TEXAS STATE UNIVERSITY

The rising STAR of Texas

RH systems

E 1.3 – Prosthetic Hand with Neuromuscular Control

Francisco Recio, Marco Alcantara, Scarlett Head

RH Systems

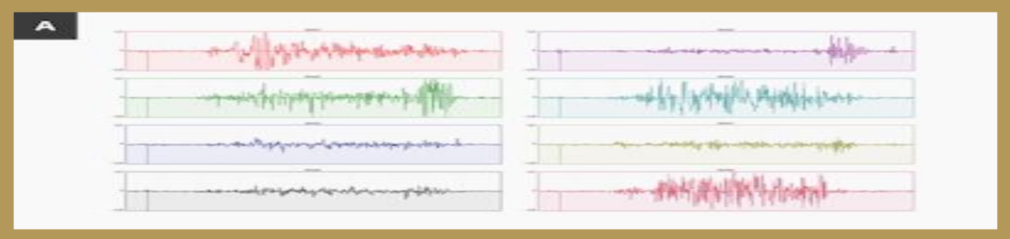
RH Systems is a privately-held, New Mexico based company that is dedicated to designing and providing temperature and humidity control metrology tools. They provide design expertise in embedded control and automation, including touch screen, user interface, and embedded firmware. Their products can be found in commercial laboratories and national metrology institutes around the world.

Overview/Description

This project will take the neuromuscular signals captured by the Myo armband, a gesture and motion control device, and produce a pulse-width modulation signal that can emulate the user's arm movements to the prosthetic arm. Our final product will be a mechanical arm that acts as a prosthetic hand and has two degrees of motion, which includes the opening and closing of the hand gripper as well as a vertical movement of the arm. Our group will create an innovative solution to aid amputees in performing everyday tasks that were unable to be done as efficiently, if at all, and to provide people with hostile work environments with a safe alternative.

Background

Electromyography (EMG) is a technique for evaluating and recording the electrical activity produced by skeletal muscles. The Myo armband is a wearable device that uses 8 electromyography (EMG) sensors and 9-axis inertial measurement unit (IMU) to capture electrical signals. The electrical sensors measure electrical signals traveling across the user's arm, which the Myo armband translates into poses and gestures.



Captured electrical signals from EMG sensors

The Myo armband connects via Bluetooth 4.0 Low Energy and provides two kinds of data, spatial data and gestural data. Spatial data informs the application about the orientation and movement of the user's arm, while the gestural data tells the application what the user is doing with their hands. The Myo armband is able to provide this data to applications in the form of events by using the free software development kit (SDK) that can be found on the Myo armband's online developer portal. The Myo SDK contains header files and link-time libraries that allows the Myo to control other application program interfaces (APIs) and includes documentation for each available function.

For this project, the Myo armband will capture electrical signals and will transmit them to an Arduino 101 microprocessor. The microprocessor will then send pulse width modulation (PWM) signals to a mechanical arm's motors, which in turn will cause the arm to mimic the hand and arm motions of the Myo wearer. Operation using PWM signals is desired because it has a very low amount of power loss.

Team Members

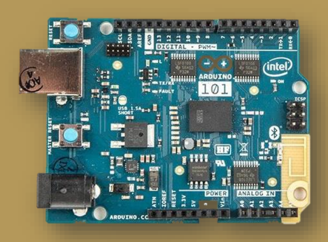
Francisco Recio
Project manager, Microprocessor code

Marco Alcantara
Assembly, Code implementation

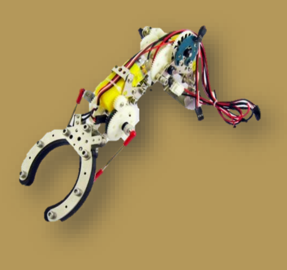
Scarlett Head
Testing, Simulating prototype

Hardware

Myo Gesture Control Armband



Arduino 101



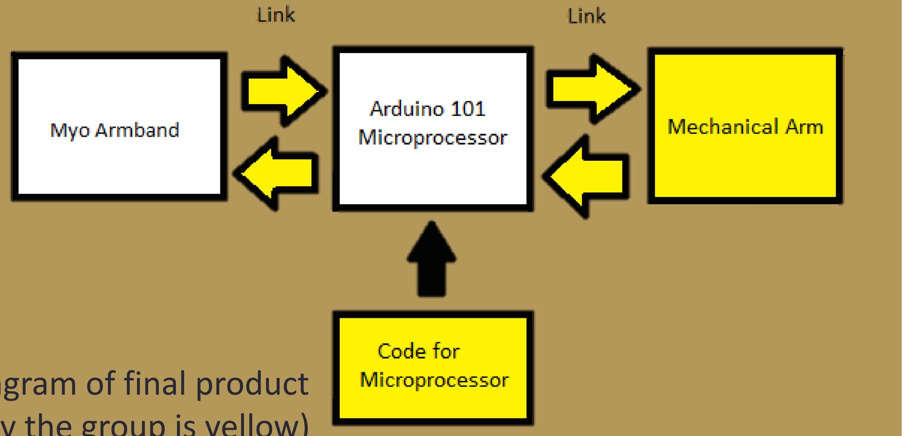
Mechanical arm

Arduino 101 Technical Specs	
Operating Voltage	3.3 V (5V tolerant I/O)
Input Voltage (recommended)	7-12V
Input Voltage (limit)	7-20V
Digital I/O Pins	14 (of which 4 provide PWM output)
PWM Digital I/O Pins	4
DC Current per I/O Pin	20 mA
Flash Memory	196 kB
Features	Bluetooth LE, 6-axis accelerometer/gyro

Approach

Our team's first requirement will be to establish communication between the Arduino 101 microprocessor and the Myo armband. This communication will be programmed with the help of the MyDuino libraries, which are libraries that have been developed to establish similar types of communication. When communication is successful, code will be implemented that reads neuromuscular signals from the Myo armband and sends corresponding PWM signals from the microprocessor to the arm. The programing language that will be used is C++. A simulated prototype will be created beforehand in order to test the motors of the robotic arm. The program will run tests in order to determine the limitations of the mechanical arm. The mechanical arm will be assembled once the motors are successfully tested. Listed below is a group of preset gestures that the Myo will be able to communicate to the mechanical arm. Once the physical limitations are recognized, error handling code will be added in order to indicate the user that a limit in the mechanical arm has been reached. Code will then be modified accordingly to smooth out movements.

<u>Preset Gestures:</u>
- Open Hand
- Close Hand
- Elbow Up
- Elbow Down



Schematic diagram of final product
(Work done by the group is yellow)

Challenges/Concerns

- Material of mechanical arm
- Simultaneous movements with mechanical arm
- Simultaneous gestures with the Myo Armband
- Bluetooth range of the armband
- Two-way communications with the Arduino 101

1st Semester Progress

Analysis of servo motor:
- Motor operating voltage = 4.8 - 6.0V
- Torque = 0.324 N·m
- Speed = 0.21 s/60°
- Direction Clockwise = 1500-1900 µs
- Rotation = 180°

Battery pack used to power servo motors:
- Chemistry: Nickel Metal Hydride
- Nominal Capacity = 2000 mAh
- Nominal Voltage = 6 V

Research completed for:
- Myo Armband capabilities
- Mechanical arm movements and constraint
- Arduino 101 capabilities

Parts procured/purchased:
- Myo Armband
- Arduino 101

Planned Work

- Purchase mechanical arm
- Design software interface by writing the code to be used for the microprocessor
- Develop communication/link between Myo armband and Arduino 101 microprocessor
- Assemble the mechanical arm
- Control motors for mechanical arm
- Test and finalize code for microprocessor
- Implement movement of arm with gestures while wearing Myo device

Expandability

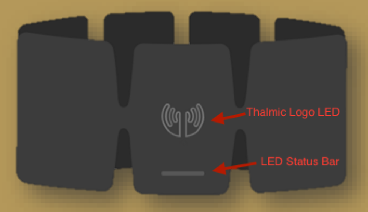
- Greater customizable gestures
- Genuine prosthetic hand
- Providing sensors on the arm to accurately determine the physical limitation of the arm under the current conditions
- Use Myo technology on other parts of the body

Boundary Conditions

- Battery life of the Myo
- Battery life of the mechanical arm
- Degrees of freedom
- Bluetooth signal distance

Error Handling

- The Myo armband will vibrate to notify the user when the mechanical arm is at its maximum range or attempts to exceed the maximum range
- The Myo armband handles its own errors through the LED Status Bar and Thalmic Logo LED on the armband



Testing Criteria

Mechanical Performance Parameters		
Function	Description	How Tested
Open Gripper Angle	A minimum angle of 150° with a ±2 tolerance	Open gripper to maximum range to measure angle
Closed Gripper Angle	A maximum of 1° with a ±2 tolerance	Close gripper to check if it closes all the way
Angle of Elbow	A minimum of 80° with a ±2 tolerance	Move elbow to maximum range. Measure the angle relative to 0°
Gripping Force	A minimum of 1 N	Place force sensor in between gripper then attempt to close
Lifting Capacity	A minimum of 100 g	Place 100 g weight in gripper and attempt to lift it

Software Performance Criteria
- Determine that a link was successfully made with the Myo armband and Arduino 101
- Determine that the Myo armband can successfully move servo motors in desired direction
- Determine that Myo can control movement of arm within desired range

Hardware Performance Criteria
- Ensure Arduino 101 has the proper voltage and current to drive servo motors
- Ensure each component is given the proper power to operate

Our project's success will be determined if the mechanical arm mimics the movements of the user wearing the Myo, and if the Myo alerts the user by vibrating when an error occurs with the mechanical arm.

Acknowledgements

Faculty Advisor: Dr. Wuxu Peng
Instructor Dr. Richard Compeau

Sponsor
Dustin Hardy
RH Systems
3416 Vista Alameda NE
Albuquerque, NM 87113