Paper ID #24691

Impoved Hardware Design of IoT Prosthetic Device

Dr. Yu Wang, The New York City College of Technology
Warren Hunter
Mr. XiaoLin Chen, NYCCT
Housney Ahmed
Haneefah Safo

Improved Hardware Design of IoT Prosthetic Device

Yu Wang, Warren Hunter, Xiaolin Chen, Housney Ahmed, Haneefah Safo

New York City College of Technology of the City University of New York

Abstract

Our previous IoT based prosthetic arm prototype used servo motors to control finger movement through an Arduino Mega, which is connected to the muscle, pulse, and temperature sensors. The Arduino Mega was also connected to a Raspberry Pi 3 model B to transfer data from/to an online web application. One major limitation encountered during testing this prosthetic device was the space occupied by these components, which makes the device bulky. In addition, these servo motors cannot control the movement of the prosthetic device precisely. In this paper, we propose to improve on the existing prosthetic limb prototype by transitioning the electromechanical system to linear actuators and replacing the larger Arduino Mega and Raspberry Pi 3 with the smaller Arduino Nano and Raspberry Pi Zero W, respectively. These changes will result in a more cost-effective, more stable, and more accurate prototype, resulting in better performance.

Keywords

Arduino, Motors, Linear Actuator, Prosthetic, Raspberry Pi, Sensor Network

1. Introduction

The field of prosthetics focuses on the development of artificial limbs to help restore functionality to persons who have lost or are missing one or more limbs. Early prosthetics made use of wood, copper, iron, and steel, but had limited functionality. At present, modern technology, such as 3D printing and connected devices, can be used to produce advanced prosthetic limbs. Nearly two million amputees are living in the US, and 54% of them has vascular diseases including diabetes and peripheral arterial disease [1]. Patients in this category can see benefits from the application of connected devices to the area of healthcare devices. These benefits may include the ability of doctors to remotely monitor patients with chronic and long-term ailments, allowing patients themselves to monitor their own data, and allowing caregivers to receive relevant data quickly. With these goals in mind, an Internet of Things (IoT) and mobile sensor platform based prosthetic device and prototype were developed [2][3][4][5]. Our undergraduate research team at the New York City College of Technology designed a prototype of an IoT system for Myo-Prosthetics [6]. Our prototype consists of microcontrollers, web-based components, electromechanical system, and sensors that can transmit the user's biosignals via an IoT system. The developed prosthetic used servo motors to drive fingers movement, an Arduino Mega to control the servo motors, and a Raspberry Pi 3 to exchange data between the Arduino and an online database. Each previous prototype of the prosthetic arm was manufactured using Fused Deposition Modeling (FDM). FDM is the most common method of 3D printing and is extremely useful for rapid prototyping [7]. The process can be cost-effective for making prototypes on a limited budget. The previous prototypes of the prosthetic design [6] were useful in understanding the process of designing and prototyping a prosthetic limb. However, the previous prototypes were less than functional as a prototype regarding their size, power consumption, and basic functionality of grasping objects precisely. At each prototype generation, the goals of the research were narrowed and refined.

This current research will improve on the existing prosthetic arm by transitioning the electromechanical system to linear actuators. The linear actuator will allow for a wider range of fingers movement with improved accuracy, as well as provide the necessary force for the hand to be able to grab objects. The muscle, pulse, and temperature sensors are included in this prosthetic arm to enable the patient or health professionals to monitor the users' health. In addition, a Raspberry Pi Zero W will replace the larger Raspberry Pi 3 Model B, and an Arduino Nano will replace the larger Arduino Mega microcontroller. The benefits of upgrading the hardware are cost-effectiveness, enhanced accuracy, and system stability.

The remainder of this paper is organized as follows: the initial hardware architecture design and research is introduced in Section 2. Improved IoT prosthetic hardware design and changes is presented in Section 3. Experiment with linear actuator and analysis is discussed in Section 4. Finally, we conclude the work in Section 5.

2. The Initial Hardware Architecture Design and Research

Microprocessors and several microcontrollers, such as Arduino Uno, Arduino Mega, the Intel Edison, the Raspberry Pi 3 Model B, or the ESP32 [7], were initially used as the central unit for the prosthetic arm. Fig. 1 shows our initial hardware design infrastructure of the Myoelectric prosthetic arm. The hardware connections are located inside the Myo-prosthetic arm. All the sensors are connected to the microprocessor situated near the center of the prosthetic arm. The

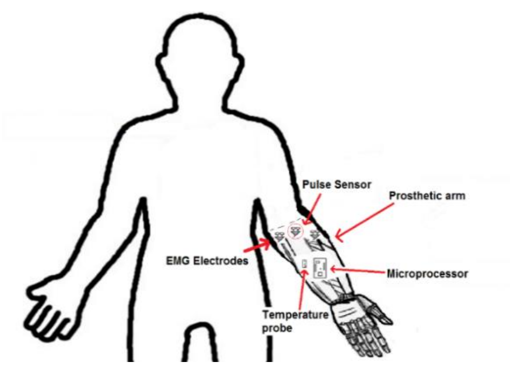

Fig 1. Distribution of bio-sensors and network

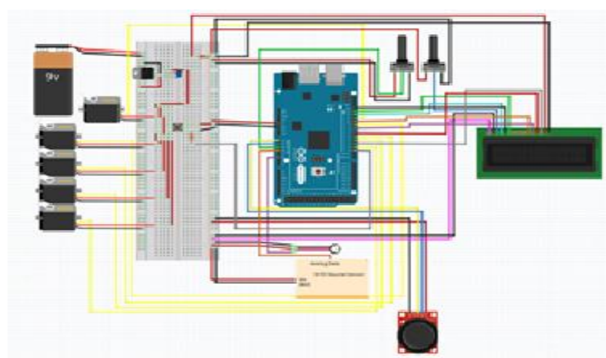

Fig. 2. The servo motors controlled by the Arduino Mega 2560

electrodes are located at the socket of the prosthetic limb. The analog signals are gathered using electrodes and sensors in the prosthetic. The acquired signals are collected and then sent to the Raspberry Pi 3 Model B located in the middle of the prosthetic arm. The myoelectric signal will then be used to control the actuators and servo motors. The sensed signal such as temperature and pulse will be uploaded to a Firebase database via Wi-Fi and then distributed to the chosen endpoints. The system allows the user to be notified and enables the data to be accessed by relevant persons to monitor the patient's health. Our initial design used servo motors attached to strings along a 3D printed hand for the electromechanical portion. The schematic in Fig 2 shows that the five servos are connected to the Arduino Mega 2560. The analog input to the system can be any sensor like a pulse sensor, joystick, or MyoWare muscle sensor. Fig.3 shows three implemented models of the myo-prosthetic device with servo motor and sensors.

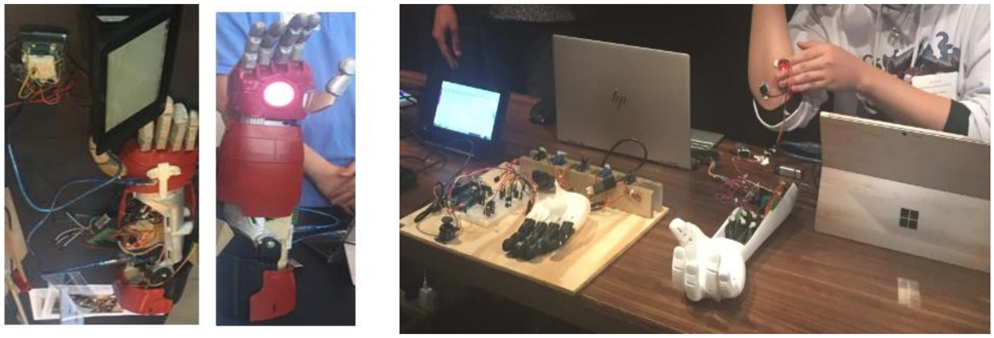

Fig. 3. Our three myo-prosthetic models with servo motors and sensors

We noticed some drawbacks in the system that need to be tackled. First, a prosthetic must be small and light enough to be wielded by the user. A prosthetic that is large and heavy does not serve its purpose very well. Previous prototypes, however, were bulky because of the design and the use of plastic in the manufacturing process. Besides, because of the desire to add specific advanced capabilities, such as patient monitoring and remote data transmission, previous prototypes erred on the side of being larger to accommodate a microcontroller and a single board computer, as well as the electromechanical components for fingers control. Second, a prosthetic must be able to emulate the capacity of a muscle to extend or contract in a linear way. Previous prototypes, although able to grasp small objects, cannot make a linear motion.

3. Improved IoT Prosthetic Hardware Design and Changes

The design modification focuses on the more efficient use of space and a wider range of fingers movement with improved accuracy. It involves replacing the current mechanical drive system and making better use of the limited space that is available inside of the prosthetic by substituting the previous electronics components with smaller components that achieve the same purpose. The new design uses a linear actuator, Arduino Nano, and Raspberry Pi Zero W. A linear actuator is an electromechanical device that translates the circular motion of a motor into linear displacement of some slider mechanism. For this prototype, we chose a dual-phase bipolar micro stepper motor-based screw and slider linear actuator. The benefit of the use of a linear actuator is that the approach allows for more accuracy of movement, resulting in more precise motor control for the fingers. Another benefit is that a linear actuator uses a screw and slider mechanism to hold its position even when power has been removed from the circuit, which results in a more energy efficient design. There is a significant benefit when dealing with battery-driven systems. Previous prototypes used Arduino Mega for the control of the mechanical drive system and for receiving the sensors data. The prototype system used a Raspberry Pi 3 Model B for the transmission of the sensors data to a remote database for remote patient monitoring. The current prototype will replace Arduino Mega with Arduino Nano, which is approximately 85% smaller than the size of the Arduino Mega [9]. Similarly, the Raspberry Pi 3 Model B will be replaced by the Raspberry Pi Zero W, which is about 50% smaller than the size of Pi 3 Model B [10]. These changes will help decrease the overall size required of the prototype.

Fig. 4 shows our new hardware design to address the issues found in the previous design. The improved hardware systems involve two new subsystems. The first subsystem encompasses the robotic hand, which uses linear actuators controlled by an Arduino Nano and a muscle sensor. Each linear actuator is stepper motor-based, and it controls one of the fingers. Each actuator

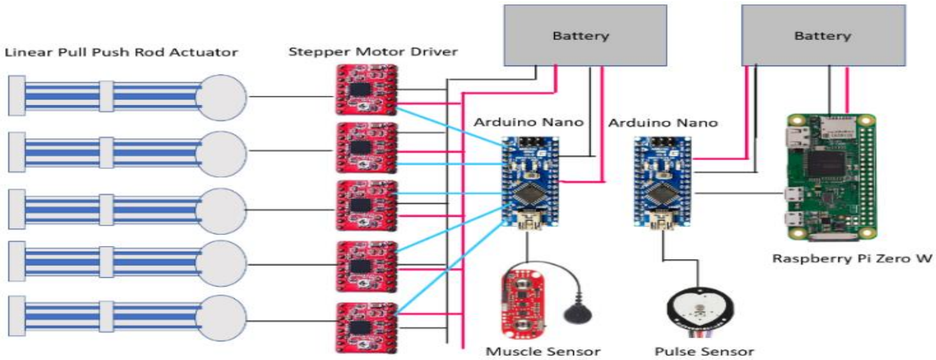

Fig. 4. Improved IoT prosthetic control system with the Arduino Nano, the linear actuator, and the sensors

requires a dedicated A4988 stepper motor driver. A single Arduino Nano can control all the five linear actuators because only two Arduino digital pins are needed to control a stepper motor driver. The step pin from the stepper motor driver controls how many steps the actuator will run, and the direction pin controls the direction the motor rotates. Greater control over the robotic hand can be achieved by calculating the number of steps required to move the linear actuator a certain distance. The displacement can be determined using the step angle of the motor and the screw pitch of the screw-slider mechanism:

$$N = \frac{R * X}{P}$$

Where N = Number of steps, R = Steps per revolution of the motor, X = Displacement, and P = Screw Pitch. Since the step angle of our linear actuator stepper motor is 18 degrees:

$$R = 360 \frac{\text{degrees}}{\text{Step angle}} = \frac{360 \text{ degrees}}{18 \text{ degrees}} = 20 \text{ Steps per revolution}$$

And since the screw pitch is 0.5mm:$N = \frac{20*D}{0.5} = 40 * X$. Therefore, based on the manufacturer's data, the theoretical formula for calculating the number of steps (N) to move the linear actuator a specific distance (D) is $N = 40 * D$. The second subsystem includes a network of sensors, including pulse and temperature sensors, whose analog data are collected by a second Arduino Nano and then transmitted to a remote database via Raspberry Pi Zero W. The Pi Zero W is programmed to act as a gateway for communicating with outside networks. Besides, two independent power supplies are needed to make sure there is no voltage or current conflict over the electrical components since the linear actuator require minimum 8V to operate, and other parts require logic 5V DC supply

4. Experiment with Linear Actuator and Analysis

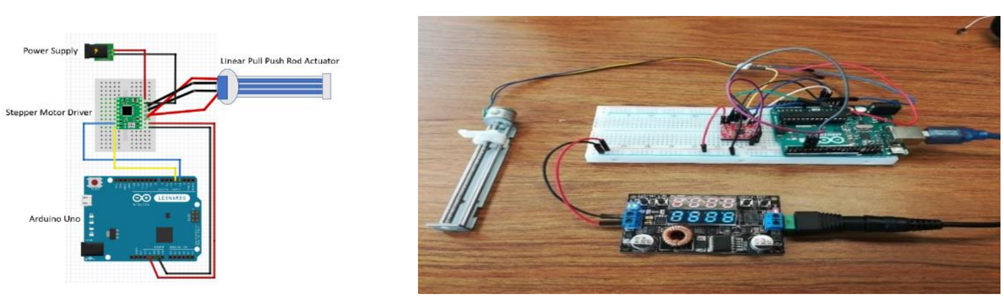

Fig.5. Linear actuator calibration and setup

```
for(int x = 0; x <670; x++) // Sends 670 pulses
{
  digitalWrite(stepPin,HIGH); //Each pulse moves the stepper motor one step backward
  delayMicroseconds(500);
  digitalWrite(stepPin,LOW); //Each pulse moves the stepper motor one step backward
  delayMicroseconds(500);
}
```

Fig. 6. The linear actuator is calibrated by empirical data: 670 pulses corresponds to 1cm of the slider movement

The linear actuator is based on a bipolar stepper motor and a slider-screw mechanism. The two phases (Motor A+, Motor A-, Motor B+, and Motor B-) of the linear actuator's bipolar stepper motor are connected to the motor driver outputs. The required 9V power supply to drive the motor is attached to the stepper driver. The stepper driver signal inputs for Step and Direction are respectively connected to digital pins 3 and 4 of the Arduino Uno. Each pulse sent to the Step pin of the driver moves the stepper motor one full step while setting the Direction pin either HIGH or LOW allows the motor to turn clockwise or counterclockwise, respectively. Finally, the VDD and GND pins of the stepper driver are connected to the 5V and GND pins of the Arduino Uno to power the stepper driver board itself.

Table 1 Linear actuator calibration

Linear Actuator Calibration Data		
No. of Pulses, n	Displacement, x (mm)	Slope $= \frac{\Delta n}{\Delta x}$
200	3	66.7
400	6	
600	9	
800	12	

The data provided by the manufacturer of the linear actuator proved to be unreliable in terms of calculating distance moved based on the number of steps. An initial theoretical calculation (N=40*X) of the number of steps needed to move the slider 1mm was not confirmed when the motor was tested. Therefore, the linear actuator had to be calibrated by empirical data.
The stepper motor calibration has been done using the following method ((Fig.5 and Fig.6): using an Arduino programming, a range of pulses from 200 to 800 are used as calibration steps. For each pulse, a measurement of the distance, in millimeters, moved by the slider is taken. Over the range of pulses, the slope of the measurements was found to be 66.7 pulse/mm (Table 1). As a result, 67 pulses corresponds to 1mm of the slider movement. Therefore, any required displacement in mm can be obtained by multiplying the desired distance by 67 pulses, and any displacement in cm can be obtained by multiplying the desired distance by 667 pulses. Fig. 7 shows control of finger movement with a linear actuator and the Arduino Nano. The finger can precisely move 1cm.

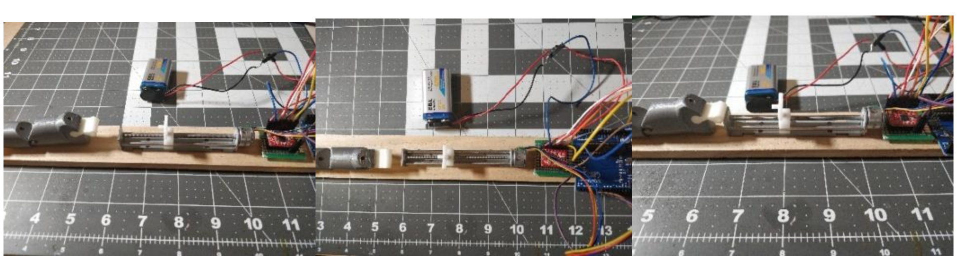

Fig. 7 Control of finger movement with a linear actuator and the Arduino Nano

5. Conclusions

This research into the creation of an IoT-based prosthetic device continues to evolve as new ideas and methods are tested and implemented. At this stage of the development, we present an approach to revising and improving the electromechanical design and functionality of our existing prototype. In addition, revising the current control system with smaller control system components will benefit the prototype by saving space while decreasing size, weight and cost. Future research will be conducted into the physical design changes that can improve the usefulness of this prosthetic device for persons requiring its use. This includes research into the ways that we can improve the 3D printed design as well as the printing process. These considerations may be able to further reduce the weight of the overall device, making it more user-friendly.

References

[1] Ziegler-Graham K, MacKenzie EJ, Ephraim PL, Travison TG, Brookmeyer R. Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050. Archives of Physical Medicine and Rehabilitation 2008; 89(3):422-9

[2] N. Mathur, G. Paul, J. Irvine, M. Abuhelala, A. Buis and I. Glesk, "A Practical Design and Implementation of a Low-Cost Platform for Remote Monitoring of Lower Limb Health of Amputees in the Developing World," in IEEE Access, vol. 4, pp. 7440-7451, 2016

[3] Neha Mathur; Ivan Glesk; Arjan Buis, "Issues in wearable mobile sensor platform for lower limb prosthetic users," 2015 17th International Conference on Transparent Optical Networks (ICTON), 2015, pp. 1 – 4

[4] S. M. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K. S. Kwak, "The Internet of Things for Health Care: A Comprehensive Survey," in IEEE Access, vol. 3, pp. 678-708, 2015

[5] R. Kumar; M. Pallikonda Rajasekaran, "An IoT based patient monitoring system using Raspberry Pi," 2016 International Conference on Computing Technologies and Intelligent Data Engineering, pp. 1 - 4

[6] Gabriel Martinez, Xiaolin Chen, Teddie Lai, Mohammed Alborati, Yu Wang, "IoT in Myo-Prosthetics," 2018 ASEE Northeast Section Annual Conference, Hartford, CT, April 27, 2018 - April 28, 2018

[7] Silva, C. and Garcia, W. (2018). SUPERHERO ENABLE 3D PRINTED PROSTHESIS [online] Thingiverse.com

[8] MySignals microprocessor. Retrieved from Libelium Comunicaciones Distribuidas S.L. [Online] [Cited: 02 23, 2017.] http://www.my-signals.com

[9] https://store.arduino.cc/usa/

[10] https://www.raspberrypi.org/products/

Acknowledgments

This research is supported by National Science Foundation ATE (#1601522). Partially supported by PSC-CUNY Grant (60310-0048), National Science Foundation Noyce grant (#1340007), and New York State Collegiate Science and Technology Entry Program (CSTEP).

Yu Wang

Dr. Yu Wang earned her Ph.D. degree in Electrical Engineering from the CUNY Graduate Center. Currently she is an Assistant Professor in the Department of Computer Engineering Technology at New York City College of Technology. Her primary area of interest includes engineering education, formal methods for modeling real-time systems, digital design, Agile testing, embedded systems, and network protocols.