Body Controlled EOD Robot
User Manual

EOD Robot Team
Cody Varner
Yazhou Li
Zening Wen
Huiwen Chu
Sixian Zhang

EE 486C – 2018
Project Sponsor: Abolfazl Razi
Sponsor Mentor: Arnau Sugranyes
Instructor: Kyle Winfree

Disclaimer

This report was prepared by students as part of a university course requirement. While considerable effort has been put into the project, it is not the work of licensed engineers and has not undergone the extensive verification that is common in the profession. The information, data, conclusions, and content of this report should not be relied on or utilized without thorough, independent testing and verification. University faculty members may have been associated with this project as advisors, sponsors, or course instructors, but as such they are not responsible for the accuracy of results or conclusions.

Team Member Contact Info

Contact Info:

Cody Varner:

1. Telephone: (951)-243-4902
2. Email: cmv246@nau.edu

Zening Wen:

1. Telephone:(520)-491-0011
2. Email: wz53@nau.edu

Yazhou Li:

1. Telephone:(928)-380-8795
2. Email: yl346@nau.edu

Sixian Zhang

1. Telephone:(928)-221-2926
2. Email: sz244@nau.edu

Huiwen Chu

1. Telephone:(928)-221-0484
2. Email: ch2435@nau.edu

Table of Contents

1 Introduction

We are pleased that you have chosen EOD robot team for your business needs. There is a strong need for EOD robot, as evidenced by wireless and remotely controlled when dealing with the explosive objects. From the recent statistics, the most common form of terrorist attack in the United States from 1970 to 2016 was bombing/explosion, and there are 1384 incidents happened. And it has been proved that under this circumstance, a well-designed, more accurate EOD robot can save many people's life. Another reason is due to making the system more user-friendly. What we mean by user-friendly is that anyone would be able to use this at their own free will. The current control system requires special training that one must go through but our system requires little to none training so it is clear that anyone would be able to put the sleeve controller on and control the robot freely.

Another reason deals with creating a low-cost human and machine interface. Current EOD's can cost nearly 6 figures but our system is a much cheaper solution as our entire project cost less than $1000. Lastly, the current control system uses a joystick to control the actions of the robot. This method allows minimal interaction between the user and robot, which is something we want to increase. So with that here is our controller for the robot that we constructed and it allows the user to give directions using the motions of their arm using sensors located all up and down the sleeve, which will, in turn, provide a greater interaction for the user itself.

We provide for you here a powerful system for design and implement a special type of robot called body-controlled explosive ordnance disposal (EOD) robot that has been custom-designed to meet your needs.

Some of the key highlights include:

1. Wireless communication;

2. Wearable controller;

3. Visualization system;

4. Affordable.

The purpose of this user manual is to help you, the client, successfully uses and maintain the EOD robot in your actual business context going forward. Our aim is to make sure that you are able to benefit from our product for many years to come!

2 Installation

2.1 Installation of the Robotic Arm

The robotic arm we had used for this project was included in a robotic arm kit that we had purchased and put together. The entire kit is consisted of:

Figure 1. Parts for the Robotic Arm

Excluded from this kit was the six servos needed for completion of this robotic arm. So, to fully build the arm we also ordered six KS-3518 waterproof servos that only went 180 degrees total. To build the robotic arm with all of these parts, follow these steps:

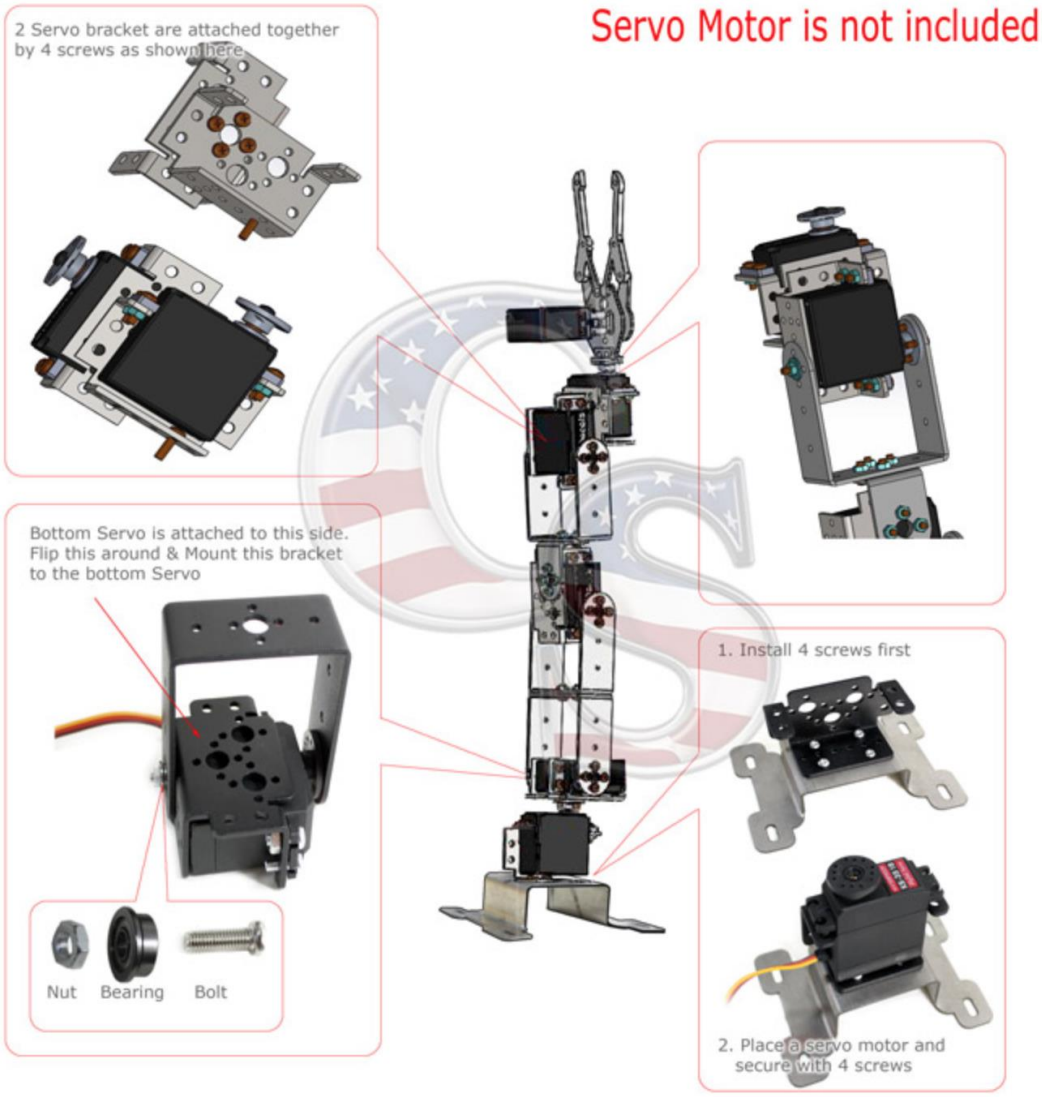


Figure 2. Connection of the Robotic Arm

If any of these parts are malfunctioning, proceed to the maintenance/troubleshooting section for the robotic arm.

2.2 Installation of the Platform

The platform of this EOD robot is using KOOKYE Robot Tank Car. The installation instructions tutorial can be find at the official website of KOOKYE:

http://kookye.com/2016/11/18/kookye-robot-smart-car-chassis-with-motor-installation-instructions/

After install the platform, two motors on the platform need to connect to the motor controller (L298N). The connection is follow the picture below or :

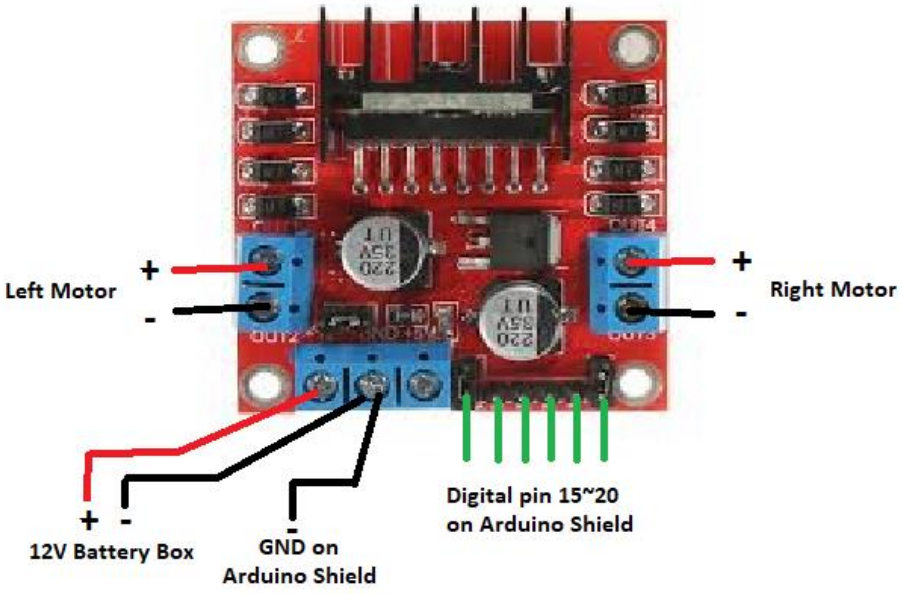


Figure 3. Connection of the L298N Motor Controller

The figure below is the instruction about how to combine the robotic arm, Arduino module and motor controller to the platform.

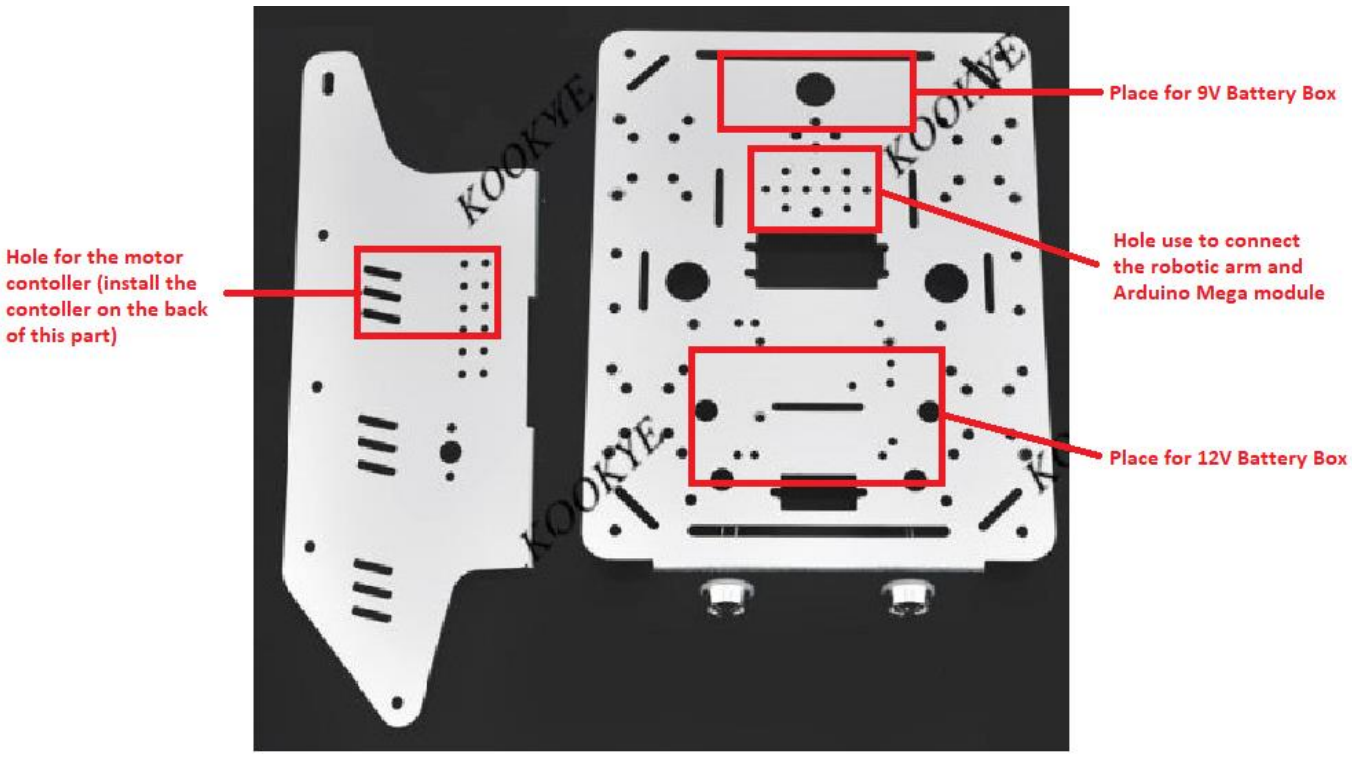


Figure 4. Connection of the Robot

2.3 Installation of the Controller

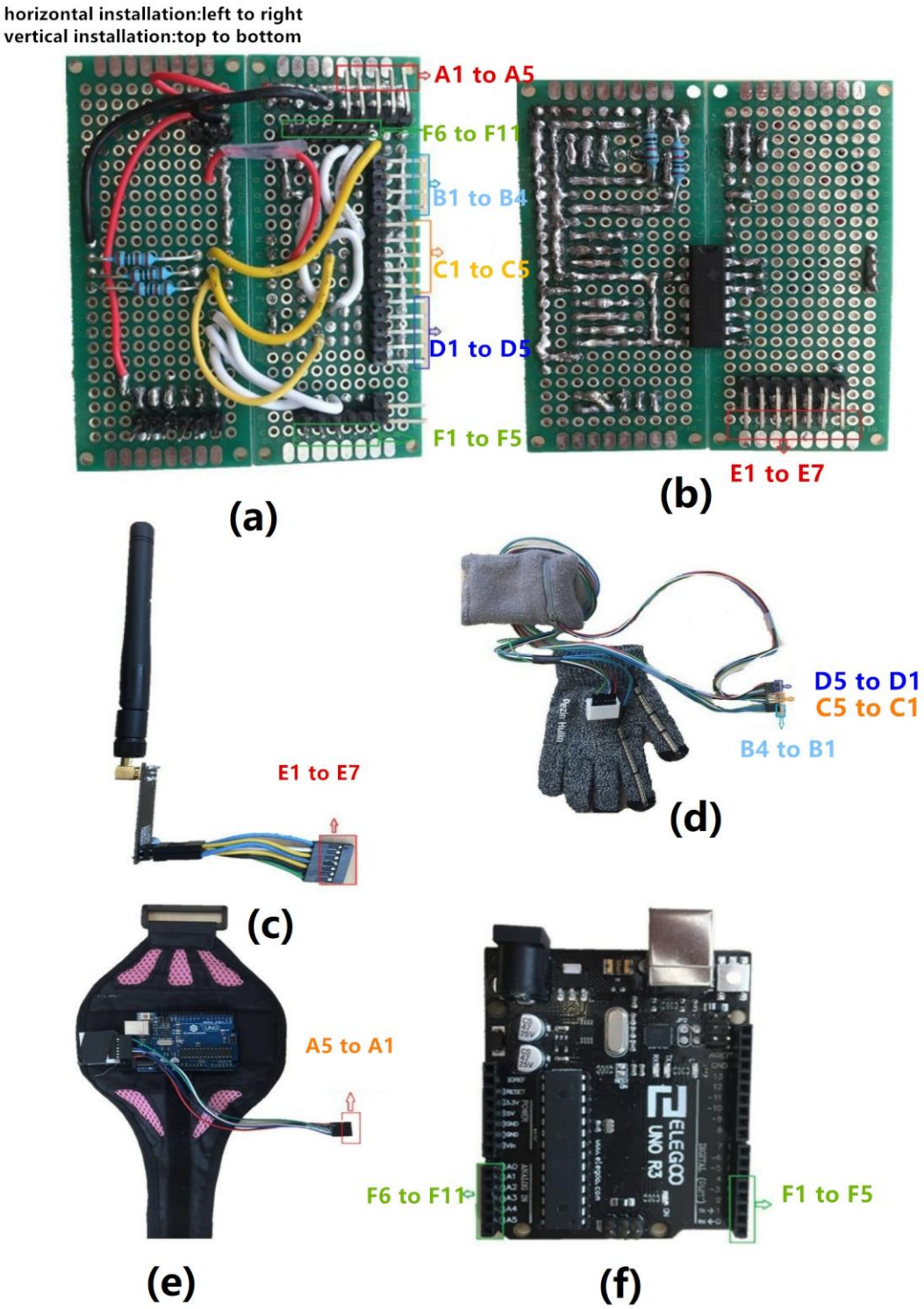


Figure 5. Connection of the Controller

Controller Components: a) Printed Circuit Board front side(PCB)
b) Printed Circuit Board back side(PCB)
c) NRF24l01 Module
d) Glove and Wrist Band
e) Armband
f) Arduino

Installation Controller: Each component has been marked (A to F). Plug male head to matched female head, eg: A1 to A1, B1 to B1, etc.Please prepare six AA batteries (9V) to battery box. Connecting battery box to Arduino power supply port.

2.4 How to turn on/off the robot and controller

When trying to turn on this product, follow these specific steps in order:

Step 1: Retrieve the EOD Robot and arm controller from their protected area and place them in a safe place where it cannot get damaged.

Step 2: Place the glove controller on the user's right hand starting with the strap that goes on the user's bicep and strap in tight so it cannot fall off or slip down. Next, put the wristband on and place it near the middle of your forearm. Once these two parts of the controller are secured, put on the glove.

Step 3: Turn on the two battery boxes located on top of the platform for the robot and then plug in the battery box to the Arduino Mega that is located under the platform. Once all these steps have been followed, insert the battery box into the Arduino in the controller.

Step 4: Everything is turned on and now you should be able to start controlling the robot freely.

Step 5: Once you are done using the product, then begin with turning off the controller first and then proceed to turn off all parts of the EOD robot.

Step 6: Place the robot and controller in a safe place once everything is properly turned off.

3 Configuration and Use

3.1 Regular Controlling methods

3.1.1 Wearing the Controller

Step 1: Wear the pink band which is fixed with the Arduino module on your right upper-arm
Step 2: Put on the gray wristband on your right forearm
Step 3: Wear the glove on your right hand
Step 4: Check the position of each sensor and its connection

Step 5: Connect the power(9V battery box) to the Arduino module

3.1.2 Switch Modes

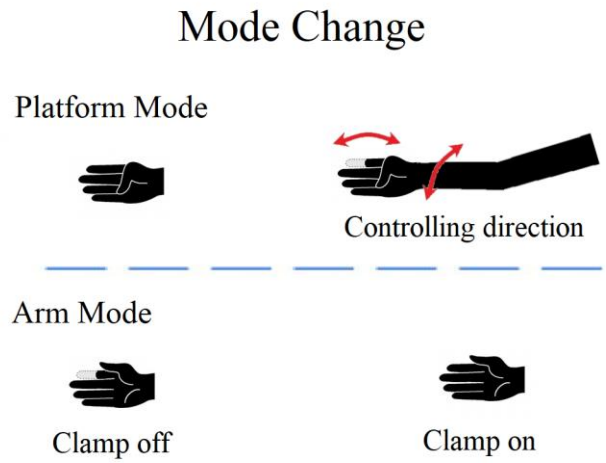


Figure 6. Mode Change Gesture

Once you have finished installation of the controller and wearing it. As above figure shows, when you bending your thumb, the robot is in platform mode. Once you lose your thumb and keep it straight, the robot will switch to arm mode.

Platform Mode:

Please keep your hand in the direction as above figure shows. The platform will turn left if you rotate your wrist left, the platform will turn right once your wrist turn right. Your index finger can control the platform moving forward, once you bending your index finger, the platform will go forward, once you lose your finger, it will stop.

Please remember to hold your thumb bending to keep in platform mode when you want to control the platform movement.

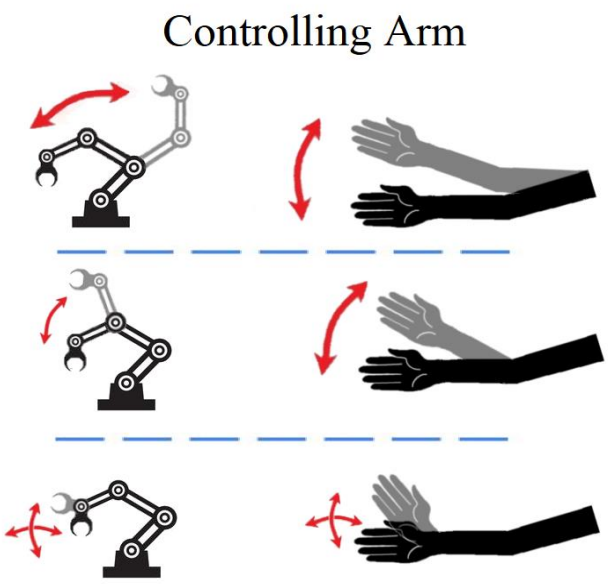


Figure 7. Arm controlling

Arm Mode:

Our EOD robot has 6-axis robotic arm.

The mpu6050 module on the glove can control the rotation and elevation of the wrist which including two servos. When user raises the hand, the clamp will go up following the hand movement. The clamp will turn left or right once the user's wrist turn left or right.

For the mpu6050 module on the forearm which is fixed in the wristband control the elevation of the servo. If the user keeps his/her hand and upper-arm still, and lift his forearm up or down, the robot elbow will rise up or go down followed by the user.

For the mpu6050 module on the upper-arm, it controls the servos on the robotic shoulder and waist, if user turn left or turn right the upper-arm, the waist which is the bottom of the robotic arm will rotate left or right. If user lift up and down his/her upper-arm, the shoulder of the robot will rise up or go down.

3.2 Visualization System

3.2.1 Initialize System

Step 1) Connect the controller with your PC through the USB port

Step 2) Find the serial port number of the controller in you PC. This will be used later.

a) Right-click on windows icon, select “Device manager”.
b) Find Arduino Uno under “Ports (COM & LPT)”. The COM number in the brace is the serial number of the controller

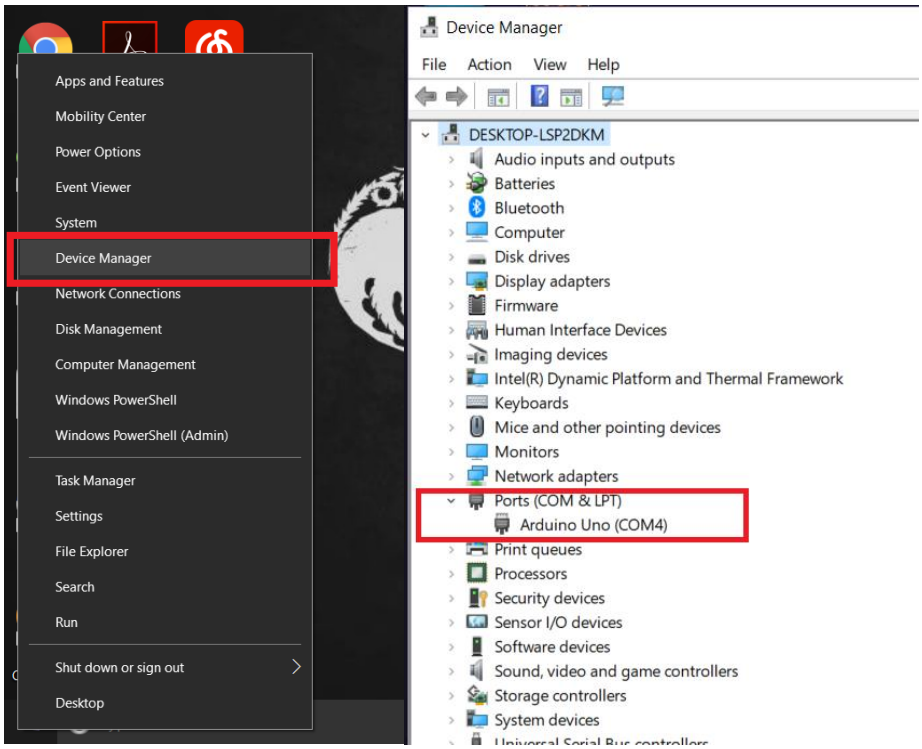


Figure 8.MPU6050 Set up(1)

Step 3) Open "EOD_Simulation_vX.X.exe". **(Attention: The executive program must be in the same directory with "EOD_Simulation_vX.X_Data" folder and "UnityPlayer.dll" file)**

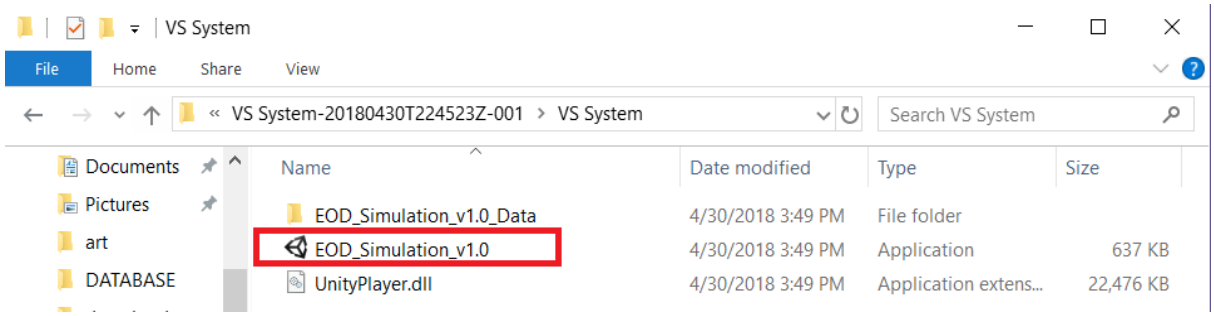


Figure 9. MPU6050 Set up(2)

Step 4) Change screen resolution to "800x600". Click "Play!"

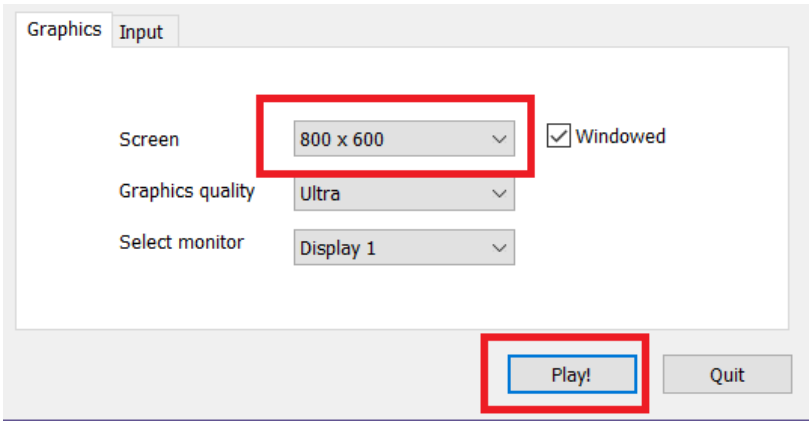


Figure 10. MPU6050 Set up(3)

Step 5) Enter the serial port number of the controller. Click “Start”

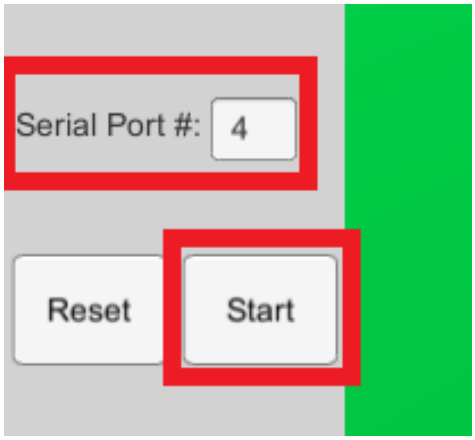


Figure 11. MPU6050 Set up(4)

Step 6) Wait for a few seconds to let the controller initialize itself. When the connection is built, you can see the controlling information on the left of the window.

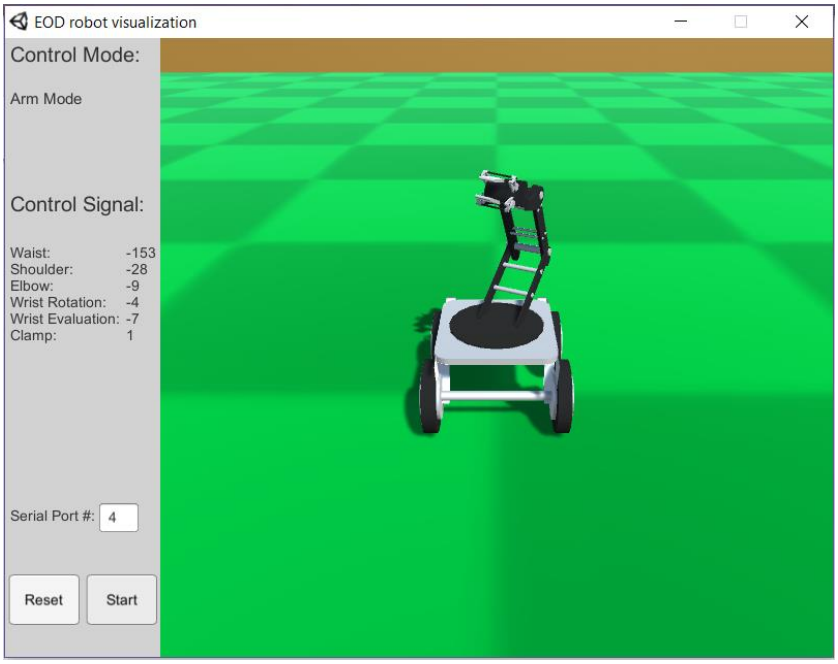


Figure 12. MPU6050 Set up(5)

3.2.2 Playing around with Visualization System

4 Maintenance

4.1 Robot

1. Please put the robot in a dry and dark environment.
2. Please keep the robot away from corrosive, explosive and inflammable.
3. Please check the screw on the robot before using the robot.
4. Please remove the battery when the robot shutdown.
5. Please do not pull out/plug in the wire frequently.

4.2 Controller

1. Please put the robot in a dry and dark environment.
2. Please keep the robot away from corrosive, explosive and inflammable.
3. Please remove the battery when the robot shutdown.
4. Please do not pull out/plug in the wire frequently.

5 Troubleshooting Operation

5.1 Battery

If the whole system is not working, please check the battery.

Battery check:

Controller: The green LEDon the Arduino will be light up when the battery is connected to the Arduino and the yellow LED on the Arduino will flash during the controller will work.

Robot: The green LED on the Arduino Shield will light up and the yellow LED on the Arduino will flash during the controller will work.

Motor control module(N298L: A red square circuit under the platform): A red LED on the circuit will light up if the power is correct to connect to L298N.

Solution:

Change the battery in the battery box.

5.2 Wireless Communication Module

If the controller cannot control the EOD robot and the batteries are changed, check the wireless communication module.

Wireless communication check: There is a red LED next to the robotic arm on the back of the robot. if the signal from the robot has been received by the robot, the red LED will be lighted up(the communication between robot and controller is very fast, so the users will the LED flashing when it works correctly).

Solution:

Step 1: Reset the board. Press the reset button for 3 seconds. The reset button is a small red button in the middle of the Arduino shield(Arduino module on the robot) or at the corner of the Arduino Uno(Arduino module on the controller). Waiting for 3 to 5 seconds, if the red LED is not lighted up, try step 2.

Step 2: Reconnect the wireless communication module by troubleshooting 5.3.

5.3 Connection

If one of the individual's parts of robot or controller is not working follow the check instruction below and the solution will after the check instruction.

Connection check:

Controller:

MPU6050(the sensor in the small white box): Open the white boxes, if the connection is right, the LED on MPU6050 will light up.

RNF24L001(the sensor with an antenna): the yellow LED will light up if its connection is correct.

Robot:

Robotic Arm: if the robotic arm does not move or shaking, it probably the connection problem.

Platform: if the robotic arm does not move or shaking, it probably the connection problem.

Solution:

Wire Connections: If you are unsure if the wires are connected properly, please check the schematics for the Robot and Controller in Appendix A.

5.4 Layout(s)

5.4.1 Robot

Robot: If there are any question regarding the layout of the Arduino Mega and the shield, then follow this picture:

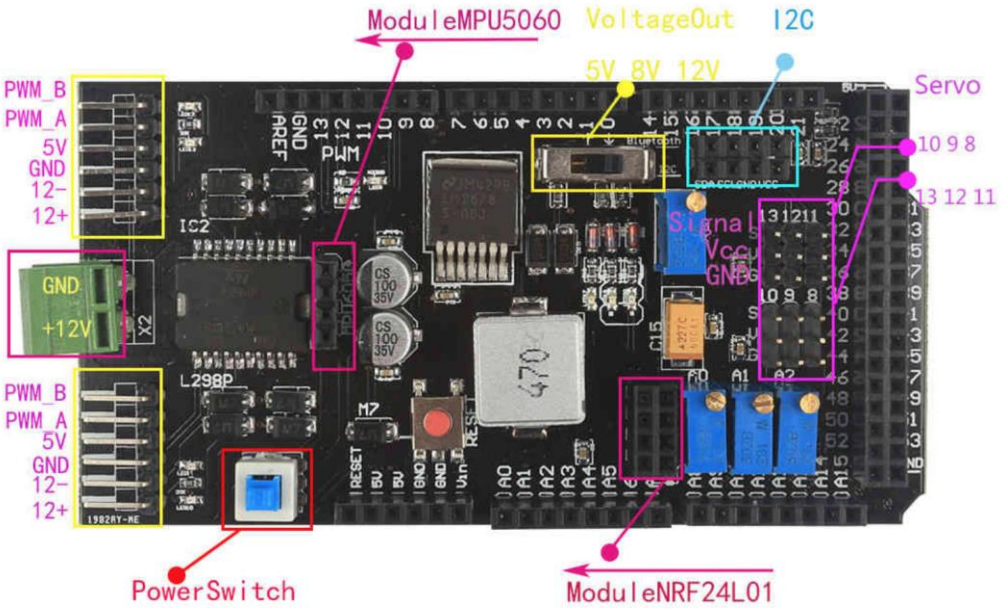
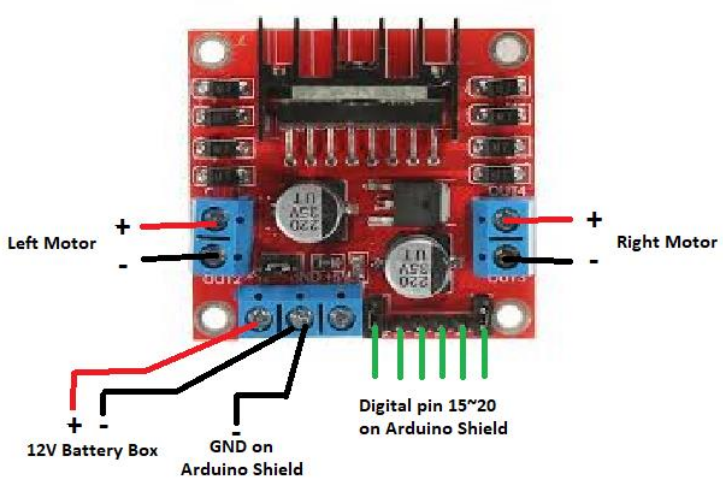


Figure 12. Arduino Mega Shield

5.4.2 Platform Motor

Motor: If there are any question regarding the layout of the motor control module(L298N), then follow



this picture:

Figure 13. Connection of L298N Motor Controller

5.4.3 Controller

Controller: If there are any question regarding the layout of the motor controller, then follow this picture:

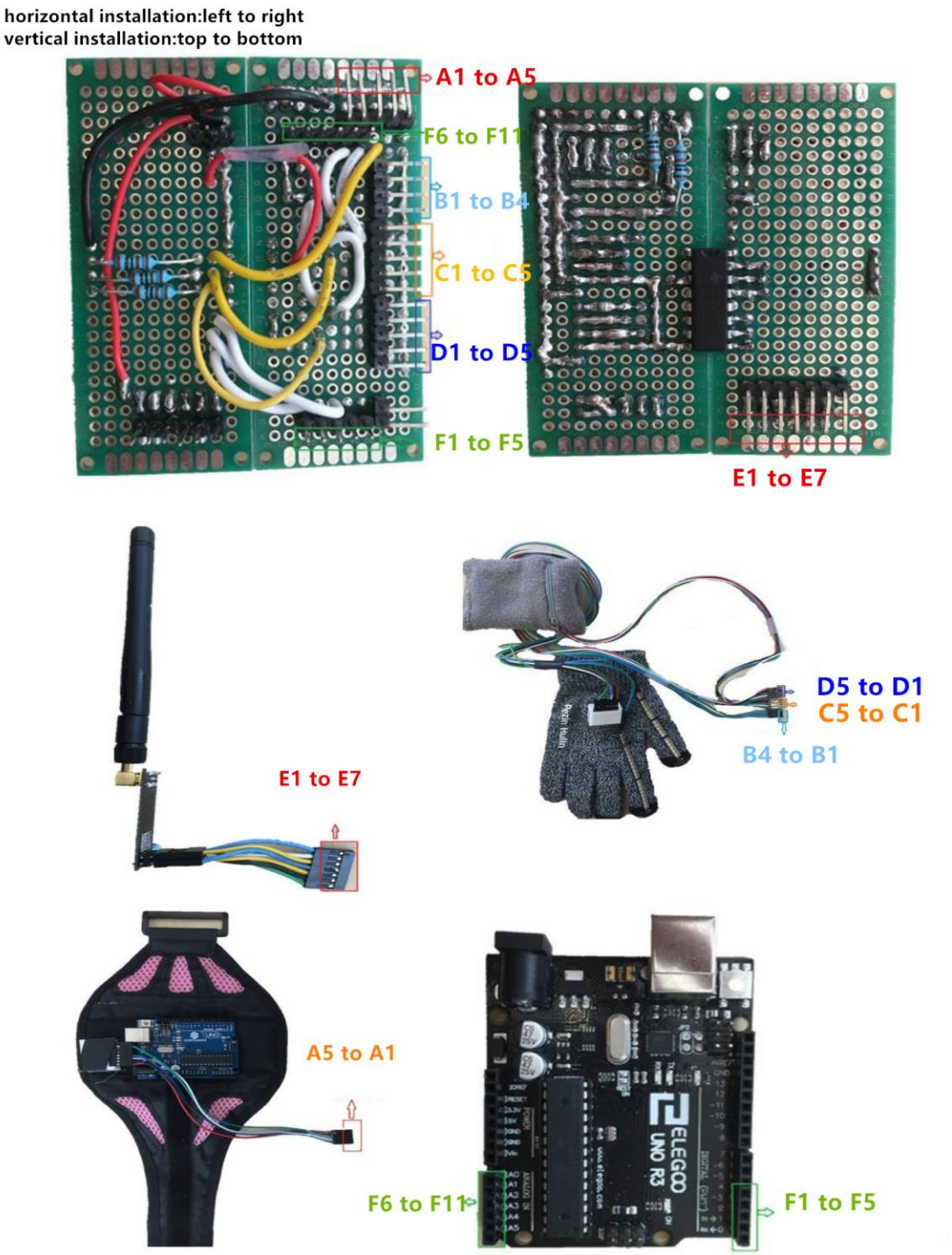


Figure 14. Connection of the Controller

6 Status of Planned Features (WBS)

6.1 WBS for Cody Varner

Within this section, you will see the Work Breakdown Structure for Cody Varner for completing this project.

Table 1. Work Breakdown Structure for Cody Varner

Person Primarily Responsible: Cody Varner				
ID	**Activity/Task**	**Description**	**Deliverable(s)**	**Other People**
1	Robotic Arm of Explosive Ordnance Disposal Robot Design			
1.1	Choose the type of the Robotic Arm and End Effector	Find a robotic arm that best suits our project and for the function of an EOD	1.Type of Robotic Arm and Gripper	
1.2	Purchase Components		1.Identify parts 2.Place order 3.Receive parts	Place order with Dr.Winfree and Dr. Razi
2	Robotic Arm Construction			
2.1	Build the Robotic Arm	Attach all components together that was provided in the kit	1.Physical part of robotic arm	
2.1.1	Attach 6 servos together using robotic arm kit	Attach the servos with the parts from the robotic arm kit	1.Correct assemelation of a 6DOF arm	

2.1.2	Test motion of arm for 6DOF			
2.2	Apply all necessary extensions/electronics to arm			
2.2.1	Attach wire extensions, hook up to arduino and arduino shield, insert battery box	Hookup everything needed to be able to make the arm work functionally	1.Fully assembled robotic arm 2.Arm is ready to start programming	
2.3	Demo Code			
2.3.1	Attach servos and set them to default positions	Initialize servos and set them to their default angles that they move to when turned on	1.Simulation of correct setup	
2.3.2	Apply code to auto control robotic arm	Set the servos in the arm to run through an auto control system	1.Simulation of autonomous control	
2.3.3	Apply code to control robotic arm with computer	Set the servos in the arm to be controlled by the user	1.Simulation of computer controlling arm	
2.3.4	Test all the code		1.Correct and working code	

2.4	Program arm for wireless control			
2.4.1	Apply code to interact with MPU 6050	Program servo motors to be controlled with the MPU sensor	1.Simulation of servo control with the 3 MPU's	Sixian Yazhou
2.4.2	Apply code to interact with Flex Sensors	Program servo motors to be controlled with the Flex sensor	1.Simulation of servo control with flex sensors	Huiwen Yazhou
2.4.3	Test and edit code			
2.5	Combine arm with robot platform			Zening Wen
2.5.1	Screw in robotic arm to platform and hookup all necessary wirings	Connect everything needed so secure the robotic arm and platform together	1.Fully assembled EOD Robot	
3	Visualize/GUI System			
3.1	Motion of robotic arm correlation	Create a system so the control of the robot in real life is transferred to control of the robot in the visual display	1.Visual Display of our Robots movement	Huiwen Chu Zening Wen Sixian Zhang Yazhou Li
4	Construct Glove for motion control			

4.1	Attach MPU and Flex Sensors to glove	Secure the flex/mpu sensors and all needed extensions to the glove/sleeve	1.Solder parts on glove	Huiwen Chu Zening Wen Sixian Zhang Yazhou Li
4.2	Construct circuit and make all correct connections	Connect all aspects together on controller	1.Fully assembled controller	Huiwen Chu Zening Wen Sixian Zhang Yazhou Li
5	Testing project			
5.1	Test all aspects of the project	Test the project to make sure it is working how it is supposed to work	1.Correct Functionality of Body Controlled EOD Robot	Huiwen Chu Zening Wen Sixian Zhang Yazhou Li

In summary, all the aspects within this WBS was completed. For starters, after all the parts were received for my part, I was able to fully build the robotic arm from all the components that I was given. Some troubles did arise from building the arm due to there was no instruction given on how to assemble it but I was able to get the arm put together how it was supposed to. Secondly, once I was able to get the arm assembled and all necessary connections made, I started to program the arm, as seen above in my WBS. There was some initial mistakes I made with programming but eventually I was able to come up with code that worked for the demo. Once this was done, I then moved on to wirelessly controlling the arm with the two sensors for this project with the help of the team. Many errors came out during this stage of the project due to the wireless system faulting, but we were all able to get the code working. The main problem with this part of the project is that the code will work but then something will happen that makes it stop working although we have improved on making this issue be resolved. Lastly, Zening and I were able to connect our parts together and connect all parts needed on the robot to where it doesn't look messy.

Furthermore, our team worked together on the last aspects of the project such as constructing the glove/sleeve, visualization system and testing the project. Our team was all able to construct the controller and were even able to get the correlation of the controller and the robot on a visual display. Lastly, we did test the project as a whole, but we didn't leave as much time to finishing the project and testing the project that we would have liked to see. This may be due to the amount of time it took us to receive our parts and because of all the problems we arose and had to fix after combining all our parts together.

6.2 WBS for Zening Wen

Table 2. Work Breakdown Structure for Zening Wen

Person Primarily Responsible: Zening Wen					
ID	**Activity/Task**	**Description**	**Deliverable(s)**	**Other People**	
1	Platform of Explosive Ordnance Disposal Robot Design				100%
1.1	Choose the type of the Platform	Find a platform which fits all requirement of this project.	1.Type of Platform	Huiwen Chu Cody Varner Sixiang Zhang Yazhou Li	√
1.2	Purchase Components		1.Identify parts 2.Place order 3.Receive parts	Place order with Dr.Winfree	√
2	Construct and Test Platform				100%
2.1	Build the Platform	Use the components we bought to build the platform.	1.Physical part of platform		√
2.2	Basic coding for Platform	Writing the code to control the platform based on Arduino	1.Basic code for platform's movement		√

2.3	Test the Platform	Use Arduino to control the platform to do some basic movement	1.Platform can do some basic movement		√
3	Visual System				100%
3.1	Choose the software for visualize system	Find a suitable software which can present the movement of the robot on the computer	1.Unity3D (visualize part)	Sixian Zhang Huiwen Chu Cody Varner Yazhou Li	√
4	Construct and Test Visualize System				100%
4.1	Connection between Arduino and Unity3D	Use Arduino to calculate date and use Unity3D to present it. We need to connect this two.	1.Translate data between Arduino and Unity3D	Sixian Zhang	√
5	Combine platform with controller				100%
5.1	Platform Movement Code	Write the code for controller to control the movement of the platform	1.Code of platform movement	Sixian Zhang	√
5.2	Test movement of the platform	Test the code on the platform	1.Movement of the platform	Sixian Zhang Yazhou Li	√
6	Robotic arm and Platform				100%

6.1	Install the Robotic arm on the platform	Use to tool to install robotic arm on platform	1.Controller can control the platform	Cody Varner	√
6.2	Control mode	Use flex sensor to control the control mode of the robot	1.Switch the control system	Cody Varner Huiwen Chu	√
7	Demonstration	Use the controller to control the robot and the robot model in visualize system	1.Use the controller control the robot 2.Use the controller to control the model in visualize system	Cody Varner Huiwen Chu Sixian Zhang Yazhou Li	**100%**

Conclusion:
As the work breakdown structure show, all of my tasks are finished. The subsystem I am responsible is the platform. For the platform part, it includes construction of the platform, chose the motor controller, codes the platform and tests the platform. After I finish my subsystem, I assist Sixian Zhang to build the Visualize system. For the visualize system, I help Sixian search some material which might help him coding the program and give some idea about the design of the visualize system. After each subsystem has been finished, I worked with other teammates to combine each subsystem together and help others to improve their parts. It includes help then install the robotic arm and Arduino on the platform, help to construct the controller and fix some error for the robotic arm. At last our team worked together to finish the demonstration.

6.3 WBS for Huiwen Chu

Table 3: Work Breakdown Structure for Huiwen Chu

Person Primarily Responsible: Huiwen Chu					
ID	**Activity/Task**	**Description**	**Deliverable**	**Other People**	Status
1	**Clamp Controller**				
1.1	Select sensor	Choose suitable sensors to implement data collection of hand movement	• Sensor type verification - flex sensor	All team member	√

1.2	Purchase Components		• Identify parts • Place order • Receive parts	Place order with Dr.Winfree	√
1.3	Construct and Test Circuits	Build and test.			
1.3.1	Flex sensor testing	Test the bending range of flex sensor	• Breadboard circuit • Test data • Bending range verification	-	√
1.3.2	Flex sensor control servo	Writing code and build circuit to implement using flex sensor to control a sensor	• Breadboard circuit • Coding • Test data • Servo moving follow by flex sensor	-	√
1.3.3	Mode change	Writing code to implement change controller when flex sensor bended in a certain range	• Breadboard circuit • Coding • Test data • Switch flex sensors to control a servo	-	√
1.3.4	Combine clamp controller with robot clamp	Writing code and building circuit to implement robot clamp controlling and arm rotation	• Breadboard circuit • Test data • flex sensor control robot clamp and arm rotation	Cody Varner	√
1.3.5	Combine clamp controller with wireless communication	Building circuit and integrating code with wireless communication	• Breadboard circuit • Test data • Flex sensor wireless control clamp and arm rotation	Cody Varner Yazhou Li	√
2	**Visualize System**				

2.1	Choose visualize system platform	Find the suitable software to show the movement of robot on the computer	• Unity 3D • MATLAB(processing data)	All team member	√
2.2	3Ds Model	Construct 3Ds model in 3Ds MAX software	• 3ds Model of robotic arm	-	√
3	**Project Test**				
3.1	Packaging	Build the 3D package for MPU6050 module	• 3D printer	-	√
3.2	Fixed Components	Fixed sensors on the glove, wrist band and arm band	-	All team member	√
3.3	Demo	Combine all parts together and test	• Test data	All team member	√

Conclusion: As the work breakdown structure shows, my part of work basically done.

For the clamp controller, I try to use the flex sensor to control a motor by writing code in Arduino and building the circuit after finished testing flex sensor. After this, I try to use two flex sensors to implement switching controlling. Because the servo of the clamp has some problem with installation, so the clamp cannot move followed by the flex sensor at the beginning. But we fixed it later.

For the visualize system part, first I try to work on using MATLAB to calculate the movement of MPU6050 with Zening Wen, but we found that we cannot implement calculation in real time and it is complicated, so we gave up this method.

For the packaging part, we sewed the modules to the glove and bands. We decide to solder most of the components on one PCB by redrew the circuit of the controller.

6.4 WBS for Sixian Zhang

Table 4. Work Breakdown Structure for Sixian Zhang

No	Activity/Task	Description of the Result	Deliverable(s)	Other People	
1	**Arm Controller**				
1.1	**Circuit Construct**	Controller circuit building	-Circuit with output data of multiple sensors		√

1.1.1	Inertial measurement unit	The basic demo is built to read out angular and acceleration data from one MPU sensor	-Breadboard circuit -Code demo -Sensor data		√
1.1.2	Multiplexer Circuit	A multiplexer circuit that can multiplex address of three different MPU sensors	-Breadboard circuit -Code demo		√
1.1.3	Integrate 1.1.1 and 1.1.2 circuit	A circuit with three MPU sensors and one multiplexer is built which can read out data from sensors simultaneously	-Breadboard circuit -Code demo -Sensor data of multiple sensors		√
1.2	**Code Demo**	Controller circuit programming	-Runnable code demo to process and package raw motion data		√
1.2.1	Raw data processing	Code is built to convert the raw data into the control signal	-Code demo -Processing result of the test data		√
1.2.2	Data packaging	Code is built to put the control signal into a form of 7 integers, where the first integer indicates the controlling mode, the rest integers indicate different controlling signal under different modes	-Code demo -Packaged controlling signal		√
1.3	**Apply code with the robot**	Combining with mechanical parts	-Robot and controller		
1.3.1	Combine the controller with the platform	Code is built for the platform to receive the control signal and operate properly	-Controller circuit demo -Platform -Functionality demonstration	Zening built the platform and helped with coding	√
1.3.2	Combine the controller with the robotic arm	Code is built for the robotic arm to receive the control signal and operate properly	- Controller circuit demo -Robotic arm -Functionality demonstration	Cody built the robotic arm and helped with coding	√

1.3.3	Calibration	Test and calibrate the motion between the controller and the robot are done	-Robot and controller -Testing result of the calibration between the controller and the robot	Test with Zening and Cody	√
1.4	**Apply code with wireless control**	Combining with wireless module	-Robot and wireless controller		
1.4.1	Combine with communication module	Code is built for both transmitter on the controller and receiver on the robot to send and receive packages. Indicator LED is also added to the robot	-Controller with communication module -Functionality demonstration	Yazhou helped with coding	√
1.4.2	Timing test	We have demonstrate the functionality of the controller and the robot but the wireless module broke down during last few weeks so not many test are done	-Robot -Wireless controller -Testing result for delay	With Yazhou, Cody, and Zening	Partly Done
1.5	**Packaging**	Solder and package the circuit	-Packaged and wearable controller		
1.5.1	Device package	Design and build the package for the controller	-Package of the controller	3D model of the package built by Huiwen.	√
1.5.2	Integrate the circuit	Soldering circuit is built that integrated multiplexer circuit, voltage dividing circuit for the flex sensors, and pin ports for sensors and communication module. The circuit board can be plugged onto the Arduino board.	-Packaged, wearable controller with soldered circuit on it.		√
2	**Visualization System**				

2.1	**Data pipeline**	Build data pipeline between sensor and visualization system	-Code demo		
2.1.1	Connect the circuit with MATLAB	Code is built to connect Arduino circuit and MATLAB	-Code demo	Zening helped with coding	√
2.1.2	Connect the circuit with visualization software	Code is built to connect Arduino circuit and Unity	-Code demo		√
2.1.3	Data pipeline between MATLAB and Unity	This task is cancelled because we found out that building a real-time data pipeline between MATLAB and Unity is far more complex.			×
2.2	**Modeling**	Visualize the motion	-Functional 3D model		
2.2.1	3D Model	3D models are imported into the system	-System demo with 3D models imported	3D Model built by Huiwen	√
2.2.2	Model control	Code is built for the system that gave the 3D models the same behavior as the real robot.	-Functionality system demo		√
2.3	**GUI**	Build interface for non-professional users	-Visualization software with GUI		
2.3.1	GUI design	GUI for the system is built so that the user can easily use it without specific runtime environment	-Executable program on Windows platform		√
2.4	**Testing**	Testing for the whole system	-Testing result		
2.4.1	Test and Simulation	The system can visualize the motion of the robot with the control signal from the controller	-Demonstration of the system		√

Conclusion:

In this capstone project, my responsibility includes MPU motion sensors on the controller and the development of the visualization system. The work I have done mainly includes building and testing circuit for multiple MPU sensors, integrating sensors and communication module onto the pluggable soldering board, programming the controller to process and package control signal, programming and testing the visualization system.

6.5 WBS for Yazhou Li

Table 5. Work Breakdown Structure for Yazhou Li

Person Primarily Responsible: Yazhou Li					
ID	**Activity/Task**	**Description**	**Deliverable(s)**	**Other People**	**Status**
1	Wireless Communication of EOD robot.				√
1.1	Looking for wireless module for this project.	Find a wireless module which can have over 10 meters range and high frequency.	One possible wireless module. (NRF24l01)		√
1.2	Purchase Components		Identify parts· Place order· Receive parts	Place order with Dr.Winfree	√
1.3	Construct and Basic test NRF24l01 module				√
1.3.1	Build a circuits to test if NRF24l01 works.	Using one flex sensor wireless control a servo motor.	The testing circuits.		√

1.3.2	Programming for testing circuits	Programming the module with Arduino for testing circuits	Basic coding for NRF24l01 module test.		√
1.3.3	Test NRF24l01 module to see the range of it.	Use flex sensor wireless control a servo motor.	NRF24l01 can do over 15 meters control.		√
1.4	Test use multiple input control multiple output.				√
1.4.1	Build a circuit to test multiple input wireless control multiple output.	Using two flex sensor control two servo motor individually.	Testing circuits		√
1.4.2	Programming for testing circuits	Programming the module with Arduino for testing circuits	Basic coding for multiple NRF24l01 module test.		√
1.4.3	Test multiple input control multiple outputs	Using two flex sensor control two servo motor individually.	NRF24l01 can send multiple signals to multiple receivers.		√
2	Combine wireless module with all other parts(arm, platform and controller)				√
2.1	Attach wireless module to controller	Connect all inputs (flex sensor, MPU6050) and wireless module to one Arduino as emitter.	The completed wireless circuits for controller part..	Sixian Zhang Huiwen Chu	√

2.1.1	Programming the wireless part for controller	Programming wireless part based on controller part's original code	Completed wireless coding part for controller	Sixian Zhang Huiwen Chu	√
2.2	Attach wireless module to receiver	Connect wireless module to arm, clamp and platform as receiver	The completed wireless circuits for receiver part	Zening Wen Cody Varner	√
2.2.1	Programming the wireless part for receiver	Programming wireless part based on receiver part's original code	Completed wireless coding part for receiver r	Cody Varner Zening Wen	√
2.3	Testing wireless part for whole project	Testing use controller wireless control receiver for whole project	Completed wireless part for whole project.		√

Conclusion:
Based on my work breakdown, my part is wireless communication part, the module is NRF24l01. First of all, I used one NRF24l01 and one flex sensor to wirelessly control one step motor. Then I used two flex sensor control two step motors to familiar fundamental functions. Then I combine wireless module to both controller and receiver, programming their parts with my NRF24l01 module, and make it work. Basically, I just change their code and add wireless code to do the whole wireless programming. Though the process, I met some problems. First, because of too much data, the transmitter always stack. We give an id to each data to solve this problem. Second, power supply, we didn't have a stable power supply to NRF24l01. It caused the transmission stack sometimes. These are two big challenges for wireless part. We also have some problems with programming, but we finally solve them. In this whole project, I more focus on wireless communication part and help controller part, building controller and clean up the wires for the receiver. After half way of this semester, we combine all our parts together. Then we have some problems, I help our group to check problems and solving problems.

7. Conclusion

We would like to take this time to personally thank Abolfazl Razi for your continued support of our team throughout the duration of this project. We enjoyed having you as our sponsor and leading us the right direction to get this project finished. We hope you are happy with our product and wish you

many happy years of productive use of our body controlled EOD robot. As our group moves onward toward our professional careers, we would be very happy to answer any questions you might have for us. We can all be reached by phone and email and all our contact information is included previously in this document. Thank you very much for this past year and we would all like to wish you the very best.

Appendices

Appendix A

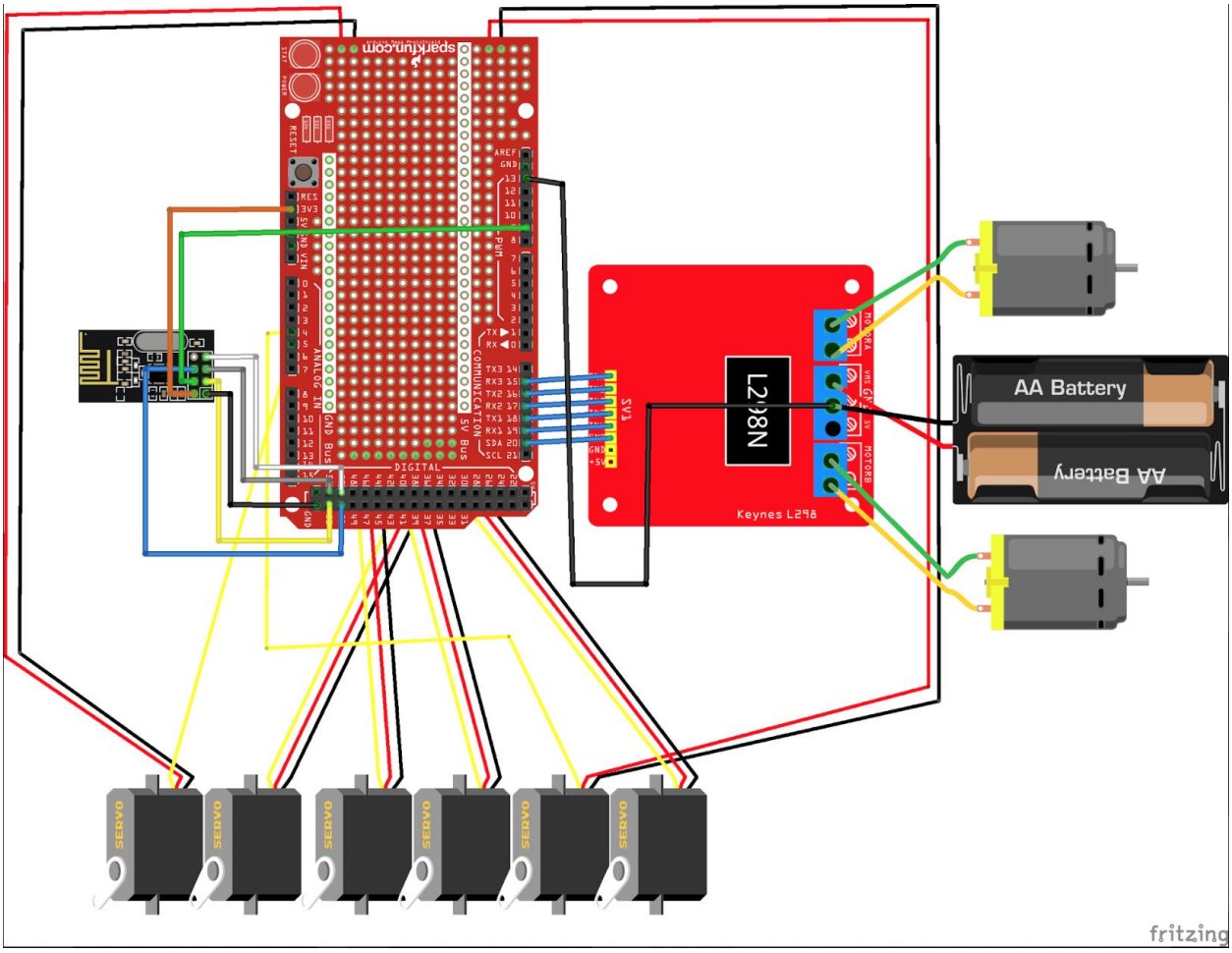


Figure 1: This image is the schematic for connections for the robotic arm and platform.

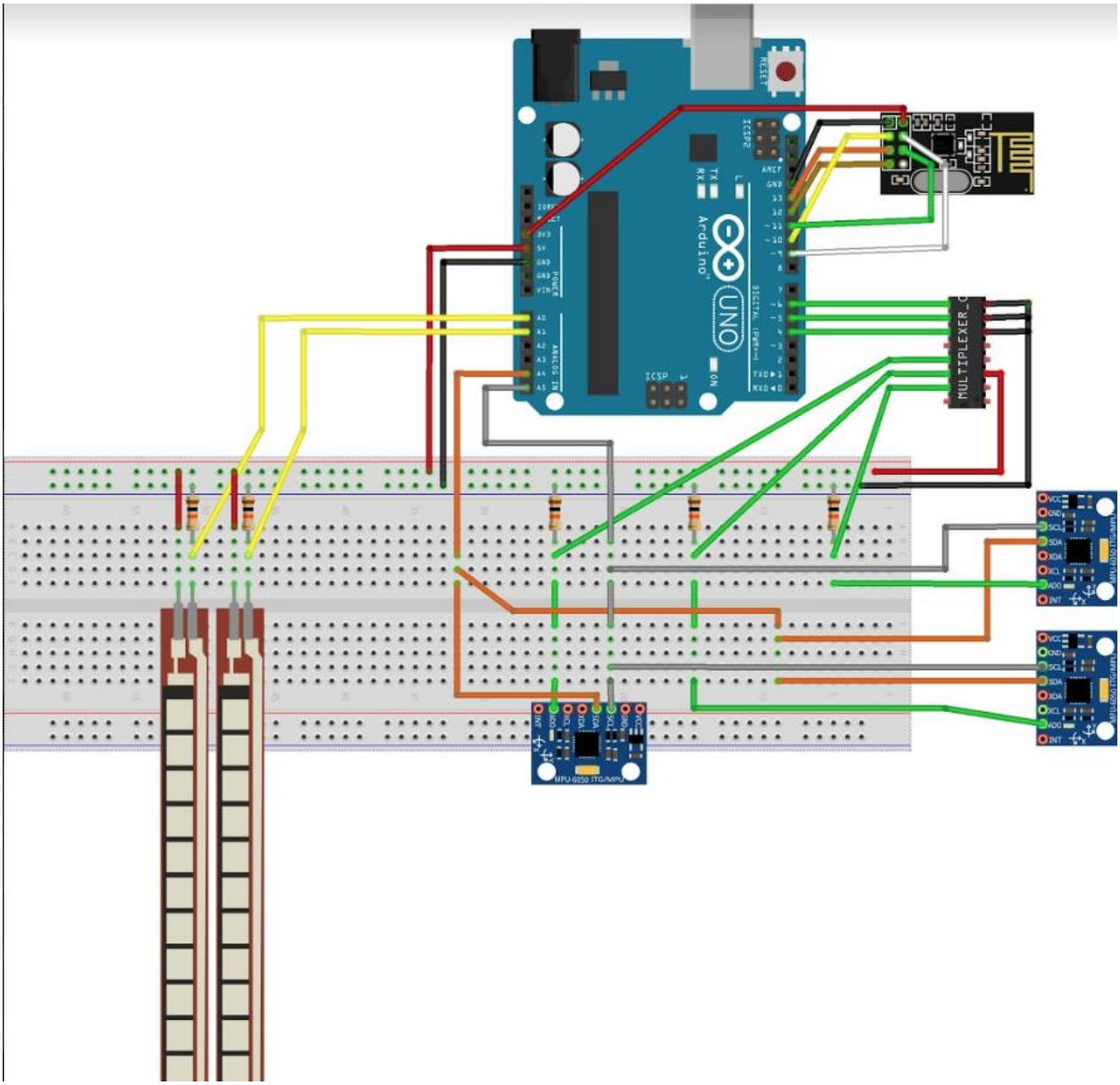


Figure 2: The schematic for the circuit of the glove/sleeve controller.