Multidisciplinary Senior Design Conference
Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York 14623

Project Number: P16062

VARM

Christopher McCabe
Electrical Engineering

Cassandra Derr
Electrical Engineering

Joshua Wilkins
Electrical Engineering

Timothy Koski
Mechanical Engineering

Hector Baez
Biomedical Engineering

Hector Terrero
Mechanical Engineering

Nicholas Barlow
Computer Engineering

ABSTRACT

It is proposed that the use of air muscles will allow for the application of forces on the human arm based on a virtual reality game input. The air muscles will be mounted on adjustable straps which allows for modular configurations based on a user's arm size. Multiple sensors will be used to control the pneumatic and arm tracking systems. A software solution is implemented to translate applied forces in the virtual world to real-world forces that will be exerted on a user's arm. A custom virtual reality scene will be created in Unity to ensure the subsystems are communicating properly and will be used for demo purposes. A proof of concept mechanical prototype has been designed and completed around a model arm with a satisfactory tracking system in place. The tracking system has been verified by the Unity demo scene to be within three degrees of tolerance.

INTRODUCTION

The gaming industry is a unique and intriguing market for corporations and investors. Amazon purchased a live-streaming platform used primarily for video games known as Twitch for over $1 billion. Google purchased YouTube for $1.65 billion which also hosts live streams and also allows gamers to upload personalized content. Reputable athletes such as Shaquille O'Neal and Rick Fox have invested into professional video game teams and help manage them. Companies such as Samsung and sports teams like the 76'ers have also invested into professional video game teams. With so much interest in the gaming industry, the race is on to discover the next big breakthrough that will revolutionize the gaming industry. Many believe that breakthrough will be virtual reality because the possibilities are endless.

The virtual reality market is expected to reach $16 million by 2017 but there is a limited amount of devices on the market today. Most devices immerse the user through visual stimulation using a virtual reality (VR) headset with little physical stimulation. Current virtual reality solutions such as the Xbox Kinect visually stimulate the user by accurately tracking their movements, but fail to provide any physical feedback from any objects the user may interact with during the game. User mounted devices such as PrioVR are able to provide the user with the sensation of being struck by virtual objects, but do not allow the user to provide any input back into the virtual system. Neither of these current solutions provides a rich and immersive VR experience; they just leave the user wanting more.

Haptic feedback is using forces and/or vibrations to recreate the sense of touch; the user is under the perception that something is manipulating their sense of touch. There are devices that exist today with a haptic engine such as the iPhone 7. The iPhone 7 doesn't have a mechanical home button that physically moves when pressed, but rather a home button that utilizes a haptic engine to simulate a button press. VArm utilizes a similar concept, applying forces to the user based on things that are happening in-game. The goal of VArm is to provide a more immersive VR experience through the use of a haptic controller worn on the user's arm which allows the user to perform an action and simultaneously feel the result of their action.

PROCESS

This project was proposed by one of the team members, thus the scope of the project as well as the customer and engineering requirements were determined by the team. Due to financial and time constraints, the scope of VArm was limited to a proof-of-concept which could be scaled up in future iterations.

The customer requirements were focused on both creating a near-realistic experience that would integrate properly with current market virtual reality headsets and on the safety of the user. Three levels of importance were assigned to these desired features to prioritize design efforts. Among the most important customer requirements were those relating to safety, including not causing bodily harm, having an emergency stop, and limiting the range of motion to that of a normal human arm. The other requirements assigned to the highest importance level related to the core functionality of VArm, including being able to apply realistic forces, being able to track the arm, and having a reasonable response time. Other less important customer requirements were those that looked forward to creating a consumer product, such as being adjustable for different arm sizes and cost of construction. These requirements were then used by the team to create quantifiable engineering requirements.

The engineering requires of this project took into consideration the list of deliverables from the customer in addition to the safety parameters and necessary specifications to accomplish those deliverables. These requirements were divided into five major functions or subsystems being mechanical, electrical, biomedical, software, and safety. Starting with mechanical, it was first determined that the device must weigh less than the ideal 5 pounds when considering the ease of use to the user and a reduced complexity in force application. It was also determined that a life test of 100-150 minutes was necessary for a standard gaming session. Next, due to concerns on safety and device life time, a requirement of no more than 10-15 pounds of force on the user's arm was applied. Within this specified force requirement, it was deemed necessary to distribute this load so as to reduce applied stress on device components. Within the electrical function, a safe power source with a V/Hz of 110/60 and a surge resistance max between 330-500 V was required so as to be safe and compatible with standard outlets. Within the biomedical function, values were determined for optimal ranges of motion for flexion and extension of the elbow and wrist. These values were limited from 90 to 200 degrees for the lower arm and 135 to 180 degrees for the wrist as they should not be fully extended nor flexed for due to safety concerns. Another requirement introduced stated that this device could not disturb physiological processes within the users arm such as circulation. Within the software functions, communication and game engine software values were taken into consideration. The tracking precision tolerance was deemed to have no more than a 3 to 5 degree of error with a maximum movement delay of 40 to 100 milliseconds. Lastly, overall system safety parameters were introduced to emphasize device safety. Although the device applies a force on the arm, the user must have 100% control of arm movement at any time, which means a limit on force applied. The device must also come equipped with an emergency stop button placed within reach of the user's free arm.

Several mechanical design concepts were considered. The three main concepts considered include a linear pneumatic design, a rotary pneumatic design, and an air muscle system. The two pneumatic systems were very similar. Sketches of these two systems are shown below in Figures 1(a) and 1(b), respectively.

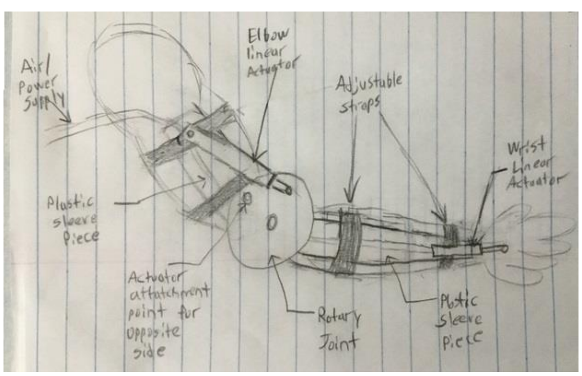

Figure 1(a): Linear Pneumatic Design Concept

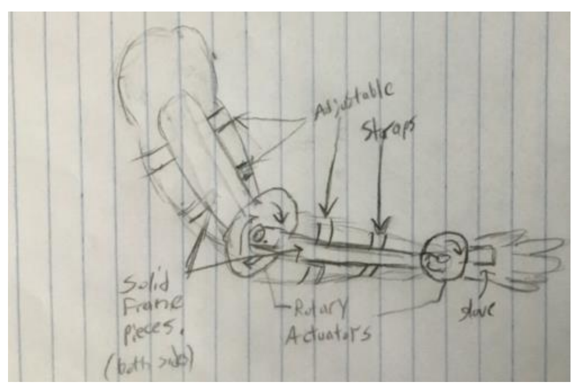

Figure 1(b): Rotary Design Concept

The linear pneumatic design would have elbow motion driven primarily by a linear actuator connected to the outer edge of the rotary joint. As the actuator extends or contracts, the force would be translated from a linear to rotational direction. At the wrist joint, a smaller linear actuator would be utilized. A rotary joint would not be used here as the end would simply push and pull the top of the hand as it extends and contracts. All parts would be attached to the arm using an adjustable strap system. The advantages of this design are that it can apply a large range of forces to the user's arm, and would very precisely be able to automate the arm to a specific angle. However, this design has several flaws, including that it would be heavy, costly, and would greatly resist the user's motions. The rotary system shown in Figure 1(b) is similar, however rather than linear actuators being translated to rotational motion, the rotational force is created directly. However, this design also comes along with the same constraints. It would resist the user's motion, not meeting the requirement for complete user controlled movement. The motors or actuators are also very expensive and bulky. This makes it difficult to meet the engineering requirements for weight, distributed load, and ergonomics.

The third main design concept, shown in Figure 2, incorporates air muscles into the design.

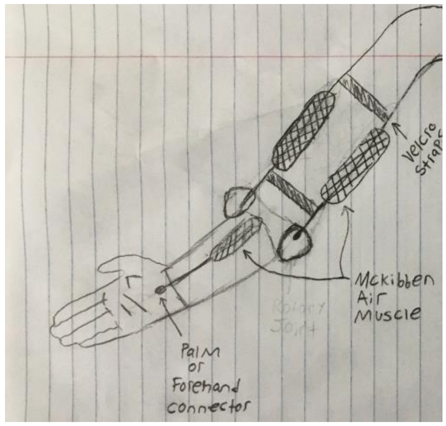

Figure 2: Air Muscle Design Concept

Air muscles consist of two primary components, a soft stretchable inner rubber tube and a braided mesh sleeve. The inner rubber tube is pressurized and expands, which in turn pushes against the nylon sleeve. The diameter of the nylon sleeve is forced to expand, which shortens the length and gives a contracting force. This provides a very high force to weight ratio. Incorporating air muscles into the design provided several advantages in terms of engineering requirements, mainly weight and ergonomics. Utilizing different sized air muscles and applying different pressures to the system allows for a range of forces, however, overall they produce much less force than the other designs. This design also has the advantage of being very low cost. The main constraint of this design is the lack of precise control over air muscle contraction. It is very difficult to characterize the exact force output at different pressures, as well as the contraction length and response time. The system would be supplied from an external air compressor with tubes connecting to the on arm system. The air supply to individual air muscles would be controlled by an off-arm manifold and valve system. Again the air muscles would be attached to the arm using an adjustable strap system. The air muscles would be attached to the straps by a custom 3D printed part. This was chosen for the final design, as it met the most important engineering requirements: weight, complete user control, and cost.

To determine what sized air muscles were required for the design, several different sized air muscles were created and tested to determine relationships between diameter size, length, contraction time, and force output. Although not all of these could be accurately measured, a general relationship between key elements was determined. From this, two 12" long, 0.375" ID air muscles were selected for the back portion of the arm, and two 6" long, 0.24" ID air muscles were chosen for the front of the arm and the wrist. These would be operated at 50 PSI as this was found to provide adequate performance with minimal leakage. Custom 3D printed attachment points, shown in Figure 3, were created and sewn onto nylon straps. The straps use Velcro to allow for various arm sizes. The connectors allow push-to-connect adapters to be screwed in. Push-to-connect adapters were chosen so that tubes and air muscles can be easily connected and disconnected from the strap system.


Figure 3: Air Muscle Attachment System

To control these air muscles, a valve system needed to be designed. The final flow It was decided that two air systems should be used, one for each joint. The final design is shown in Figure 4. The air from the compressor is kept at 60 psi with the use of a regulator, and the feed is then split to two proportional valves, allowing us to manipulate the system pressure inside each manifold. One manifold feeds the elbow joint air muscles, and the other feeds the wrist joint air muscles, all of which can be individually activated. The original design also included a third proportional valve at the exhaust, which gives the ability to control the speed of air muscles relaxation. Due to budget constraints this was not incorporated in the final design.

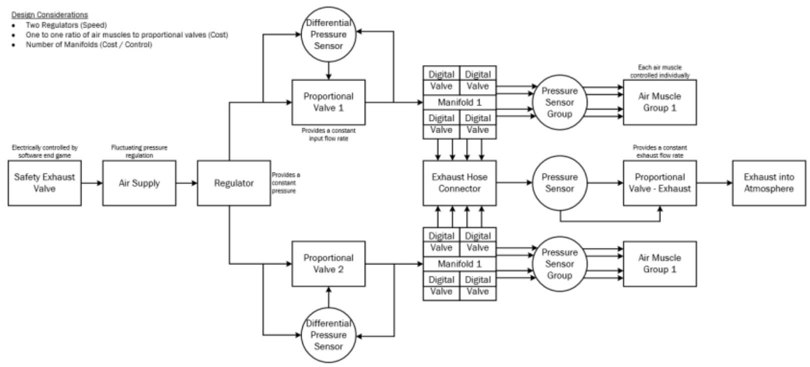

Figure 4: Valve system design

One of the main electrical design decisions was to use either a breadboard or a printed circuit board (PCB) to house the electronics. This choice in turn determined electrical component selection: through-hole or surface mount. A PCB was chosen due to several reasons, including that surface mount components are smaller than through-hole components, PCB's tend to be more organized than a breadboard system, and the higher signal integrity and speed they allow. This is due to the ability to have a ground plane to minimize resistance between different points in the circuit. Separate analog and digital ground planes were used to prevent switching noise in digital circuits from affecting the more sensitive analog circuitry.

The purpose of the PCB electronics was two-fold: to control the system to produce the forces commanded by software, and to monitor the system to be sure these forces were being applied correctly and detect any hardware faults. The former system consisted of simple transistor circuits to turn the two-way valves on and off as well as control circuitry to supply the current to the proportional valves necessary to produce the correct system pressure. The latter system included analog pressure sensors to measure the pressure in individual air muscles as well as the system pressures for each joint. This circuit also included an analog to digital converter (ADC) to digitize these sensor readings for the microcontroller.

Microcontroller selection was difficult because of the many different constraints. The microcontroller needed to be small enough to fit on the PCB, needed enough pins to control everything, needed to be fast so there would be little latency, and it must be cheap because of the limited available budget. The microcontroller that was selected was the Teensy LC. The Teensy is responsible for the control of the overall board. It will respond to commands from the game engine to apply specific forces to the arm, as well as monitor the status of the board and ensure that commands are being executed as expected and to protect the safety of the user.

The other major task for the electrical sub-team was to accurately track the arm to ensure that the in-game model of the person is up to date and that the forces are not applied that will injure the user. The initial concept was to use a PixyCam. The PixyCam is a vision sensor that uses color for object detection and tracking. It can process up to 50 frames per second at 640x400 or 30 frames per second at 1280x800. It idles at low power and it consumes little power during operation. It places a bounding box around any object that it detects and it outputs the x and y coordinates as well as the height and width to the microcontroller.

The next concept to explore was the use of Inertial Measurement Units (IMU), a collection of digital sensors for the measuring of acceleration, rotational speed, and magnetic field strength along 3 spatial axes. The IMU's accelerometer is primarily used to derive orientation with respect to gravity, while the magnetometer's readings are used to derive orientation with respect to magnetic north. Combined with a gyroscope, which measures the rotational velocity of the IMU, a reliable orientation vector can be calculated. The Madgwick Attitude and Heading Reference System provided the raw mathematical formulae for these calculations. The MPU-9250 IMU was initially chosen, due to its low cost and relatively wide usage.

A testing environment was designed to provide an immersive environment to showcase the integration of these technologies. Using the Unity game engine, a mini-game was designed that set up an in-game user avatar, configured so that their arm's orientation matches that of the user, using the tandem of IMU sensor modules. The user is tasked to aim a virtual blaster connected to their arm to shoot targets within a time limit. The resting weight of the blaster, and recoil from shooting is sent to the on-arm force actualization system.

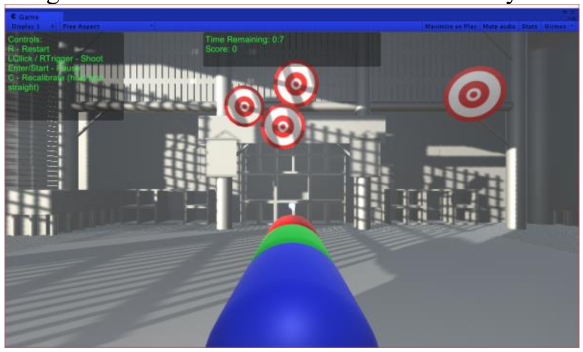

Figure 5: Unity Demo Program Image

Custom software components were designed to allow for non-blocking serial communication with both external systems from within the multi-threaded Unity engine.

RESULTS AND DISCUSSION

The completed mechanical design consisted of three sets of two air muscles, one for elbow contraction, one for elbow extension, and one for wrist extension. It was determined that these axes would be sufficient for a proof-of-concept effort, and limited the scope to be achievable given the constraints. Two air muscles were used for each

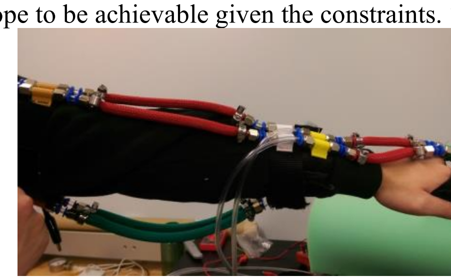

Figure 6: Completed Mechanical Design

actuation to increase the amount of force applied. These were mounted on Velcro straps to give flexibility of both placement and arm size, as requested by the customer. The completed mechanical system is shown in Figure 6.

The user-mounted system needed to be comfortable for the user to wear for a gaming session. Several of the engineering requirements address this, including the weight, contact surface temperature, distributed load, and ergonomics. The weight of the user-mounted system had to be ideally less than 5 lbs, but worst case less than 10 lbs. All components that would be mounted to the user were placed on a scale, and it was determined that the entire prototype weighed only 1.72 lbs, well within the specification. The only user-mounted components capable of generating heat were the electrical components of the tracking system. These did not get above room temperature (72°F), and were comfortable to wear during testing of the tracking system. Thus, this specification was also met. The distributed load test was removed due to the fact that the prototype is unable to violate this specification by design. No forces are generated in a constricting fashion, only a linear one. Thus, an inward pressure on the user's arm is impossible. The prototype was worn in an inactive state during tests, and it was found to be comfortable by the tester. Thus, the ergonomics requirement was also met. It was also determined that the system was unable to disturb any physiological processes, such as restricting circulation or interfering with pacemakers. Thus, this specification was met as well.

The remaining mechanical specifications were related to the performance of the system. The primary requirements were the range of force applied, the range of motion on the actuated joints, and the life test to ensure it will continue to function after continuous use. The life test was not performed due to time constraints. The force test could not be completed due to the inability to accurately measure force output of the arm. It is estimated that the prototype would not be able to apply the desired force. However, it was determined that the sudden impulse applied to the user is more important to the user experience than the steady state force applied. The life test was not performed due to time and resource constraints. The air compressor available to the team for use was not capable of maintaining system pressure for long enough to complete a life test, and the other air supplies available for use operate at a much higher pressure than was utilized for the prototype. The range of motion test was performed using a goniometer, applying maximum force in each direction of actuation and measuring the maximum angle achieved.

The range of motion achieved was much smaller than originally specified. For the elbow, the full range of motion between extension and flexion was only 29°, much smaller than the specified 85°. The wrist range of motion was also much smaller than specified, only 12° compared to the desired 85°. This is partially due to the fact that the wrist was only actuated in one direction for the prototype. The wrist radial and ulnar deviation was not included in this prototype design, as previously discussed. This is partially due to the fact that the air muscles are able to produce less force than desired. It is also partially due to the fact that air muscles are only able to contract by about 15% of their total length.

The overall valve system design performed well, with some minor issues. The small manifold orifices slightly slowed down overall flow, reducing system response. Another issue came from a faulty proportional valve, which slowly leaked air into the manifold.

Once the PCB was assembled, the major subsystems were tested. Most of the subsystems were found to perform as expected. The proportional valve sub-circuit was found to have a design flaw; thus modifications were made to ensure proper functionality. The complete PCB is shown below in Figure 7.

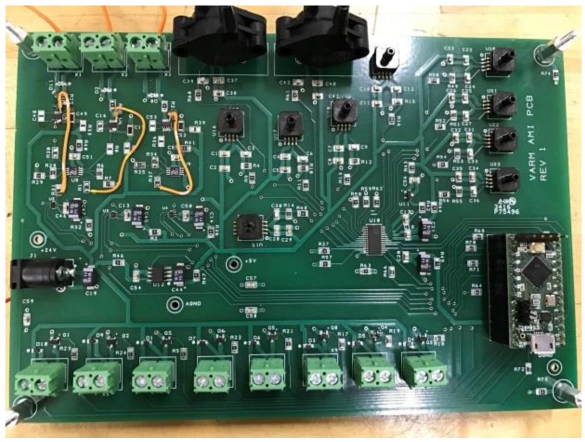

Figure 7: Completed PCB

The pressure sensors were tested using a known pressure source. Both the differential and absolute pressure sensors were found to have a very linear response that closely matched the datasheet. The resulting plots are shown below in Figure 8.

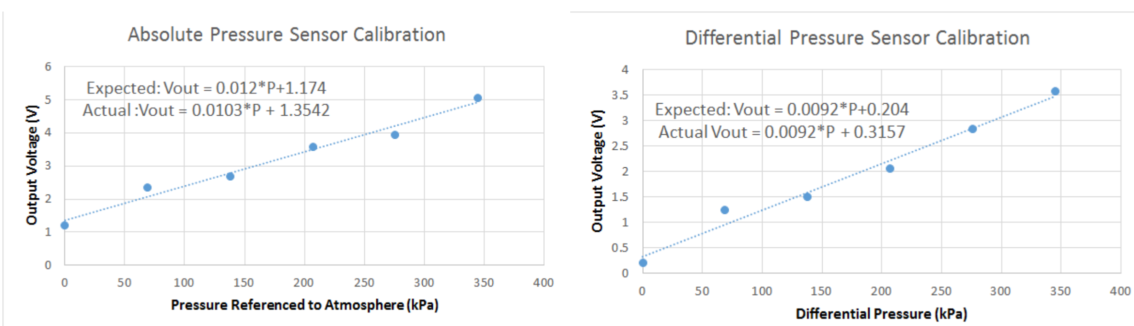

Figure 8: Absolute and Differential Pressure Calibration Plots

An emergency stop button was included in the design to give the user a way to stop the system if they ever felt unsafe. This was connected directly to the air muscle control PCB to allow for the shortest delay between the user activating the button and the system shutting down. Once an emergency stop was asserted, no commands would be accepted until the condition was cleared by the host PC. This is to ensure that the system does not begin normal operation until the user is ready to begin again, and that if the host PC sends any commands before detecting the emergency stop, the system does not continue to operate. This was tested during operation, and it was found that all air muscles immediately exhausted, and did not begin actuating again until the emergency stop condition was cleared. Thus, the engineering and customer requirement of an emergency stop was met.

It was also required by the customer that the electrical system be powered by a safe power source and be electrical surge resistant. Both of these requirements were met by design, as the PCB was powered by a standard wall outlet, which regulated the AC wall supply to a 24V DC supply. The wall wart meets the electrical surge resistance standard, and thus also the customer's requirement.

The final electrical requirement was that the system be able to track the user's arm to within 3-5° of accuracy. Originally, the tracking system was going to have two parts, on-arm mounted IMUs and an off-arm mounted PixyCam. A sample image taken from the PixyCam is shown below in Figure 9.

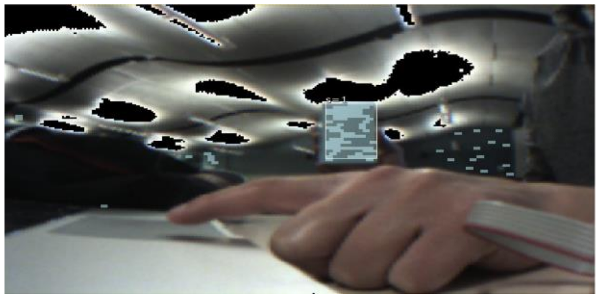

Figure 9: PixyCam Output Image

Despite using a different lens to improve resolution, the PixyCam still did not produce great results. It has difficulty under high white light conditions and was unable to accurately track objects at a distance comparable to that of a gamer playing their game. Thus, only IMUs were used to track the arm.

Difficulties were encountered when integrating the MPU-9250 IMU. Its electronic sensors were particularly prone to noise; stray magnetic noise disrupted the sensor's readings, manifesting as an imprecise "drifting" orientation. A change in operating temperature or environment would also require a complete recalibration to ensure accurate performance. Thus, an alternate was found in the BNO-055 absolute orientation IMU. The BNO-055 did not encounter any calibration-related issues, however, the effects of gimbal locking were apparent, as the IMU was not able to detect any planar rotation among its axes once one of its axes were rotated 90˚. Further investigation is necessary to try to minimize or counteract this occurrence. Gimbal locking primarily occurred in this application due to the IMU's roll (forward axis rotation) axis becoming aligned with its yaw (horizontal planar rotation) axis. The magnetic compass component of the IMU measures the relative strength of the magnetic field around it. Since this is primarily earth's natural magnetic field, the magnetometer typically measures yaw in its initial orientation. Once pitched 90°, earth's magnetic field is now inline with the IMU's roll axis, causing inaccuracy of any yaw axis rotation once in this new position. This was primarily caused by a conversion from the initial 4 axis value provided by the BNO0855 into a 3axis yaw-pitch-roll representation used to convert between the IMU's right hand rule X,Y,Z axis representation to Unity's left hand rule configuration. This was resolved via swapping the IMU's X and Z axes, and inverting the Y axis to match the expected orientations in Unity. The tracking precision was measured using the goniometer. The IMUs and the goniometer were mounted to the test arm, and the arm was placed at different angles. The resulting measurements from both systems were compared, and it was found that the tracking system was able to match the goniometer measurements to a worst case of 3°, within the specification.

The Unity demo worked as expected, serial interfaces perform relatively quickly, without introducing excessive latency. The communication had to be rate-limited, however, as a 115200 baud channel is not broad enough to handle the relatively large ASCII text communications more frequent than every 3 frames (50ms). Unity also introduces additional latency due to the limitation in its internal update rate, performing serial communication requires one frame to send a serial command, and an additional frame to process a received command, adding an additional 33ms of latency onto each communication. Another problem that introduced latency to the system was that the hardware was unable to reach the commanded force profile before the next command was sent. In order to maintain the speed of commands from the host machine to provide the user a more reactive experience, it was accepted that the commanded force would not be fully achieved, and that the system feedback to ensure commands were executed properly was limited to ensuring the safety of the user, rather than the performance of the system.

The customer also required that there be a "software backdoor" into the arm control system, such that it could be debugged and used independent of the Unity game engine. This was achieved by using a serial interface to mimic the Unity game engine's serial commands. In addition to the operational commands applying force to the arm, applying a software emergency stop, or clearing an emergency stop condition, a few debug commands were implemented as well. These included a command to read the state of all pressure sensors and a command to check the status of the system.

Another requirement of the system was that the prototype's movement be 100% user controlled. From a software standpoint, force commands are only sent if the user has done something in game to trigger those motions. From a physical standpoint, the prototype is flexible enough that the user is able to move freely while playing it, though their motions may be resisted due to actions happening in-game.

CONCLUSIONS AND RECOMMENDATIONS

The concept of performing an action and simultaneously feeling the results of that action has been partially realized. The Unity demo scene can accurately track the arm and when the user fires a blast in the game, the mechanical prototype actuates the model arm. The prototype was able to meet all of the safety specifications, which were among the most important. It meets all of the specifications for the wearable portion of the device, such as surface temperature and weight. It is also completely controllable by the user; all forces are prompted by actions performed in-game, their natural motions are not inhibited unless by an in-game action, and the user is able to completely stop the system at any time with the emergency stop. However, the prototype fell short in the desired performance, being unable to apply the desired force or range of motion, and taking much longer than desired to completely execute these force commands.

The shortcomings of the system can be attributed largely to the constraints of the project, including a small starting budget of only $500, and a time constraint of only two semesters during which no team members were able to work full-time on the project. These two constraints acted in tandem to negatively affect the team; it took time to request and secure an increased budget, which in turn reduced the time available to purchase major components, then build and test the system. The major components affected by this delay were the custom PCB and components, and the proportional valves used to regulate system pressure. The valves needed to be custom ordered to fit the specifications of the project and took a month and a half to arrive. If a larger budget had been secured sooner, the project could have gone smoother with more time for testing and validation at the end.

In the end, the system created is a successful proof of concept. Though it did not meet the specifications desired, it could be improved with a larger budget and a longer project timeline as well as designers with more time to contribute. Larger valves run at a higher pressure would allow more air through faster, and would significantly improve system performance. This would also allow for larger air muscles, generating greater force. In addition, a larger software team could have produced a better demonstration of the capabilities of the project, and wider distribution of software responsibilities could have allowed for the development of a full API and software spec of the system, for easier distribution of the project as an iterable open source project.

ACKNOWLEDGMENTS

Louis Beato,
Dr. Lamkin-Kennard
RIT MSD