

UNIVERSITY OF ILLINOIS AT URBANA - CHAMPAIGN

Earthworm Robot

Design Document

By

Kunakorn Puntawong
Zehua Li

TA:
Luke Wendt

Final Report for ECE 445, Senior Design, Spring 2017
TA: Luke Wendt

3 May 2017
Project No. 56

Abstract

Complex narrow terrains are difficult to traverse using mechanical robotics, and the locomotion patterns of earthworms provides biological inspirations for adaptable engineering solutions for these conditions. This project set out to construct a soft robot with shape memory alloy actuators and the controls for it to traverse a 2D environment. The capability of the SMA's to emulate the worm's circular compression and longitudinal compressions are implemented, while attempt on the worm locomotion is also experimented with. Additionally, a UI solution is proposed to control and communicate with the worm robot.

Keywords: robotics, bioinspiration, shape memory alloy, earthworm

List of acronyms:

SMA -	shape memory alloy
MCU -	micro-controller unit
NiTi -	Nickel titanium alloy, Nitinol
UI -	User Interface
PWM -	Pulse Width Modulation
UART -	Universal Asynchronous Receiver/transmitter
PCB -	Printed Circuit Board

Table of content

1. Introduction	4
1.1 Purpose	4
1.2 Objective	4
1.2.1 Goals and Benefit	4
1.2.2 Functions and Features	4
2. Design Overview	5
2.1 Block diagram	5
2.2 Hardware Designs	6
2.2.1 Actuator	6
2.2.2 Mesh	7
2.2.3 Arrangement	8
2.2.4 On board control	9
2.3 Software Designs	10
2.3.1 User Interface	10
2.3.2 MCU code	12
2.3.3 Control system	14
3. Requirements and Verification	15
3.1 System wide requirements	15
3.2 Control requirements	15
3.3 Actuation requirements	15
3.4 Sensor requirements	16
3.5 Power requirements	16
4. Costs	17
5. Conclusion	18
5.1 Accomplishments	18
5.2 Uncertainties	18
5.3 Ethical considerations	18
5.4 Future Work	19
6. Citations	20
6.1 Other Referenced Sources	20
7. Appendix	21
7.1 Appendix A: Requirements and Verifications	21

7.2 Appendix B: Schematics	24
7.3 Appendix C: Flowcharts	25
7.4 Appendix D: Rhombi Mathematics	27

1. Introduction

1.1 Purpose

The purpose of this project is to construct a soft robot designed to emulate the earthworms' shape and muscle with artificial material, and their peristalsis movement with electrically powered actuators and computer controlled locomotion. The peristalsis movement approach allows the robot to traverse through rough terrain, goes through rubble and even climb tubes; a task that can be problematic for conventional robots. The soft robot construction is also more durable and are less prone to hardware failures due to its flexibility, its ability to absorb impact and its lack of mechanical element.

1.2 Objective

1.2.1 Goals and Benefit

- Robot design mimics the earthworm movement and body shape changes
- Higher durability compared to that of conventional “hard” robot

1.2.2 Functions and Features

- Ability to traverse flat terrain with obstacles
- Ability to deform into different shape and configuration
- Remote control from desktop using bluetooth protocol

2. Design Overview

2.1 Block diagram

The design has four main modules as shown in Figure 1: control, actuation, sensor and power. The control modules receives the command from the user from software UI, processes the command and sends the appropriate command over to the microcontroller unit (MCU) over bluetooth protocol. The actuation module which contains the MCU processes the command, taking sensors data into consideration and drives the pwm signals to the actuators. The sensor modules detects the change in the state and send the information over to MCU. Lastly, the power modules provides 5V power to the MCU and sensors and 9V to the darlington power transistor.

Details of each module are explained below in the requirement and verifications section, while we also attempt to explain the system from a bottom up view, from the mechanical components to the electronics to the controls.

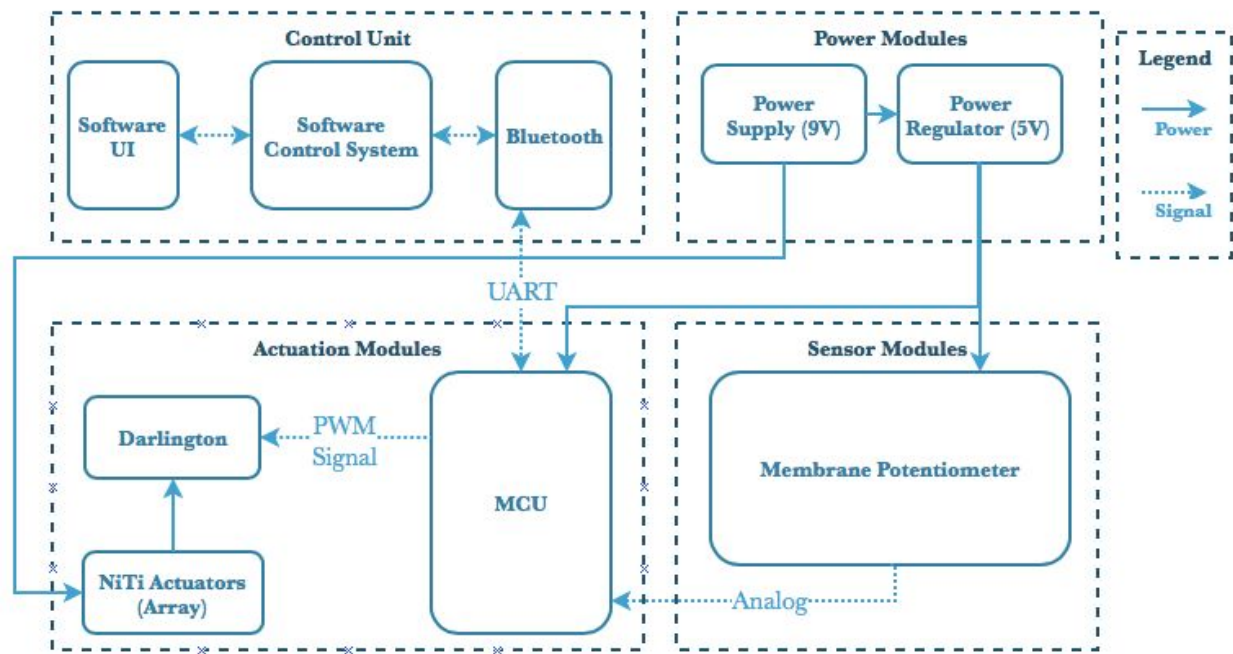


Figure 1. Block diagram of the robot and its control system

2.2 Hardware Designs

2.2.1 Actuator

The prototype actuator was manufactured with McMaster Carr Shape-Memory Nitinol Wire of 100um diameter into coils with the specifications of 200um inner diameter and 4cm in length, and baked at 400 degrees celsius to program the shape.

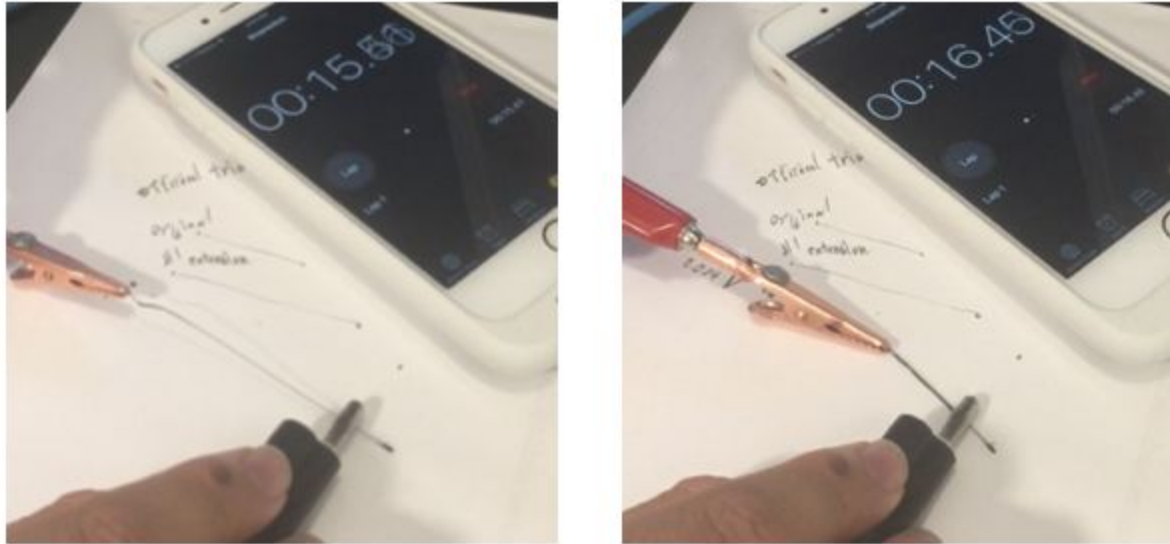


Figure 2. Nitinol coil before and after compression

The prototype actuator measured to have a resistance of 17 Ohms when contracted and 32 Ohms when extended further, without heating. It was tested with a voltage of 4.5V supplied by the lab power supply through the wire into the darlington pair, on the surface and suspended with wire. In both sets of trials, the actuator pulls itself and the segment of wire connected to it, with an average time of around 1 second for the wire to go from the extended form to the contracted form. In the suspended trials, the wire is able to retain the contracted form as long as current is supplied, and it falls back to the extended form when the current is no longer supplied as the weight of itself and mostly the wire pulls it down.

Due to an unaccounted for sudden decrease of human resources in our machine shop, it was no longer able to create more coils with the original setup. We instead used a 200um diameter wire from Dynalloy INC. The thicker wire allowed our machine shop to manufacture actuators with less time consumption, and allowed us to experiment with finished actuators during the time allowance of this project.

2.2.2 Mesh

The mesh body is an integral design to the earthworm. The mesh has a property that when it is compressed circumferentially, it expands longitudinally. The mesh body is composed of multiple rhombuses, so we can model the mesh property using a single Rhombi. In Figure. 3, when the rhombi compresses vertically, it expands horizontally. The mathematics of the rhombii compression is demonstrated in Appendix D.

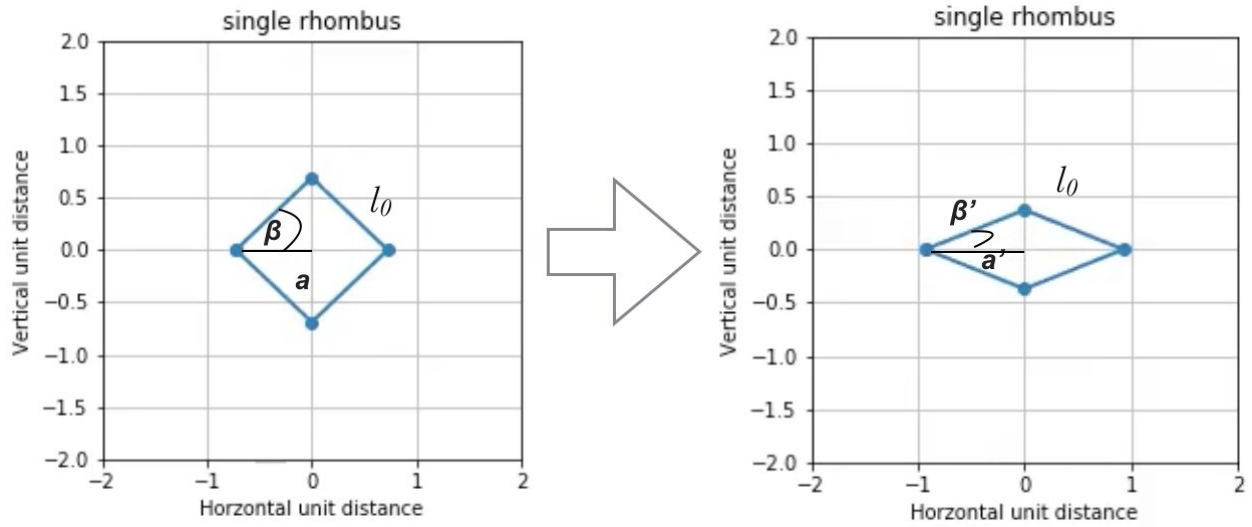


Figure 3. Rhombi vertical compression

After figuring out the mathematics of the compression, we began searching for the appropriate mesh body that has the right stiffness and flexibility. After experimenting with 30 different types of meshes, we discovered the meshes that work in the application of cable management, particularly braided sleeving. However, the polyethylene braided sleeving that we ordered were not capable of withstanding the heat generated by the actuator and melted the mesh.

In the end, the body of the earthworm robot is made from a Polyether ether ketone (PEEK) braided sleeving that is widely used in aerospace industry. The PEEK braided sleeving has a melting of 343 degree celsius and is extremely flexible[1]. Experimentally, it has worked well with the nitinol actuator and has not shown any sign of fatigue after multiple used.

2.2.3 Arrangement

The four circumferential actuators are arranged around the mesh sleeve along the worm body on the outside, woven with coated wires in regular intervals so they don't come in contact with themselves, short out part of the actuator and bypass the current. The longitudinal actuators are placed inside the worm body and fastened onto the side in regular intervals. They are placed on the horizontal bisecting plane of the worm to bend it left and right. They also serve to maintain the length of the worm actively while the circumferential actuators compress to change the shape of the mesh sleeve locally.

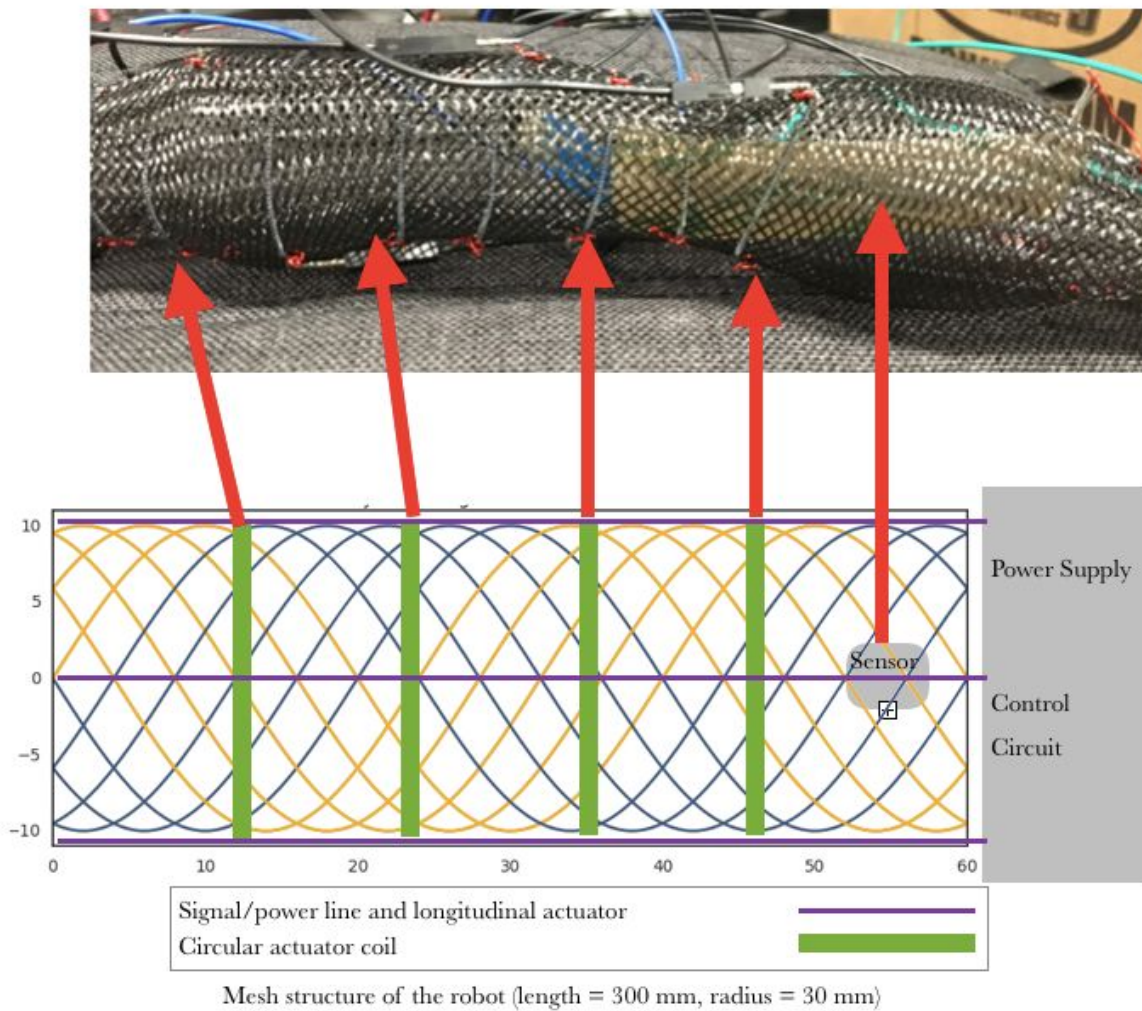


Figure 4. Arrangement of worm body, actuators, and sensors

2.2.4 On board control

The onboard control Unit communicates via serial communications with the control unit. It processes the commands sent from the computer, signals from the sensors, and and feedbacks from the actuation modules. It controls the behavior of the robot and sends feedbacks back to the computer regarding the status of the robot and the execution of commands.

The MCU (dsPIC30f4011) receives the commands from the computer through the communication link and transfers it into the corresponding PWM signal to drive the actuators via darlington transistor pairs. The power of the actuator varies according to the duty cycle of the input signal.

The Darlington transistor (ULN2003A) [2] receive the control signal from the MCU and regulates the current supplied to the NiTi actuators. The current is supplied from a 9V source, passes through the actuators in parallel, reaches each of the darlington transistor pairs controlled by the output PWM of the MCU, and then to the collector of the darlington transistor chip. Schematics of the PCB is included in Appendix B.

2.3 Software Designs

The software unit helps to realize the user commands within the capabilities of the robot. The software flowchart is included in Appendix C.

2.3.1 User Interface

The user interface runs on the computer which the user has access to. We constructed the user interface with python, and it runs on jupyter notebook, with the elements rendered with ipywidget [3]. Essentially, the user interface executes code in the kernel and is connected to the local web interfaces elements that are displayed in the notebook interface.

User interface has 4 major purposes:

- 1) to communicate with the worm
- 2) to accept user inputs
- 3) to display system outputs, and
- 4) to convert user input into a format the MCU can successfully interpret.

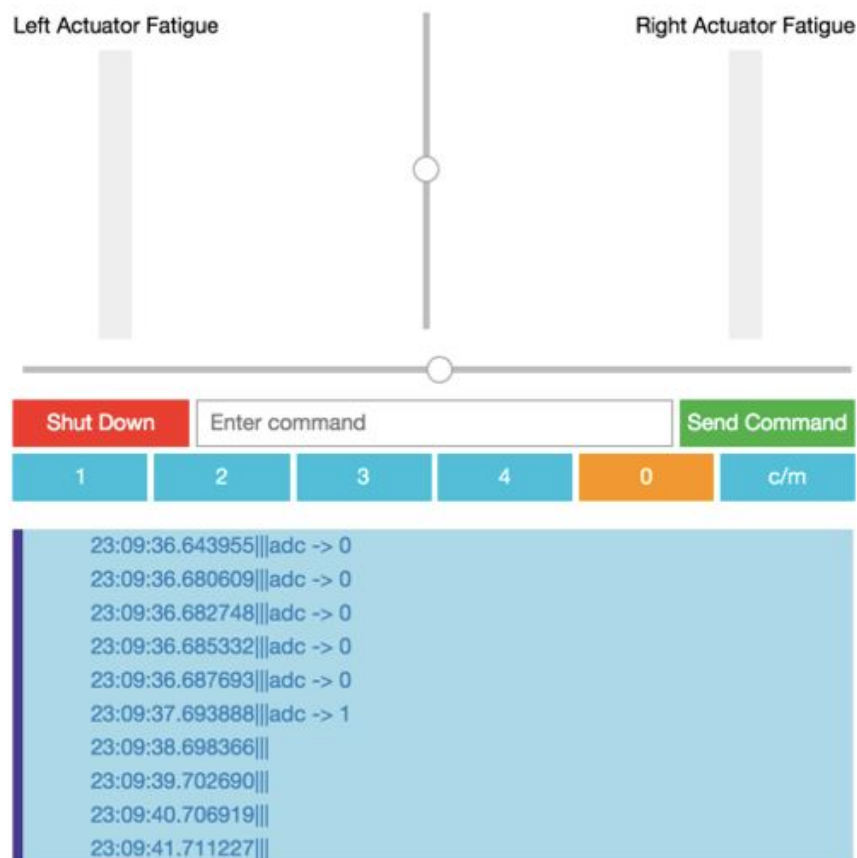


Figure 5. Python Based User Interface

The UI communicates with the on board system (MCU) via a HC-06 bluetooth chip with UART. The communication uses a baud rate of 9600, and once connected, the python program operates on two threads, one receiving thread that listens from the MCU, and one transmitting thread that listens from the jupyter notebook interface. Information coming from the MCU are processed on the receiving thread, and the result is then reflected onto the UI in the blue console area. The flowchart is included in Appendix C.

The UI is capable of interpreting three different schemes of user input: slider, text input, and button presses. The vertical and horizontal sliders are used to control the locomotion directions (forward/neutral/backward) and turning directions(left/middle/right), respectively. The same functionality can be accomplished with “WASD” inputs in the command text box. These keystrokes are detected and processed without the need to press the “Send Command” button. More complex text input can also be executed, and longer input are started with the character “!” to distinguish them from the one key keystrokes. The buttons 1-4 control individual circumferential actuators, 0 resets these actuators, and “c/m” contracts/releases both longitudinal actuators and provides a way to manually tune the actuation sequence and the expansion/contraction of the worm body. The flowchart is included in Appendix C.

For the “WASD” inputs, the locomotion controls and the turning controls are then processed via a set of separated state machines to decide what command to send to the worm. The left and right actuations are additionally controlled via individual timers to prevent overheating. The timers model the heating and cooling processes crudely via a linear model due to how rapid they are (both timers last 10 seconds in total). In more precise applications, it is advised to determine the this via feedback control or a more accurate model.

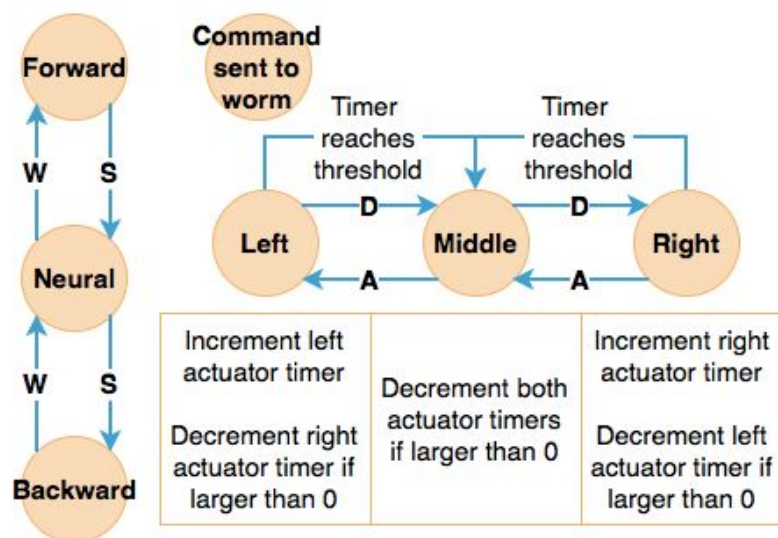


Figure 6. control state machines

2.3.2 MCU code

The program onboard the MCU perform two major task: communication and motion and body planning. The communication between the user interface and the MCU happens through the bluetooth protocol. The user interface program sends specific command (forward, turn right, etc) to the HC-06 bluetooth. The HC-06 then communicate with the MCU using UART protocol which enables the interrupt. The algorithm in Figure 7 determines whether the commands sent is a direction change command or whether it a locomotion change. If it is the former, the worm will immediately change direction. However, if it is the latter, the worm will wait for the worm to finish its locomotion cycle then execute the new locomotion that was sent.

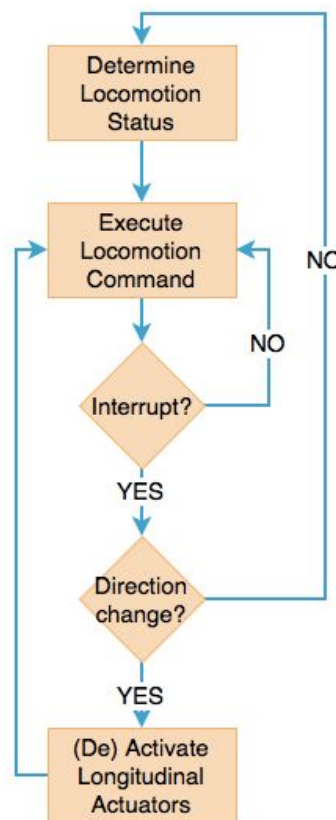


Figure 7. Execution Algorithm

The second task is for the MCU to execute the motion that was commanded. There are several function in the program that contains a sequence that sends an appropriate PWM signal to the actuators as shown in Figure 8. If the commands are either to turn left or right, the actuator along the length of the body will be immediately contracted. However, if the commands are for the worm to either go forward or go backward, the MCU will send a series of pwm signal to each of the actuator for the worm as shown in figure 9.

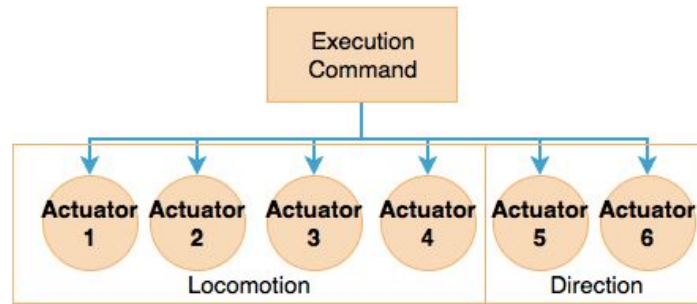


Figure 8. Actuators partition

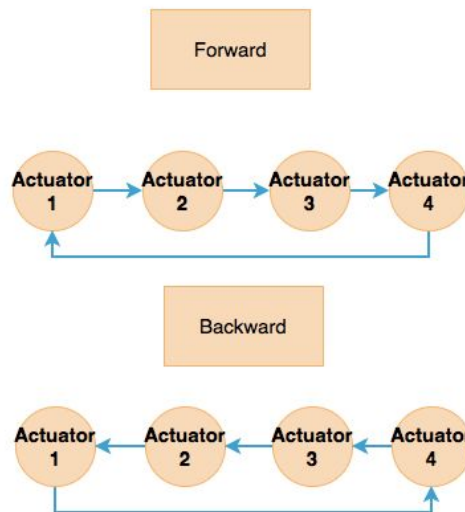


Figure 9. Forward/backward locomotion sequence

2.3.3 Control system

Our placement of the potentiometer resulted in the sensor bending during the actuation sequence and the force with the contact point on the worm fluctuating too much to provide good readings. Therefore we took out the potentiometer and worked with a closed loop control system. Continue using the model representation of the mesh as rhombuses, we model each segment controlled by the actuator as a single rhombus as shown below. Given that L is the side length; d_{max} is the maximum angle achievable with horizontal axis, and d_{min} is the minimum angle achievable with horizontal axis, then:

$$\text{Displacement per sequence of movement} = 2(L * \cos(d_{min}) - L * \cos(d_{max})).$$

However, due to the stiffness of the wires, the individual rhombuses were unable to spring up to the fully relaxed states after their activation periods. Resulting in a lack of displacement of the vertices touching the ground, and subsequently the lack of displacement of the worm as a whole.

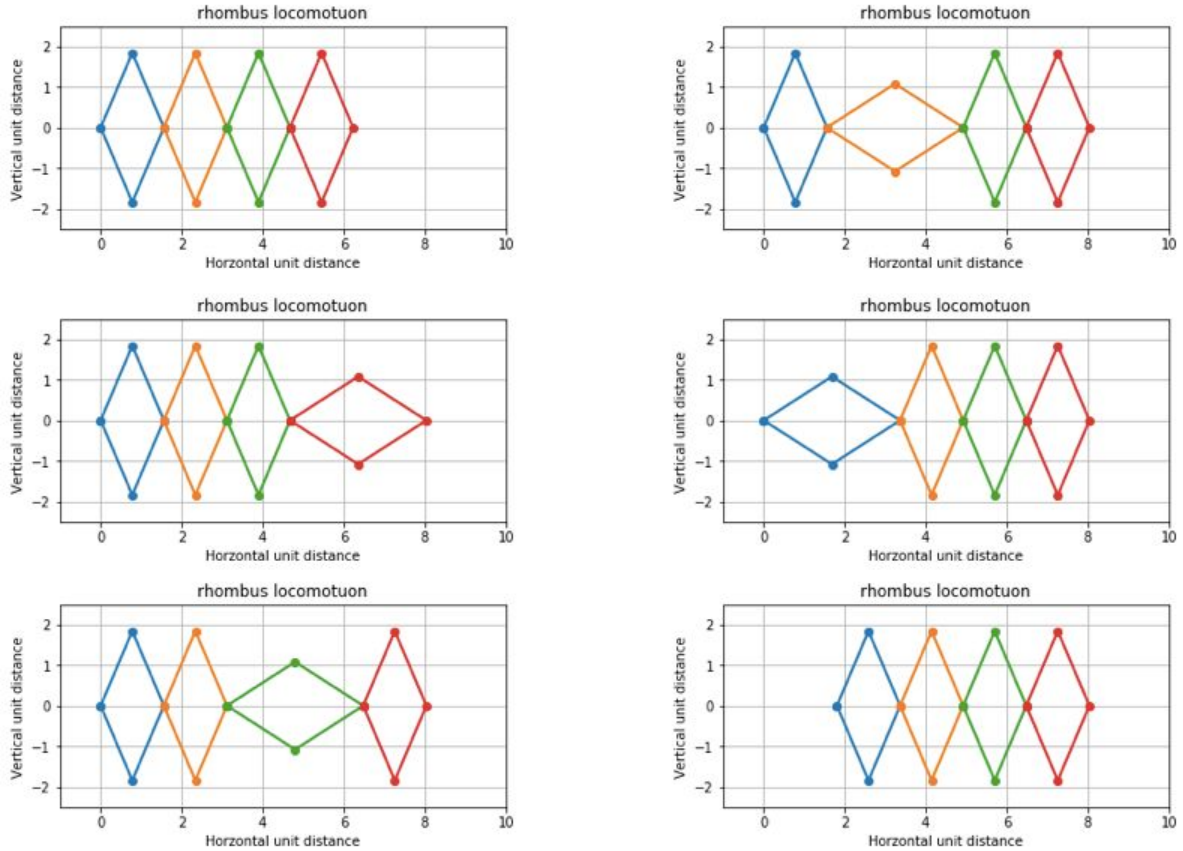


Figure 10. Locomotion pattern implemented in the open loop control

3. Requirements and Verification

3.1 System wide requirements

There are several tests required to verify that the overall earthworm robot works and these tests required the use of every module in the system. First, we verified that our earthworm robot can perform locomotion by sending a command from the worm to move forward and measure that the speed of the robot meet or exceeds the required speed. In the end, this earthworm robot did not fulfill the requirement due to failures to manufacture a small enough nitinol wires as per original design.

However, we were able to fulfill the other system wide requirement. To do this, we send commands for the worm to perform the corresponding action and take measurement on the worm to verify that the worm meets the requirement. The verification process are shown on the table in Appendix A

3.2 Control requirements

To verify the requirement that the sensor feedback response time is less than 150 ms, we create a customize function. This function, inside the MCU, measures the time difference between two events: reading the potentiometer value of the buffer and the time that the buffer is being filled again. Similarly, to verify the requirement that the user input response is less than 150ms, our function measure the two way communication time between the MCU and the computer UI. This function measure the time difference between sending a message from the computer UI and receiving an acknowledgement message from the MCU.

3.3 Actuation requirements

To verify that both the longitudinal and circumferential actuator meets the requirement, we pass power from the bench power supply to the nitinol actuator. The voltage is then increased until either the actuator burned out or the actuator was able to compressed the worms to meet the requirement. We also verify that the driver module met the requirements by driving a simulated load for 30 minutes. In the end, all of the actuation requirement was fulfilled.

3.4 Sensor requirements

After initial testing of the band flexible potentiometer, we found out the sensor setup that we initially planned was not responsive enough and not accurate enough. So, we abandoned the sensor and the open loop control system and go with the simpler closed loop control system. As a result, the sensor requirement was not met.

3.5 Power requirements

To verify that the power regulator works, we uses it to power the circuit and make sure that it worked after 30 minutes. Lastly, to verify that the power supply works, we connect to a simulated load and make sure that it worked after 30 minutes. The overall complexity of the power module is very low, so we did not allocate any points toward it.

4. Costs

Distributor	Items	Quantity	Cost per unit	Total
CableOrganizer.com	PET braided sleeving 3/4"	5	\$1.30	\$6.50
CableOrganizer.com	PEEK braided sleeving 3/4"	5	\$4.80	\$24.00
Digi-key.com	Darlington transistor: ULN2003A	4	\$0.50	\$2.00
Dynalloy, inc	Flexinol Acuator Wire: 200micron 70 Celcius	50	\$1.37	\$68.50
Dynalloy, inc	Flexinol Acuator Wire: 100micron 70 Celcius	80	\$1.37	\$109.60
ECE Shop	PCBs	3	\$4.70	\$14.10
McMaster Carr	PEEK mesh	4	\$30.81	\$123.24
McMaster Carr	Nitinol Wire	30	\$1.19	\$35.70
Microchip	PICkit 3 In-Circuit Debugger	1	\$47.95	\$47.95
Mouser.com	Microcontroller: DSPIC30F4011-30I/P	4	\$6.84	\$27.36
Mouser.com	Clock Oscillator: SG-615P	4	\$2.25	\$9.00
Mouser.com	Passive Elements (Resistors, Capacitor, etc)	1	\$10.00	\$10.00
Sparkfun	SoftPot Membrane Potentiometer 50mm	5	\$4.95	\$24.75
Sparkfun	IMU: MPU-9250	2	\$14.95	\$29.90
Parts Sub-total			\$132.98	\$532.60
Service	personal	Hours	Hourly rate x2.5	Total
Computer Engineer	Kunakorn Puntawong	225	\$100.00	\$22,500.00
Computer Engineer	Zehua Li	225	\$100.00	\$22,500.00
Labor			\$200.00	\$45,000.00
Grand Total				\$45,532.60

5. Conclusion

5.1 Accomplishments

We were able to create a robot that is inspired by the peristalsis movement of the earthworm through the use of artificial muscle constructed from nitinol actuator. The robot was able to contracts and expands in each of its section which is an action required for locomotion.

Furthermore, the robot and its PCB was fully constructed with the control, actuation and power modules in place as per design. Our web application and program were able to communicate with each other to control the robot.

5.2 Uncertainties

We were not able to get the robot to perform locomotion which is the initial goal. This is mainly contributed by the lack of precision machining small actuator ,so we were forced to used actuator that is twice as large and four times as stiff. The elastic force of the braided sleeving was not able to counter the extra stiffness of the actuator and caused the movement to be different than initially planned. As a result, our robot, instead of moving forward, twitches in place.

5.3 Ethical considerations

Abiding to IEEE code of Ethics [4], the experiments were conducted with sufficient safety mechanisms (insulation to shock and heat, power regulations). If conducting similar experiments, please keep in mind the potential shock and fire hazard the live and heated actuators may bring, and deploy proper safety measures. Demonstration and experimentations should not be done in public without ensuring the safety measures of the location and consulting local authorities.

The device has a rather large scope and we welcome all criticism and suggestions to its improvements and regarding its flaws. If circumstances where specific modification or improvement are required, we are responsible to provide assistance. Similarly, we welcome any inquiries regarding the research process and any specific aspects.

Our software design and implementation will abide to the ACM Code of Ethics and Professional Conduct [5]. The control system and algorithms draw inspirations from various previous works and we honor the property right including copyrights and patent. Similarly, we will keep an open attitude towards future projects and provide references to replicate any experimentations and to facilitate further improvement of designs and implementations.

5.4 Future Work

In the future, we would like to improve the robot by using an actuator which is smaller and is better manufactured. This would result in a robot that is more energy efficient, has faster response time and has improved locomotion ability. Also, with the improvement in actuators' energy efficiency and future battery technology, we are confident that in the near future, we will be able to eliminate the power adapter and make the worm completely wireless. Last but not least, we would like to redesign the PCB so that it is small enough to be place within the worm's body.

In addition to improving the robot design, we also would like to apply the nitinol actuator technology to other application such as lightweight prosthetic and other soft robots as well. Nitinol actuator has an extremely power to weight ratio which allows them to potentially replace conventional mechanical part such as motors and solenoid. However, they have an extremely poor power efficiency so further research in the field of shape memory alloy is needed to improve the performance.

6. Citations

- [1] “Flexo PEEK Advanced Engineering Technical Data Sheet” [Online]. Available: <http://cdn.techflex.com/assets/pdfs/specification-sheets/pek.pdf>. [Accessed: 03-May-2017].
- [2] “Flexinol Control Circuit Using PIC 16F690 and ULN2003A.” [Online]. Available: <http://robotics.hobbizine.com/ulndrive.html>. [Accessed: 03-May-2017].
- [3] “ipywidgets — ipywidgets and jupyter-js-widgets 7.0.0a1 documentation.” [Online]. Available: <http://ipywidgets.readthedocs.io/en/latest/>. [Accessed: 03-May-2017].
- [4] “IEEE Code of Ethics,” in *Encyclopedia of Software Engineering*, 2002.
- [5] R. E. Anderson, “ACM code of ethics and professional conduct,” *Commun. ACM*, vol. 35, no. 5, pp. 94–99, 1992.

6.1 Other Referenced Sources

- [6] H. Song, E. Kubica, and R. Gorbet, “Resistance Modeling of SMA Wire Actuators” NDT in Canada, Nov 2011 [Online]. Available: http://www.ndt.net/article/ndtcanada2011/papers/66_Gorbet_Rev2.pdf Accessed: Feb. 5, 2017
- [7] "Independent segments enable burrowing through narrow spaces: Earthworm," in *asknature.org*, AskNature, 2016. [Online]. Available: <https://asknature.org/strategy/independent-segments-enable-burrowing-through-narrow-spaces>. Accessed: Feb. 2, 2017.
- [8] K. Ōtsuka and C. M. Wayman, *Shape memory materials*. Cambridge: Cambridge University Press, 1998.
- [9] Microchip, *dsPIC30F Family Reference Manual High-Performance Digital Signal Controllers*, Microchip Technology Inc., 2006.
- [10] Microchip, *dsPIC30F4011/4012 Data Sheet High Performance Digital Signal Controllers*, Microchip Technology Inc., 2005.
- [11] T. Burke, "dsPIC30F4011 Examples," 2012. [Online]. Available: https://batchloaf.wordpress.com/dspic30f4011_super_example/. [Accessed 15 March 2017].
- [12] L. Xin, E. Huang and S. Mok, *Product Data Sheet Module HC-06*, Guangzhou HC Information Technology Co., Ltd., 2011.
- [13] S. Seok, C. D. Onal, K.-J. Cho, R. J. Wood, D. Rus, and S. Kim, “Meshworm: A Peristaltic Soft Robot With Antagonistic Nickel Titanium Coil Actuators,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 5, pp. 1485–1497, 2013.
- [14] Krishnan G, Bishop-Moser J, Kim C, Kota S. Kinematics of a Generalized Class of Pneumatic Artificial Muscles. *ASME. J. Mechanisms Robotics*. 2015;7(4):041014-041014-9. doi:10.1115/1.4029705.

7. Appendix

7.1 Appendix A: Requirements and Verifications

System wide requirements					
Requirement	Quantification	Verification	Completeness	Results	Points
Maximum movement speed	≥ 10 m/h	1. Drive the robot with the signal to achieve the highest velocity over a certain distance and measure the elapsed time. 2. Use the elapsed time to calculate the speed.	The objective of the project is to create a robot inspired by an actual earthworm so having locomotion in the forward direction is a major requirement. The speed we choose is that of a small earthworm found in nature.	Failed, the robot could not perform locomotion.	7
Maximum bending radius	≤ 0.5 meter	1. Command the robot to perform maximum bends. 2. Measure the radius of arc of the bends	The robot, when deployed, should be able to navigate in environment with obstacle. So, having a small bending radius is necessary.	Success, the maximum bending radius is 0.6 meters	4
Average cross-sectional weight	≤ 0.6 kg/m	Weight the robot and divide by length	The robot has to be light and small enough to traverse into rough and dense terrain without being noticed.	Success, the average cross-sectional weight is 0.3 kg/m	4
Maximum longitudinal extension (from contracted form)	$\geq 20\%$	1. Send the command to the robot to perform maximum extension. 2. Measure the length before and after the extension	This is to test that the longitudinal muscle has enough force for the worm to extend longitudinally: an action needed for locomotion. It is also to test to overall system requirement: bluetooth, pwm, power driver and actuator.	Success, the maximum longitudinal extension is 27% from its contracted form.	6
Maximum circumferential contraction (from extended form)	$\geq 10\%$	1. Send the command to the robot to perform maximum contraction. 2. Measure the circumference before and after the contraction.	This is to test that the circumferential muscle has enough force for the worm to compress the cross section of the worm: an action needed for locomotion. It is also to test to overall system requirement: bluetooth, pwm, power driver and actuator.	Success, the maximum circumferential contraction is 68%.	6

Control Module requirements					
Requirement	Quantification	Verification	Completeness	Results	Points
Sensor feedback response time	≤ 150 ms between sensor input and control output	1. Provide artificial input to the MCU from a waveform generator, process the data with the algorithm, output the data to the user interface, and record latency between every entry. (Experiment for 1 minute.) 2. Verify that the average time difference between the two signals is less than 150 ms.	The worm sensors must be able to detect change in the state of the system and send the feedback for the control algorithm to respond appropriately in time.	Success, the average sensor feedback response time is 10.3 ms	4
User input response time	≤ 150 ms between user input and control output	1. Send a series of commands from the user interface to the MCU. 2. The MCU should attempt to execute the commands and provide feedback to the user interface. 3. Verify that the average latency between issuing a command and receiving the corresponding feedback is less than 150ms	The system should have reasonably fast user input response time such that the user can control the worm without much delay.	Success, the user input response time is 72.3 ms	4

Actuator requirements					
Requirement	Quantification	Verification	Completeness	Result	Points
Maximum body extension (longitudinally)	$\geq 20\%$ body length	1. drive all the actuator with the maximum current that the actuator can handle before permanent deformation 2. Measure the length before and after.	The nitinol actuator and mesh has to work together to contract worm body longitudinally. This is action is required for locomotion to work.	Success, the maximum longitudinal extension is 27%	4
Maximum body contraction (cross-section)	$\geq 10\%$ body circumference	1. drive all the actuator with the maximum current that the actuator can handle before permanent deformation 2. Measure the circumference before and after.	The nitinol actuator and mesh has to work together to contract worm body circumferentially. This is action is required for locomotion to work.	Success, the maximum circumferential contraction is 68%	4
Driver module	able to drive all actuator at 200mA peak-to-peak with 50% duty cycle over 30 minutes.	1. Send the command to the robot to generate a 200 mA peak-to-peak signal to drive the nitinol actuator. 2. Wait for a period of 30 minutes or until the transistor stop working. 3. Verify that the driver module is still operational after cooling down.	The driver module has to provide enough power to heat up the nitinol actuator for it to contract. The driver also has to be reliable enough to withstand user abuses.	Success, the driver module is still functional after 30 minutes.	4

Sensor requirements					
Requirement	Quantification	Verification	Completeness	Results	Points
Band membrane potentiometer accuracy	$\pm 10\%$ horizontal extension measurement	1. Send the command to the robot to perform maximum extension. 2. Measure the length before and after the contraction. 3. Use the band variable resistor to infer the length before and after the contraction. 4. Verify that both inference is within 10% of the actual length.	The sensor is used to measure the accuracy and to feed data to closed loop control system to get a better performance than that of a closed loop.	Failed., the sensor did not meet the response time and was not accurate enough.	3

Power requirements					
Requirement	Quantification	Verification	Completeness	Result	Points
Power regulation	Able to provide 2.7 V to 3.8 V to onboard chips for 30 minutes	1. Drive the circuit with simulated commands for several 30 minutes intervals, and monitor the circuit voltage. 2. Verify the expected voltage is provided.	The voltage regulation has to provide enough voltage for the control module to operate.	Success, the power regulator was able to provide constant power to the logical component for over 30 minutes.	N/A
Power supply	Able to provide 4 W to the actuators for 30 mins	1. Drive the power supply with adequate load (or the actuators themselves) repeatedly for several 30 minutes intervals, and monitor the circuit. 2. Verify the expected power is provided.	A constant high current power is needed for the actuator to operate.	Success, the power supply was able to provide over 4 watts of power for over 30 minutes.	N/A

7.2 Appendix B: Schematics

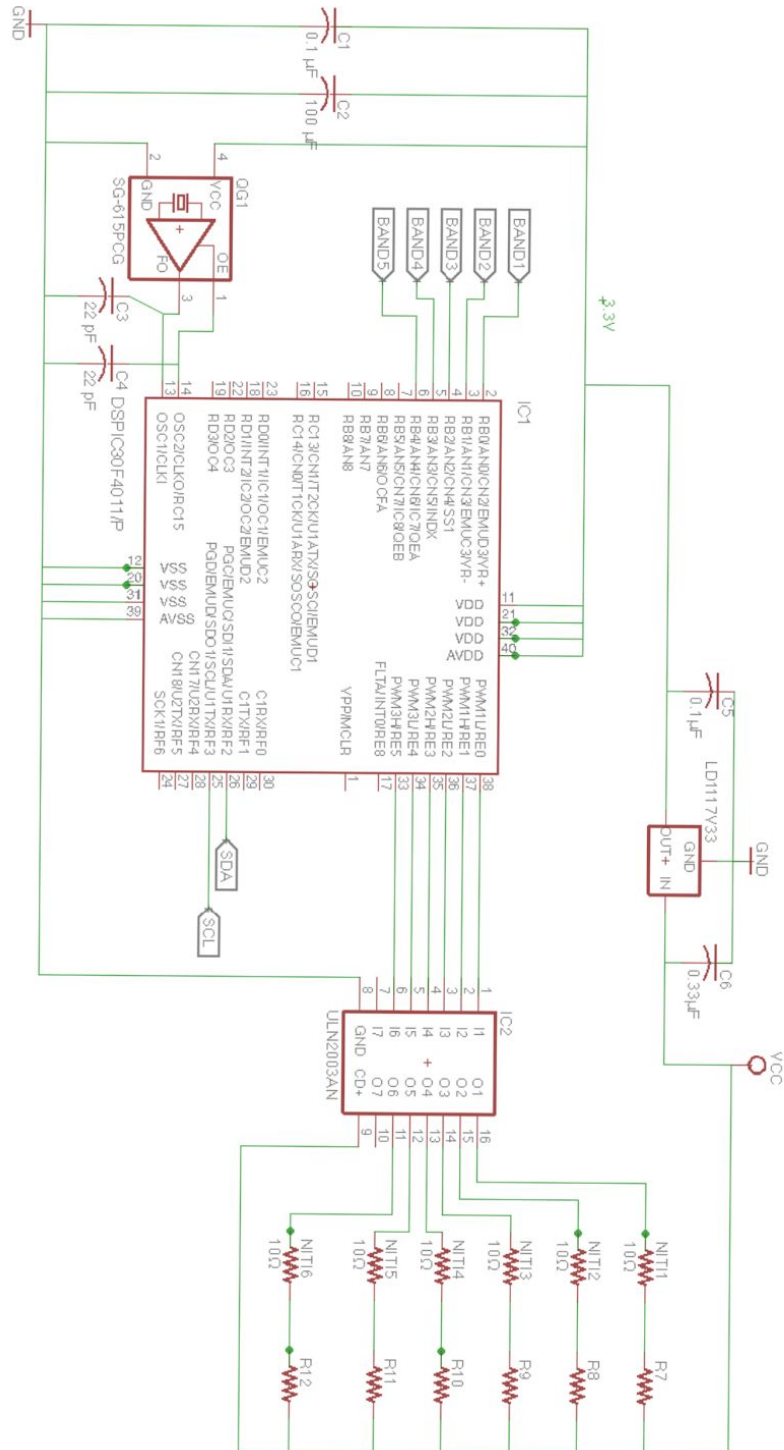


Figure 10. Circuit diagram

7.3 Appendix C: Flowcharts

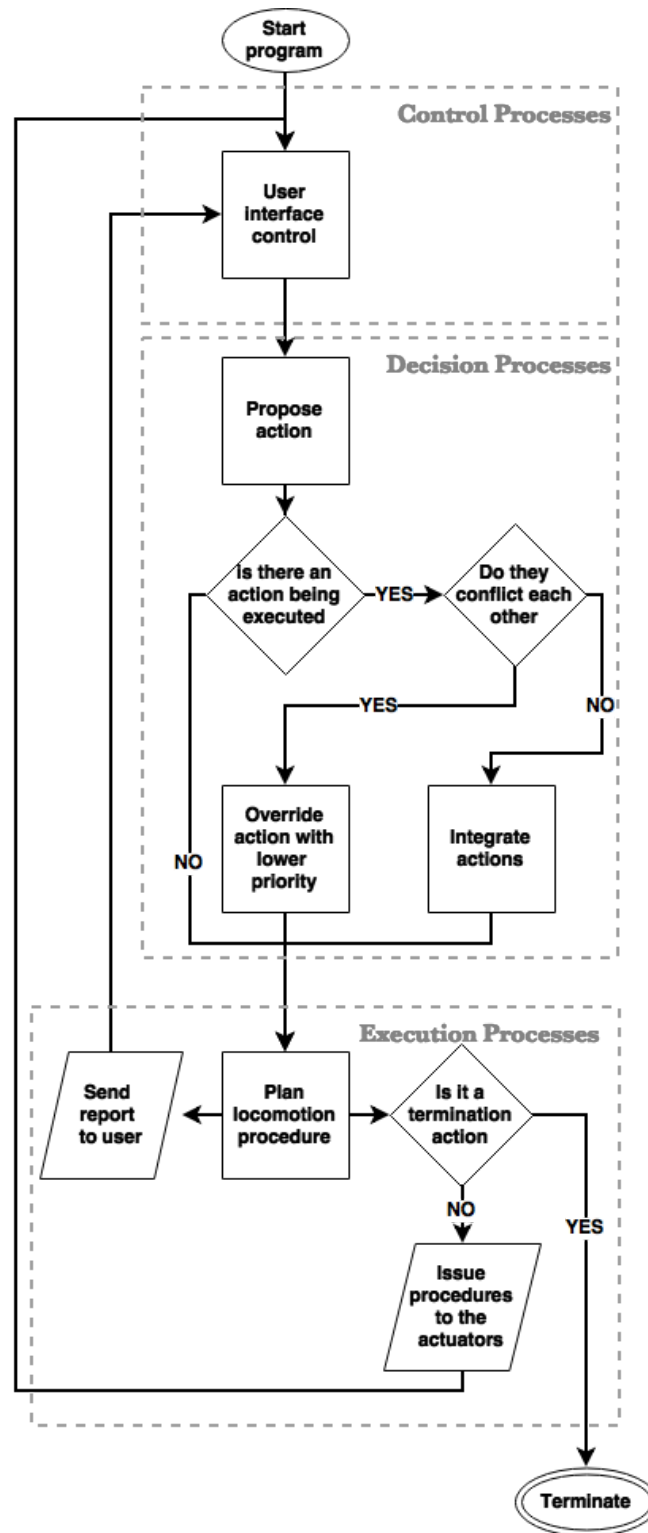


Figure 11. Software flow chart

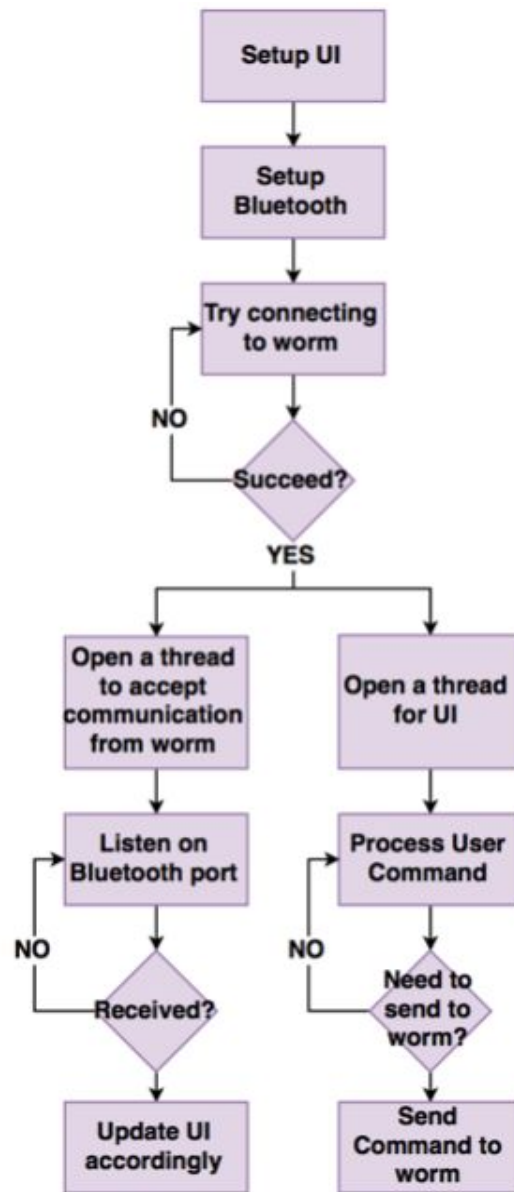


Figure 12. Communications flow chart

7.4 Appendix D: Rhombi Mathematics

β = angle between the side of the rhombus and the horizontal axis

$t = l_0 \sin \beta = \frac{1}{2}$ length of the vertical axis of the rhombus

l_0 = length of the side of the rhombus

C = Circumference of the Mesh

ΔC = Compression along the circumference of the Mesh

$a = \frac{1}{2}$ length of the horizontal axis of the rhombus

$\delta = \frac{\Delta C}{C} l_0 \sin \beta$ = compression per $\frac{1}{2}$ rhombus

$$t - \delta = l_0 \sin \beta'$$

$$\sin \beta' = \frac{l_0 \sin \beta - \delta}{l_0} = \sin \beta - \frac{\delta}{l_0}$$

$$a' - a = \Delta a = l_0 (\cos \beta' - \cos \beta) = l_0 \cos \left(\sin^{-1} \left(\sin \beta - \frac{\delta}{l_0} \right) \right) - l_0 \cos \beta$$

$$= l_0 \cos \left(\sin^{-1} \left(\sin \beta - \frac{\Delta C}{C} \sin \beta \right) \right) - l_0 \cos \beta$$