Smart Saw

Final Report

SD32

Brady Haugo

Long Que

Mitchell Kistner: Project Manager/Power

Ethan Bauman: Controls/Task Manager

David Kruse: Electrical/Power

Austin Mackedanz: Electrical/Controls

Lance Longhorn: Electrical

Patrick Pham: Electrical/Power

Jace Fedler: Electrical/Controls

Team Email: sdmay23-32@iastate.edu

Team Website: https://sdmay23-32.sd.ece.iastate.edu

Revised: 12/2/2022 Version 1

Executive Summary

Development Standards & Practices Used

Standard 1: 1679.1-2017 - IEEE Guide for the Characterization and Evaluation of Lithium-Based Batteries in Stationary Applications:
https://ieeexplore.ieee.org/document/8262521

This article is the guide to the IEEE standards on the safe use of Lithium Batteries. This project uses a pair of lithium batteries to power itself so this guide was helpful in the safe use of said batteries.

Standard 2: 1118.1-1990 - IEEE Standard for Microcontroller System Serial Control Bus
https://ieeexplore.ieee.org/document/159173

This article describes the IEEE standards for the use of microcontrollers in a variety of uses. Our project uses arduino's as control devices which is covered within this guide.

Summary of Requirements

Functionality

- Wireless control of branch saw, rotation, and saw clamp
- Improved user experience through use of controller

Resources

- Two sets of eight relay switches
- Arduino Uno and Arduino Mega
- A pair of Xbee transceivers and shields

Aesthetics

- Easy to understand design
    - Warning and danger indicators for components that can cause harm(jaws and chainsaw)
- Organized and covered wires/electronics

User Experiential

- Easy to understand and ergonomic controller
- Readily available safety features

Economic/market value

- Main chainsaw chassis made from plastic
    - Design will be 3d printed for the scope of this project
- Chainsaw is electric

Environmental

- Chainsaw is electric

UI

- Intuitive controls
- Easy to connect saw and controller

Applicable Courses from Iowa State University Curriculum

EE 333: Electronic Systems Design, CPRE 288: Embedded Systems I, EE 201: Electric Circuits, EE 230: Electric Circuits and Systems, EE 285: Problem Solving Methods and Tools for Electrical Engineering, EE 185: Introduction to Electrical Engineering and Problem-Solving I

New Skills/Knowledge acquired that was not taught in courses

Project documentation, project planning , building a BOM, material acquisition and research, and team communication.

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

BOM - (Bill of Materials)

IEEE - (Institute of Electrical and Electronics Engineers)

1 Team

1.1 Team Members

David Kruse

Mitchell Kistner

Austin Mackedanz

Jace Fedler

Lance Longhorn

Patrick Pham

Ethan Bauman

1.2 Required Skill Sets for Your Project

Software Programming, Electrical Circuit Design, CAD design, PCB Design and Controls.

1.3 Skill Sets covered by the Team

(for each skill, state which team member(s) cover it)

Software: Ethan Bauman, Austin Mackedanz, Jace Fedler

Circuit Design: Ethan Bauman, Mitchell Kistner, David Kruse, Jace Fedler

CAD Design: Lance Longhorn, Mitchell Kistner, Patrick Pham

PCB Design: Mitchell Kistner, Ethan Bauman, Austin Mackedanz, David Kruse

Controls: Ethan Bauman, Jace Fedler, Austin Mackedanz

1.4 Project Management Style Adopted by the team

Our project management style is an agile style that has weekly meetings with our client to discuss ideas. There is more continuous planning than front heavy planning.

1.5 Initial Project Management Roles

Mitchell Kistner : Project Manager and CAD Design

Ethan Bauman : Controls and Software

David Kruse : Circuits and PCB Design

Austin Mackedanz : Software and Circuit

Lance Longhorn: CAD Design and Circuit

Patrick Pham : CAD Design and Circuit

Jace Fedler : Controls and Software

2 Introduction

2.1 Problem Statement

-Our client currently has a prototype for a branch saw, a device that can clamp onto a branch and allows the user to saw off the branch from a safe distance. The problem that the client has brought to us is the saw currently is not as user friendly as he wants. He would like the saw to be upgraded so that the status of the clamp, saw, and the arm that moves the saw are all monitored. That data should be used to optimize the saw's performance, ie, safer cuts, more efficient use of battery. He would also like to greatly reduce the steps a user has to take in order to operate the saw safely and effectively.

2.2 Intended Users and Uses

The intended users of the product are currently in the commercial and consumer market. The product is targeted towards tree trimming businesses and crisis response organizations. Because of its compact and affordable size, it also provides a cheaper alternative to expensive tree trimmers and dangerous home tree trimming methods.

1. Tree Trimming Business
    a. **Characteristics:** Relies on people needing trees trimmed, wants to provide quality and reliable work
    b. **Needs:** Equipment to trim trees, people to use tree trimming equipment safely and effectively
    c. **Uses and benefits:** Reduces the equipment load the business has to worry about, less safety risk, reliable work

2. Crisis Response Organization
    a. **Characteristics:** Need for as many saws for downed limbs and other limbs in the way.
    b. **Needs:** Safe way to cut apart trees for removal
    c. **Uses and benefits:** Easy to move product, anyone can operate the product
3. Tree Trimming Enthusiast
    a. **Characteristics:** wanting to cut down trees in order to avoid paying tree trimming business, possibly wants either firewood or regular wood for future carpenter work
    b. **Needs:** Safer tree cutting equipment as well as being user friendly to the enthusiast
    c. **Uses and benefits:** Cheaper than hiring a company to trim trees and safer than trimming trees with a standard saw.

2.3 Requirements & Constraints

Functionality

- Wireless control of branch saw, rotation, and saw clamp
- Improved user experience through use of controller

Resources

- Two sets of eight relay switches
- Arduino Uno and Arduino Mega
- A pair of Xbee transceivers and shields

Aesthetics

- Easy to understand design
    - Warning and danger indicators for components that can cause harm(jaws and chainsaw)
- Organized and covered wires/electronics

User Experiential

- Easy to understand and ergonomic controller
- Readily available safety features

Economic/market value

- Main chainsaw chassis made from plastic
    - Design will be 3d printed for the scope of this project
- Chainsaw is electric

Environmental

- Chainsaw is electric

UI

- Intuitive controls
- Easy to connect saw and controller

2.4 Engineering Standards

Standard 1: 1679.1-2017 - IEEE Guide for the Characterization and Evaluation of Lithium-Based Batteries in Stationary Applications:
https://ieeexplore.ieee.org/document/8262521

This article is the guide to the IEEE standards on the safe use of Lithium Batteries. This project uses a pair of lithium batteries to power itself so this guide was helpful in the safe use of said batteries.

Standard 2: 1118.1-1990 - IEEE Standard for Microcontroller System Serial Control Bus
https://ieeexplore.ieee.org/document/159173

This article describes the IEEE standards for the use of microcontrollers in a variety of uses. Our project uses arduino's as control devices which is covered within this guide.

Standard 3 :1910.266(e) - OSHA Standard for Safe Operation of a Chainsaw
https://www.osha.gov/etools/logging/manual-operations/logger/chain-saw/saw-operation

These standards show what a user should follow when using a chainsaw or something similar in order to facilitate safe usage. Our project uses a chainsaw in which these standards apply to it.

3 Project Plan

3.1 Project Management/Tracking Procedures

-We are using the agile project management style, as our project has many independent modules, which can be worked on independently of each other, with them all coming together at the end to form the final project.

-We are using Asana to track progress throughout the project. We are using Drive for document sharing between us. We are using Slack to communicate within the group and with the client as well.

3.2 Task Decomposition

First Semester

- Created team contract and assigned roles
- Team began planning what features would be implemented
- Controller was researched and a rough outline was designed
- Parts were researched and ordered
- Design document was created to show process

Second Semester

- Individual functions were tested to see how to activate them
- Controller box was 3D printed
- Code for both Arduino and controller was designed
- Wireless communication between two Arduinos was achieved
- Controller was assembled
- Design was slightly revised and relay switches were ordered to allow arduino control over the saw's functionalities
- The saw's "brain box" was 3D printed to hold the saw's Arduino and relay switch networks
-

3.3 Other Resource Requirements

-Other resources which may prove helpful to this project are going to be expertise regarding arduino coding, general software expertise, and radio communication expertise.

4 Design

4.1 Design Context

4.1.1 Broader Context

Area	Description	Answers
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities)	One of the main objectives of the Smart Saw project is to increase the safety of chainsaw users. The product however does not protect bystanders from any falling branches
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures.	The product will allow homeowners to have a sense of ownership over their land care by allowing them to trim trees by themselves while still being safe. Equipping crisis response teams with this product also gives them one more layer of security and versatility when facing a situation.
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement.	The Smart saw is entirely electric with a rechargeable battery. Reducing any emissions. The saw also guarantees a straight cut, reducing the amount of waste if the lumber is going to be used.
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups.	The Smart Saw is meant to be a cheaper alternative to a tree trimming company. The product being safer than trying to complete a job with a normal chainsaw, and cheaper than a tree trimming service makes it economically viable. The product can be marketed to both homeowners, and tree trimming companies, as well as crisis response organizations that help with clearing out debris.

4.1.2 Prior Work/Solutions

-While the market does not currently have a remote, smart chainsaw, we are able to compare to the prototype we are improving upon. Our main objective is to make the smart saw able to receive commands wirelessly. The main advantage of this would be how it is much safer to trim trees while being out of the tree and away from the saw compared to being directly under the tree branch to control the saw arm and clamp. A shortcoming with this decision is the need for multiple sensors in order to read and control power, saw position, and clamp position.

-Our product also provides consumers with another choice between hiring a tree trimming service or using a safe, easy to use solution that our saw provides. An average tree-trimming service costs $70-$200 per hour. The smart saw would be priced under $1000. This would be a good option that would allow a consumer to avoid paying for such an expensive service and still keep themselves safe should they not have the experience to use a regular saw.

https://www.gotreequotes.com/how-much-does-an-arborist-cost/per-hour/#:~:text=With%20a%20professional%20arborist%2C%20your,the%20cost%20will%20be%20higher.

4.1.3 Technical Complexity

-Our design requires wireless communication, motor control, automation, and circuit theory. Wireless communication, motor control, and automation all require a proficient background in programming and knowledge of how sensors operate and which sensors make the most sense to use for this application. Circuit theory is necessary to understand how the electronics communicate with each other as well as how to power the components safely. Advanced math knowledge may also be necessary in order to optimize the battery the saw is running on so that we can get the maximum amount of cuts out of one charge.

4.2 Design Exploration

4.2.1 Design Decisions

Wireless communication method

-We had a few options on how we want to remotely communicate with the saw but we have landed on using radio frequency and creating a physical custom controller. We had considered using bluetooth and creating an app, but through discussion, we found we are more familiar with physical electronics rather than app development. If we had chosen an app, we would have spent more time than necessary learning how to create an app.

Microcontroller

-Originally we were going to use two Arduino Unos, but to ensure we can accommodate for the motor controllers and wireless communication we've decided to increase our available I/O pins by upgrading to an Arduino Mega for the saw only. Choosing to move forward with the Uno may prevent us from being able to reach full functionality.

4.2.2 Ideation

-Another option was adding a PCB for long term use and organization.

-some of our options in wireless Controllers are the RC car controller or creating our own controller using a game controller as a base

4.2.3 Decision-Making and Trade-Off

Criteria	Weight	Bluetooth App	RF Controller
		Score|Total	Score|Total
User Interface	.3	5|1.5	4|1.2
Cost	.1	5|.5	4|.4
Effectiveness	.3	3|.9	5|1.5
Ease of Implementation	.3	2|.6	3|.9
Total	1	3.5	4

4.3 Proposed Design

4.3.1 Overview

-For our project's wireless addition, we plan on making a radio controller to communicate with the saw. The controller would have a trigger system similar to other yard working tools where releasing the trigger will immediately stop the saw.

4.3.2 Detailed Design and Visual(s)

Hardware:

-For our controller, we would use an arduino to operate a transceiver to send and receive commands to the arduino on the saw. The controller would have an indicator for power and connectivity to the saw. For its functions, it would have a switch to set connection to a specific saw and two three position rocker switches to control the clamp and the saw blade. There is also a safety switch which must be on in order to operate the saw. The saw's arduino would be in charge of receiving and interpreting data from the sensors on the saw. There is also a visual indication showing that there is connection between the saw and controller and a kill switch on the saw for safety concerns.

Software:

-For our project the software will control the functionality of the devices in the say and operate for the connection between the saw and remote. In the saw when a function is called from the remote it will check to make sure proper power is on and that the saw is currently in a safe operating state. If it passes both instances the function will move on to executing the command while constantly checking for the previous requirements to be met. With the Remote the code will constantly check if a button is being pressed to send data to the saw. For turning on the saw, the remote will constantly send a run command to the saw and as soon as connection is lost or the run button is let go the remote will stop sending the run command and the saw will shut the blade off as soon as it stops receiving that signal.

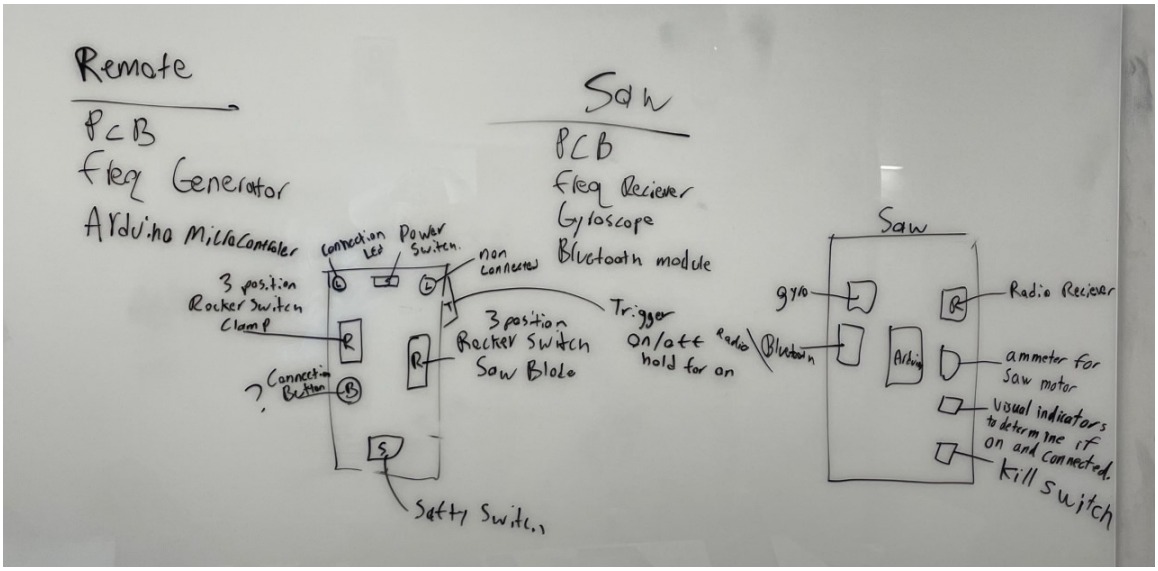


Figure 2: Initial drawing of remote and saw connections/switches

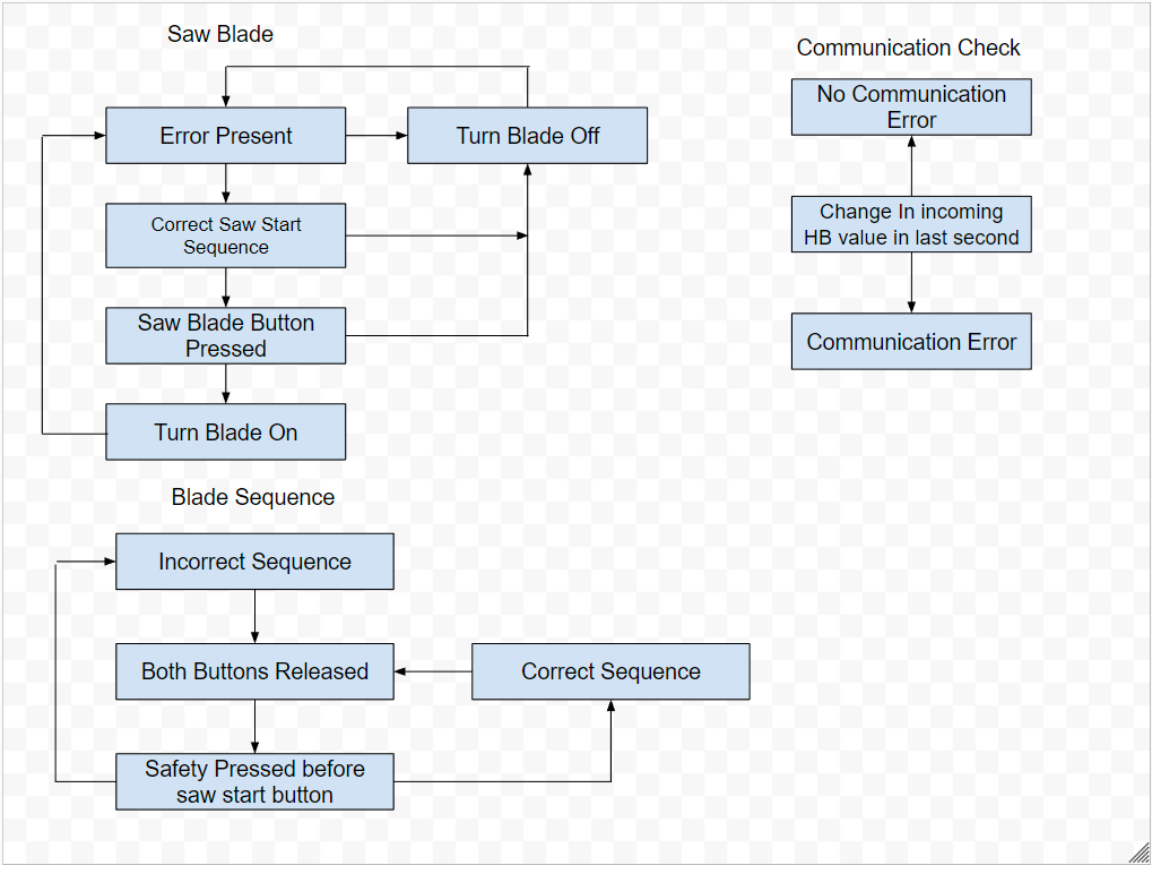


Figure 3: Programming Flowchart

To send data between the two arduinos, a data package is sent using the xbees that contains 2 Bytes of data. Because the xbee communicates with each other with chars, the outgoing data was contained in a package so that the receiving arduino could detect the package and convert the input chars back into the correct byte values contained in the data. When an arduino had successfully read data from another arduino it would then send its own data the next time the program ran through a cycle. If no data had been received, the arduino would send out data at a rate of only 1-2 times per second. This was done to establish proper handshake between the two arduinos so they could transfer data without issues.

```
if (XBee.available())//Read incomming Data from other controler
{ // If data comes in from XBee, send it out to serial monitor
  cTempDataCollect = XBee.read();
  if(cTempDataCollect == '*')
  {
    iDataInCounter = 0;
    bReadComplete = LOW;

    while((bReadComplete == LOW) && (iDataInCounter <= 5))
    {
      if(XBee.available())
      {
        cTempDataCollect = XBee.read();
        //Serial.write(cTempDataCollect);
        if(cTempDataCollect == '!')
        {
          bReadComplete = HIGH;
          uiDataIn = DataToByte(arCharCollect);
          arCharCollect[0] = ' ';
          arCharCollect[1] = ' ';
          arCharCollect[2] = ' ';
          arCharCollect[3] = ' ';
          arCharCollect[4] = ' ';
          iXbeeCounter = 0;
        }
        else
        {
          arCharCollect[iDataInCounter] = cTempDataCollect;
          iDataInCounter = iDataInCounter + 1;
        }
      }
    }
  }
}//end of data read loop


//Output Data signal through xbee
  if(iXbeeCounter >= 0)
  {
    XBee.print('*');
    XBee.print(uiDataOut);
    XBee.print('!');
    iXbeeCounter = -100;
  }
  iXbeeCounter ++;
```

Figure 4: Data Communication Code

To read values from the data bytes and to set values to them, we used and and or gates to read and set values to the bits we wanted to use while leaving others at their current value. Do send data for how we wanted the saw to act based on an input to the controller, each bit had a use that was defined in the data based on if it was going from saw to controller or controller to saw. The first byte in the data is an ID byte to allow the controller to pair up with a saw and so that the saw only takes commands from a controller it is paired with. Both arduinos also contained a heartbeat value in its data so the other arduino can check that value and see if there is a communication error that happens between the two arduinos.

Controler -> Saw : Byte 1								
BIT #	16	15	14	13	12	11	10	9
BIT Use	Matched Controller ID #	Matched Controller ID #	Matched Controller ID #	Matched Controller ID #	Matched Controller ID #	Matched Controller ID #	Matched Controller ID #	Matched Controller ID #
When Value 0								
When Value 1								

Controler -> Saw : Byte 0								
BIT #	8	7	6	5	4	3	2	1
BIT Use	Heartbeat Signal	Error	Saw Blade	Clamp Open	Clamp Close	Blade Rotate Forward	Blade Rotate Reverse	Error Reset
When Value 0	HB Off	No Active Error	Blade Off	No Movement/Close	No Movement/Open	No Movement/Reverse	No Movement/Forward	No Reset Of Saw Error
When Value 1	HB On	Active Controler Error	Blade On	Open Clamp	Close Clamp	Clockwise Motion	Counter Clockwise Motion	Reset The Saw Error

Saw -> Controler : Byte 1								
BIT #	16	15	14	13	12	11	10	9
BIT Use	Saw ID #	Saw ID #	Saw ID #	Saw ID #	Saw ID #	Saw ID #	Saw ID #	Saw ID #
When Value 0								
When Value 1								

Saw -> Controler : Byte 0								
BIT #	8	7	6	5	4	3	2	1
BIT Use	Heartbeat Signal	Error	Reset Ack	Blade On				
When Value 0	HB Off	No Active Error	Reset Bit not Recieved	Blade Is Off				
When Value 1	HB On	Active Saw Error	Reset Bit Recieved	Blade Is Running				

Figure 5: Datasheet for data bytes

For the sequence to run the saw, the controller looked to see if both the buttons start out off. For the controller to transmit to run the saw, the safety button must first be pressed. Once the safety is pressed and held, the blade will start when the run button is pressed. Once the blade is running the safety can be released and the blade will stop whenever the run button is released. If the sequence is done improperly, the saw will flash and indicator with the flashing yellow led.

```
if((digitalRead(iBladeOn) == LOW) && (digitalRead(iBladeSaftey) == HIGH) && (bSafteyReset == HIGH) && (bError == LOW))//Determine if Saftey is Pressed
{
  bSafteyPressed = HIGH;
}

if ((bSafteyPressed == HIGH) && (digitalRead(iBladeOn) == HIGH) && (bError == LOW))//If saftey active and saw on button pressed start saw
{
  uiDataOut = uiDataOut|0x0020;
  bSafteyReset = LOW;
}
else if(((digitalRead(iBladeOn) == LOW) && ((uiDataOut&0x0020)>>5 == 1)) || (bError == HIGH))
{
  bSafteyPressed = LOW;
  uiDataOut = uiDataOut&0xFFDF;
}

if((digitalRead(iBladeOn) == LOW) && (digitalRead(iBladeSaftey) == LOW))//if both buttons released undo saftey bool
{
  bSafteyReset = HIGH;
  bSafteyPressed = LOW;
}
```

Figure 6: Saw Blade Sequence

4.3.3 Functionality

-The user would first make sure that the connection between the controller and the smart saw is secure and that they are communicating with each other. Then you would attach the smart saw to the branch, get away to a safe area from the branch, and use the controller to cut the tree branch. Then the smart saw would be detached and lowered down safely.

4.3.4 Areas of Concern and Development

-Some of our safety features include two triggers where one trigger has to be held down in order to turn on the saw blade. That'll fall in line with similar safety features on other saws and utilities. The prototype that's given to us is already designed to hold in place while it's operating so we won't have to worry about designing any mechanical safety features.

-Our primary concern is establishing effective communication between the saw and controller and addressing the possibility that the connection is broken.

-We plan on having the saw automatically shut down if connection is broken for an extended period of time (2-3 sec).

4.4 Technology Considerations

-We are choosing to use radio frequency to communicate between the smart saw and the controller that we are using. The strengths of radio frequency is that it has a higher range and is more secure, while disadvantages are that it is more inconsistent. It could be more consistent if it was higher power. A possible alternative is bluetooth.

5 Testing

5.1 Component Testing

- **Xbees**: Initial testing for the xbee's consisted of having one Xbee connected to an arduino and the other on a programming board connected to a laptop. Using the serial terminal on the arduino, we were able to send characters back and forth between Xbees. We then altered the code and moved the second Xbee to another arduino. One arduino was hooked up to a circuit with a button while the other was given an LED. The goal for this test was to use one circuit to turn the led on, on the other circuit which we were able to do.
- **Saw Blade**: The motor controller that came with the prototype we were given was our main method of controlling the saw. The motor controller has a +5V, ground, and signal wire. After testing, the chainsaw is off when the signal is an open circuit. When the signal moves close to ground, the saw will begin running clockwise and vise versa as the signal line moves toward +5V. However, during testing and planning how to wire the relay switches, the motor controller broke causing the saw portion to become inoperable.
- **Clamp and Saw Bar**: Both the clamp and the saw bar run in very similar ways. When the motor receives ∓ 14V, the clamp will open or close and the saw bar will rotate left or right.
- **Relay Switches**: To test the relay switches we wired up the switches to the saw with no power to the saw to see what leds on the relay network would turn on.

5.2 Controller Testing

- **Distance Testing**: After the initial tests with the Xbees, we created a breadboard prototype of the controller in order to test the range we could reasonably get. The goal of this test was to use the switches to turn on led on a separate board and have one person walk until communication between the Xbees was lost. Walking from the main entrance to Coover toward Sweeny was done in order to have a clear unobstructed view between the two Xbees. Using this method we were able to measure a range of over 350 feet while maintaining connection between the two Xbees.

- **Functionality Testing:** Here we had our final draft of our controller wired in its 3D printed box. The goal here was to see if we could actuate the relay network using the controller. After this we then plugged in the batteries to make sure the saw would run properly with the relays. By this point the saw motor controller had broken so we could not run that functionality. After dry running the relays and confirming the right switches were activating at the right times we plugged in the batteries to confirm the saw functions worked which was successful.

6 Implementation

-We are planning on making the controller, programming the controller to be able to communicate with another arduino in the Smart Saw. Making sure that the Smart Saw arduino can activate the clamp, bar rotation, and main chain. Our last part would be to combine the controller, Smart Saw Arduino, and other safety features together to finish our project.

7 Professional Responsibility

7.1 Areas of Responsibility

-One of the codes of ethics described in ACM version is to uphold and respect the code while holding each violator responsible to them. It addresses the seven professional responsibilities by combining them into a list that each member within the ACM or a professional should uphold to in order to uphold the expectation that is expected of them.

7.2 Project Specific Professional Responsibility Areas

-Work Competence will be a high responsibility to uphold due to the many moving elements that this project will demand. Most of our group have had experiences in CAD design, Arduinos, and electrical appliances so most of the roles needed for the project can be fulfilled to its best ability. Next on the important scale will be social responsibility, where we'll have to create a design that'll be relatively easy to mass produce and be durable enough to do its purpose: cut trees. Financial responsibility is already being handled via the BOM. Health and Safety is also being considered when testing the project as programming when to start and stop the saw was a part of our client's concern. Communication between our client and us still remains a high priority for us as we keep asking which components do they like and dislike. Finally our lowest priority in responsibility areas are sustainability and property ownership. Right now we're concerned about getting the project to work, and we believe that those two areas mentioned before wouldn't fit right in our workload.

8 Closing Material

8.1 Discussion

-The main results of our project will be a better product than the one that we started with. We will have started with a manually controlled Smart Saw that uses pre existing rc controllers to a wirelessly controlled Smart Saw with custom controllers. The requirement was making a wireless controller for the Smart Saw so that it can be used from a safe distance from the tree branch being cut down. We can achieve this by the end of next semester and have planned on how it will get done.

8.2 Conclusion

-Our goal was to design and create a controller for our client's branch saw and implement wireless communication between the saw and controller. Through the use of a pair of Arduinos, Xbee's, and a relay network we were able to complete our client's goals to integrate a wireless system into the saw so that we can communicate with it using a controller from a distance. Unfortunately, due to the motor controller breaking during testing, we were unable to run the chain motor to run wirelessly.. We were at least able to set up the saw in a way that once a new motor controller is acquired, the client will be able to regain that functionality.

8.3 Appendices

8.3.1 Team Contract

Team Name ___EE Gang_____

Team Members:

1) ____Mitchell Kistner_____ 2) _____Ethan Bauman_____

3) ____Austin Mackedanz_____ 4) _____David Kruse_____

5) ___Patrick Pham_____ 6) _____Lance Longhorn_____

7) ____Jace Fedler_____ 8) _____

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings: Tuesday from 2:10 to 4:00

2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face): Face-to-face, Asana, Phone based.

3. Decision-making policy (e.g., consensus, majority vote): Majority Vote

4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived): David Kruse

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings: Show up to every meeting and if can't show, notice 1 hour in advance.

2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines: If assigned a task it is on you to get it done without constantly needing to be reminded or pushing it to someone else without reason.

3. Expected level of communication with other team members: Don't be afraid to communicate when needing help or giving assistance or updates.

4. Expected level of commitment to team decisions and tasks:Communicate any decisions or tasks.

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):

Mitchell Kistner : Project Manager and CAD Design

Ethan Bauman : Controls and Software

David Kruse : Circuits and PCB Design

Austin Mackedanz : Software and Circuit

Lance Longhorn: CAD Design and Circuit

Patrick Pham : CAD Design and Circuit

Jace Fedler : Controls and Software

2. Strategies for supporting and guiding the work of all team members:

Putting team tasks and subtasks on Asana task board.

3. Strategies for recognizing the contributions of all team members:

Uses the Asana Task Board where team members are assigned tasks which can be seen by everyone.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.

Mitchell Kistner: Project management experience and a wrench turner

Ethan Bauman: Controls experience and experience with task management software.

David Kruse: Soldering skills in PCB testing and Quality Control experience

Austin Mackedanz: Electrical and programming experience

Jace Fedler : Electrical, controls, and programming experience, project and time management, specialization in computer engineering. Magic.

Lance Longhorn: Circuitry, Programming, and 3D modeling skills. Basic app development knowledge.

Patrick Pham: Linux, some programming experience, Small circuit knowledge.

2. Strategies for encouraging and supporting contributions and ideas from all team members:

Asana workspace are areas for collaboration on all the team tasks.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

Have a meeting to talk through the options.

Goal-Setting, Planning, and Execution

1. Team goals for this semester: Get part of the project done and have a gameplan set to finish the project second semester.

2. Strategies for planning and assigning individual and team work: Team decision on who does what and project manager records and assigns tasks through asana.

3. Strategies for keeping on task:

Weekly meetings going over the tasks and current progress towards completion.

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?

They will buy large Culver's cheese curds.

2. What will your team do if the infractions continue?

More cheese curds and discussion with the advisor if problems continue.

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*

b) *I understand that I am obligated to abide by these terms and conditions.*

c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) Jace Fedler

DATE 9/13/22

2) Austin Mackedanz

DATE 9/19/22

3) Ethan Bauman

DATE 9/19/22

4) Lance Longhorn

DATE 09/19/22

5) Patrick Pham

DATE 9/19/22

6) David Kruse

DATE 9/19/22

7) Mitchell Kistner

DATE 9/19/22

8.3.2 Instructions

"Smart" Saw operations manual

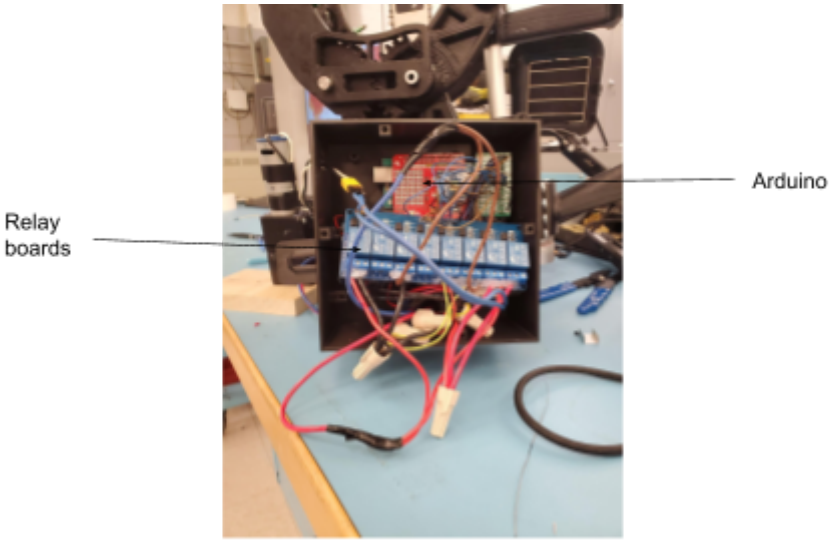
The "Smart" Saw is the next stage of Pachyderm's branch saw series that uses a wireless controller to operate the saw blade and clamps in order to cut down tree branches. Please read the following before use and have the

appropriate safety equipment on you at all times. WARNING: This manual was made to instruct users how to operate the Smart Saw safely. We are not responsible for any injuries due to mismanagement of the saw or horseplay. Please use the Smart Saw as intended: to cut down tree branches.

The Saw

The saw itself is one of Pachyderm's Branch saws utilizing clamps and a rotating saw bar with separate motors to attach itself to a branch then cut the branch.

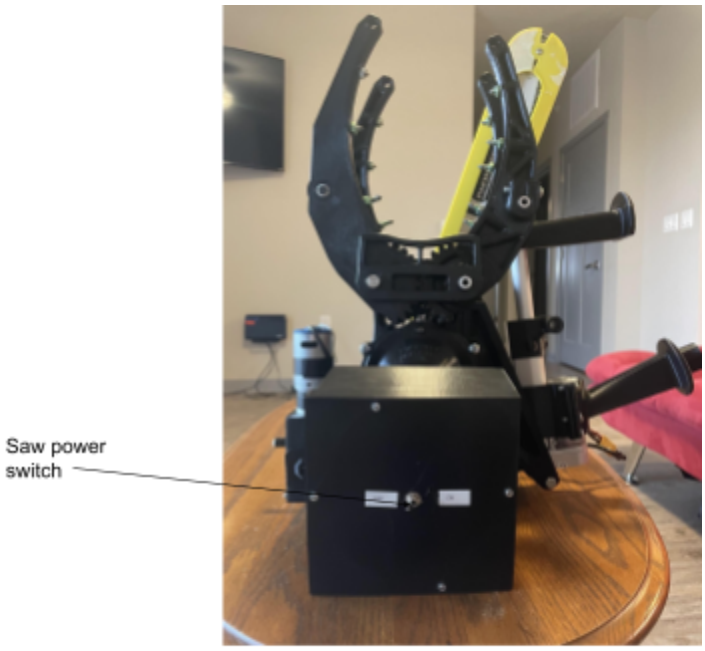
There's also a dead man switch located on the other side of the brain that will cease all operations of the saw; pressing down on the switch turns off the blade, pulling up turns on the blade.



Saw brain (lid off)

The saw uses Arduino uno that is responsive to the wireless controller which then uses a series of relays to operate the clamps and saw blade all located in a

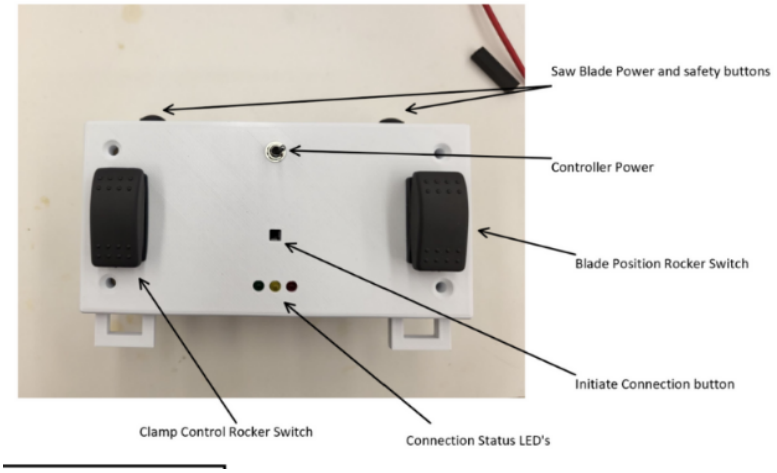
3D printed box called “the brain”. Within the brain, the relays are then connected to the clamp and saw motors accordingly.



Saw brain (Lid on)

The Controller

The controller is made up of an Arduino and an Xbee transceiver housed in a 3D printed box, which are used to send communications to the saw. The controller has two rocker switches on the top of it, one to control the clamps and another to control the saw bar. There’s two buttons on the front of the controller, both are used to control the saw blade. The buttons use a two stage process in order to operate the saw blade, one is used as a safety



button and must be pressed down first then the power button is pressed thus activating the saw blade. The controller power switch is in the middle along with the connection button and the LEDs. The power switch turns on the controller and the connection button is pressed until the LEDs show that the controller is connected to the saw. The LEDs also have other indications in order to show the user the states the controller is in. There's also loop holes at the bottom of the controller for lanyards if desired.

LED Indicator Lights	
Red Flashing	Connection Error
Red On	No Connection
Amber Flashing	Blade Start Pressed Without Using Saftey
Amber On	Connected Idle Mode
Green Flashing	Input Data detected Blade is Off
Green On	Blade In operation

Steps on operating the Smart Saw

1. Move the saw to desired position on the tree branch safely as possible
2. Turn on the saw via power switch located on the box
3. Make sure dead man switch is pulled up
4. Turn on the controller via power switch on the controller
5. Hold down the connection button on the controller until the amber LED is on.

Note: if the red LED is flashing or on then the controller is not connected. Turn off the controller and repeat step 3

6. Once the controller is connected, move away from the saw, with controller in hand, to a safe distance.
7. Press on the left rocker switch to move the clamps. Make sure the clamps close onto the branch as tight as possible
8. Press on the right rocker switch to move the saw bar. Move the saw bar to a starting cutting position.
9. Press the right safety button in front of the controller before pressing the left button (Green LED will be flashing to indicate the left button). Then press the left button to engage the saw blade (Green LED will be on).

Note: if pressed incorrectly, the amber LED will start flashing indicating that the blade start button was pressed incorrectly

10. While holding down the front buttons, press on the right rocker switch to slowly move the saw bar down to begin cutting
11. After cutting the branch, let go of both buttons to disengage the saw blade
12. Safely move the clamps so is to remove the saw from the tree branch

In case of malfunctions

Connection Issue

Turn off and on the controller to reset connections then hold down the connection button to reconnect to the saw

Saw blade unresponsive

Press down the dead man switch then establish the connection (see **Connection Issue**). After solving the connection, pull up the Dead Man switch and test the blade