



PROJECT PLAN

Team May15-03

Shawn LaGrotta	(lagrotta@iastate.edu)
Jacob Mayer	(jdmayer@iastate.edu)
Trevor Boone	(tdboone@iastate.edu)
Matt Eckes	(mweckes@iastate.edu)
Jacob Schulz	(jschulz@iastate.edu)

TABLE OF CONTENTS

1	List of Figures	2
2	Definitions	2
3	Problem Statement	3
4	Design Requirements	3
4.1	Functional Requirements	3
4.2	Non-Functional Requirements	3
5	Operating Environment.....	4
6	Market and Literature Survey	4
6.1	Current Industry Solutions	4
7	System Description	5
8	System Block Diagram.....	6
8.1	Hardware.....	6
8.2	Software	7
9	Concept Sketch.....	8
10	User Interface Description	9
11	Deliverables.....	10
11.1	First Semester	10
11.2	Second Semester.....	10
12	Work Plan.....	11
12.1	Resource Requirements	11
12.1.1	Provided By Client	11
12.1.2	Purchased by Client.....	11
12.2	Project Schedule.....	12
12.2.1	First semester	12
12.2.2	Second semester	12
13	Testing.....	13
13.1	Software Communication	13
13.2	File Parsing Gerber Files	13
13.3	Motor Controls.....	13
13.4	X, Y, and Z Motion	13
13.5	Transition Time	14
13.6	Locating Test Points	14
13.7	Risks.....	14

1 LIST OF FIGURES

Figure 1: Bed of nails PCB tester4
 Figure 2: SPEA S2 Flying Probe Testers4
 Figure 3: System Block Diagram6
 Figure 4: Software Block Diagram7
 Figure 5: Image of overall system8
 Figure 6: Images of robot LH and RH sides8
 Figure 7: Image of jig LH and RH sides9
 Figure 8: GUI Mockup 10
 Figure 9: Fall Semester Schedule 12
 Figure 10: Spring Semester Schedule..... 12

2 DEFINITIONS

Term	Description
<i>HDD</i>	Hard Disk Drive
<i>HDD Jig</i>	Device used to hold the HDD in place
<i>PCBA</i>	Printed Circuit Board Assembly
<i>Oscilloscope</i>	Device used to view electrical signals
<i>LCD</i>	Liquid-Crystal Display
<i>LED</i>	Light Emitting Diode
<i>HTTP</i>	Hypertext Transfer Protocol
<i>G-Code</i>	Numerical control programming language used in the controlling automated machine tools
<i>Gerber Files</i>	The standard used by printed circuit board industry software to describe the printed circuit board images: copper layer, solder mask, legend, etc.
<i>Drill Files</i>	Defines hole locations, tool numbers and finished hole sizes (X&Y coordinates) for PCBA design
<i>Arduino</i>	A single board microcontroller
<i>Raspberry Pi</i>	A single board computer
<i>RAMPS</i>	RepRap Arduino Mega Pololu Shield (controller shield)
<i>UI</i>	User Interface
<i>GUI</i>	Graphical User Interface
<i>Stepper Motor</i>	Motors which operate only in discrete increments of rotation (controlled by electrical signals)
<i>Carriage</i>	The moving middle assembly on the x-axis of a RepRap which holds the probe
<i>Via</i>	Electrical connection between layers on a circuit board
<i>GPIO</i>	General Purpose Input/Output

3 PROBLEM STATEMENT

Hard disk drives (HDDs) are complicated systems that require significant amount of investigation and debugging during development. One important tool used in this work is using oscilloscopes to capture and examine waveforms that describe the behavior of the device. The waveforms are on specific test points, which are vias on the HDD PCBA. A typical HDD can have hundreds of these vias.

The goal of this project is to create a robotic oscilloscope probe positioning system to make examining these signals 1: easier, and 2: possible from a remote location. This will be accomplished by creating a robotic system capable of positioning an oscilloscope probe onto any of the vias on the drive.

The robotic system must be capable of positioning an oscilloscope probe anywhere on the 2.5" form factor drive and contacting a via on the PCBA. The system must be able to provide enough pressure on the probe to create a good electrical connection without damaging the HDD under test or the oscilloscope probe.

4 DESIGN REQUIREMENTS

4.1 FUNCTIONAL REQUIREMENTS

- There shall be a probe positioning device and a control system to control it.
- The probe positioning device shall:
 - Consistently hold a 2.5" form factor HDD in the same position
 - Hold, move, and manipulate the LeCroy Oscilloscope Probes
 - Place the probes on HDD PCBA test points with enough force to make a good electrical contact, yet not enough to damage the PCBA
 - Take commands from the control system and act accordingly
 - Have a calibration method that ensures accurate placement of the probes on the PCBA
 - Allow probes to be grounded to the test fixture and HDD
 - Support positioning of at least one probe
- The control system shall:
 - Be remotely accessible over the HGST intranet
 - Send commands to the probe positioning device (client)
 - Receive commands from the remote operator over the intranet (web server)
 - Translate commands from remote operator to change the probe position

4.2 NON-FUNCTIONAL REQUIREMENTS

- The device shall move the probe to any point on the board within 60 seconds
- The device shall have LEDs that show the operational status of the device
- The device shall have an LCD to display the coordinates of the probe and operational status
- The robot will return status information back to the client
- It shall take up a volume less than 4 ft³
- The device shall be easily maintained and built from standard parts when possible
- The device should support a manual operation mode

5 OPERATING ENVIRONMENT

The robot will be placed on a debugging cart in a HGST laboratory. The lab is climate controlled, therefore the testing platform can be open to the environment. Design requirements for harsh conditions do not need to be considered. The client's interaction with the GUI is through the HGST's intranet and will need to be non-intrusive as possible to the company network.

6 MARKET AND LITERATURE SURVEY

6.1 CURRENT INDUSTRY SOLUTIONS

Bed of nails testing: Uses a single, flat collection of pogo pins placed in a matching pattern of the circuit board's nodes. Presses down onto the board and all pins come in contact with nodes. Individual pins are wired to another module that aggregates, measures, and gives the readings of the pins to the operator. This method has a long turnaround time for different layouts (programming and setup) and is very expensive.

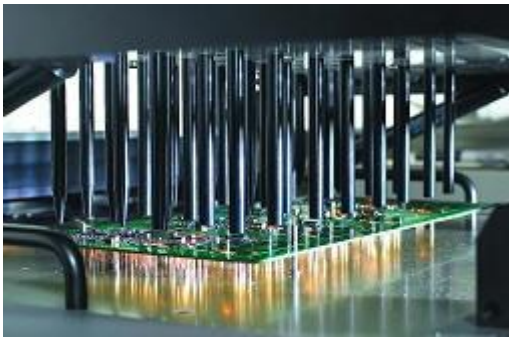


Figure 1: Bed of nails PCB tester

Flying probe: Uses a single pogo pin moved by an electromechanical system like a coordinate measurement machine. Simply moves the pin to the desired node on the circuit board and performs a measurement. It sacrifices the sheer performance of the bed-of-nails method for extremely quick turnaround (virtually no setup and no programming) and low cost.

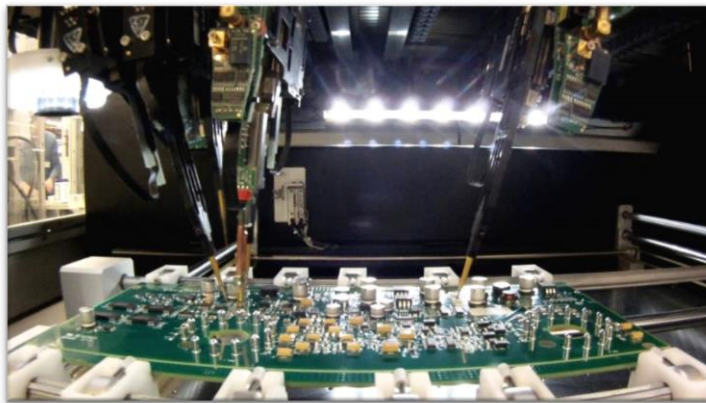


Figure 2: SPEA S2 Flying Probe Testers

7 SYSTEM DESCRIPTION

In order to meet the accuracy requirements and still stay within budget, we decided to modify the flying probe test method by leveraging 3D printing technology to accurately move the probe around the HDD. The downside to this decision is that we will only be able to test one via at time.

To use the system the user will manually place a 2.5" HDD into a jig and then plug in the HDD development electrical harness. Next, the user will manually calibrate the robot through a LCD UI. The user will then access the robot remotely through the HGST Intranet using a web browser. Upon logging in, the user will then import the drill and Gerber board files into the web application. Next the Raspberry Pi web server will parse the file; thus finding all the via locations.

Next, the web application will plot all test via locations onto a template HDD in the GUI. Once the file is parsed and the test points are displayed, the user can then proceed to select a via to test. The web server will then take the test point via and current probe coordinates to generate the corresponding G-code commands. The Arduino receives these commands, converts the commands to a waveform, and then sends the signal to the RAMPS 1.4 motor board. The RAMPS board sends power to the stepper for the correct duration.

Upon receiving a test command, the Raspberry PI will activate the status LEDs, and indicate others that the robot is in use. The current coordinates of the oscilloscope probe are displayed on the LCD by the RAMPS 1.4 shield. Operational status messages will be sent back through the web server to the client at all times.

8 SYSTEM BLOCK DIAGRAM

8.1 HARDWARE

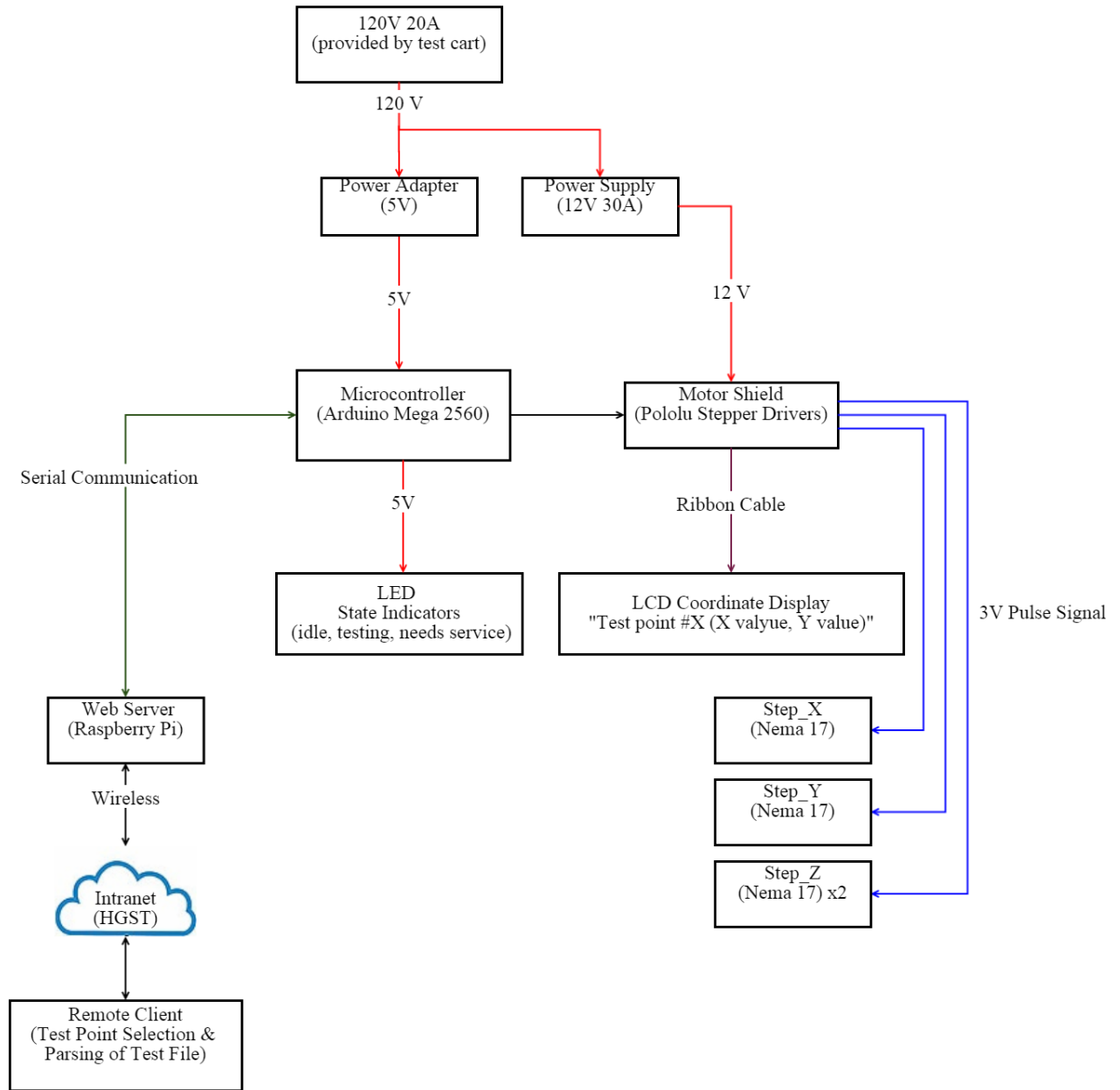


Figure 3: System Block Diagram

Software Modules and Data Flow

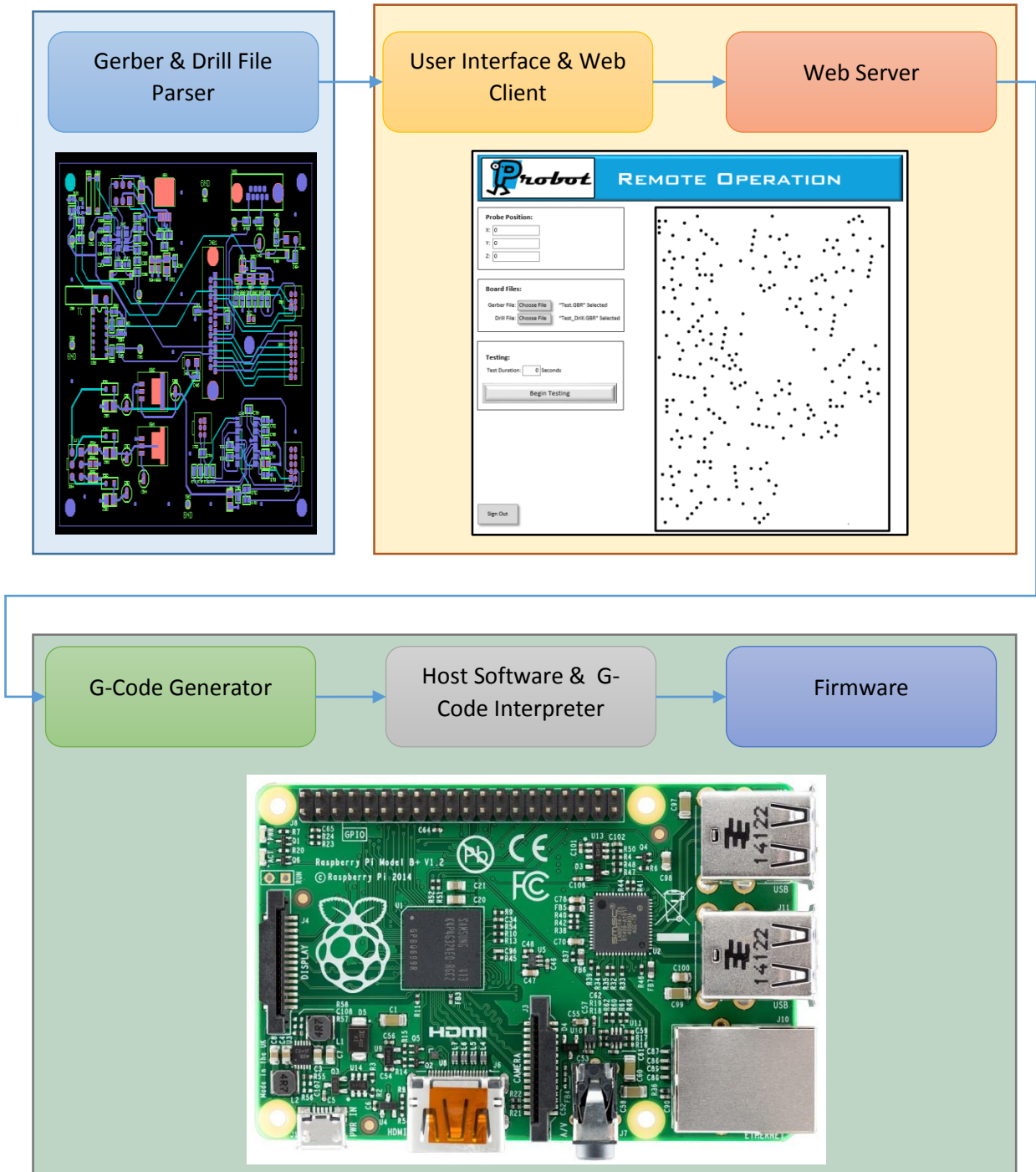


Figure 4: Software Block Diagram

9 CONCEPT SKETCH

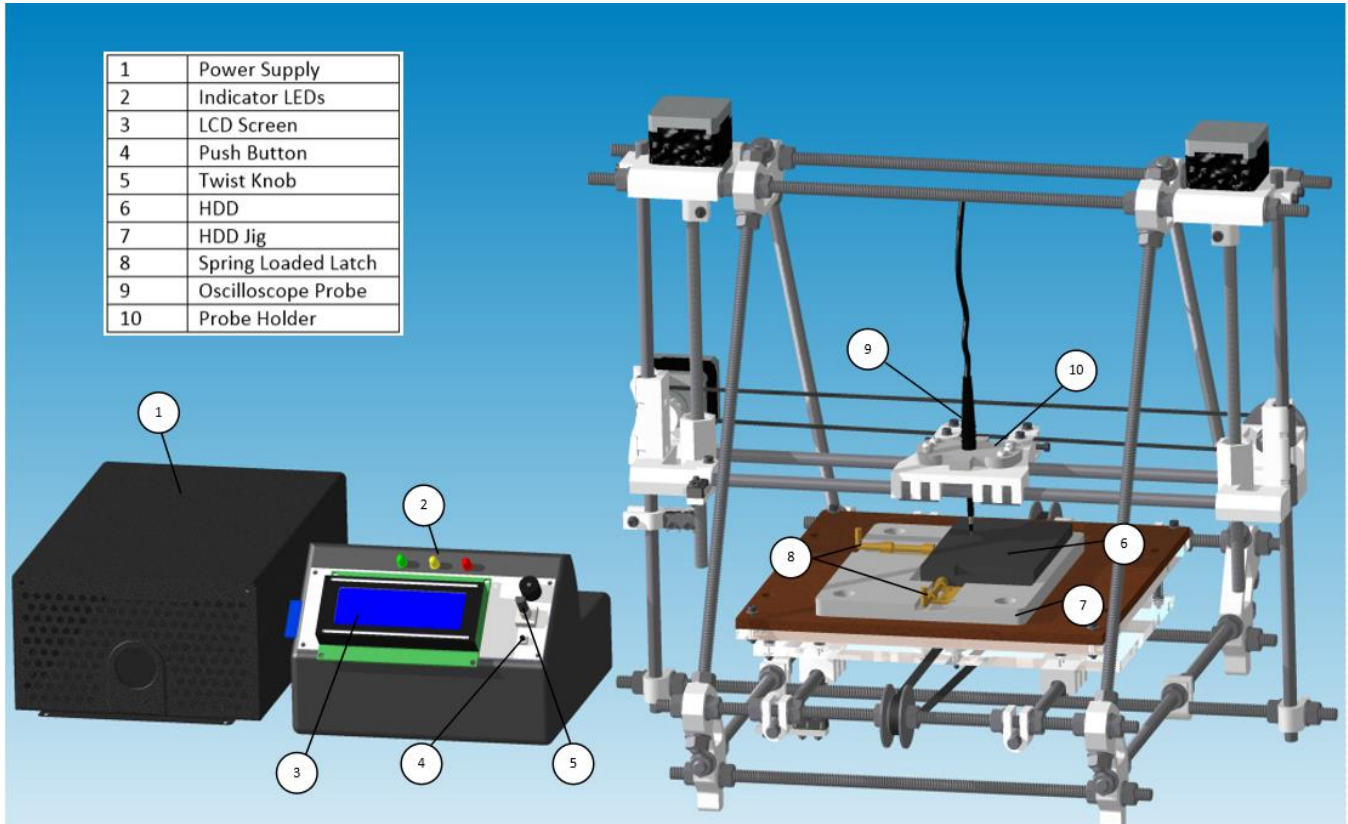


Figure 5: Image of overall system

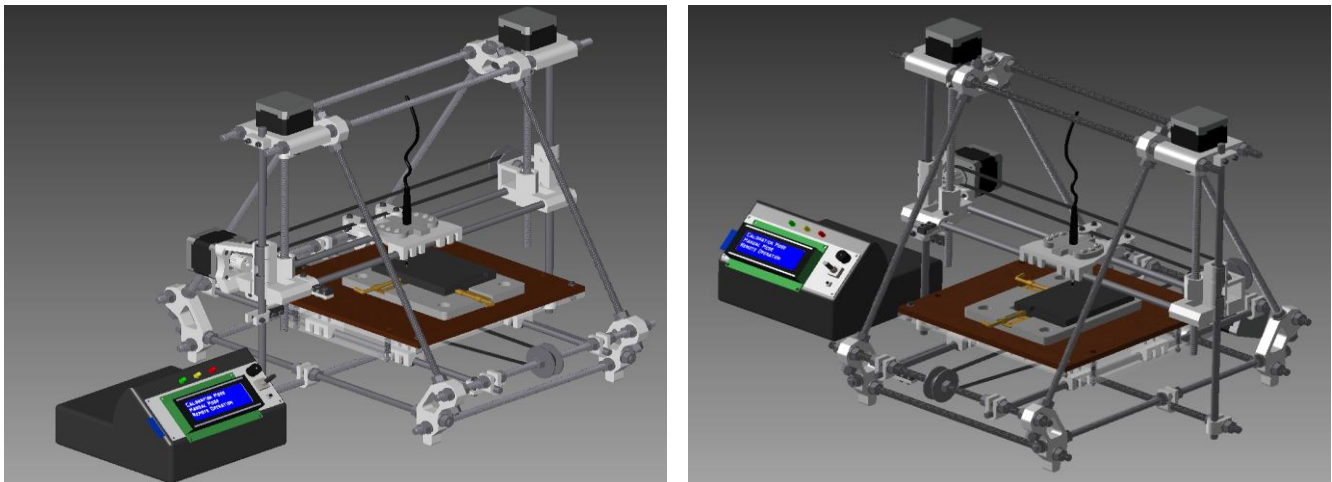


Figure 6: Images of robot LH and RH sides

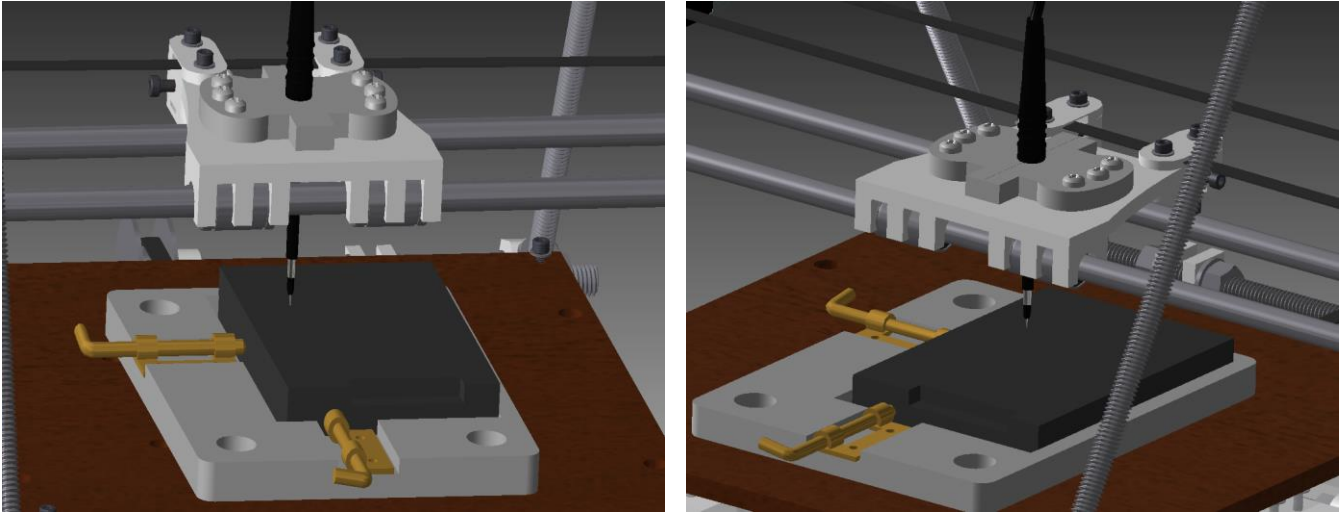


Figure 7: Image of jig LH and RH sides

10 USER INTERFACE DESCRIPTION

For the interface we decided to build a web application and serve it off of a computer that is connected by a serial connection to the hardware controller. A web interface has an advantage over native applications, in the fact that there is already a rich ecosystem of tools to help develop web apps. This allows for a solution that is cross platform compatible and doesn't require the user to install any extra software on their computer. On the implementation level, the plan is to program the web server in python using the tornado library, as it is both lightweight and allows for the asynchronous handling of requests. This server will serve a simple HTML/CSS page with the GUI functionality implemented in JavaScript.

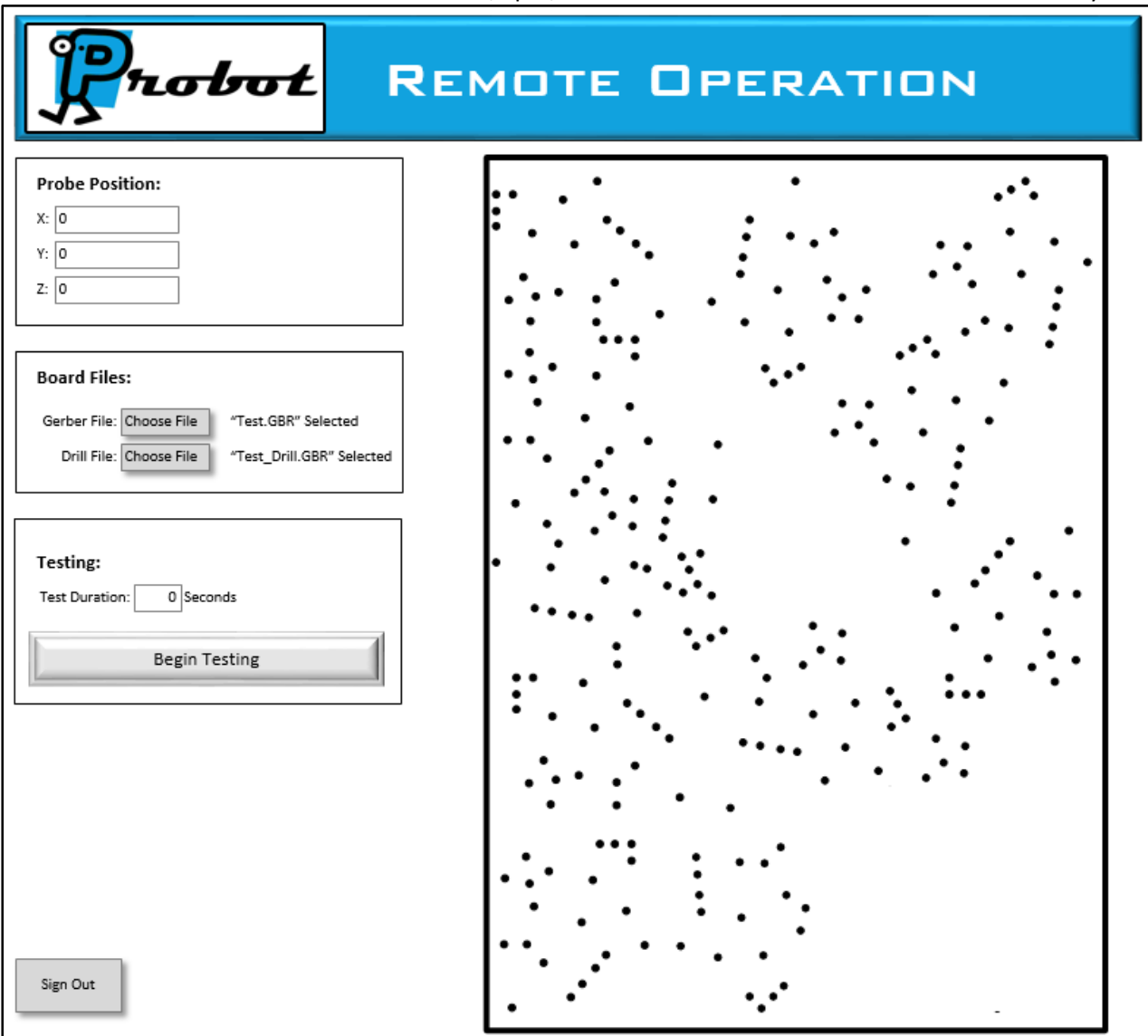


Figure 8: GUI Mockup

11 DELIVERABLES

11.1 FIRST SEMESTER

- CAD design for HDD testing robot
 - HDD jig
 - Probe holder
- Prototype of user interface
- Strategy for parsing Gerber and Drill files

11.2 SECOND SEMESTER

- Functioning prototype of HDD testing machine
- Web based interface to for movement of probes

- Full documentation for the machine and all related components
 - Be able to recover from software and hardware errors
 - Minimally intrusive to companies software

12 WORK PLAN

12.1 RESOURCE REQUIREMENTS

12.1.1 Provided By Client

HGST is to provide hard disk drives that are typically used during testing. The oscilloscope probe is also provided in order to accurately create a holder for it. In addition, Gerber board files will be provided so that the test point locations can be extracted and used for the generation of G-code. For testing we are using our own laptops, but the final version will run off a HGST laptop on the debugging bench.

12.1.2 Purchased by Client

12.1.2.1 3D Printer Kit

The printer kit will include all necessary components for X, Y, and Z translation. These components include, but are not limited to, stepper motors, motor belts, slide rails, motor controllers, and frame.

Estimated Cost: \$619.37

12.1.2.2 LCD Screen

A LCD screen will be used to provide information to the user, such as, current probe location, calibration protocol, and what test point the probe is currently traveling to. The user will also be able to see whether the robot is currently in operation through the LCD.

Estimated Cost: \$32.00

12.1.2.3 Web Server

A Raspberry Pi and Pi Wi-Fi will provide communication from the user to the robot controllers. The Wi-Fi capabilities will allow for remote control of the robot.

Estimated Cost: \$60.00

=

12.2 PROJECT SCHEDULE

12.2.1 First semester

First Semester

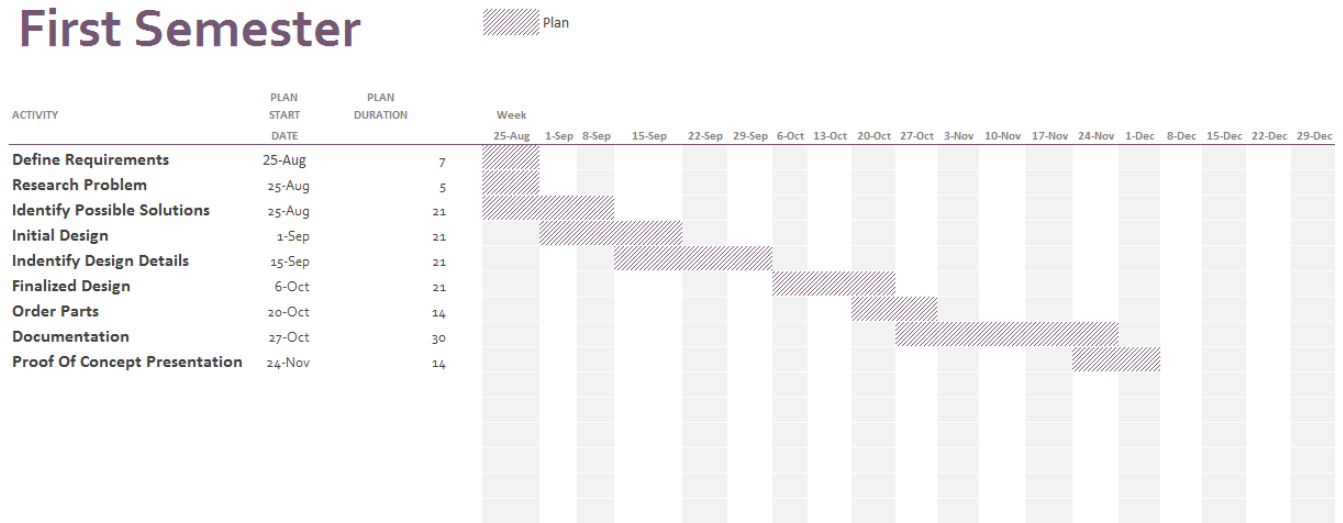


Figure 9: Fall Semester Schedule

12.2.2 Second semester

Second Semester

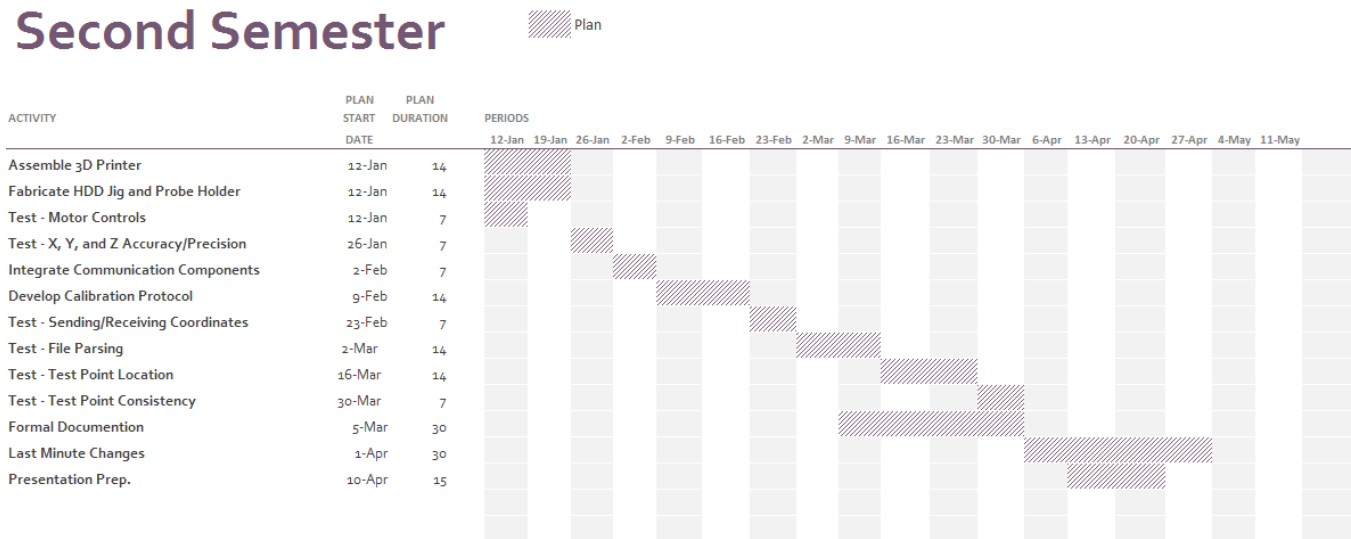


Figure 10: Spring Semester Schedule

13 TESTING

13.1 SOFTWARE COMMUNICATION

Without communication between the client and the server, the user's command will never make it to the robot. To test that the communication module is working correctly, first a connection will be obtained and verified. From there, the client and server agree to go into test mode in which each client command will be sent to the server, the server will note the command, and return a successful status to the client. This will ensure that communication between the two is correct and that the server can handle any requests the client is capable of sending.

13.2 FILE PARSING GERBER FILES

The success of this project lies in the ability to accurately locate test vias on a HDD. If we are unable to accurately locate the test points with regard to a reference point then the whole system fails. Due to the importance of this step, we must be able to show that the GUI accurately represents the test vias on the HDD. After the software generates a representation of the HDD with the test points located, the coordinates will be compared to the locations given in both the gerber file and the measured values.

13.3 MOTOR CONTROLS

Testing of motor controls is an important step in ensure all components are working properly. Before significant time is spent is mounting the motors to the robotic system, we first would like to check that they are operating correctly. RepRap software has an application, Pintrun, that allows for motor commands to be sent to the Arduino, and then to the specified motor (X, Y, or Z). For this testing procedure the motor control components (Arduino, RAMPS, 3 Nema 17 stepper motors) will be connected. Commands, sent through Pintrun, will control each individually and test for clockwise and counter clockwise motion. This test is simply to ensure that all motor controls components are working correctly and the Arduino motor pins are correctly identified.

13.4 X, Y, AND Z MOTION

Accuracy and precision in the X, Y, and Z direction is an absolute must for this project. The robotic system needs to provide significant reliability in order to be useful. For this reason extensive testing will be done to ensure both accuracy and precision. For this testing procedure, we will give the robot a command to travel in one cardinal direction. The movement will then be accurately measured using calipers. The actual moving distance will be compared to the expected value. To get a high degree of statistical importance a large sample size will be taken. This procedure will be done at a variety of lengths as well as compared against different speed ratings. Once the accuracy and precision are documented, we will be able to publish an optimum speed rating that will allow for accurate movement with limited traversing time.

13.5 TRANSITION TIME

A non-functional requirement the client expects is the movement from one test point to another to take no more than 60 seconds. The accuracy of the movement has priority over the speed requirement, but if we can show that the robot can move quickly from one test point to another, the robot's efficiency will greatly increase. For this test we will create a random list of test points that the robot must travel to. The time it takes the robot to move from one test point to another will be measured. Once a large sample size is taken the average will be published.

13.6 LOCATING TEST POINTS

The final test of the system is a complete test of the devices capabilities. A HDD will be placed in the HDD jig. The HDD Gerber files will be parsed and the GUI will indicate where the test points are located. The user will then select randomly a test point to measure. The robot's performance will then be measured on the basis of locating the test point and making sound electrical contact with the test via. We will do a large sample size in order to simulate the debugging process a HGST employee may encounter. We will then find an operational limit in how precise and accurate we can be over number of test points and recommend how many vias a user can accurately test before recalibrating the robot.

13.7 RISKS

- **Calibration:** It is crucial that the system stay calibrated as the user is likely to not be around the machine to fix the problem. Additionally, small calibration errors (< 1mm) cannot be tolerated as it would exceed minimum feature separation. However, our team has no experience with calibration methods and we don't know how well our equipment will cooperate in regards to both being calibrated and staying calibrated.
- **Accuracy:** The machine has to be accurate within less than 1 mm. This means that our algorithms, motors, and construction must meet strict tolerances that might be difficult to obtain.
- **Customization:** The foundational bulk of the work is going to be purchased in order to best utilize our time and improve the quality of our final solution. However, in order to build exactly what we need, there is some custom mechanical work that needs to be done. Due to the lack of practical mechanical engineering experience on our team, it is possible that problems will arise with our design that we did not foresee.

14 CONCLUSION

Overall, our robot is designed to leverage the highly accurate but cheap technology of 3D printing to design a HDD development tool. The robot will hold an HDD in a jig securely, all while an engineer at a remote location commands the robot to move the probe to different vias for debugging of the HDD. The via locations will be identified by parsing Gerber and Drill files and then displayed on a GUI so that the user can identify which via to test.