessary connections. 1: The enclosure provides enough of an opening for all connections to be made, if two or less modules are in use. 2: The enclosure provides enough of an opening for all necessary connections to be made, independent of how many modules are in use.
Optional:			
15	There should be an enclosure for the system.	The microcontroller and all other hardware should be globally encased.	0: All components of the platform are exposed. 1: Most of the platforms components are hidden within a case. 2: All of the platforms components are hidden within a case.

APPENDIX F: ENGINEERING DRAWINGS

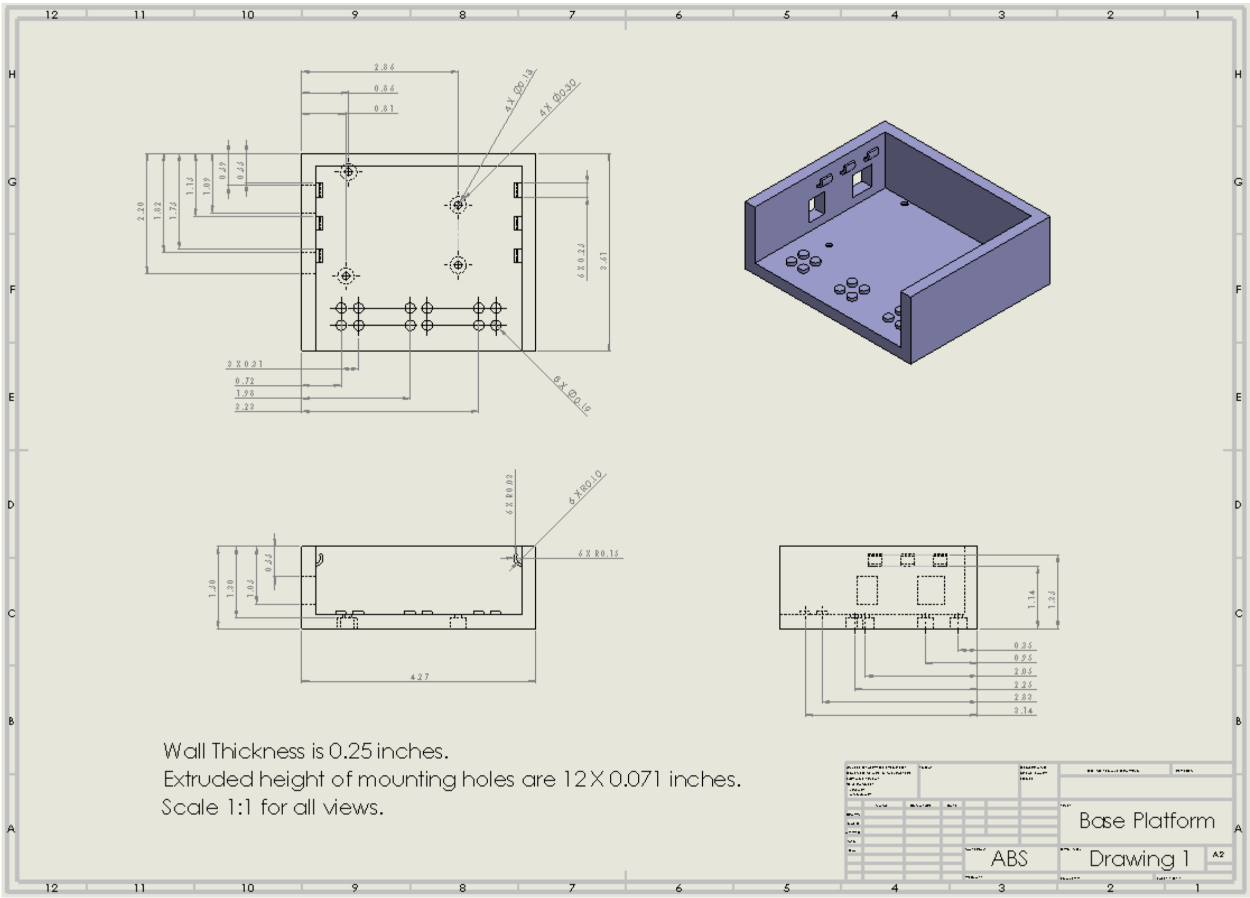


Figure 50: Engineering Drawings for Main Platform

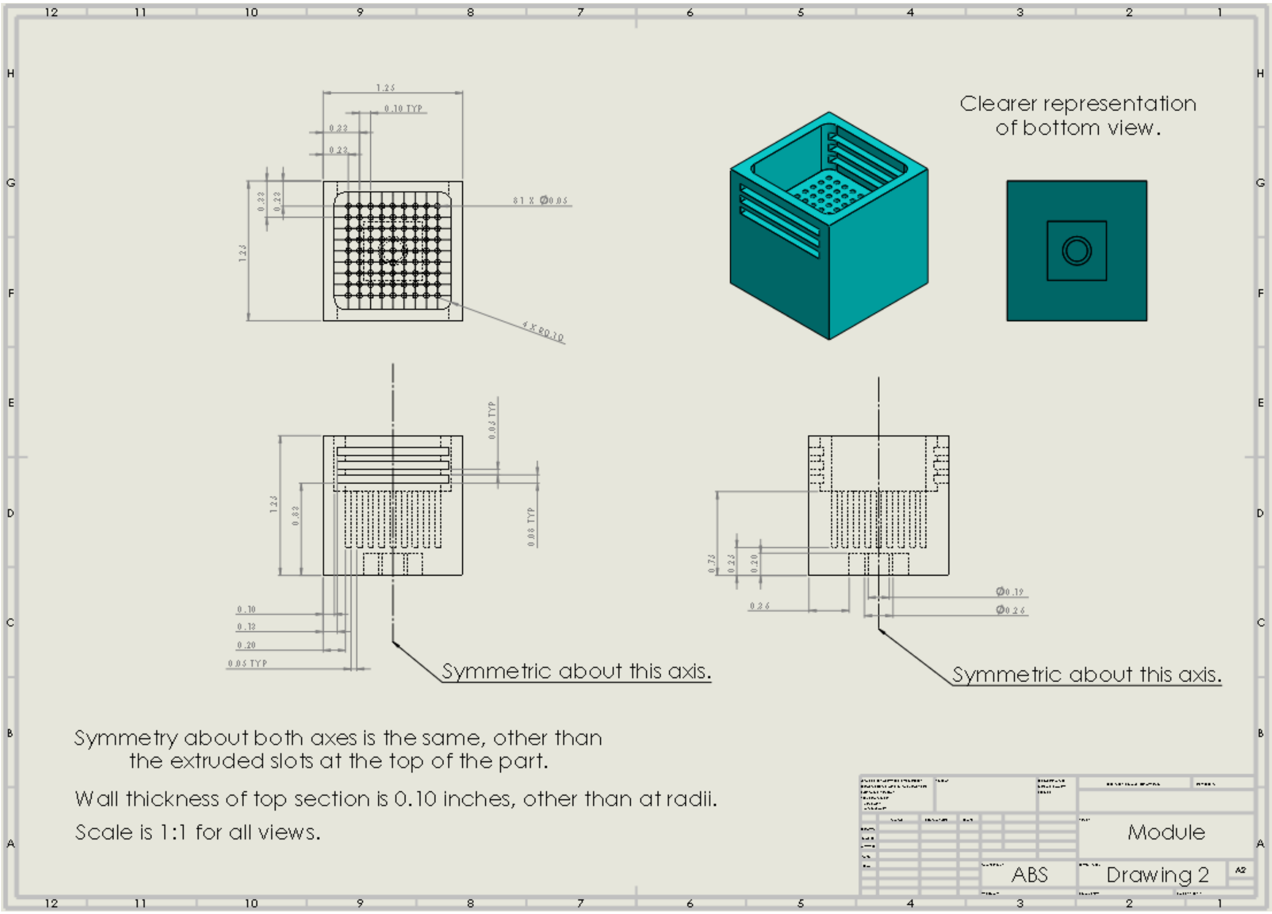


Figure 51: Engineering Drawings for Module


```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  Most Arduinos have an on-board LED you can control. On the Uno and
  Leonardo, it is attached to digital pin 13. If you're unsure what
  pin the on-board LED is connected to on your Arduino model, check
  the documentation at http://www.arduino.cc

  This example code is in the public domain.

  modified 8 May 2014
  by Scott Fitzgerald
 */


// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

Figure 52: Arduino Code "Blink"