Luke's Hand

Jose N. Suarez, Cesar Oleas, Esteban Guerra, Alexander Bermudez

School of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

Abstract **— This paper documents the design of Luke's Hand project, which is able to simulate basic movements of a human hand. Luke's Hand is a robotics application implemented with servomotors, a sonar sensor, and a multiprocessor microcontroller to achieve faster responses of the real time system. The robotic hand is mainly controlled by three different modes: a sonar distance detector, a 4-bit switch, and an analog-digital conversion module using a glove interface.**

Index Terms **— Analog-digital conversion, detector, microcontroller, multiprocessor, servomotor, sonar distance measurement, switch.**

I. Introduction

Luke's Hand is a robotic implementation to emulate basic movements of a human right hand. The inspiration of the project comes from looking at the statistics of people who lost an upper limb. Those increasing numbers are attached to traumatic changes in life that touches not only the person who suffers such injuries, but also their families. Thus, the project is considered as a small contribution to the scientific community, which has been working to give amputees a second chance to improve their quality of life.

In General, the field of robotics has been rapidly growing during the last decade, opening a new window to bring robotics into everyday applications. The range of applications goes from simple toys, to medical equipment and complex space missions conducted by NASA. Hence, the members of the team have decided to become active participants with Luke's Robotic Hand application.

The idea behind the robotic hand is to provide the user with an economic while efficient alternative to similar products currently available on the market. Its most valuable use is in the prosthetics industry, but it can also be used on industrial processes that cannot be done directly by humans (extreme temperatures), or as marketable toy.

II. Specification and Requirements

The goal of this project is to recreate human like movements of the hand. Those movements are accomplished by combining several electronic components, a mechanical limb, and the appropriated algorithm to program the microcontroller, which is the heart of Luke's hand implementation. The size of the fingers as well as the palm's of this robotic hand are smaller than a real human hand, which constrains a more realistic solution with the application to emulate a real human hand. The decision to use aluminum at the beginning or plastic at the final prototype was taken to exploit the natural characteristics of such materials, which helps to construct the skeleton of hand and fingers strong, resistant, and tremendously light in weight.

The size of the robotic hand is about 15 cm long and about 7 cm wide. The dimensions of the palm are 5.5 cm long in the front of the hand. The joints of phalanges as well as articulations between fingers and palm should not add friction to the mechanism that executes flexion and extension movements. Each finger has to have a string attached internally from its tip and run down internally until the arm. Those strings would need to act as tendons to implement finger movements.

Luke's hand would be able to grab objects of up to 2 pounds of weight. The hand is also required but not limited to hold objects of different shapes between 3.0 cm and 6.0 cm of diameter at the section where the finger make contact with the test object.

Electronics components, as describe above, are fundamental for the application to perform desired activities, and a microprocessor is needed to coordinate such activities by administrating the other hardware resources. Choosing a microcontroller for the project was critical, and with endless options due to the wide spectrum of microchips available in the market. Thus, a research was required to decide the best option according to the application to be developed among with its specifications. Characteristics desired of the microchip are: low power consumption, well documented embedded applications, and USB interface capabilities would be required.

The specifications of the required microchip would be:

- Low power consumption
- At least 30 I/O pins
- USB to Serial communication
- 256KB of RAM and ROM memory space
- Control of clock speed by software allows dynamic trading of power vs. speed
- DIP package, convenient for prototyping
- Good documentation available

The microprocessor is going to help the senior design group to control the robotic hand; however, it is intended to build a custom made Printed Circuit Board (PCB) for the application. Additionally, an USB interface is going to be needed in order to connect the PCB to the computer and program the microprocessor using the appropriated programming language for the microchip selected.

Custom PCB board is also required as a central part of the system. The electronics components board will either be ordered from a specialized vendor, or designed and printed on the lab depending on the complexity of the final design of such project, budget, and time available.

The robotic hand would have three different modes of control and those are: using a 4-bit switch, a sonar detector that activates the grabbing functionality, and a glove interface. With the 4-bit switch, depending on its position, short programs previously loaded from the computer, tested, and save would be executed to perform different movements of the fingers.

A single battery of 12Volts might be required as a power source for the robotic hand. The power is going to be distributed to the microcontroller, sensors, and the different electronic components of the PCB. In addition, it would be necessary to build a voltage converter and/or voltage regulator, which is going to takes the voltage supply and convert it to the right voltage needed by each component. A separated battery will be required as power supply for the servomotors.

Software Requirements:

- Well-known programming language like C or Java
- Robust Integrated Development Environment (IDE)
- Low cost IDE
- Instruction Set Architecture (ISA) support.
- Memory allocation through assembly instructions
- Libraries should include support for modern interfaces (USB, FireWire)
- Modularity of the program.

III. Application Components

A. Hand

The hand itself is a mechanical limb with twelve (12) degrees of freedom, three for each finger, which are controlled by a single servomotor per finger; the thumb is a static finger fixed against the palm. The hand is commercially available and inexpensive. This component is made out of plastic, which is a lightweight material, and therefore, less force is required when executing movement instructions. It also has a built in tendons system that contracts all the fingers simultaneously, but with a small modification of the original design, the movement of fingers get isolated one from the others. Figure 1 shows the chosen plastic hand before modifications to be used for this project.

Figure 1. Plastic Hand commercially available

B. Microcontroller

A fundamental component of this project is the Propeller microchip P8X32A-D40, designed and built by Parallax Inc. This microcontroller, shown in Figure 2, was chosen for its truly pipeline design of eight (8) 32-bit processors in series called cogs, which can execute different tasks in parallel simultaneously, giving a throughput of up to 160 Millions of Instructions per Second (MIPS), 20 MIPS per cog. Those cogs may work independently or in cooperation's by sharing resources through a hub, which is the coordinator of assets in the chip. All the cogs share the same clock signal to give them a needed synchronization.

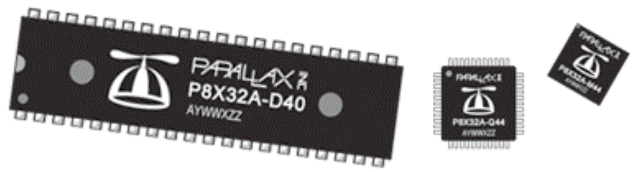

Figure 2. These are the Different IC Packages offered by Propeller. Reprinted with permission of the copyright owner, Parallax, Inc © 2008. All rights reserved.

Propeller's internal RC oscillator runs at a maximum of 12 MHz, but with some hardware manipulation achieves up to 80MHz of external clock speed. The chip requires an input voltage of 3.3 volts, producing very low power consumption at impressive high speed. The microcontroller has 32 I/O pins and the current source sink per I/O is 40 milliamps. Such characteristics made the Propeller chip ideal for a real time system with instructions to be processed simultaneously at the level of each finger of the robotic hand while reading and processing data from other inputs.

Parallax provides the Propeller tool, which is an Integrated Development Environment (IDE), with two different programming languages to code the Propeller: Spin, which is an object-oriented programming language, and Propeller Assembly as the second alternative. Those two programming languages can also be combined in the same source code if desired. Spin, is a programming language developed exclusively for Propeller by its manufacturer as a response to other complicated IDEs, which some times turn away users even from good products.

C. Servomotors

Hobby servomotors were selected to control Luke's hand finger movements. This type of motors requires a constant 4.8 to 6.0 volts in addition to a digital signal, which provides the pulse width modulation (PWM) required to control the movement of the motor in order to determine the position of the servo. Hence, servomotors are composed of three cables: a positive voltage, ground, and signal. These devices are designed to achieve accurate rotation and precise positioning over the range of rotation.

The chosen servomotor, Futaba S3004, has a maximum degree of rotation of One hundred and sixty degrees when a 2ms square wave signal is applied. It has a speed of 0.11/0.09 seconds per sixty degrees. This servo can deliver a 2.6/3.0 Kilograms per centimeter of torque. This torque is enough for contracting the fingers and giving the force necessary to grab objects with in specifications of the project. Futaba S3004w servomotor used in this application is shown below in Figure 3.

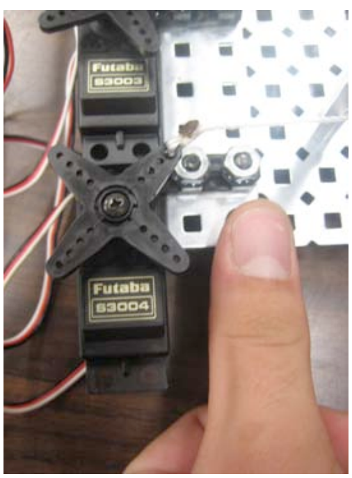

Figure 3. Futaba S3004 servomotor

The required pulse for servomotors varies from 1 ms to about 2 ms, which means that when 1ms pulse is sent, the servo is commanded to turn all the way in one direction, but when 2 ms pulse is sent, then it moves all the way in the other direction. So a pulse of 1.5 ms instructs the servo to move to its center or neutral position. The utilization of those characteristics would be discussed in the design and implementation section.

D. Flex Sensor

The flex sensor manufactured by Spectra Symbol is utilized in the globe interface designed as one of the control modes for the robotic hand. It is commercially available in Trossen Robotics and comes in 4.5 inches of length. The flex sensor varies its resistance across the pads while it is bent. The flex sensor resistive statistics are:

Straight (extended): about 9000 Ohm; 90 degree bent: about 14000 Ohm; 180 degree bent: about 22000 Ohm.

The flex sensors runs with 3.3 volts and they are connected directly to an A/D converter that will give a 12-bit representation of the voltage variation.

E. Sonar Sensor

Ping Ultrasonic Sensor is a parallax product that assists the application to detect an object within certain distance range and triggers the robotic hand to react according to the program loaded in the microcontroller. This type of detector is very effective and works with more accuracy than IR sensors to detect objects because the light of the room where the test is performed is not a factor, also the objects to be detected can be of white color or transparent, which is an inconvenient for IR sensors. This sonar sensor can perform measurements for moving or stationary objects. The pulse is transmitted from the component and distance-to-target is determined by measuring the time required for the echo to go and return to the receiver portion of the device.

The sonar sensor provides precise, non-contact distance measurements in the range from 2 cm to 3 m, and requires 20 mA to function with very low power consumption.

F. Power Supply

Electronic components of the robotic hand require DC power for proper operation, and a good power supply circuit has to maintain a constant voltage across the load with changes in load current. The circuit designed for the application is divided in different block, which are: Power, Power Servos, EEPROM, Sonar Sensor, Flex Sensors, Servos, USB to Serial Communication, and the microcontroller. There are also some LEDs used to indicate certain operations. Figure 4 below shows the circuit designed for the Luke's hand, which was implemented in a single Printed Circuit Board.

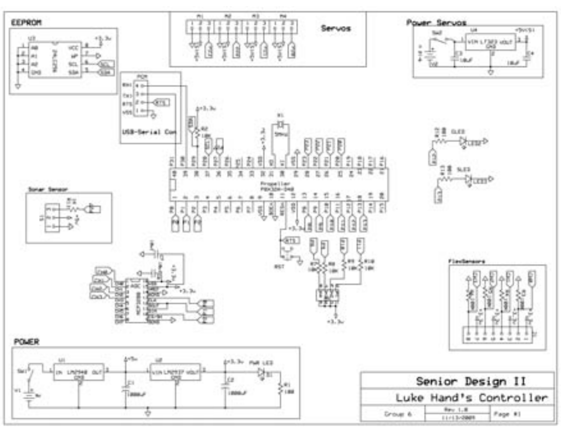

Figure 4. Circuit Design

Power block is the combination of two Linear Voltage Regulators. Those voltage regulators are used to reduce the 9 volts input to the voltages required by the electronic components of Luke's hand. The importance of IC linear voltage regulators is their built-in protection from over current conditions and overheating. The first regulator is LM2940, which outputs 5 V-1A to the Ping Ultrasonic Sensor. The second regulator is LM2937, which outputs 3.3V-500mA to power the microcontroller (P8X32A-D40), the Flex Sensors, the EEPROM, and the Analog-Digital Converter (MCP3208). Also the power block has a LED, which is used to indicate when power is on.

Power Servos Block is used to power the four servomotors, which are used to control the movements of each finger. The IC used is LT323, which deliveries 5V and load current of up to 3A. The maximum current draw for all four servos can reach up to 1.5A. Hence, it is recommended to have higher current source than the actual current calculated in order to prevent any lack of power in the motors. The power source is going to be two 9V batteries connected in parallel in order to obtained higher ampere-hour (Ah) ratings.

Flex Sensors block requires 3.3 volts. It has an eight-pin connector and four voltage divider circuits. The four flex sensors that are located in the glove interface are connected to the voltage dividers. Each flex sensor is a resistive sensor, which increases its value as the bent increases. Flex sensors are connected to a fixed 100Kohms resistor. When the Flex Sensor is straightened up, the resistance is about 9Kohms, and when it is 180 degrees bend its resistance increases to 22Kohms.

The following calculations show the voltage changes as the bent increases.

$R_{unflex}=9K\Omega$
$R_{fixed}=100K\Omega$
$R_{flex}=22K\Omega$

$$V_i = 3.3V * 100K\Omega/109K\Omega = 3.03V \quad (1)$$

$$V_f = 3.3V * 100K\Omega/122K\Omega = 2.70V$$

A/D converter changes the input voltage to a digital value. The A/D is a MCP3208, which is a 12 bit, 8-channel, Serial Peripheral Interface Analogue to Digital Converter. So, the theoretical digital output code produced by the A/D converter is a function of the analog input signal and the reference input, as shown below.

Definitions: V_{in}=Analog Input Voltage; V_{ref}=Reference Voltage (3.3V); ADC bits resolution is 2^12=4096; Digital Output Code (DOC)

$$DOC = Vin*4096/V_{ref} \quad (2)$$

$$DOC=3.03*4096/3.3=3761$$
$$DOC=2.70*4096/3.3=3351$$

With the digital code is easier to calculate the pulses that the microcontroller has to send to the servomotor in order to move the fingers as accurate as possible to emulate the globe user's movements.

USB-Serial Communication block is used as a bridge to connect the microcontroller with the computer in order to load various programs to the microchip and then to the EEPROM. The main device in this block is Propeller Plug, which is connected to P30 and P31 of the microcontroller. The Plug is capable of asynchronous communication up to 3M baud with 3.3V and 5V devices such as Propeller microcontroller. Propeller Plug is power up by the computer's USB port with 5V needed for the plug to work.

EEPROM block is a 32KB (24LC256) device used to store the programs loaded in the microcontroller. The power supplied is 3.3V and it is connected to pins (P28, P29) of the microcontroller. Servo block is just a set of four servo connectors, which are fed by 5V, ground, and signal pins from the microcontroller. The Sonar Sensor Block is a three-pin connector that is going to be connected to the Ping ultrasonic sensor, which is located at the palm of the robotic hand.

The PCB design was done using Express PCB, which is a software application easy to use and free to download. The final design is shown below in figure 5.

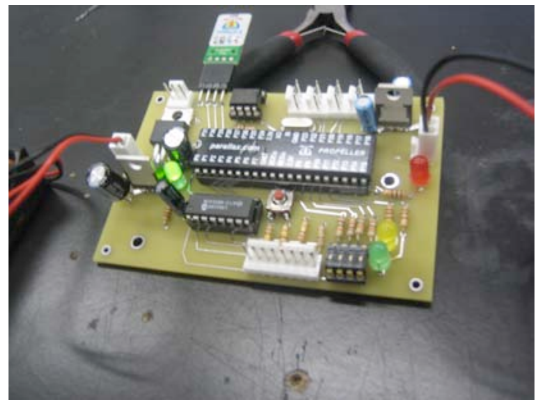

Figure 5. Final PCB Assembled

G. 5 MHz Crystal

Another important component in the circuit design is the 5 MHz crystal, which is connected to X0 and X1 of the microcontroller. It is recommended to used this crystal size for application that are timing sensitive like serial communication, tone generation, servo control, and time keeping, because instructions have to be done fast. So, by using an external crystal the microcontroller can have a system clock of 80MHz. This crystal is efficient and inexpensive.

IV. Design and Implementation

After selecting a commercially available plastic hand, which in its original design moves all its fingers simultaneously, it was tore apart to eliminate the actuator mechanism and isolate the fingers from each other. Inside the hand, there was a plastic string that connected the tip of the finger with the extracted actuator mechanism internally at the base of the fingers. Each plastic string was then attached to a nylon string and extended internally to the arm where the servomotors were placed. The intended purpose of isolating the fingers was to give them independence at the time of executing different computer programs to control the robotic hand.

The previously mentioned torque produced by servomotors is used to pull the string system acting as tendons attached directly from the tip of the fingers to the arm of each servo in order to contract the fingers of the robotics application. Such servomotor's functionality is desired because they pull the strings straight down with out the need of any additional mechanism for wrapping those strings on a rod like the DC motors would need to do. It is worthy of mentioning that those wrapping mechanism requires extra space and some times they become the source for many failures.

The characteristics of servomotors described above in the application components section are exploited in the robotic hand to represent the contraction and extension of the fingers, so when the fingers are extended, the signal reaches its maximum value, and for contracted finger the opposite would be true, which makes the 1.5 ms half way of the full range in a finger movement.

Luke's hand will feature an array of functions that would be control essentially by three different modes:

1. Switch Mode

The main mode is a 4-bit switch that at position 0000, where zero represents a low signal and one means high voltage, the application stays in standby mode. Positions from one to four are reserved for the hand to show the implementation of counting, which is executed by flexing some fingers and keeping the others extended to signal those given numbers. The robotic hand has also other programs to arrange the finger in such a way that it shows the sequence operation, which is counting from one to four in a single loop; the "love sign" is made by contracting all the fingers but the pinky, which in conjunction with the fixed thumb give such representation; and "greetings sign" is put into operation by waving the four fingers. Additionally, for positions eight and nine of the switch, the grabbing operation was implemented for objects of different shapes.

2. Sonar Detector

Another control mode is implemented with the utilization of a sonar detector and it was located in position 1010 of the 4- bit switch. Such sensor provides feedback to the users and system to perform task such as sensing objects within 2 cm from the palm, and signaling the servomotors to grab the detected object. The sonar sensor was attached to the palm of Luke's hand. After opening the hand's palm, measuring empty space and adapting palm with respective holes for the receiver and transmitter; the sonar sensor was attached without obstructing the tendons channels.

Along with the sonar detector, there was the idea of using force sensors in the tip of the fingers to avoid any damage on fragile objects that could be used for testing the application. Unfortunately, the chosen plastic hand limits the application from installing the force sensors because in the tip of the fingers there is a metallic rivet that attaches the tendon mechanism to the actual finger. Those rivets are fundamental for the hand functionality and cannot be removed or relocated without seriously compromising the movement of the fingers.

3. Globe Interface

The third mode of control implemented for the application is a globe interface, which was located in 1011 of the 4-bit switch. The interface between the user and Luke's hand is a smart flex globe. This globe contains four flex sensors attached to the back of each globe's finger. The flex sensor's output is a direct correlation between the change of human hand's position controlling the globe interface and the voltage drop generated at the flex sensors. While the finger flexes and extends, the voltage output changes and sends a signal to the analog to digital converter (A/D converter), where a 12-bit representation of the voltage is calculated. When the fingers are completely extended, the usual 12-bit value of the voltage output is about 3550. Subsequently, the usual 12-bit value of the voltage when the fingers are completely flexed is about 2100.

A simple relation between the range of finger's flexion and the range of the servos' motion is calculated on the globe controller program. The program calculates a percentage of the voltage difference. Since the lower and higher boundaries are set for the 12-bit voltage value, a percentage from their difference is used to make a correlation with the servo's motion. The range of the pulse signal given to the servos to rotate clockwise and counterclockwise is also set. In conclusion, if the finger flexes 25% of the total flexing range, then the same percentage would be used to rotate the servomotor in order to flex the robotic hand's fingers and somehow emulate the human hand in the globe interface.

V. Test Cases

The following is a comprehensive of the test cases that the hand underwent during and after the design process.

- Grab small lightweight object with rough surface: Cell phone (successful)
- Grab small lightweight object with smooth surface: Candy bar (successful)
- Small object with non-uniform shape: Set of keys (successful)
- Hold heavier object: Soap bottle (successful)
- Sense an object with in 2cm of the palm and grab it: Red bull can (successful)
- Grab a cell phone while controlling the hand with the glove: successful
- Count to four using the switch interface: successful
- Switch between all operating modes using the DIP switch: successful
- Test reaction time of the servos relative to the glove's input signal: Less than 100ms (successful)

References

[1] Jame Cox, (2002). Fundamentals of Linear Electronics Integrated and Discrete, 2nd Edition. USA: Delmar Thomson Learning.

[2] Bernard Grob & Mitchel E. Schultz (2003). Basic Electronics, 9th Edition. USA: Glencoe/McGraw Hill.

[3] Gordon Mc Comb & Myke Preoko, (2006). Robot Builder's Bonanza, 3rd Edition. USA: McGraw-Hill Companies.

[4] Ben-Zion Sandler, (1991). Robotics: Designing the Mechanism for Automated Machinery. USA: Prentice-Hall, Inc.

[5] Parallax. Web site: http://www.parallax.com/

[6] Microchips. Website: http://www.microchips.com/