

Summer 8-31-2003

## A biologically inspired joint model using engineering methods to enhance understanding of muscle activity

Richard Paradiso  
*New Jersey Institute of Technology*

Follow this and additional works at: <https://digitalcommons.njit.edu/theses>



Part of the [Biomedical Engineering and Bioengineering Commons](#)

---

### Recommended Citation

Paradiso, Richard, "A biologically inspired joint model using engineering methods to enhance understanding of muscle activity" (2003). *Theses*. 657.  
<https://digitalcommons.njit.edu/theses/657>

This Thesis is brought to you for free and open access by the Electronic Theses and Dissertations at Digital Commons @ NJIT. It has been accepted for inclusion in Theses by an authorized administrator of Digital Commons @ NJIT. For more information, please contact [digitalcommons@njit.edu](mailto:digitalcommons@njit.edu).

## Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

**Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation**

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

## **ABSTRACT**

### **A BIOLOGICALLY INSPIRED JOINT MODEL USING ENGINEERING METHODS TO ENHANCE UNDERSTANDING OF MUSCLE ACTIVITY**

**by  
Richard Paradiso**

Compliant actuators and control methods have been known to exhibit similarities in human musculoskeletal systems. In order to better understand and improve the effects of force optimization under closed-loop conditions, a physical joint model was constructed with an agonist and an antagonist muscle operating under linear control. Utilizing LabVIEW software, compliant McKibben air muscles and Merlin stretch sensors, the author was able to incorporate a Bang-Bang controller and propose the inclusion of pulse width modulation (PWM) as well as a Proportional-Integral-Derivative (PID) controller to study the reflex loops under various levels of feedback sensitivity. The feedback mechanism, similar to proprioceptive muscle spindle feedback, will be based on the input given from the stretch sensors to achieve a desired movement.

Motions are controlled by the Central Nervous System (CNS). Demonstration of an open-loop muscle model working in conjunction with an inertial system would be extremely difficult. However, understanding the control process of a muscle can be demonstrated through the closed-loop control method. This was accomplished by constructing a linear bang-bang controlled computer program to simulate a musculoskeletal model that incorporates proprioceptive feedback. Simulink software was also utilized to show how the damping and stiffness coefficients, as in human muscle spindle and Golgi Tendon Organ feedback loops, can be adjusted to optimize stability.

**A BIOLOGICALLY INSPIRED JOINT MODEL USING ENGINEERING  
METHODS TO ENHANCE UNDERSTANDING OF MUSCLE ACTIVITY**

**by  
Richard Paradiso**

**A Thesis  
Submitted to the Faculty of  
New Jersey Institute of Technology  
in Partial Fulfillment of the Requirements for the Degree of  
Master of Science in Biomedical Engineering**

**Department of Biomedical Engineering**

**August 2003**

**APPROVAL PAGE**

**A BIOLOGICALLY INSPIRED JOINT MODEL USING ENGINEERING  
METHODS TO ENHANCE UNDERSTANDING OF MUSCLE ACTIVITY**

**Richard Paradiso**

**Dr. Richard A. Foulds, Thesis Advisor** **Date**  
**Associate Professor of Biomedical Engineering, NJIT**

**Dr. Stanley Reisman, Committee Member** **Date**  
**Professor of Biomedical Engineering, NJIT**

**Michael Bergen, Committee Member** **Date**  
**Adjunct Professor of Biomedical Engineering, New Jersey VA Medical Center**

## **BIOGRAPHICAL SKETCH**

**Author:** Richard Paradiso

**Degree:** Master of Science

**Date:** August 2003

### **Undergraduate and Graduate Education:**

- Master of Science in Biomedical Engineering,  
New Jersey Institute of Technology, Newark, NJ, 2003
- Bachelor of Science in Mechanical Engineering,  
New Jersey Institute of Technology, Newark, NJ, 1999

**Major:** Mechanical Engineering

### **Presentations and Publications:**

Richard Paradiso and Richard A. Foulds,

“A biologically inspired joint model to enhance understanding of spasticity,”  
The 29<sup>th</sup> Annual Northeast Bioengineering Conference, Newark, NJ, March 2003.

**To my beloved family  
and in honor of my father**

## **ACKNOWLEDGMENT**

I would like to convey my deepest appreciation to Dr. Richard Foulds, who not only served as my research advisor, providing invaluable resources and vast insight, but also encouraged and supported my efforts. Special thanks are given to Dr. Stanley Reisman and Michael Bergen for participating in my committee. I could not have completed this degree without the generous funding from my employer, Merck & Co., Inc., for which I will be forever grateful.

Special thanks go to the knowledgeable engineers at the National Instruments Inc. Company, who have shared their expertise and professionalism. My fellow graduate students in the Biomedical Engineering laboratory are deserving of recognition for their support.

## TABLE OF CONTENTS

<b>Chapter</b>	<b>Page</b>
1 INTRODUCTION .....	1
1.1 Objective .....	1
1.2 Background Information .....	1
1.3 Summary .....	6
2 COMPONENT INTEGRATION.....	8
2.1 McKibben Air Muscles.....	8
2.2 Merlin Stretch Sensors.....	11
2.3 Wheatstone Bridge.....	15
2.4 Transistor Circuit .....	19
2.5 Mead Fluid Dynamic Valves .....	21
2.6 Data Acquisition Card.....	24
2.7 Summary .....	27
3 METHODOLOGY AND RESULTS .....	28
3.1 Problem Statement.....	28
3.2 Musculoskeletal Model.....	30
3.3 Bang-Bang Control .....	33
3.4 Pulse Width Modulation .....	38
3.5 PID Control.....	39
3.6 Summary .....	41

**TABLE OF CONTENTS**  
**(Continued)**

<b>Chapter</b>	<b>Page</b>
4 CONCLUSIONS.....	42
4.1 Discussion .....	42
4.2 Future Directions .....	43
APPENDIX A STATIC PHYSICAL MODEL OF MCKIBBEN MUSCLES .....	44
APPENDIX B SPECIFICATIONS OF NI-DAQCARD-AI-16E-4 .....	46
APPENDIX C I/O CONNECTOR BOARD FOR DAQCARD-AI-16E-4 .....	54
APPENDIX D DIFFERENTIAL VOLTAGE COMPARISON TABLE.....	59
APPENDIX E BANG-BANG CONTROLLER PROGRAM .....	60
APPENDIX F PULSE TRAIN GENERATION PROGRAM .....	67
APPENDIX G E-SERIES PID CONTROL PROGRAM.....	72
REFERENCES .....	79

## LIST OF TABLES

<b>Table</b>	<b>Page</b>
2.1 Physical Properties of Merlin Stretch Sensors.....	11
2.2 Flow Control Patterns of a Single Air Muscle.....	22
2.3 NI-DAQCard Input Configurations .....	26
C.1 I/O Connector Signal Descriptions .....	55
C.2 I/O Connector Signal Descriptions (Continued).....	56
C.3 I/O Signal Summary for the DAQCard-AI-16E-4 .....	57
C.4 I/O Signal Summary for the DAQCard-AI-16E-4 (Continued).....	58

## LIST OF FIGURES

<b>Figures</b>	<b>Page</b>
1.1 Block diagram of whole system.....	6
1.2 Output plot of Bang-Bang control program.....	7
2.1 McKibben air muscle.....	9
2.2 Pressure force transformation principle.....	10
2.3 Merlin stretch sensor concept.....	12
2.4 Merlin stretch sensor overshoot problem.....	14
2.5 Merlin stretch sensor release problem.....	14
2.6 Fixed displacement using linear actuator.....	15
2.7 Basic Wheatstone Bridge circuit.....	16
2.8 Wheatstone Bridge circuit on breadboard of model indicated by circle.....	17
2.9 Circuit diagram of Wheatstone Bridge used for model.....	18
2.10 Transistor power amplifier circuit board.....	19
2.11 Motorola P2N2222A transistor.....	20
2.12 Darlington Pair arrangement of transistors.....	20
2.13 Mead Fluid Dynamics solenoid three way valve.....	21
2.14 Circuit diagram for buffer circuit.....	23
2.15 DAQCard-AI-16E-4 low-level block diagram.....	25
2.16 CB-68LP I/O connector block.....	27
3.1 Biologically inspired single joint model.....	28
3.2 Musculoskeletal model with intrinsic and proprioceptive feedback.....	31
3.3 Simulink musculoskeletal model with intrinsic and proprioceptive feedback.	32

**LIST OF FIGURES**  
**(Continued)**

<b>Figures</b>	<b>Page</b>
3.4 Joint angular position of Simulink model.....	33
3.5 Joint angular velocity of Simulink model.....	33
3.6 NI MAX analog input configuration.....	34
3.7 NI MAX digital output configuration .....	35
3.8 Front panel of NI LabVIEW Bang-Bang control program.....	36
3.9 VI of NI LabVIEW Bang-Bang control program .....	37

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1 Objective**

The objective of this master's thesis is to present a working model of a biologically inspired joint utilizing an agonist and an antagonist muscle under the Bang-Bang control method. Proportional-Integral-Derivative (PID) control and pulse width modulation (PWM) techniques are proposed for optimum results.

For the Bang-Bang control method, different fixed perturbations were exhibited by the system. The data from the sensors used in this model were determined to provide adequate feedback for achieving integration with the McKibben air muscles. The control of air pressure to each muscle was established through the reading of two Merlin stretch sensors.

For the PID control method, the advantages of better performance and stability are measured over the Bang-Bang method. Also, PWM is introduced to further add as a powerful technique for controlling the analog circuit with digital outputs. A LabVIEW program incorporating PWM, as well as PID control, is proposed.

### **1.2 Background Information**

In an effort to better understand muscle activity, it is important to note how muscle activation occurs, from a biological viewpoint. Muscle spindles are important proprioceptors. The muscle spindle is located between and among extrafusal fibers. Their

job is to detect muscle kinematics (length and velocity). The intrafusal fibers associated with a spindle can themselves contract under central nervous system (CNS) control which can alter their sensitivity to absolute muscle kinematic changes. The intrafusal fibers, also called spindle fibers, are located within a connective tissue envelope. Muscle spindles signal muscle length and velocity, while Golgi Tendon Organs signal muscle tension [1]. Human motion has often been compared to robot motion. The underlying systems however, differ dramatically. Human actions consist of highly non-linear elements such as multiple degree of freedom joints, muscles, ligaments and sensory organs. Furthermore, the system has kinematic and actuator redundancies. Robot movement takes on a much simpler form which incorporates a trajectory planner algorithm and a position (and sometimes velocity) feedback control. Major differences include the use of agonistic muscles as the biological actuator, compared with torque motors used in robots [23].

Each muscle spindle consists of sensory nerve endings wrapped around special muscle fibers called spindle fibers (also called intrafusal fibers) [7]. Stretching a spindle fiber initiates a volley of impulses in the sensory neuron (called an "I-a" neuron) attached to it. This I-a neuron contains velocity information. The impulses travel along the sensory axon to the spinal cord where they form several kinds of synapses:

1. Some of the branches of the I-a axons synapse directly with alpha motor neurons. These carry impulses back to the same muscle causing it to contract.
2. Some of the branches of the I-a axons synapse with inhibitory interneurons in the spinal cord. These, in turn, synapse with motor neurons leading back to the antagonistic muscle, otherwise known

as a flexor. By inhibiting the flexor, these interneurons aid contraction of the extensor, or the agonist.

3. Still other branches of the I-a axons synapse with interneurons leading to brain centers, e.g., the cerebellum that coordinates body movements.

The contractions of all muscles are triggered by electrical impulses, whether transmitted by nerve cells, created internally (as with a stimulator) or applied externally (as with an electrical-shock stimulus) [1]. The electrical signal sets off a series of events that lead to cross-bridge cycling between the thick myosin and thin actin filaments, which in turn, generates force. The series of events is slightly different between skeletal, smooth and cardiac muscle [7]. The focus of this work is on the skeletal muscle portion where the thick and thin filaments do the actual work of a muscle. Thick filaments are made of a protein called myosin. At the molecular level, a thick filament is a shaft of myosin molecules arranged in a cylinder. Thin filaments are made of another protein called actin. The thin filaments look like two strands of pearls twisted around each other [1].

During contraction, the myosin thick filaments attach to the actin thin filaments by forming crossbridges [13]. The thick filaments pull the thin filaments past them, making the sarcomere shorter. In a muscle fiber, the signal for contraction is synchronized over the entire fiber so that all of the myofibrils that make up the sarcomere shorten simultaneously [8]. There are two structures in the grooves of each thin filament that enable the thin filaments to slide along the thick ones: a long rod-like protein called

tropomyosin and a shorter bead-like protein complex called troponin. Troponin and tropomyosin are the molecular switches that control the interaction of actin and myosin during contraction [7].

In order to better understand and replicate the complex control of the CNS, investigation was required into several musculoskeletal system models that biomedical researchers developed. These systems are then transformed from highly non-linear into linear to give the ability to pursue a quantitative approach to investigate the CNS and the peripheral system. Although much consideration has been given to the individual components of these systems, little research has examined their contribution to the system as a whole [23].

Voluntary movement is accomplished by the execution of motor programs for planned forces and corresponding EMG patterns in the muscles. The CNS uses relatively simple coordination rules among muscles and joints that greatly simplify the problem of finding muscle activation patterns to satisfactorily approximate kinematic goals. The well-known kinematic features of movements result from a trial and error tuning of force profiles based upon visual and kinesthetic feedback [26].

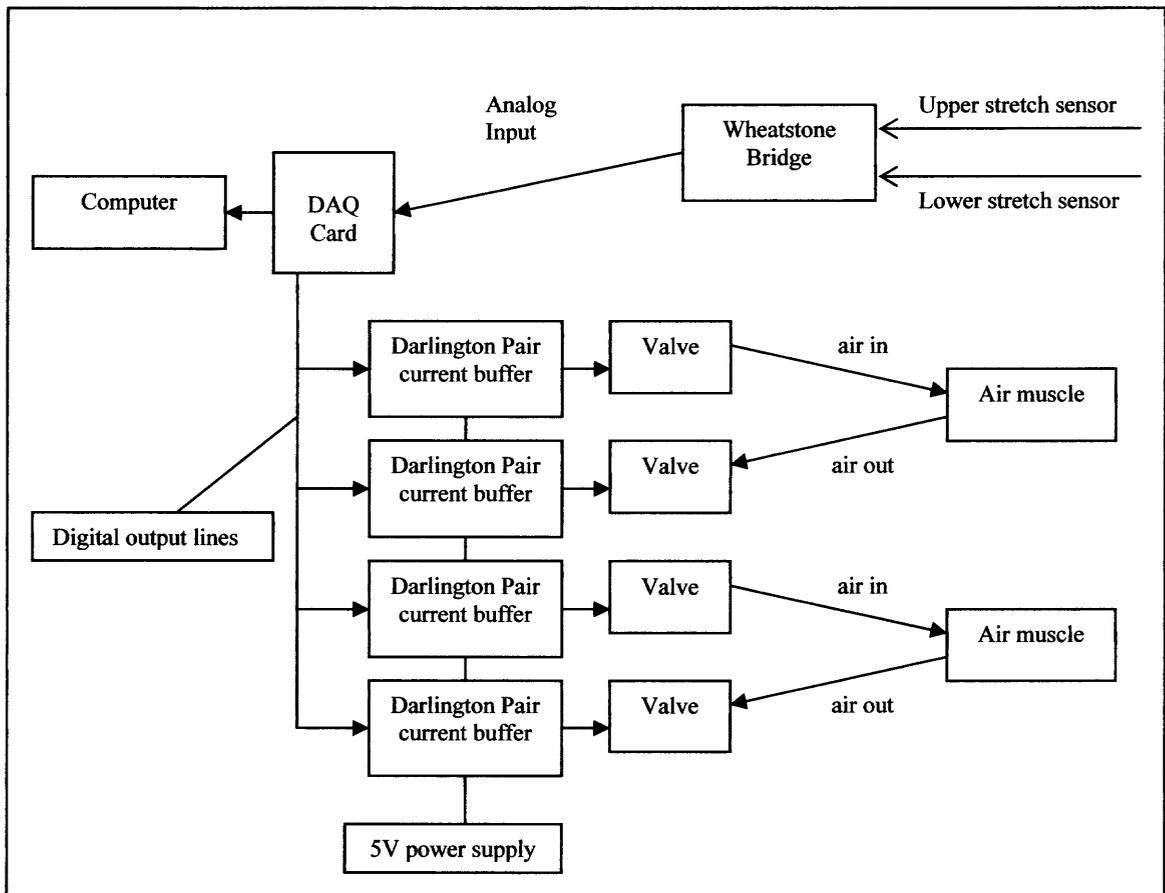
In order to fully exploit musculoskeletal modeling systems, researchers have suggested that open-loop muscle models (i.e. without proprioceptive feedback) must incorporate an inertial component [6]. Whenever open-loop muscle models are applied in a musculoskeletal model, the permanency of the whole model depends completely on the intrinsic viscoelastic properties of the muscles, resulting from the cross-bridge stiffness

(if represented) [17]. Also, much consideration has been given to combining the force-length and force-velocity relationships, in relation to the passive dynamic properties such as segment inertia and joint viscoelasticity, to ensure stability in the system. Inverse dynamic simulations can result in changeable solutions, and forward-dynamic simulations will tend to be borderline stable systems, since no additional effort will be spent on stability [13]. Therefore, open-loop muscle models will typically underestimate the physical energy needed for stabilization of the limbs, since there is likelihood that human beings will always keep a certain safety region from an unstable arrangement of the body and its limbs. In open-loop musculoskeletal systems, co-contraction is the only means for increasing the impedance (i.e. the stiffness and viscosity) of the system [27]. The stability of the optimized solutions has never been a subject of discussion.

From a controls engineering viewpoint, the author is only interested in the control of simple positioning tasks within the sagittal plane. Morasso noted that the common features among the different reaching movements are the single peaked shape of the hand tangential velocity and the straight shape of the hand trajectory. Soechting and Lacquaniti further noted that these properties were unaffected by changes in the load held in the hand or by the intended speed of movement. These properties of straightness and bell-shaped velocity profiles have become defining features of unconstrained joint movements, even though Hollerbach noted that movements in the sagittal plane tended to be more curved than those in the horizontal plane [23].

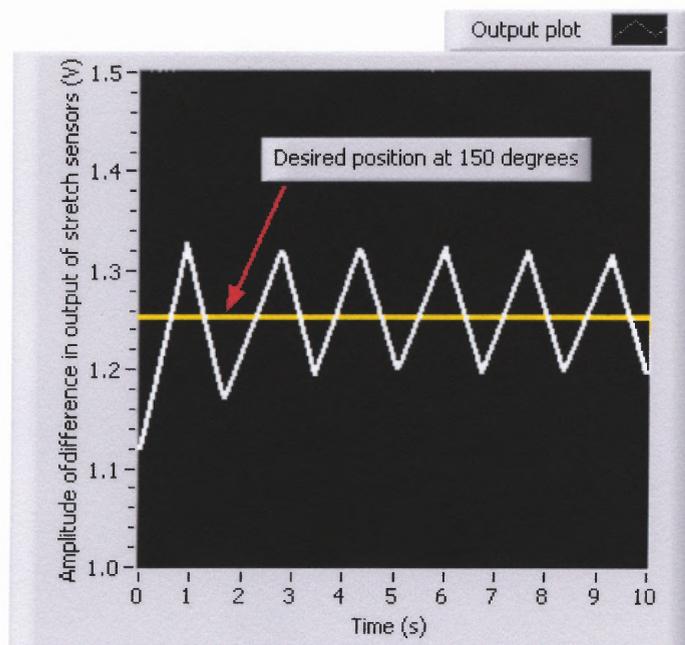
### 1.3 Summary

In this experiment, the physical model is constructed in a manner that minimizes the effect of non-linear properties. Examination of a single degree-of-freedom movement, and limited range of motion enable optimization of feedback loops to achieve stability. Stiffness and damping are accomplished by adding muscle force-length and force-velocity relationships to the closed-loop system. A block diagram of the whole system is shown below.



**Figure 1.1** Block diagram of whole system.

The control of movement in this model requires co-activation of agonist and antagonist muscles. This is done by applying a force in one muscle to get the joint moving, and turning on the other muscle to act as a brake to slow the joint to achieve the desired target angle. Although co-activation was executed in this experiment, through the Bang-Bang control method, further investigation was needed to increase the amount of overall stability. The plot below demonstrates the oscillations achieved about a desired destination using the on/off method of control at a sample rate of 250 kHz.



**Figure 1.2** Output plot of Bang-Bang control program.

Furthermore, the implementation of a finer tuned control system is difficult with a compliant air muscle. The activation of an air muscle does not require current, but uses air volume to vary muscle force. Air volume, since not controlled by pressure, is then related to time. For these reasons, other time based control methods are proposed.

## **CHAPTER 2**

### **COMPONENT INTEGRATION**

Many elements were included in constructing this single joint air muscle model. The system includes the use of two McKibben AM-02 air muscles and a 5hp compressor to emulate antagonist and agonist muscle activation. The control of the air pressure to the muscles is accomplished by four Mead Fluid Dynamics isonic three-way solenoid valves, which were modified to two-way control in order to maintain position. Merlin stretch sensors were affixed alongside the air muscles to behave as strain gauges. The strain was measured by constructing a Wheatstone Bridge circuit on a breadboard, as a means of converging the resistance change of the stretch sensors into voltage. Measuring these voltages requires the use of two analog inputs to a data acquisition system. The joint model also requires the use of four digital outputs to trigger the on/off state of the pneumatic valves. Implementation of the Bang-Bang control system was accomplished through the use of National Instruments (NI) LabVIEW software v6.0 and a NI-DAQCard-AI-16E-4 data acquisition device. The power requirements were supplied by a separate continuous 5V DC amplifier. In this chapter, the joint model composition, component limitations and effects on stability are explored.

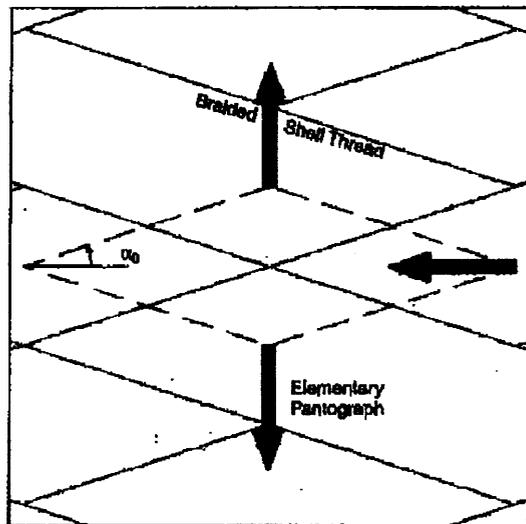
#### **2.1 McKibben Air Muscles**

The McKibben pneumatic artificial muscle was developed in artificial limb research in the 1950s by Joseph L. McKibben. They have recently been commercialized by the Bridgestone Rubber Company of Japan for robotic applications, and re-engineered by

Prof. Jack Winters for construction of bio-mechanical realistic skeletal models [21]. McKibben muscles consist of an internal bladder surrounded by a braided helical mesh shell (with flexible yet non-extensible threads) that is attached at either end to fittings or to some tendon-like structure (Figure 2.1) [20]. When the internal bladder is pressurized, the high pressure gas pushes against its inner surface and against the external shell, and tends to increase its volume. The cylindrical shape of the inner tube allows this flexible pantograph network to convert circumferential pressure forces into axial contraction force (Figure 2.2) [21]. Due to the non-extensibility (or very high longitudinal stiffness) of the threads in the braided mesh shell, the actuator shortens according to its volume increase and/or produces tension if it is coupled to a mechanical load. This physical configuration causes McKibben muscles to have variable-stiffness spring-like characteristics, non-linear passive elasticity, physical flexibility, and very light weight compared to other kinds of artificial actuators.



**Figure 2.1** McKibben air muscle.



**Figure 2.2** Pressure force transformation principle by means of the pantograph network of the muscle braided shell.

The relationships between tension, length, velocity, and activation are major characteristics of actuators which vary greatly from type to type. Human skeletal muscle also has its own particular characteristics: for example, the convex shape active tension-length relationship, the non-linear passive tension-length relationship, and the hyperbolic tension-velocity relationship. Each of these properties is also a function of activation level. In order to find the tension in a McKibben muscle as a function of pressure and actuator length without considering the detailed geometric structure, a theoretical approach based on energy conservation is introduced (Appendix A) [21].

The properties of McKibben air muscles vary according to size. The air muscles used in this experiment measure 210mm in length. The diameter is 20mm and the muscle can pull a load of 45lbs. at the maximum 50psi pressure. At 25% contraction, the air muscle can pull a load of 25lbs. A fully stretched muscle creates the largest force [20].

## 2.2 Merlin Stretch Sensors

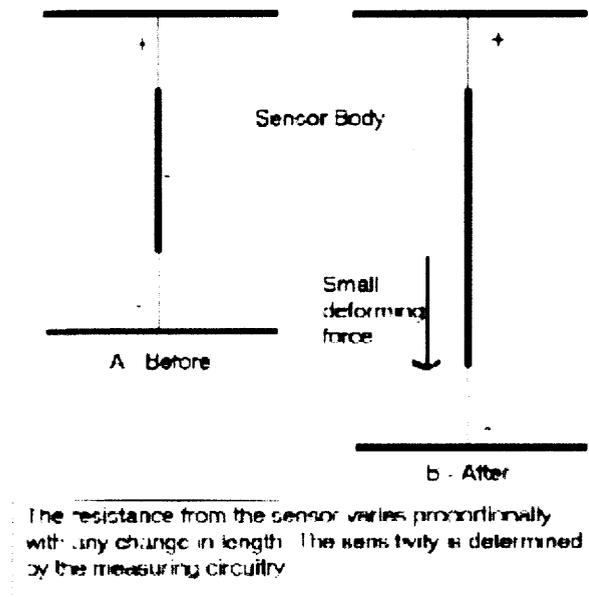
The Merlin stretch sensor is a new way to measure stretch, displacement and compression. This sensor is economical, easy to use and lightweight. The sensor comes in various shapes and sizes and can be customized to suit any application. The construction is comprised of a flexible cylindrical tube with electrical connections at both ends of the sensor body. The joint model uses the sensors to measure the elongation of the antagonist and agonist muscles and provides a means of feedback into the control mechanism. The physical properties of these sensors are shown on the following table (Table 2.1) [12].

**Table 2.1** Physical Properties of Merlin Stretch Sensors

<b>Property</b>	<b>Value</b>
Diameter	2mm (5mm at connections)
Resistance	1cm equivalent to 100 ohms
Elongation @ Breaking	300%
Recommended Max. Extension	100%
Resolution	infinite (performance varies according to capacitor value selected)
Repeatability	40-50%

The stretch sensor behaves like a variable resistor. As the length of the stretch sensor changes, so does its resistance (Figure 2.3). For each centimeter of sensor length change, there is a change in resistance of about 100 ohms. The stretch sensor can be measured a number of different ways. One method consists of connecting the stretch sensor directly to an analog multimeter to see how the resistance changes with length.

Another method is to use the sensor as part of a resistor/capacitor (RC) circuit, which will have a known decay time. If C is fixed, the decay time will be proportional to R. Similarly, a Basic Stamp 2 (BS2) system can provide instruction to measure the decay time of a resistor/capacitor circuit called RCTime. The counter provides a value that is proportional to the resistance which can be used as a simple means to take measurements from the stretch sensor. This model uses a Wheatstone Bridge, in which the resistors in two arms were replaced by the stretch sensors [12], thus producing a voltage proportional to stretch.



**Figure 2.3** Merlin stretch sensor concept.

The dynamic measurements of the stretch sensor play an important role in justifying some of the undesirable perturbations obtained. The stretch sensor has a continual decay to a resting position (rest position varies with extension). The stretch sensor will normally return to its rest position in around a minute or less (depending upon the amplitude of movement). This implies that the stretch sensor, if held in one position, will not hold a constant resistance. When taking measurements dynamically there are several key artifacts to consider:

1. **Overshoot** - when stretched to a position and held the sensor will decay rapidly to a resting position. The initial increase in value is known as overshoot (Figure 2.4).
2. **Release artifact** - when stretched and held for a time, and then released the resistance may increase before reverting to a correct value (Figure 2.5).
3. **Decay** - when being stretched or released the sensor is subject to a continual decay (Figure 2.6). The figure depicts a series of fixed displacements plotted against a known reference sensor.

The overshoot and release artifacts tend not to cause too many problems for most applications, but the continual decay means that movements of varying speed give different results [12]. However, the decay times of the stretch sensors are negligible when considering the constant velocity movements from the single-joint air muscle model.

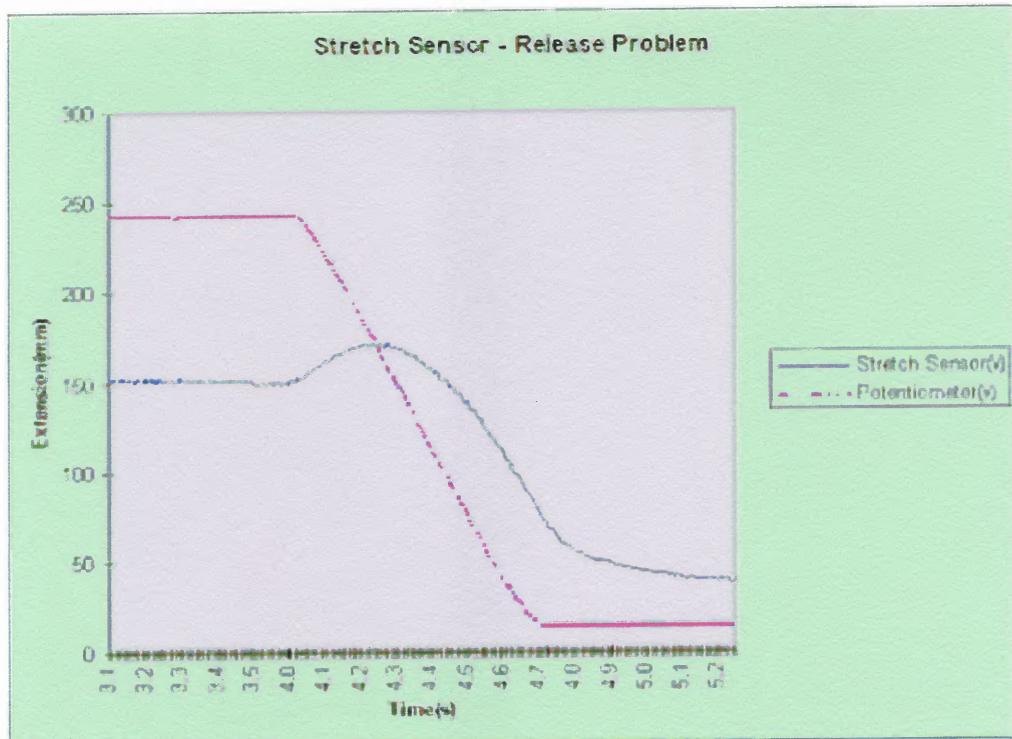


Figure 2.4 Merlin stretch sensor release problem.

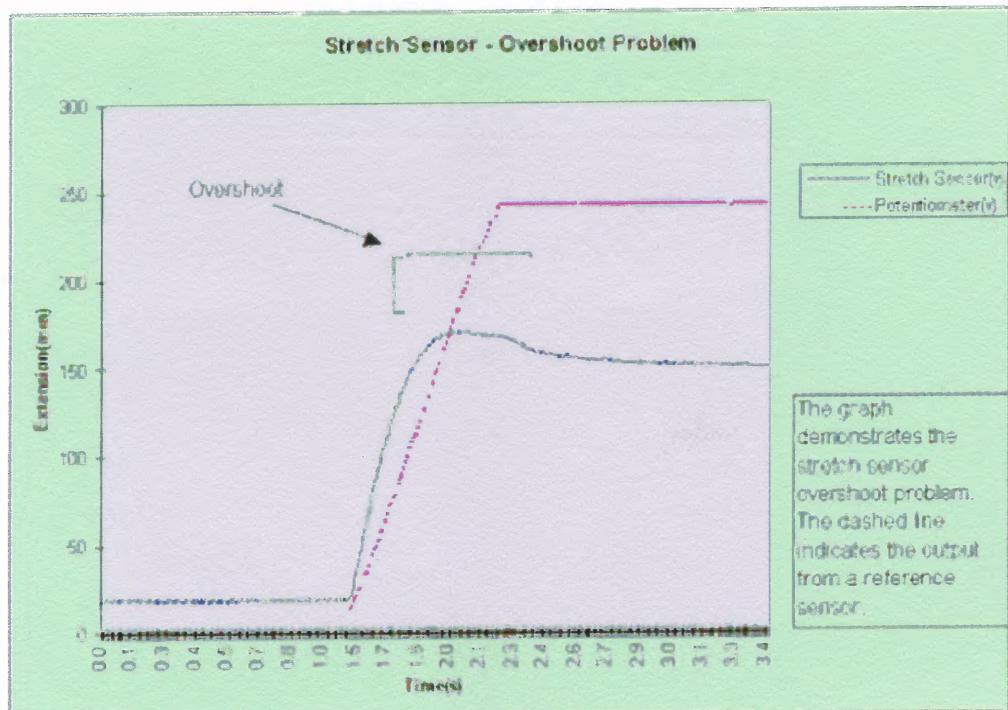
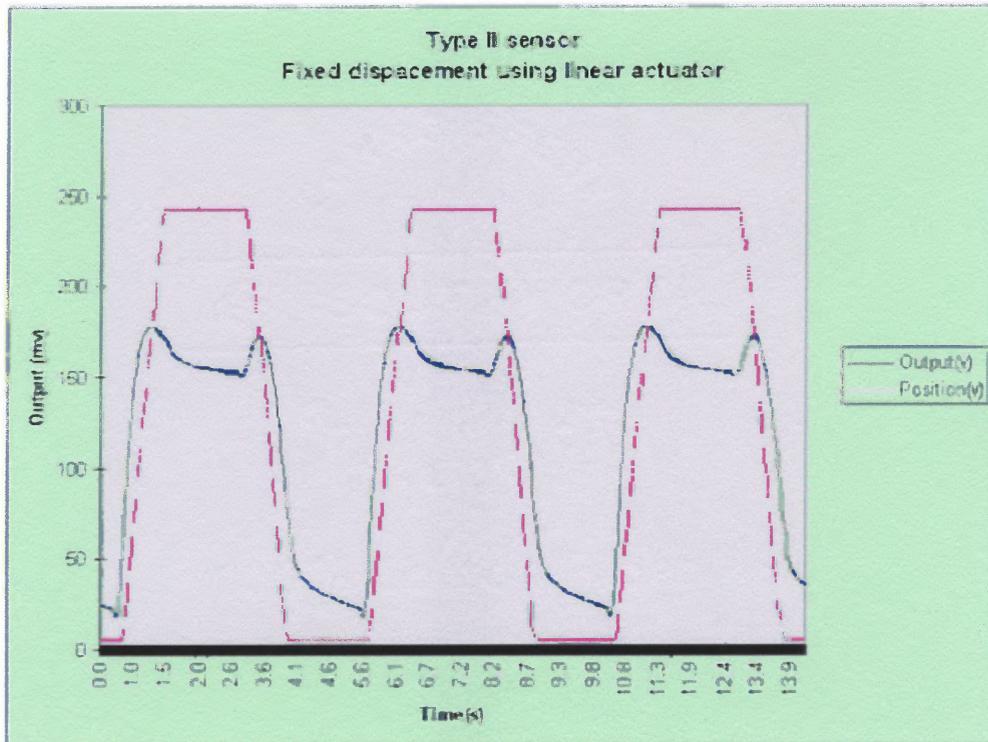


Figure 2.5 Merlin stretch sensor overshoot problem.



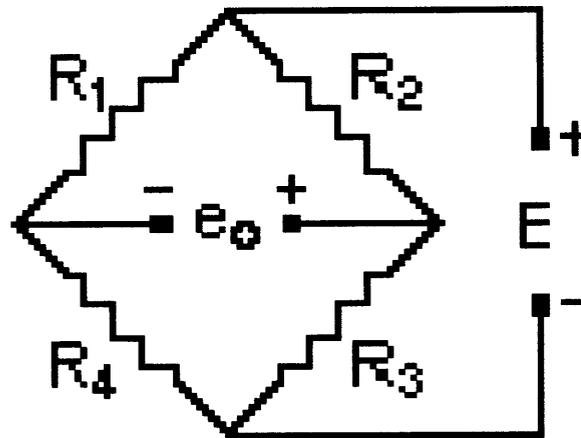
**Figure 2.6** Fixed displacement using linear actuator.

### 2.3 Wheatstone Bridge

The Wheatstone Bridge was invented by a British scientist named Samuel Hunter Christie. Ironically, the Wheatstone Bridge gets its name from Sir Charles Wheatstone, a British physicist and inventor who first applied it for measuring the resistance in an electric circuit [9]. The author incorporates a Wheatstone Bridge into the circuit to measure the variable resistance of the Merlin stretch sensors.

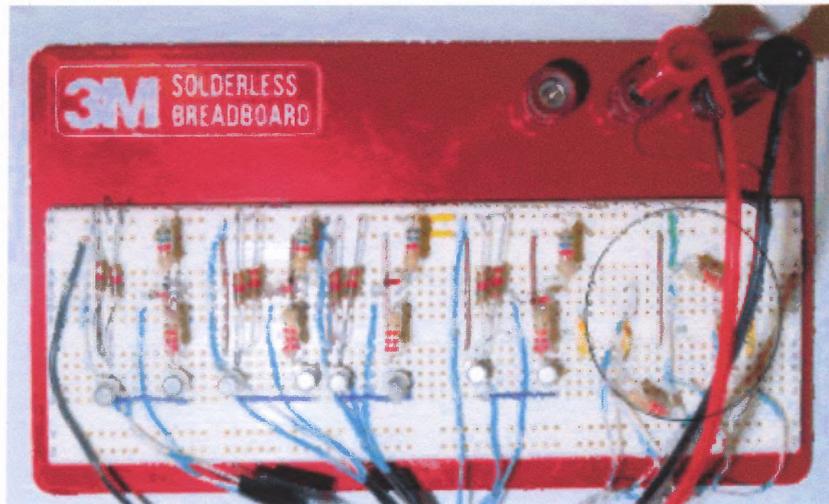
The full Wheatstone bridge circuit, in its simplest form, consists of four resistive elements, or bridge arms ( $R_1$ ,  $R_2$ ,  $R_3$ ,  $R_4$ ), connected in a series-parallel arrangement, and an excitation voltage source ( $E$ ). The electrical connections where pairs of bridge arms are joined to the lead wires from the excitation voltage source are referred to as

input corners of the bridge (Figure 2.7). A differential output voltage ( $e_o$ ) is measured at the two remaining bridge corners, referred to as output or signal corners [9].



**Figure 2.7** Basic Wheatstone Bridge circuit.

A true balanced Wheatstone Bridge can be shown if the arm resistances are chosen such that the bridge is resistively symmetrical about an imaginary line drawn through the bridge output corners. Then, the differential output voltage ( $e_o$ ) will be identically equal to zero, regardless of the value of the excitation supply voltage ( $E$ ). In the Wheatstone Bridge of the circuit, two 5.6 kohm resistors and two Merlin stretch sensors, were utilized to balance the bridge, indicated by R2, R4 and R1, R3, respectively (Figure 2.7). If the bridge were not in balance, a differential voltage will be present at the output corners of the bridge, and the magnitude of this output voltage will be proportional to the amount of unbalance [25].



**Figure 2.8** Wheatstone Bridge circuit on breadboard of model indicated by circle.

### Voltage Divider Formula

The bridge circuit diagram used for the joint model is depicted in Figure 2.9. The voltage on the right side of the voltmeter (marked + for the red or positive lead) with respect to ground can be determined using the voltage divider formula as:

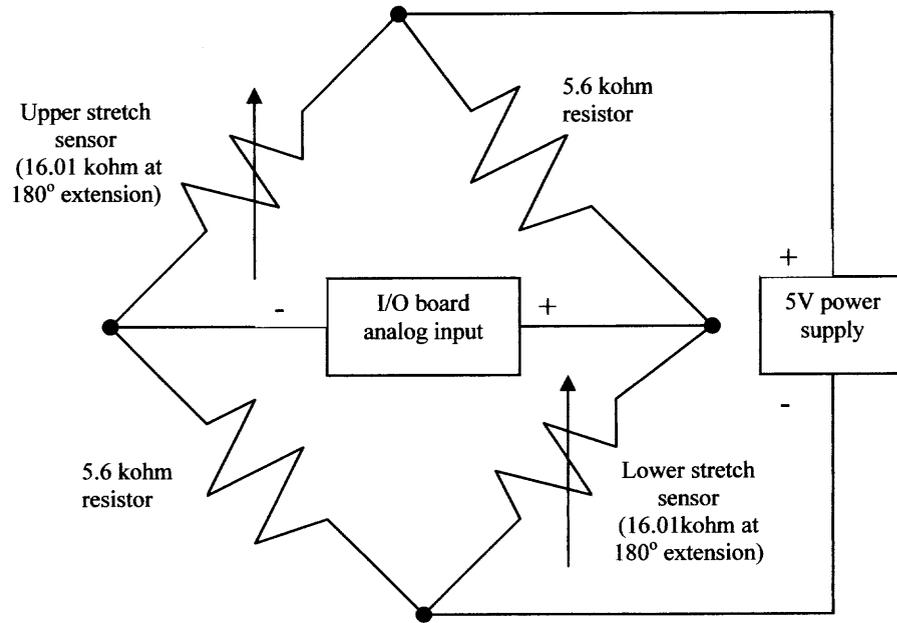
$$V_R = (R_2/(R_2+R_3))*E = (5.6K/16.01K)*5V = +1.749 \text{ volts}$$

The voltage on the left side of the voltmeter (marked - for the black, negative, or reference lead) with respect to ground can also be determined using the voltage divider formula as:

$$V_B = (R_4/(R_1+R_4))*E = (5.6K/16.01K)*5V = +1.749 \text{ volts}$$

The 16.01K value is representing the resistance value of the upper and lower stretch sensors at rest (i.e. at 180 degrees).  $V_{RB}$  is the voltage at point R with respect to

point B. Further,  $V_{RB} = V_R - V_B$ . Therefore,  $V_{RB} = 0$  volts. This indicates that the bridge is balanced [25].

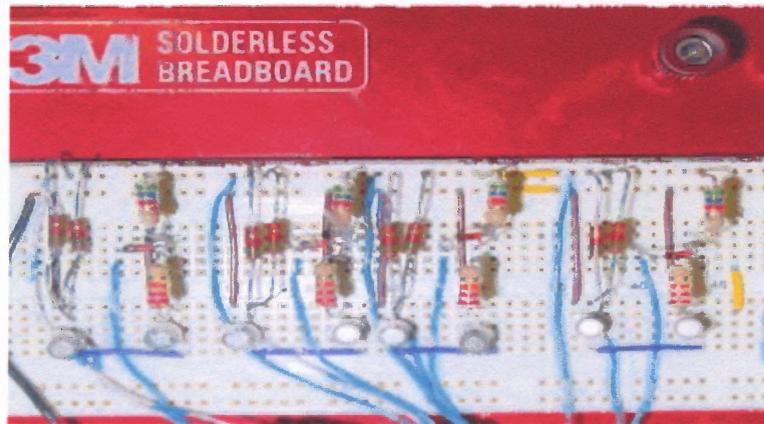


**Figure 2.9** Circuit diagram of Wheatstone Bridge used for model.

The complete Wheatstone Bridge is excited with a regulated 5V DC power supply. Through balancing, the voltage can be zeroed at the null point of measurement. As strain is applied to the Merlin stretch sensors, a resistive change takes place in both sensors and unbalances the Wheatstone Bridge. This results in a signal output, related to the strain value. This analog measurement is converted to a digital value for processing. The voltage derived from these resistances is converted into an angular position of the joint and is used in conjunction with the desired position to control operation of the pneumatic valves.

## 2.4 Transistor Circuit

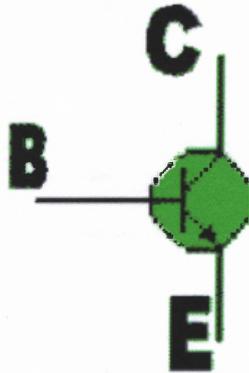
A digital input/output (I/O) block is used to transmit the signal to the four solenoid valves. In order to accomplish this, four digital lines must transmit a signal back to the computer to activate a high logic “one” state or a low logic “zero” state [18]. The digital value will be incorporated into the program for processing. The digital I/O circuit includes N-P-N transistors, various resistors and ½ Watt diodes to provide a means of switching current to the valves since the data acquisition card does not produce sufficient output current (Figure 2.10).



**Figure 2.10** Transistor power amplifier circuit board.

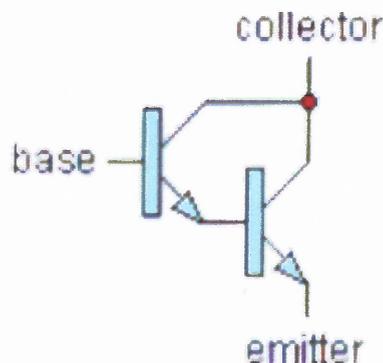
The job of a transistor is to allow the small amount of current that enters its 'base' terminal to control the amount of current flowing from its 'collector' terminal to its 'emitter' terminal (Figure 2.11). This allows a low current circuit to control a higher current circuit, either in an on/off fashion or linearly. Junction transistors consist of two junctions made from N-type and P-type semiconductor materials and are called bipolar

transistors (two polarities). Motorola P2N2222A silicon transistors were used to switch and amplify the current [22].



**Figure 2.11** Motorola P2N2222A transistor.

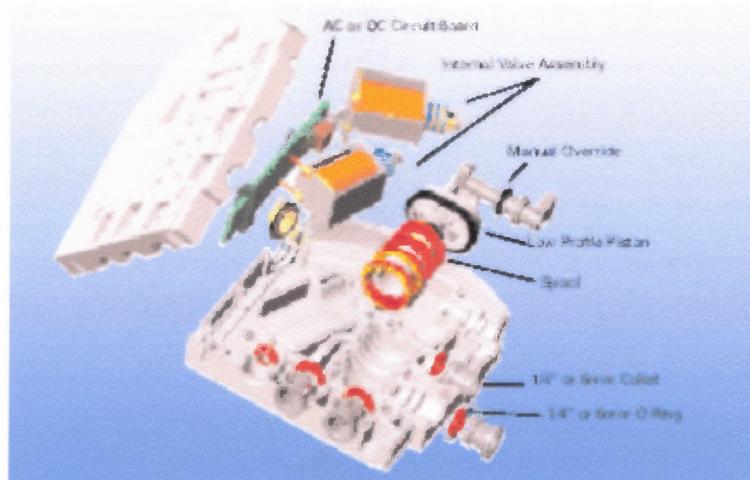
A single transistor arrangement in the circuit lacks the gain to drive the pneumatic valve. A Darlington Pair is needed to amplify the current and this is achieved by the first transistor's emitter feeding into the base of the second transistor (Figure 2.12). The current is amplified to an overall gain of 100. The total gain can be calculated by multiplying the gains of the two transistors. Similarly, the voltage drop across the transistor pair at the base doubles from 0.7V for one transistor to 1.4V for two transistors [10].



**Figure 2.12** Darlington Pair arrangement of transistors.

## 2.5 Mead Fluid Dynamics Valves

The solenoid valves used in the model contain an integrated electronics board with surge suppression and a LED. The half-shell design of the three-way valves allows flow channels and component compartments to be designed directly into the body. The body halves are joined by ultrasonic welding, creating a strong bond and hermetic seal. Quick connect collets allow easy joining of tubes and manifolds [11]. An exploded view diagram of the valve is provided (Figure 2.13).



**Figure 2.13** Mead Fluid Dynamics solenoid three-way valve.

The three-way design incorporates an exhaust, inlet and outlet port [11]. The valves have been modified to facilitate individual flow control patterns by implementing a two-way design. This was accomplished by plugging the exhaust port of each valve and using a pair of valves to control each air muscle. The flow control patterns relating to the control of air pressure into each air muscle can be seen in Table 2.2.

**Table 2.2** Flow Control Patterns of a Single Air Muscle

<i>Sequence</i>	<i>Valve 1</i>	<i>Valve 2</i>	<i>McKibben Air Muscle</i>
<i>1</i>	<i>on</i>	<i>on</i>	<i>air flows in up to max pressure, contracts</i>
<i>2</i>	<i>off</i>	<i>on</i>	<i>air flows out of Valve 1, holds contracted</i>
<i>3</i>	<i>off</i>	<i>off</i>	<i>air flows out of Valve 1 &amp; 2, releases</i>
<i>4</i>	<i>on</i>	<i>off</i>	<i>air flows into Valve 1, holds contracted</i>

The electrical characteristics of the solenoid valves play an important role in developing the transistor circuit [22]. The circuit was built to amplify the current supply needed to actuate the valves. Using a multimeter, the resistances of the valves were measured to be 19 ohms. The current draw required to turn on the valves was then calculated using Ohm's Law:

$$E = I \times R$$

$$I = 5V / 19\text{ohms}$$

$$I = 263\text{mA}$$

Therefore, the circuit was built so that the transistors were capable of switching the minimum 263mA rating. The transistor's maximum collector current ( $I_C$ ) rated 600mA as a result of the overall 100 gain rating achieved through the Darlington Pair arrangement. From this, calculations can be made for both the first and second of the transistor's base currents ( $I_{B1}$  &  $I_{B2}$ , respectively) to switch the 600mA maximum collector current using the following equations:

$$\text{First transistor Gain} = I_{C1} / I_{B1} = 10$$

$$I_{B1} = 60\text{mA} / 10$$

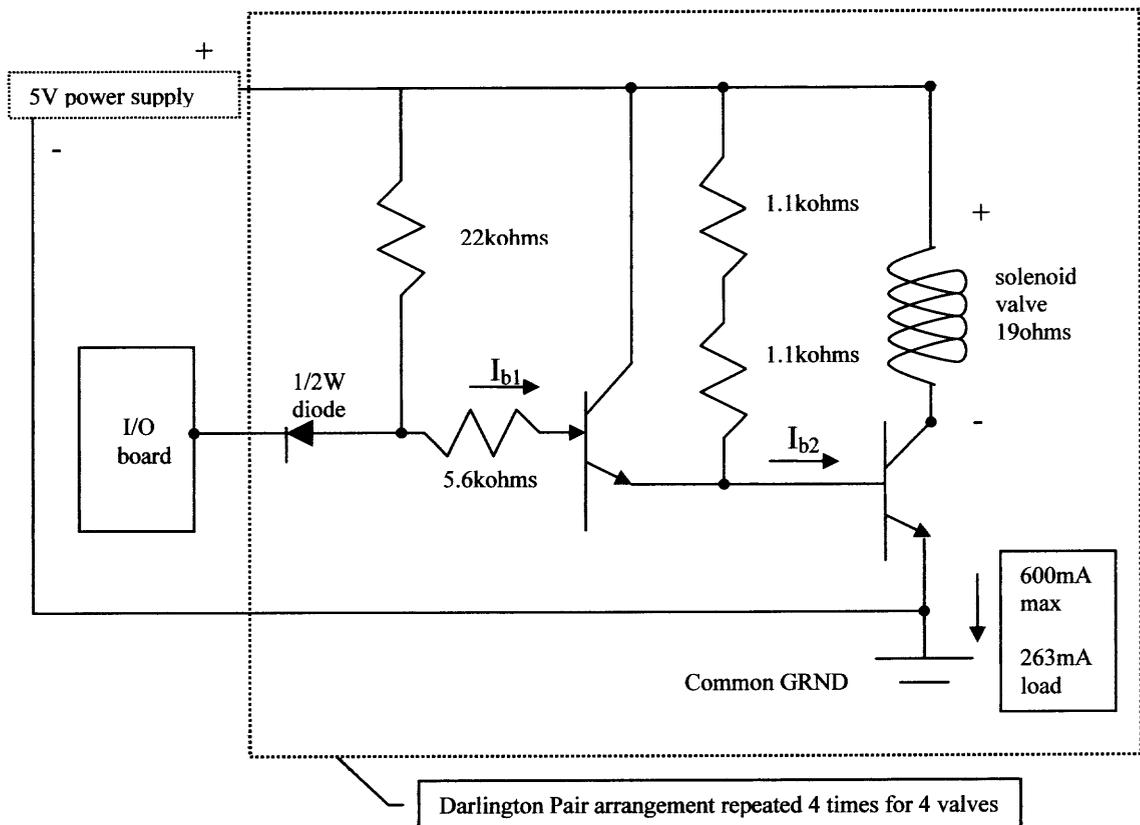
$$I_{B1} = 6\text{mA}$$

$$\text{Second transistor Gain} = I_{C2} / I_{B2} = 10$$

$$I_{B2} = 600\text{mA} / 10$$

$$I_{B2} = 60\text{mA}$$

The Darlington Pair circuit is then repeated four times for the activation of the four valves into the buffer circuit. Only one of four duplicate Darlington Pair arrangements is shown in Figure 2.14.



**Figure 2.14** Circuit diagram for buffer circuit.

## 2.6 Data Acquisition Card

The software used to control the model was written in National Instruments (NI) LabVIEW v.6.0. In order to read the stretch sensors and activate the pneumatic valves, a data acquisition system was needed for program control. The DAQCard E Series cards are multifunction analog, digital, and timing I/O cards for computers equipped with Type IIPCMCIA slots. The card that was selected features 12-bit ADCs with eight lines of TTL-compatible digital I/O, and two 24-bit counter/timers for timing I/O [15].

LabVIEW and LabWindows/CVI are program development software packages for data acquisition and control applications. LabVIEW uses graphical programming, whereas LabWindows/CVI enhances traditional programming languages. Both packages include extensive libraries for data acquisition, instrument control, data analysis, and graphical data presentation. LabVIEW features interactive graphics, a state-of-the-art user interface, and a powerful graphical programming language. The LabVIEW Data Acquisition VI Library, a series of VIs for using LabVIEW with National Instruments DAQ hardware, is included with LabVIEW. The LabVIEW Data Acquisition VI Library is functionally equivalent to the NI-DAQ software. The detailed specifications of the 6024E DAQCard used in the model can be found in APPENDIX B [15].

NI-DAQ has both high-level DAQ I/O functions for maximum ease of use and low-level DAQ I/O functions for maximum flexibility and performance. Examples of high-level functions are streaming data to disk or acquiring a certain number of data points. An example of a low-level function is writing directly to registers on the DAQ device. NI-DAQ does not sacrifice the performance of National Instruments DAQ devices because it allows multiple devices operate at their peak performance. NI-DAQ

also internally addresses many of the complex issues between the computer and the DAQ hardware such as programming interrupts. NI-DAQ maintains a consistent software interface among its different versions to decrease code modifications. A low-level diagram of the NI-DAQ software used in the model is illustrated in Figure 2.15 [15].

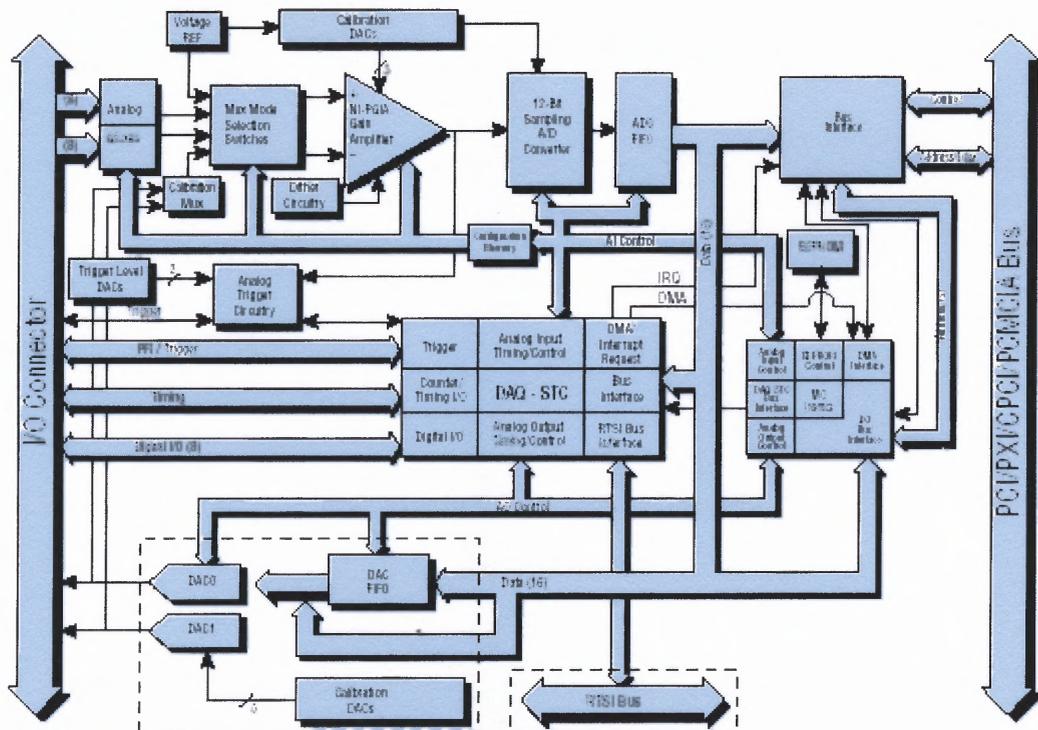


Figure 2.15 DAQCard-AI-16E-4 low-level block diagram.

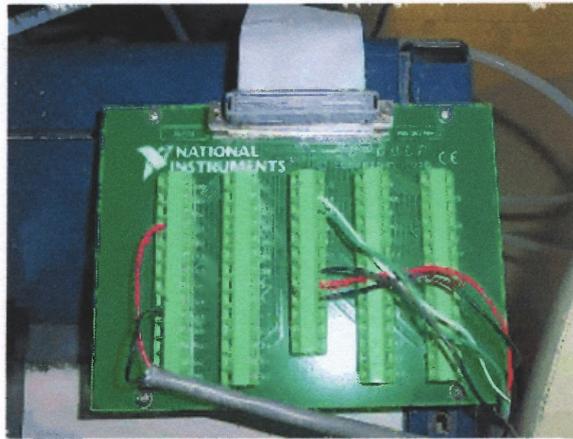
The DAQCard's analog inputs have three different input modes: non-referenced single-ended (NRSE) input, referenced single-ended (RSE) input, and differential (DIFF) input. The single-ended input configurations use up to 16 channels. The DIFF input configuration uses up to eight channels. Input modes are programmed on a per channel basis for multimode scanning [15]. Table 2.3 on the following page describes the three available input configurations.

**Table 2.3 NI-DAQCard Input Configurations**

Configuration	Description
DIFF	A channel configured in DIFF mode uses two analog channel input lines. One line connects to the positive input of the DAQCard programmable gain instrumentation amplifier (PGIA), and the other connects to the negative input of the PGIA.
RSE	A channel configured in RSE mode uses one analog channel input line, which connects to the positive input of the PGIA. The negative input of the PGIA is internally tied to analog input ground (AIGND).
NRSE	A channel configured in NRSE mode uses one analog channel input line, which connects to the positive input of the PGIA. The negative input of the PGIA connects to the analog input sense (AISENSE) input.

A differential connection is one in which the DAQCard computes the difference between two analog inputs. These connections are available when the selected channel is configured in DIFF input mode. The input signal is tied to the positive input of the programmable gain instrumentation amplifier (PGIA), and its reference signal, or return, is tied to the negative input of the PGIA. Each signal uses two multiplexer inputs; one for the signal and one for its reference signal.

The I/O connector board used for the DAQCard E Series cards has 68 pin connections (Figure 2.14). The arrangement and details of the I/O board pin connections can be found in Appendix C. It is important to note that exceeding the differential and common-mode input ranges distorts the input signals. Also, exceeding the maximum input voltage rating can damage the DAQCard E Series card and other connected hardware [15].



**Figure 2.16** CB-68LP I/O connector block.

## 2.7 Summary

In order for the circuit to switch a current, a voltage source, a ground and a load are required. NI-DAQCard's digital I/O lines have limited current sinking capability. Therefore, the buffer circuit that was incorporated into the system required the DAQCard to drive the base current of the first transistor to provide higher base current for the second transistor, and allow sufficient current to activate the valve.

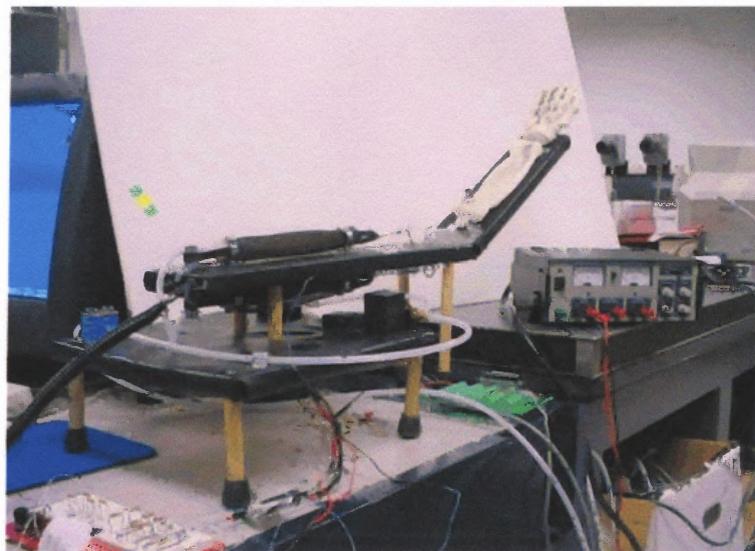
The solenoid valves provide linear motion when powered. The solenoid armature moves inward when the signal coming into the driver is high [10]. The solenoid is connected between the supply voltage and the transistor collector. This acts as load on the driver. When the input signal coming into the solenoid subsystem is low, a potential difference across the solenoid causes current to flow [2]. It is this flow of current (>263mA) that causes the solenoid armature to move. A spring in the solenoid will cause the armature to move back as soon as the current is turned off [10].

## CHAPTER 3

### METHODOLOGY AND RESULTS

#### 3.1 Problem Statement

Bang-Bang control and Proportional-Integral-Derivative control methods have been known to exhibit similarities in human musculoskeletal systems. These systems are very difficult to replicate because the components that comprise them are highly non-linear [27]. This dilemma was approached by constructing a single degree-of- freedom joint model (Figure 3.1). The single joint enables us to concentrate on fewer joint torques and forces that are abundant in biological systems. Although this approach may not be ideal for modeling such a system, the model was intended to examine force optimization techniques from a control engineering point of view for quick quantitative analysis.



**Figure 3.1** Biologically inspired single joint model.

The first approach was the ON/OFF (or Bang-Bang) controller, which is a closed-loop controller. A closed-loop uses a feedback signal to maintain the controlled process at (or as close as possible to) the desired value. This type of controller is operated by turning the actuator (valve) ON when the desired joint angle is less than the actual angle value, and OFF when it is greater. The actual angle is measured by the two Merlin stretch sensors while the desired angle is user selectable on the LabVIEW program's front panel.

Unfortunately, the Bang-Bang control method used in this experiment resulted in hysteresis. Some overshoot occurred due to the limitations of the valves and the control of the process variable. This is because the resulting position of the joint exceeded the desired position before the valve could be turned off. Likewise, undershoot occurred because of end point positions moving lower than the desired position.

Secondly, A PWM program is proposed to digitally encode the analog signal and hence, reduce overall instability. Although perturbations are expected to be reduced, the system should fail in reaching a critically damped state. A PWM program has been generated for incorporation into the existing Bang-Bang control program.

In addition, a PID control program is proposed to further alleviate the hysteresis. The system does this by responding to the magnitude of the error, the duration of the error, and the rate of change of the process variable (gas) or the error (position). The integral term will work to eliminate the offset. The derivative term will work on the rate of change of the process variable or the rate of change of the error to correct the process variable before it strays too far. This method should work quite well but complicates the mathematical analysis slightly because the system is now third order [14].

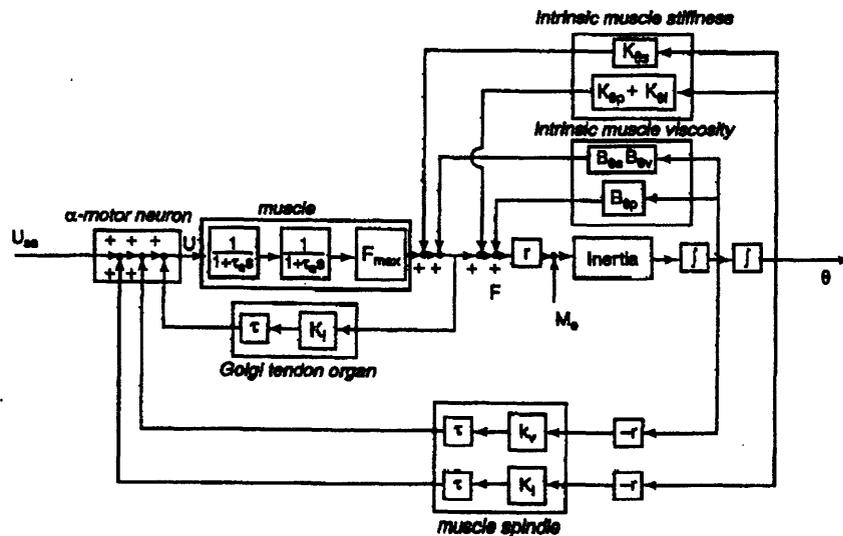
The physical model was constructed of a single joint with an agonist and an antagonist muscle operating under linear control. Utilizing LabVIEW v6.0 software, McKibben air muscles and Merlin stretch sensors, a Bang-Bang closed-loop controller was implemented to exhibit end point position control. The controller was found to be inadequate in maintaining stability in the system. In an effort to alleviate the perturbations, a proportional-integral-derivative (PID) controller and a program to include pulse width modulated (PWM) output are introduced. Through both system approaches and investigation into the musculoskeletal model, research can be done on the reflex loops under various levels of feedback sensitivity.

### **3.2 Musculoskeletal Model**

Much research work has been done in the development of muscle models [17]. However, many of these muscle models are open-loop models, transferring neural input into force. Stability of these models depends completely on the intrinsic viscoelastic properties of the muscles. Contrarily, experiments on reflexive muscle actuators and intact limbs revealed that the emerging viscoelastic behavior of muscles is the result of the proprioceptive feedback of muscle spindles and Golgi Tendon Organs. A closed-loop model incorporating intrinsic and proprioceptive feedback has been developed by Van der Helm and Rozendaal and is shown in Figure 3.2 [23].

The linear musculoskeletal closed-loop model represents a single degree-of-freedom (DOF) system and its proprioceptive feedback loops. The input ( $u_{ss}$ ) is the supraspinal neural input to the closed-loop actuator (a muscle inside its proprioceptive feedback loops). The moment arm  $r$  transfers muscle force into moment, and joint angle

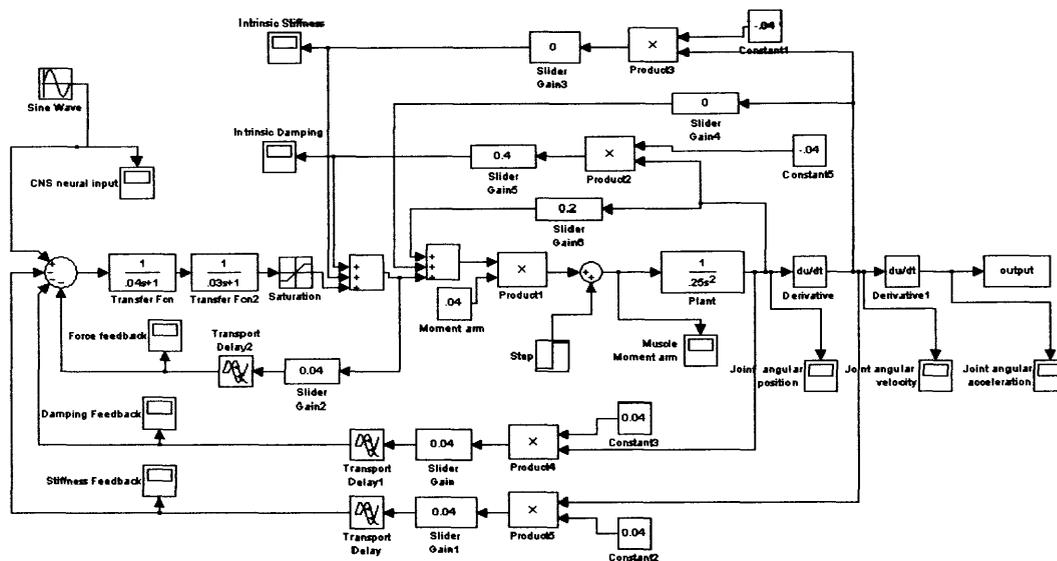
into muscle length. Neural signal transport and processing are represented by a time delay ( $\tau$ ). Joint angle and angular velocity are fed back through the intrinsic muscle properties (force-length and force-velocity relation) and through the muscle spindle, resulting in I-a and II-a muscle spindle afferents. The Golgi Tendon Organ is sensitive to muscle force. The output of the model is the joint angle of a limb [23].



**Figure 3.2** Musculoskeletal model with intrinsic and proprioceptive feedback.

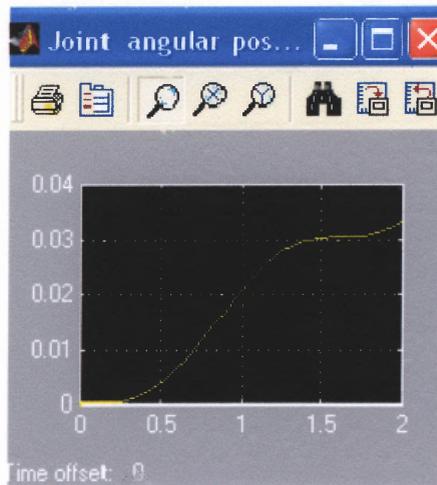
Mathematical simulations show that using only length feedback results in a nearly unstable system with a very small bandwidth (less than 0.5Hz.). Using length and velocity feedback stability limits results in a maximal length feedback which is not sufficient to obtain the reference stiffness [13]. Only the combination of length, velocity and force feedback results in a system that is an acceptable approximation of the desired impedance. Furthermore, the force feedback gain only depends on the muscle activation

dynamics and maximal isometric force. The ratio between velocity and length feedback gains is approximately 1 to 10 [28].

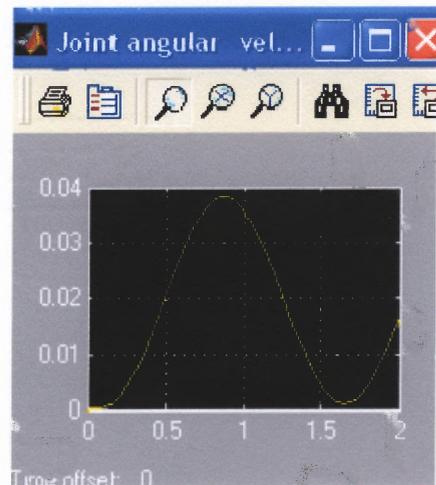


**Figure 3.3** Simulink musculoskeletal model with intrinsic and proprioceptive feedback.

Through the use of Matlab (Simulink) v.6.1 software, Van der Helm and Rozendaal's model are utilized to produce similar experimental findings (Figure 3.3). All gains were adjusted to achieve the most stable system possible. Hysteresis was minimized and joint angular position (Figure 3.4) and joint angular velocity (Figure 3.5) profiles resulted.



**Figure 3.4** Joint angular position of Simulink model.



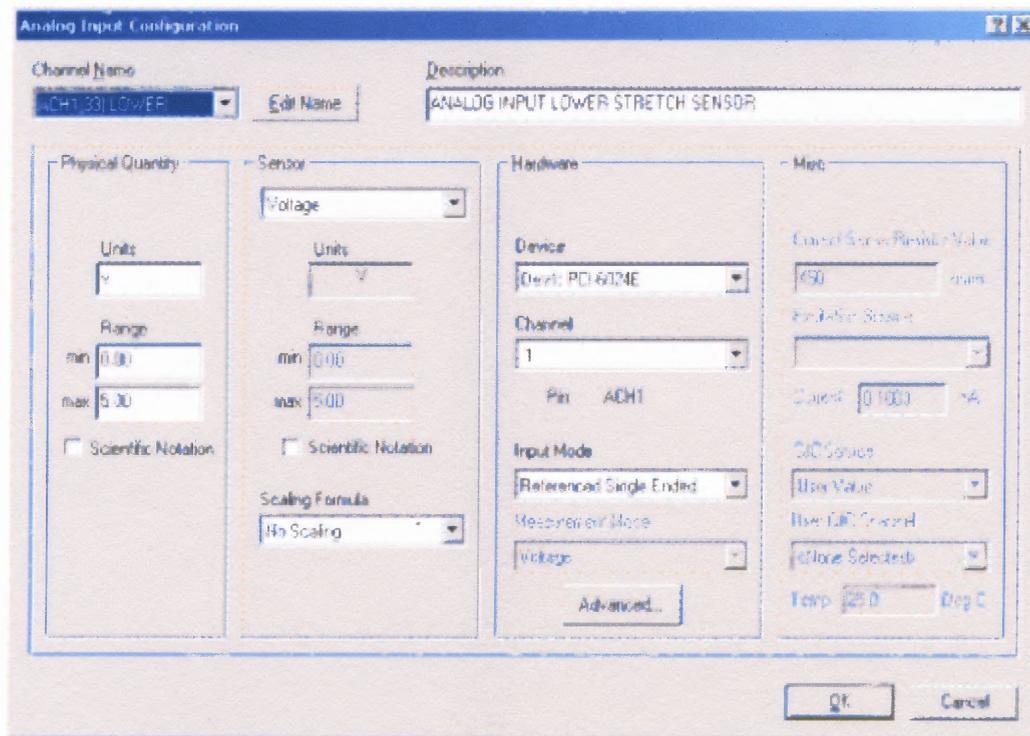
**Figure 3.5** Joint angular velocity of Simulink model.

Past research has shown that it is feasible to implement proprioceptive feedback loops and the accompanying gains in a musculoskeletal model [6]. The Simulink model demonstrates a more desirable joint angular position plot than the results (Figure 1.2) achieved with the Bang-Bang method of control for the mechanical joint. The amount of hysteresis is minimized in Figure 3.4. This suggests that the Bang-Bang method of control is unable to anticipate the desired location of the moving joint and another form of control is needed to stabilize the system.

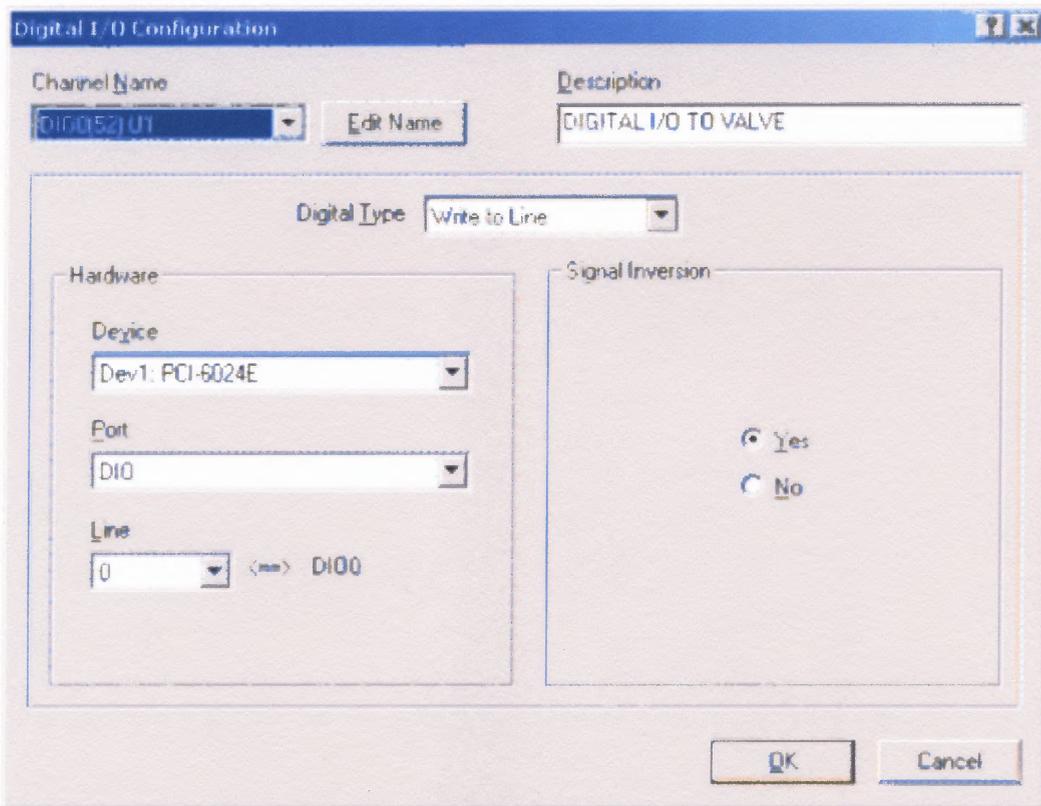
### 3.3 Bang-Bang Control

The Bang-Bang method used to control the joint model was accomplished by using NI LabVIEW v.6.0 software (Appendix F). In order for the system to respond appropriately, configuration of the analog inputs were required. This was accomplished through the use of NI Measurement and Automation Explorer (MAX) v.2.0 tool that is

built in to NI-DAQ driver software. The configuration process allowed us to set the parameters of the analog I/O and digital I/O channels. The configuration setup also allowed testing of system diagnostics. A panel view of the system setup for the analog inputs of the stretch sensors is shown in Figure 3.6 and corresponding digital outputs of the solenoid valves in Figure 3.7 [15].

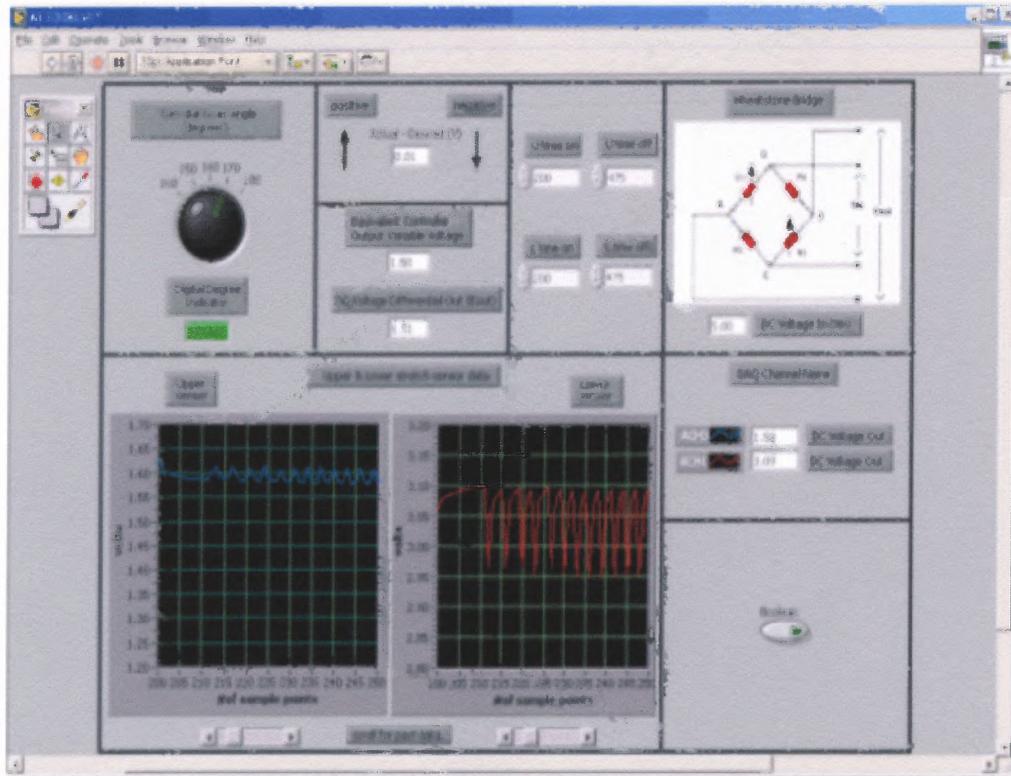


**Figure 3.6** NI MAX analog input configuration.



**Figure 3.7** NI MAX digital output configuration.

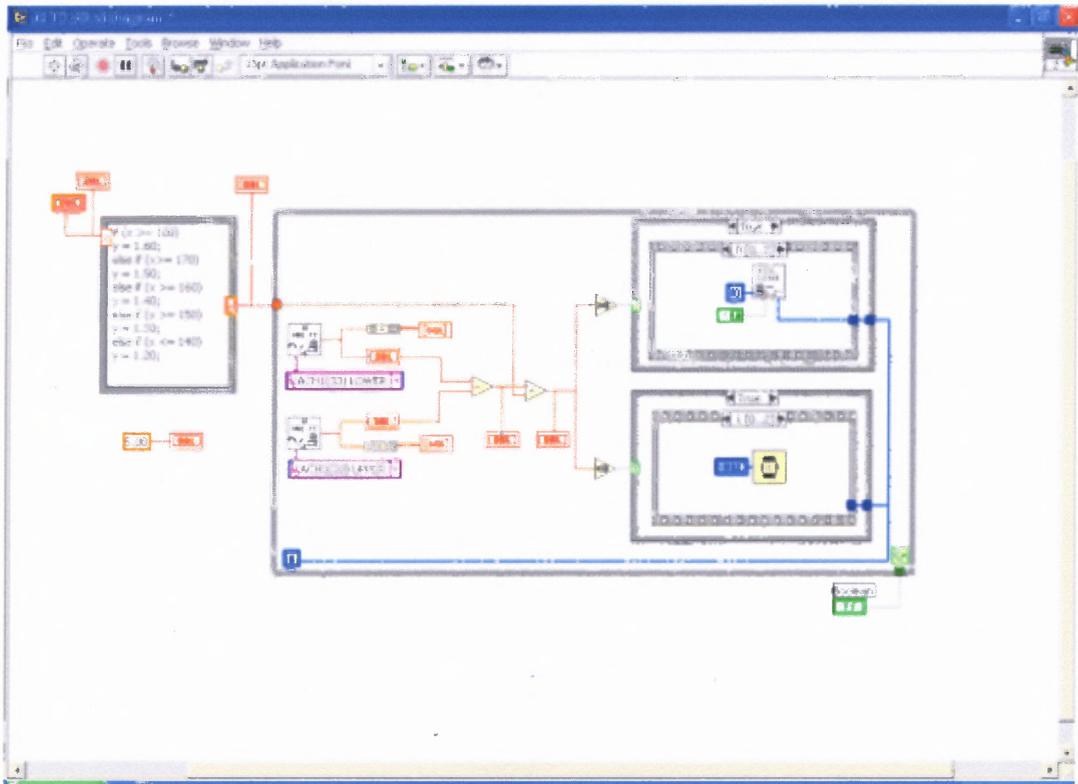
The results of the program show that for Bang-Bang control, the system is underdamped. This is an undesirable form of control for the single joint model. Upon entering an end point position located within the  $140^{\circ}$ - $180^{\circ}$  range, at 10 degree increments, the joint moved through the desired location and then back past this point in the opposite direction causing it to oscillate. The amplitude of the stretch sensors was captured in the program for analytical interpretation (Figure 3.8). The output plot of the Bang-Bang control program demonstrates the hysteresis achieved using this control method which resulted in perturbations around the desired position (Figure 1.1). However, the program VI brought forth some of the constraints involved in utilizing a Bang-Bang control methodology to approach a stable system in the physical model (Figure 3.9).



**Figure 3.8** Front panel of NI LabVIEW Bang-Bang control program.

The limitations of the potentiometers used decreased the reliability of the feedback signals. The continual decay of the Merlin stretch sensors to a rest position resulted in premature program results [12]. Other limitations to consider involve the overshoot and the release artifact problem mentioned earlier. There is also an error attributed to the physical measurements taken of the stretch sensors differential output voltage at arbitrary 10 degree increments needed for program triggering. The differential output voltage values obtained from the stretch sensors were compared to the values in the formula node, obtained through preliminary testing. The differential output voltages of the stretch sensors were measured at 10 degree increments to obtain a value for the computer program to compare the actual versus the desired voltage. The calibration table

created for implementing the differential voltage into the program can be found in Appendix D.



**Figure 3.9** VI of NI LabVIEW Bang-Bang control program.

The constant air pressure (50psig) into the air muscles posed yet another problem. The full duty cycle greatly attributed to the instability of the joint. Limitation of the modulating signal output is needed so that the pulses can be better controlled. This would ultimately result in a reduction of the amount of air coming into the air muscles, which would significantly reduce the hysteresis.

### 3.4 Pulse Width Modulation

Pulse Width Modulation (PWM) is a way of digitally encoding analog signal levels. Through the use of high-resolution counters, the duty-cycle of a square wave is modulated to encode a specific analog signal level. The PWM signal is still digital because, at any given instant of time, the full DC supply is either fully “ON” or fully “OFF” [16]. The voltage or current source is supplied to the analog load by means of a repeating series of on and off pulses. The on-time is the time during which the DC supply is applied to the load, and the off-time is the period during which that supply is switched off. As long as the bandwidth is sufficient, any analog value can be encoded with PWM [18].

The joint model lacks a pulse width modulated output. However, a versatile PWM program is proposed to reduce power consumption and hence, reduce overall instability. The goal will be to limit the amount of current supply to the valves by varying the duty cycle to 50%. In doing this, the air muscles will incur a reduction in overshoot and undershoot. Although perturbations are reduced, the system is not expected to reach a critically damped state. A PWM program has been generated for easy implementation into the VI of the existing NI Bang-Bang control method (Appendix F).

The NI-DAQ driver has a special feature that allows reprogramming of the counters "on-the-fly" during continuous pulse train generation. At the low level, updating of the count registers implies that the next pulse generated after the update will have the new parameters. This means that the number of pulses per second and the pulse-width can be changed while the pulse train generation is in progress [16].

This example program demonstrates this capability. It can be used with E-Series or 660x devices. The accuracy of the pulse train frequency and duty cycle is limited only by the resolution of the internal time base. Depending on the time base selected, only certain frequencies will be produced [16].

### 3.5 PID Control

PID stands for Proportional, Integral, and Derivative. A PID feedback controller is designed to generate an output that causes some corrective effort to be applied to a process so as to drive a measurable process variable towards a desired value known as the set-point [24]. The controller uses an actuator to affect the process and a sensor to measure the results. For the joint model, a NI program of a PID controller is proposed to minimize the perturbations (Appendix G).

Most feedback controllers determine their output by observing the error between the current and desired locations and a measurement of the process variable. Errors occur when the set-point is changed or when a process load affects the process variable. PID controllers are identified by the following equations:

$$P = \text{Proportional Band} = 100 / \text{gain}$$

$$I = \text{Integral} = 1 / \text{reset (units of time)}$$

$$D = \text{Derivative} = \text{rate} = \text{pre-act (units of time)}$$

The PID controller provides quick acting corrective control of most process variables. Adding integral control to a proportional controller will eliminate the steady state error, but will increase overshoot and settling time, but by adding derivative control, the overshoot and settling time can be reduced [24].

The proposed Real-Time PID Control VI is designed to be connected to the external system. Once the device number and rate are established, the connection of the analog input signals must be maintained to the stretch sensors. The PID output will control the digital output channels to the valves. A set-point value can then be selected to adjust the system. The Tuning Parameters can also be used to adjust the PID output to control the system [14].

This is a hardware timed control system which means that counter/timers on the DAQ device control the rate of acquisition and generation. The acquisition and generation are also controlled with the same clock, so the PID output occurs at the same time the Process Variable is acquired [5]. While hardware controls the timing of analog-to-digital and digital-to-analog conversions, this data must be written to and read from software buffers. This part of the control is software dependent. The “Keeping Real-Time?” Boolean indicates that the software is able to keep up with the hardware rate. The help window description on each control is available for more information about its use [14].

The PID controller must be tuned to incorporate gravity and inertia of the mechanical joint. Theoretically the program should reduce fluctuations in the system but, accurate tuning of the P, I, and D modes are essential. The PID program will eventually replace the Bang-Bang method of control. The end result will be to generate a controller output that steadily drives the process variable in the direction required to eliminate the error.

Implementation of the PID, or even a PD controller is difficult with an air muscle. The activation of an air muscle does not require current, but uses air volume to vary muscle force. Air volume, if not controlling pressure, is then related to time.

### 3.6 Summary

Musculoskeletal closed-loop models were analyzed to represent similar systems as constructed in the single DOF joint model. Matlab software was utilized to incorporate these musculoskeletal models for a quantitative approach. Results show that by adjusting the feedback gains, there is an increase in stability. Then, this same approach was applied to an ON/OFF (or Bang-Bang) closed-loop controller. Some overshoot occurred due to the limitations of the valves and the control of the process variable. Likewise, undershoot occurred because of end point positions moving lower than the desired position (the lower trip point). This resulted in a feedback value incapable of maintaining the controlled process at the desired value.

The concept of PWM was introduced to alter the digital output signal in an attempt to better control the system. Theoretical calculations conclude that at a 50% duty cycle, the reduction of air coming into the air muscles greatly enhances stability [16]. Furthermore, the implementation of a PID control program is suggested as a more appropriate means of controlling the single DOF joint because of the tuning functionality in the process parameters.

## CHAPTER 4

### CONCLUSIONS

#### 4.1 Discussion

In the biologically inspired joint model, the author demonstrated control by the Bang-Bang method. Results conclude that this method of control for achieving single plane movements is not optimum due to the resulting perturbations present. In an effort to reduce the hysteresis, NI programs of a Pulse Width Modulated output and Proportional-Integral-Derivative controller are proposed. Reproducing non-linear biological systems for quantitative analysis encompasses detailed knowledge of the control mechanisms in biological systems as well as an understanding of control engineering methodology.

A look into musculoskeletal models provided insight into how human single degree-of-freedom joint systems operate. Demonstration of an open-loop muscle model working in conjunction with an inertial system would be extremely difficult but, similarities of how the control process of a muscle works can be demonstrated through the closed-loop program [23]. This was accomplished by making all parameters linear to simulate a musculoskeletal model that incorporates proprioceptive feedback. Matlab software was also utilized to show how the damping and stiffness coefficients, as in human muscle spindle and Golgi Tendon Organ feedback loops, can be adjusted to optimize stability in single DOF joints moving in sagittal plane trajectories.

Controlling movement of a joint requires co-activation of agonist and antagonist muscles. Essentially, a force needs to be applied to one muscle to get the joint moving, and activation of the other muscle to act as a brake. This in turn, slows the joint, letting it

achieve the desired target angle. This form of control was unattainable by the Bang-Bang control method.

Limitations of the Bang-Bang control method brought forth investigation into other possible methods of control. The concept of Pulse Width Modulation is introduced as a powerful technique for controlling the digital outputs. Furthermore, a Proportional-Integral-Derivative control method is introduced as a future means of increasing stability.

## **4.2 Future Directions**

To date, very little is known about proprioceptive feedback and the dynamics of musculoskeletal systems. It has been suggested that much research should be devoted into this area which will show the importance of stability for muscle co-ordination. As an approach to optimization of feedback gains in biological systems, perturbations of robot manipulator systems can provide further insight into the research of muscle spindles and Golgi Tendon Organs [23].

## APPENDIX A

### STATIC PHYSICAL MODEL OF MCKIBBEN MUSCLES

This Appendix describes a theoretical approach, based on energy conservation, to find the tension as a function of pressure and actuator length without considering the detailed geometric structure.

The input work ( $W_{in}$ ) is done in the McKibben muscle when gas pushes the inner bladder surface. This is

$$dW_{in} = S_i \int (P - P_o) d_{li} \cdot d_{si} = (P - P_o) S_i \int d_{li} \cdot d_{si} = P' dV \quad A.1$$

where  $P$  is the absolute internal gas pressure,  $P_o$ , the environment pressure (1 atm = 1.0336 bar),  $P'$ , the relative pressure,  $S_i$ , the total inner surface,  $d_{si}$ , the area vector,  $d_{li}$ , the inner surface displacement, and  $dV$ , the volume change. The output work ( $W_{out}$ ) is done when the actuator shortens associated with the volume change, which is

$$dW_{out} = -FdL \quad A.2$$

where  $F$  is the axial tension, and  $dL$ , the axial displacement. From the view of energy conservation, the input work should equal the output work if a system is lossless and without energy storage. Assume the actuator is in this ideal condition. Then, the “virtual work” argument becomes:

$$dW_{out} = dW_{in} , \quad A.3$$

thus, from equ. 1 and equ. 2,

$$-FdL = P'dV , \quad A.4(a)$$

$$F = -P'dV/dL . \quad A.4(b)$$

To estimate  $dV/dL$ , assume the extensibility of the shell threads is very low (as it has to be chosen), so the actuator volume will only depends on its length. In addition, the

middle portion of the actuator is modeled as a perfect cylinder with zero-wall-thickness, where  $L$  is the length of the cylinder,  $\theta$ , the angle between a braided thread and the cylinder long axis,  $D$ , the diameter of the cylinder,  $n$ , number of turns of a thread, and  $b$ , the thread length.  $L$  and  $D$  can be expressed as functions of  $\theta$  with constant parameters  $n$  and  $b$ ,

$$L = b \cos \theta , \quad \text{A.5}$$

$$D = b \sin \theta / \pi n \quad \text{A.6}$$

The volume of the cylinder is

$$V = 1/4\pi D^2 L = (b^3 / 4\pi n^2) \sin^2 \theta \cos \theta \quad \text{A.7}$$

So, from equ. 4b,  $F$  can be expressed as a function of and  $P'$  and  $\theta$ ,

$$F = -P' \frac{dV}{dL} = -P' \frac{dV/d\theta}{dL/d\theta} = \frac{P'b^2(2 \cos^2 \theta - \sin^2 \theta)}{4\pi n^2} = \frac{P'b^2(3 \cos^2 \theta - 1)}{4\pi n^2} , \quad \text{A.8(a)}$$

which is equivalent to

$$F = \pi D_o^2 P' / 4 (\cos^2 \theta - 1) \quad \text{A.8(b)}$$

where  $D_o$ , is the diameter when  $\theta$  equals  $90^\circ$ . The tension is thus linearly proportional to the pressure, and is a monotonic function of the braid angle ( $0^\circ < \theta < 90^\circ$ ). The maximal shortening will be reached when  $F = 0$ , that is,  $\theta = 54.7^\circ$ .

Notice that although the tension was obtained with the assumption of an ideal cylinder here, the tension can always be derived by knowing the arbitrary shape of the actuator with equation 4b without the assumption [18].

## APPENDIX B

### SPECIFICATIONS OF NI DAQCARD-AI-16E-4

This Appendix lists the specifications of DAQCard E series devices. These specifications are typical at 25°C unless otherwise noted.

## DAQCard-AI-16E-4

---

### Analog Input

#### Input Characteristics

Number of channels .....	16 single-ended, 16 pseudo-differential, or 8 differential (software-selectable on a per channel basis)
Type of ADC .....	Successive approximation
Resolution .....	12 bits, 1 in 4,096
Max sampling rate .....	250 kS/s guaranteed

## Input signal ranges

Board Gain (Software Selectable)	Board Range (Software Selectable)	
	Bipolar	Unipolar
0.5	±10 V	—
1	±5 V	0 to 10 V
2	±2.5 V	0 to 5 V
5	±1	0 to 2 V
10	±500 mV	0 to 1 V
20	±250 mV	0 to 500 mV
50	±100 mV	0 to 200 mV
100	±50 mV	0 to 100 mV

Input coupling ..... DC

Max working voltage  
(signal + common mode) ..... Each input should remain within  
±11 V of ground

Overvoltage protection ..... ±25 V powered on, ±15 V  
powered off

Inputs protected ..... ACH<0..15>, AISENSE

FIFO buffer size ..... 1,024 samples

Data transfers ..... interrupt, programmed I/O

Configuration memory size ..... 512 words

## Transfer Characteristics

Relative accuracy .....	$\pm 0.5$ LSB typ dithered, $\pm 1.5$ LSB max undithered
DNL .....	$\pm 0.5$ LSB typ, $\pm 1.0$ LSB max
No missing codes .....	12 bits, guaranteed
<b>Offset error</b>	
Pregain error after calibration .....	$\pm 16$ $\mu$ V max
Pregain error before calibration .....	$\pm 4.0$ mV max
Postgain error after calibration .....	$\pm 1.0$ mV max
Postgain error before calibration .....	$\pm 265$ mV max
<b>Gain error (relative to calibration reference)</b>	
After calibration (gain = 1) .....	$\pm 0.02\%$ of reading max
Before calibration .....	$\pm 2.5\%$ of reading max
Gain $\neq 1$ with gain error adjusted to 0 at gain = 1 .....	$\pm 0.02\%$ of reading max

## Amplifier Characteristics

<b>Input impedance</b>	
Normal powered on .....	100 G $\Omega$ in parallel with 100 pF
Powered off .....	1 k $\Omega$ min
Overload .....	1 k $\Omega$ min
Input bias current .....	$\pm 200$ pA
Input offset current .....	$\pm 100$ pA
<b>CMRR (all input ranges, DC to 60 Hz)</b>	
Gain $\leq 1$ .....	85 dB
Gain = 2 .....	95 dB
Gain $\geq 5$ .....	100 dB

## Dynamic Characteristics

<b>Bandwidth</b>	
Small signal (-3 dB) .....	800 kHz
Large signal (1% THD) .....	400 kHz

## Settling time for full-scale step

Gain	Accuracy	
	$\pm 0.012\%$ ( $\pm 0.5$ LSB)	$\pm 0.024\%$ ( $\pm 1$ LSB)
0.5	4 $\mu$ s typ, 8 $\mu$ s max	4 $\mu$ s max

## System noise in LSB rms, not including quantization

Gain	Noise, dither off	Noise, dither on
0.5 to 10	0.2	—
0.5 to 20	—	0.5
20	0.25	—
50	0.5	0.7
100	0.9	1.0

Crosstalk ..... -80 dB, DC to 100 kHz

**Stability**

## Offset temperature coefficient

Pregain .....  $\pm 5 \mu\text{V}/^\circ\text{C}$ Postgain .....  $\pm 240 \mu\text{V}/^\circ\text{C}$ Gain temperature coefficient .....  $\pm 20 \text{ ppm}/^\circ\text{C}$

## Digital I/O

Number of channels ..... 8 input/output

Compatibility ..... TTL/CMOS

### Digital logic levels

Level	Min	Max
Input low voltage	0 V	0.8 V
Input high voltage	2 V	5 V
Input low current ( $V_{in} = 0$ V)	—	-320 $\mu$ A
Input high current ( $V_{in} = 5$ V)	—	10 $\mu$ A
Output low voltage ( $I_{OL} = 24$ mA)	—	0.4 V
Output high voltage ( $I_{OH} = 13$ mA)	4.35 V	—

Power-on state ..... Input (High-Z)

Data transfers ..... Programmed I/O

## Timing I/O

Number of channels ..... 2 up/down counter/timers,  
1 frequency scaler

### Resolution

Counter/timers ..... 24 bits

Frequency scalars ..... 4 bits

Compatibility ..... TTL/CMOS

### Base clocks available

Counter/timers ..... 20 MHz, 100 kHz

Frequency scalars ..... 10 MHz, 100 kHz

Base clock accuracy .....  $\pm 0.01\%$

Max source frequency ..... 20 MHz

Min source pulse duration ..... 10 ns in edge-detection mode

Min gate pulse duration .....10 ns in edge-detection mode

Data transfers .....interrupts, programmed I/O

## Triggers

### Analog Trigger

Source .....ACH<0..15>, external trigger  
(PF10/TRIG1)

Level .....± full-scale, internal;  
±10 V, external

Slope .....Positive or negative  
(software selectable)

Resolution .....8 bits, 1 in 256

Hysteresis .....Programmable

Bandwidth (-3 dB) .....2.0 MHz internal,  
3.0 MHz external

#### External input (PF10/TRIG1)

Impedance .....10 kΩ

Coupling .....DC

Protection .....±35 V powered off,  
-0.5 to VCC when configured as  
a digital signal,  
±35 V when configured as an  
analog trigger signal or disabled

### Digital Trigger

Compatibility .....TTL

Response .....Rising or falling edge

Pulse width .....10 ns min

## Calibration

Recommended warm-up time .....15 min

Calibration interval .....1 year

External calibration reference .....	>6 and <9.999 V
Onboard calibration reference	
Level .....	5.000 V ( $\pm 2.5$ mV) (actual value stored in EEPROM)
Temperature coefficient .....	$\pm 5$ ppm/ $^{\circ}$ C max
Long-term stability .....	$\pm 15$ ppm/ $\sqrt{1.000}$ h

## Power Requirement (from PCMCIA I/O channel)

+5 VDC ( $\pm 5\%$ ) .....	280 mA typ in operational mode, 400 mA max in operational mode, 70 mA in power down mode
----------------------------	--

Power available at I/O connector ..... +4.65 to +5.25 V at 250 mA



Note

*These power usage figures do not include the power used by external devices that are connected to the fused supply present on the I/O connector.*

*Note also that under ordinary operation, the DAQCard has a current requirement of 270–290 mA; but if the analog inputs being sampled are overdriven at high gains, or if the analog inputs are left floating when the DAQCard is not in use, the current may increase to 400 mA.*

*You can save current by using the NI-DAQ power down utility when your DAQCard is not in use.*

## Physical

PC Card type .....	Type II
I/O connector .....	PCMCIA 68-position female connector

## Environment

Operating temperature .....	0 $^{\circ}$ to 55 $^{\circ}$ C
Storage temperature .....	-55 $^{\circ}$ to 150 $^{\circ}$ C
Relative humidity .....	5% to 90% noncondensing

## Programmable Function Inputs

The 10 PFIs are connected to the signal routing multiplexer for each timing signal, and software can select one of the PFIs as the external source for a given timing signal. It is important to note that any of the PFIs can be used as an input by any of the timing signals and that multiple timing signals can use the same PFI simultaneously. This flexible routing scheme reduces the need to change physical connections to the I/O connector for different applications.

You can also individually enable each of the PFI pins to output a specific internal timing signal. For example, if you need the UPDATE\* signal as an output on the I/O connector, software can turn on the output driver for the PF15/UPDATE\* pin.

## DAQCard Clocks

Many functions performed by the DAQCard E Series cards require a frequency timebase to generate the necessary timing signals for controlling A/D conversions, DAC updates, or general-purpose signals at the I/O connector.

A DAQCard can directly use its internal 20 MHz timebase as the primary frequency source.

## APPENDIX C

### I/O CONNECTOR BOARD FOR DAQCARD-AI-16E-4

This appendix describes the I/O pin assignment for the NI-DAQCard-AI-16E-4 connector board.

ACH8	34	68	ACH10
ACH1	33	67	AIGND
AIGND	32	66	ACH9
ACH10	31	65	ACH12
ACH3	30	64	AIGND
AIGND	29	63	ACH11
ACH4	28	62	AISENSE
AIGND	27	61	ACH12
ACH13	26	60	ACH5
ACH6	25	59	AIGND
AIGND	24	58	ACH14
ACH15	23	57	ACH17
Reserved	22	56	AIGND
Reserved	21	55	Reserved
Reserved	20	54	Reserved
DIO4	19	53	DGND
DGND	18	52	DIO0
DIO1	17	51	DIO5
DIO6	16	50	DGND
DGND	15	49	DIO2
-5 V	14	48	DIO7
DGND	13	47	DIO3
DGND	12	46	SCANCLK
PF 0/TRIG1	11	45	EXTSTROBE
PF 1/TRIG2	10	44	DGND
DGND	9	43	PF 2/CONVERT
+5 V	8	42	PF 3/GPCTR1_SOURCE
DGND	7	41	PF 4/GPCTR1_GATE
PF 5/UPDATE	6	40	GPCTR1_OUT
PF 6/WETR G	5	39	DGND
DGND	4	38	PF 7/STARTSCAN
PF 9/GPCTR0_GATE	3	37	PF 8/GPCTR0_SOURCE
GPCTR0_OUT	2	36	DGND
FREQ_OUT	1	35	DGND

Table C.1 I/O Connector Signal Descriptions

Signal Name	Reference	Direction	Description
AIGND	—	—	Analog Input Ground—These pins are the reference point for single-ended measurements and the bias current return point for differential measurements. All three ground references—AIGND, AOGND, and DGND—are connected together on your DAQCard E Series card.
ACH<0..15>	AIGND	Input	Analog Input Channels 0 through 15—Each channel pair, ACH< <i>i</i> , <i>i</i> +8> ( <i>i</i> = 0..7), can be configured as either one differential input or two single-ended inputs.
AISENSE	AIGND	Input	Analog Input Sense—This pin serves as the reference node for any of channels ACH<0..15> in NRSE configuration.
DGND	—	—	Digital Ground—This pin supplies the reference for the digital signals at the I/O connector as well as the +5 VDC supply. All three ground references—AIGND, AOGND, and DGND—are connected together on your DAQCard.
DIO<0..7>	DGND	Input or Output	Digital I/O signals—DIO6 and 7 can control the up/down signal of general-purpose counters 0 and 1, respectively.
+5 V	DGND	Output	+5 VDC Source—These pins are fused for up to 250 mA of +5 V supply. The fuse is self-resetting.
SCANCLK	DGND	Output	Scan Clock—This pin pulses once for each A/D conversion in the scanning modes when enabled. The low-to-high edge indicates when the input signal can be removed from the input or switched to another signal.
EXTSTROBE*	DGND	Output	External Strobe—This output can be toggled under software control to latch signals or trigger events on external devices.
PFIO/TRIG1	DGND	Input  Output	PFIO/Trigger 1—As an input, this is either one of the PFIs or the source for the hardware analog trigger. PF1 signals are explained in the <i>Timing Connections</i> section later in this chapter. The hardware analog trigger is explained in the <i>Analog Trigger</i> section in Chapter 3, <i>Hardware Overview</i> .  As an output, this is the TRIG1 signal. In posttrigger data acquisition sequences, a low-to-high transition indicates the initiation of the acquisition sequence. In pretrigger applications, a low-to-high transition indicates the initiation of the pretrigger conversions.
PF11/TRIG2	DGND	Input  Output	PF11/Trigger 2—As an input, this is one of the PFIs.  As an output, this is the TRIG2 signal. In pretrigger applications, a low-to-high transition indicates the initiation of the posttrigger conversions. TRIG2 is not used in posttrigger applications.

Table C.2 I/O Connector Signal Descriptions (Continued)

Signal Name	Reference	Direction	Description
PF12/CONVERT*	DGND	Input	PF12/Convert—As an input, this is one of the PFIs.
		Output	As an output, this is the CONVERT* signal. A high-to-low edge on CONVERT* indicates that an A/D conversion is occurring.
PF13/GPCTR1_SOURCE	DGND	Input	PF13/Counter 1 Source—As an input, this is one of the PFIs.
		Output	As an output, this is the GPCTR1_SOURCE signal. This signal reflects the actual source connected to general-purpose counter 1.
PF14/GPCTR1_GATE	DGND	Input	PF14/Counter 1 Gate—As an input, this is one of the PFIs.
		Output	As an output, this is the GPCTR1_GATE signal. This signal reflects the actual gate signal connected to general-purpose counter 1.
GPCTR1_OUT	DGND	Output	Counter 1 Output—This output is from the general-purpose counter 1 output.
PF15/UPDATE*	DGND	Input	PF15/Update—As an input, this is one of the PFIs.
		Output	As an output, this is the UPDATE* signal. A high-to-low edge on UPDATE* indicates that the analog output primary group is being updated.
PF16/WFTRIG	DGND	Input	PF16/Waveform Trigger—As an input, this is one of the PFIs.
		Output	As an output, this is the WFTRIG signal. In timed analog output sequences, a low-to-high transition indicates the initiation of the waveform generation.
PF17/STARTSCAN	DGND	Input	PF17/Start of Scan—As an input, this is one of the PFIs.
		Output	As an output, this is the STARTSCAN signal. This pin pulses once at the start of each analog input scan in the interval scan. A low-to-high transition indicates the start of the scan.
PF18/GPCTR0_SOURCE	DGND	Input	PF18/Counter 0 Source—As an input, this is one of the PFIs.
		Output	As an output, this is the GPCTR0_SOURCE signal. This signal reflects the actual source connected to general-purpose counter 0.
PF19/GPCTR0_GATE	DGND	Input	PF19/Counter 0 Gate—As an input, this is one of the PFIs.
		Output	As an output, this is the GPCTR0_GATE signal. This signal reflects the actual gate signal connected to general-purpose counter 0.
GPCTR0_OUT	DGND	Output	Counter 0 Output—This output is from the general-purpose counter 0 output.
FREQ_OUT	DGND	Output	Frequency Output—This output is from the frequency generator output.

Table C.3 I/O Signal Summary for the DAQCard-AI-16E-4

Signal Name	Drive	Impedance Input/ Output	Protection (Volts) On/Off	Source (mA at V)	Sink (mA at V)	Rise Time (ns)	Bias
ACH<0..15>	AI	100 G $\Omega$ in parallel with 100 pF	25/10	—	—	—	$\pm 200$ pA
AISENSE	AI	100 G $\Omega$ in parallel with 100 pF	25/10	—	—	—	$\pm 200$ pA
AIGND	AI	—	—	—	—	—	—
DGND	DO	—	—	—	—	—	—
VCC	DO	0.45 $\Omega$	Short-circuit to ground	250 at $V_{cc}$	—	—	—
DIO<0..7>	DIO	—	$V_{cc} + 0.5$	13 at ( $V_{cc} - 0.4$ )	24 at 0.4	1.1	50 k $\Omega$ pu <sup>1</sup>
SCANCLK	DO	—	—	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu
EXTSTROBE*	DO	—	—	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu
PFI0/TRIG1	ADIO	10 k $\Omega$	$V_{cc} + 0.5/\pm 35$	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu <sup>2</sup>
PFI1/TRIG2	DIO	—	$V_{cc} + 0.5$	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu
PFI2/CONVERT*	DIO	—	$V_{cc} + 0.5$	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu
PFI3/GPCTR1_SOURCE	DIO	—	$V_{cc} + 0.5$	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu
PFI4/GPCTR1_GATE	DIO	—	$V_{cc} + 0.5$	3.5 at ( $V_{cc} - 0.4$ )	5 at 0.4	1.5	50 k $\Omega$ pu



## APPENDIX D

### DIFFERENTIAL VOLTAGE COMPARISON TABLE

This appendix describes the experimental differential voltages taken at ten degree increments from the stretch sensors. This value was then added to a program equation to relate to a desired position.

DESIRED ANGLE	ACH3 (UPPER) differential voltage	ACH1 (LOWER) differential voltage	difference (ACH3-ACH1)	METER (Eo)	IF METER = diff(ACH3-ACH1)	IF METER < diff(ACH3-ACH1)	IF METER > diff(ACH3-ACH1)
180	3.73	2.07	1.66	1.7	U1,U2,L1,L2 ON	U1,U2,L1,L2 ON	U1,U2,L1,L2 ON
171	3.72	2.08	1.64	1.65	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
162	3.71	2.09	1.62	1.6	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
153	3.7	2.1	1.6	1.55	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
144	3.69	2.11	1.58	1.5	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
135	3.68	2.12	1.56	1.45	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
126	3.67	2.13	1.54	1.4	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
117	3.66	2.14	1.52	1.35	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
108	3.65	2.15	1.5	1.3	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
99	3.64	2.16	1.48	1.25	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
90	3.63	2.17	1.46	1.2	U1,U2,L1,L2 ON	U1,U2 ON; L1 OFF; L2 ON	L1,L2 ON; U1 ON; U2 OFF
U1 = first valve in sequence of upper muscle from compressor input							
U2 = second valve in sequence of upper muscle from compressor input							
L1 = first valve in sequence of lower muscle from compressor input							
L2 = second valve in sequence of lower muscle from compressor input							

## APPENDIX E

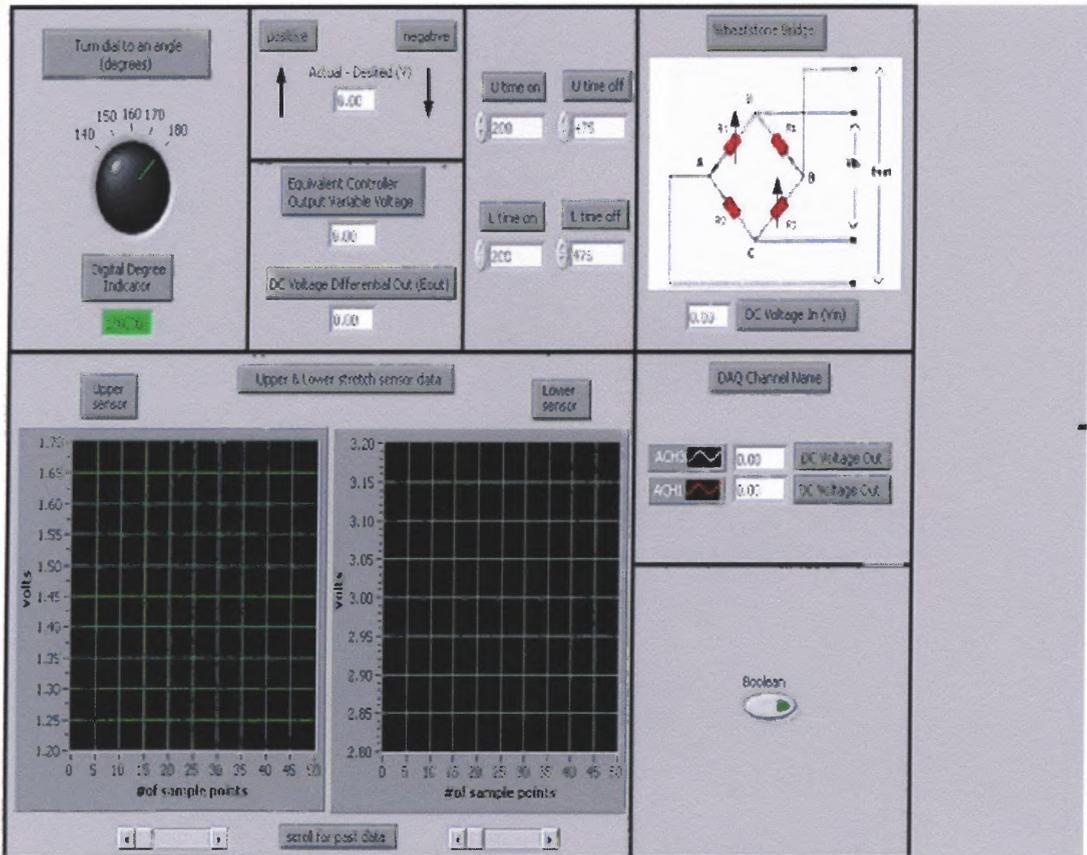
### BANG-BANG CONTROLLER PROGRAM

This appendix describes all of the program contents and icons used in the Bang-Bang control program.

#### Connector Pane



#### Front Panel



### Controls and Indicators

---



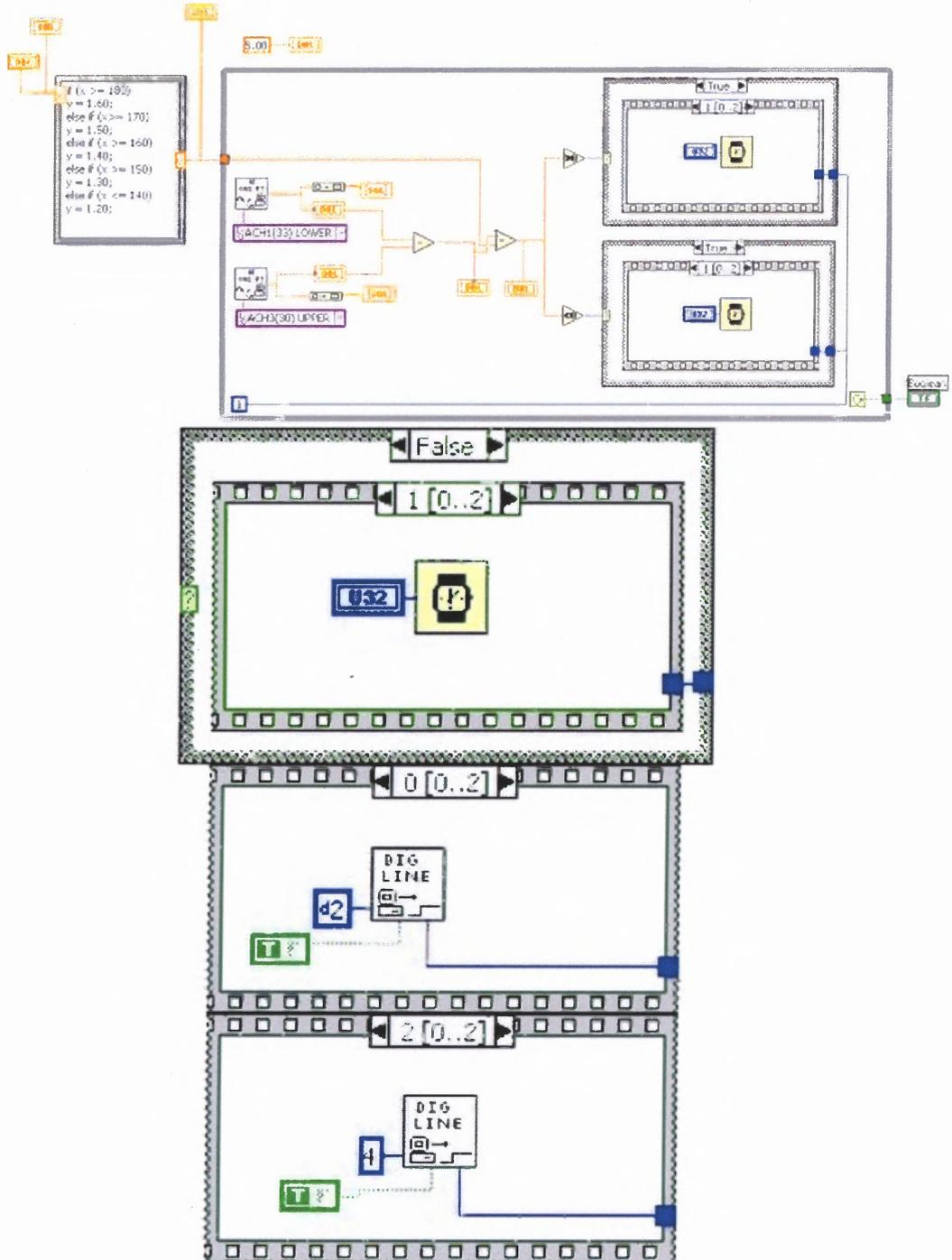
Boolean True/False

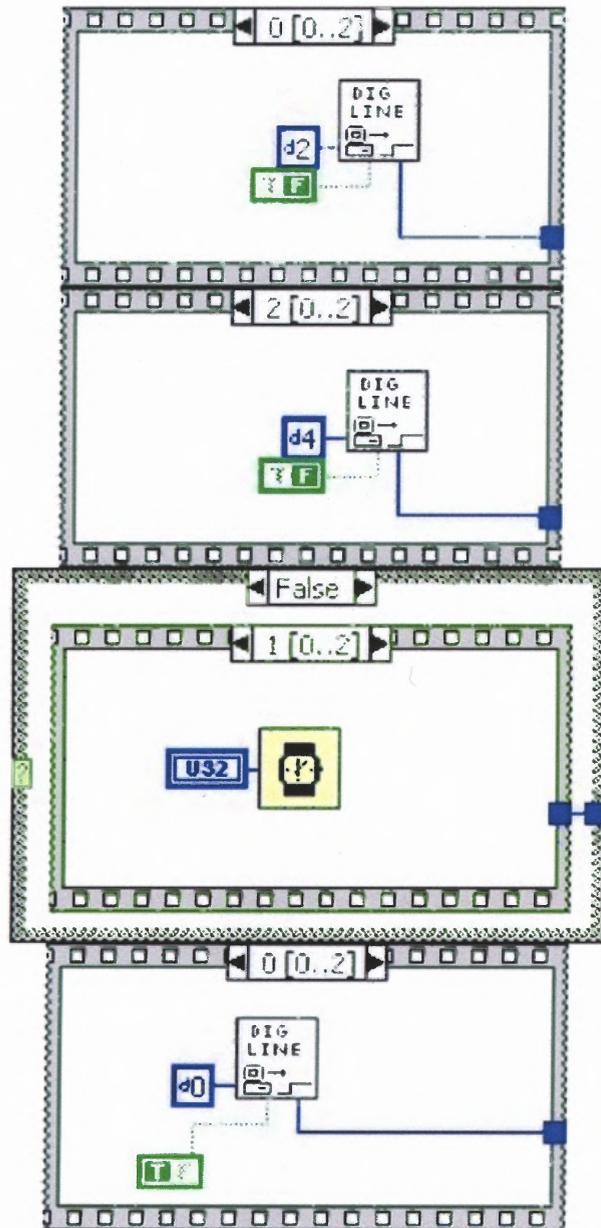


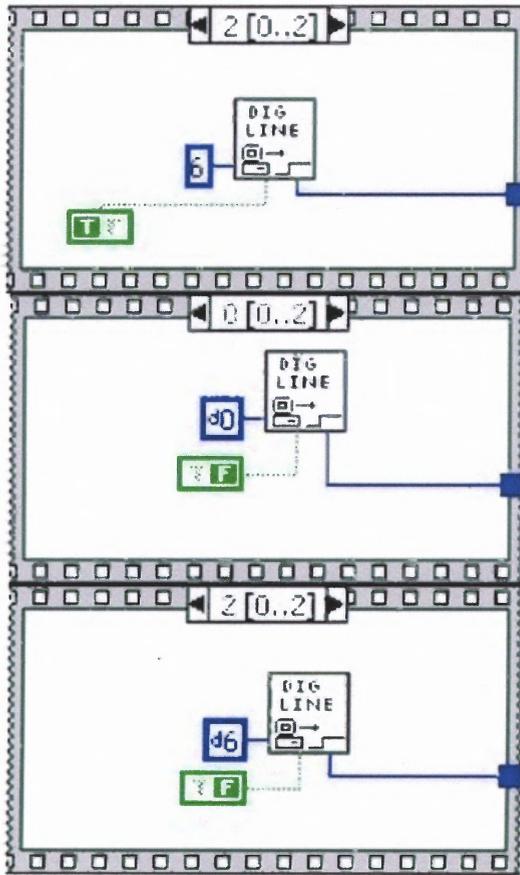
Actual - Desired (V)



## Block Diagram







## List of Sub VIs

---



### **AI Sample Channel.vi**

**C:\PrProgram Files\National Instruments\LabVIEW  
6\vi.lib\DAQ\1EASYIO.LLB\AI Sample Channel.vi**



### **AI Sample Channel (scaled value).vi**

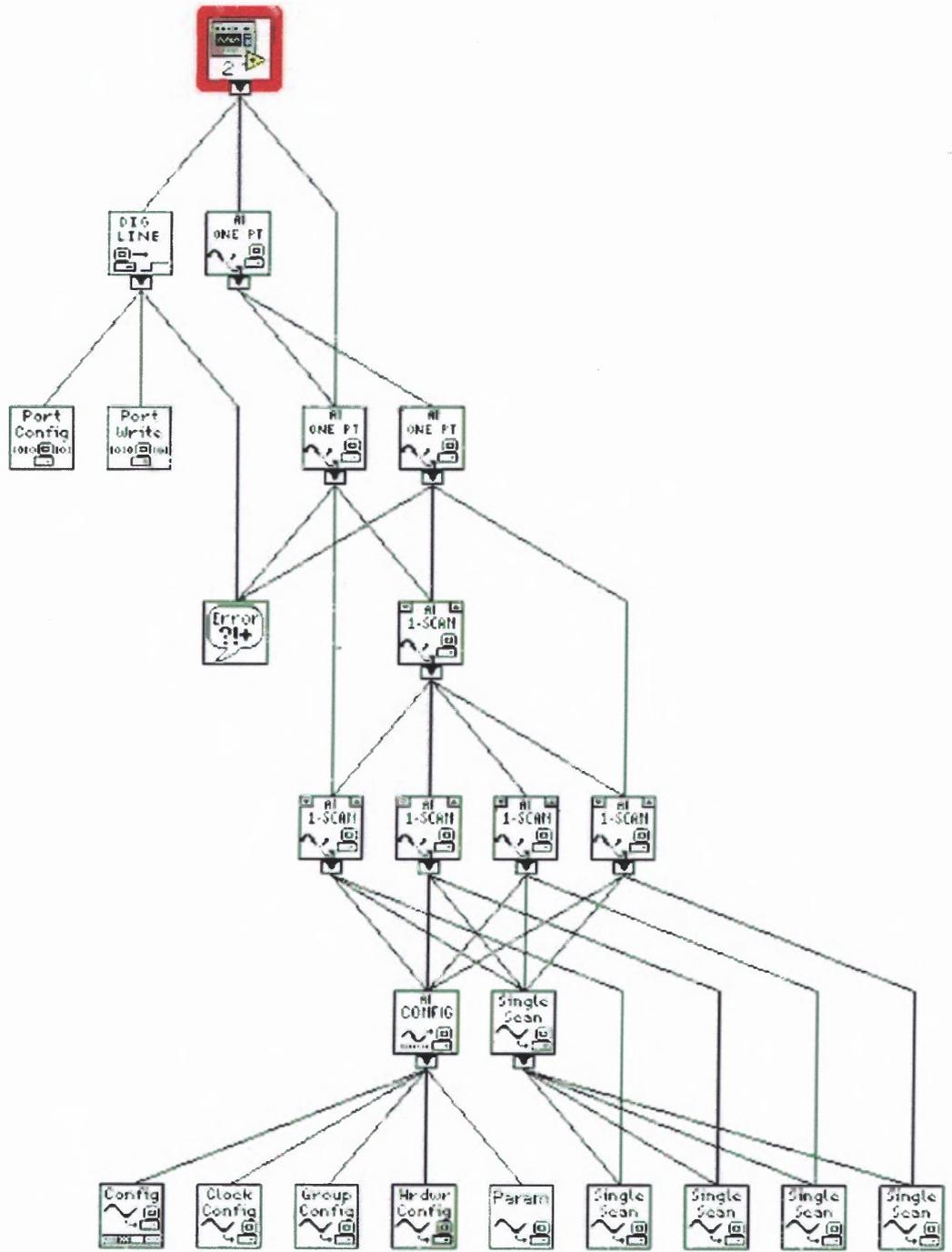
**C:\PrProgram Files\National Instruments\LabVIEW  
6\vi.lib\DAQ\1easyio.llb\AI Sample Channel (scaled  
value).vi**



### **Write to Digital Line.vi**

**C:\PrProgram Files\National Instruments\LabVIEW  
6\vi.lib\DAQ\1EASYIO.LLB\Write to Digital Line.vi**

### Position In Hierarchy



## APPENDIX F

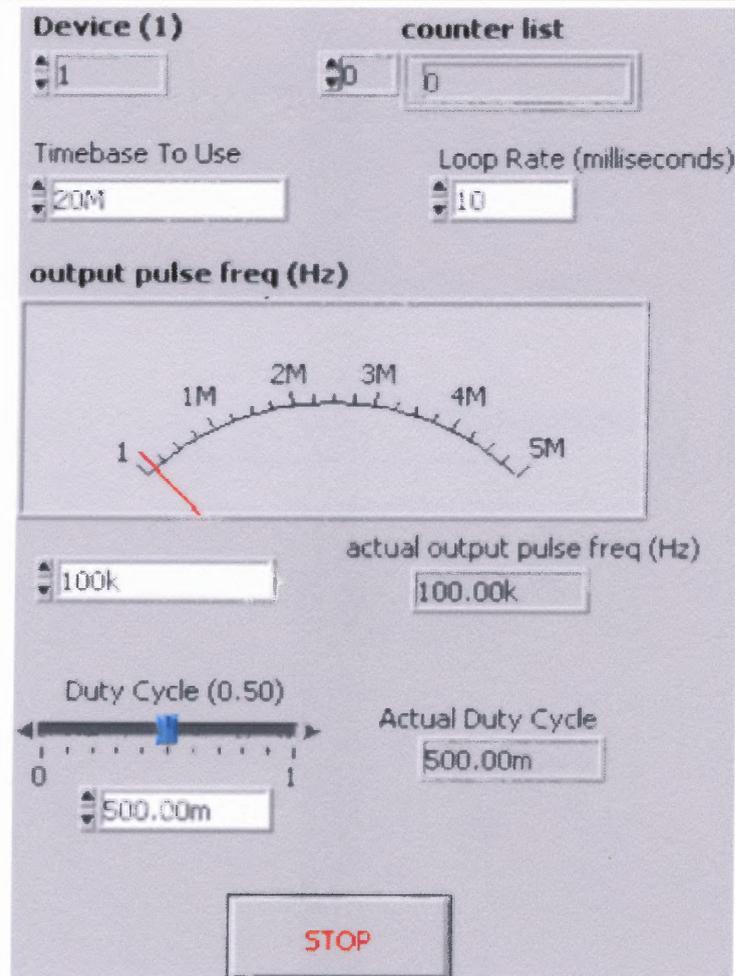
### PULSE TRAIN GENERATION PROGRAM

This appendix describes the entire program contents used in the Pulse Train Generation.

#### Connector Pane



#### Front Panel



## Controls and Indicators

---



**stop**

**(TF) STOP: stop acquiring and generating data and clear the board resources for these operations.**



**output pulse freq (Hz)**

**(SGL) update rate: the number of updates to generate per second. The default input is 1000 updates/s.**



**Device (1)**

**Enter the device number you assigned to the plug-in data acquisition board or other device when you configured it.**



**counter**

**Counter list is an array of strings that specifies the counters the VI assigns to the group.**



**Counter list is an array of strings that specifies the counters the VI assigns to the group.**



**Timebase To Use**



**Loop Rate (milliseconds)**



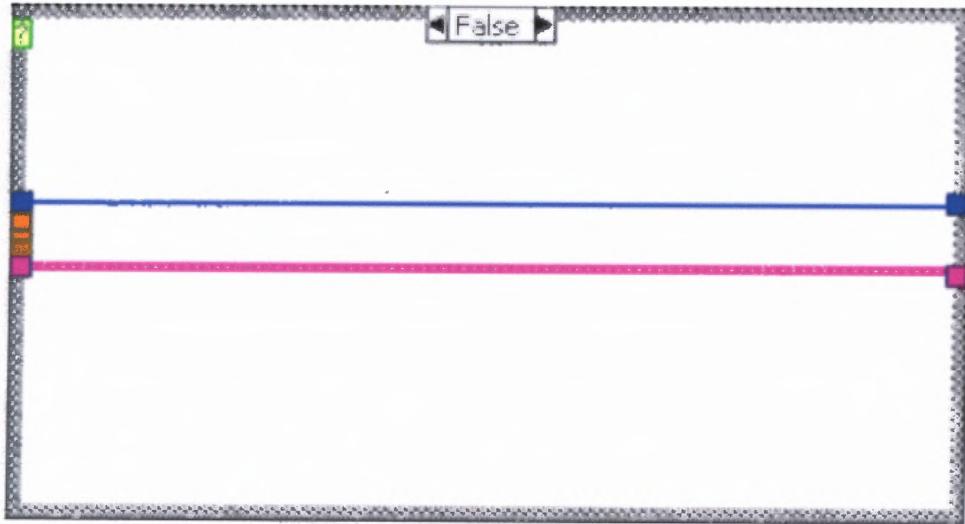
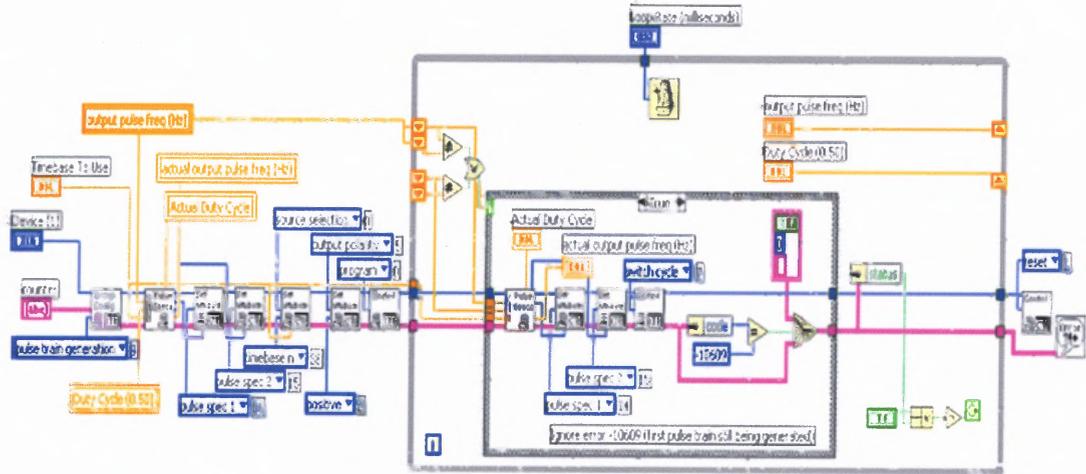
**Duty Cycle (0.50)**



**actual output pulse freq (Hz)**

**Enter the scan rate of the acquisition in Hz, that is, the number of times to read all the channels each second.**

### Block Diagram



## List of Sub VIs



**Counter Set Attribute.vi**  
**C:\Program Files\National Instruments\LabVIEW**  
**6\vi.lib\DAQ\zadvdctr.llb\Counter Set Attribute.vi**



**Counter Group ConFigure.vi**  
**C:\Program Files\National Instruments\LabVIEW**  
**6\vi.lib\DAQ\zadvdctr.llb\Counter Group ConFigure.vi**



**Counter Control.vi**  
**C:\Program Files\National Instruments\LabVIEW**  
**6\vi.lib\DAQ\zadvdctr.llb\Counter Control.vi**



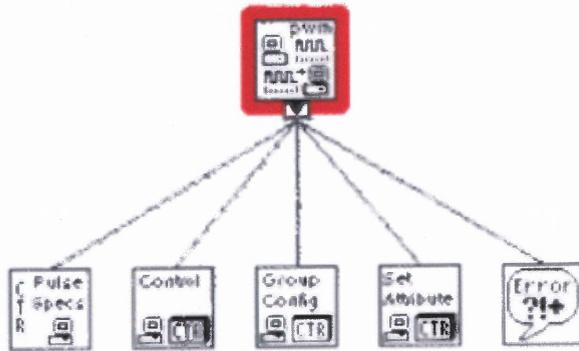
**General Error Handler.vi**  
**C:\Program Files\National Instruments\LabVIEW**  
**6\vi.lib\Utility\error.llb\General Error Handler.vi**



**Calculate Pulse Specs (Fixed Timebase).vi**  
**C:\Documents and Settings\Richard\Desktop\Pulse Train**  
**Generation with Changing Pulse Specs.llb\Calculate Pulse Specs**  
**(Fixed Timebase).vi**

### Position in Hierarchy

---



## APPENDIX G

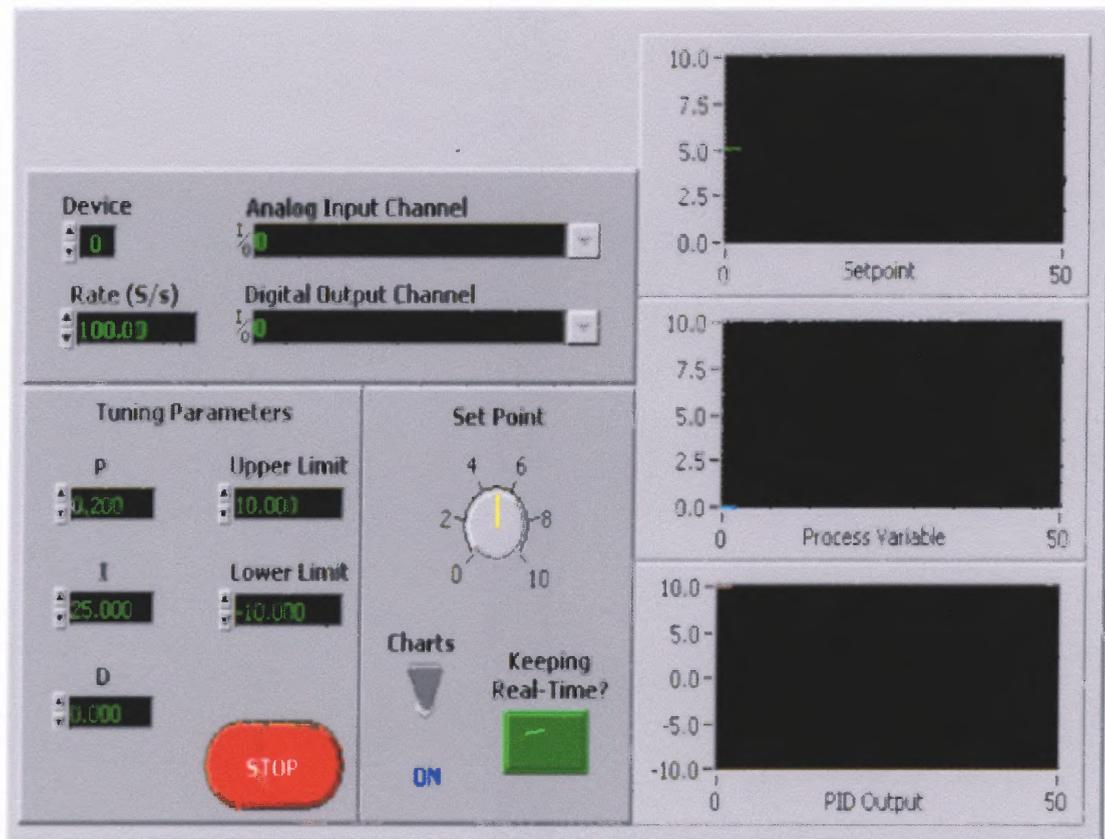
### E-SERIES PID CONTROL PROGRAM

This appendix describes the proposed program to be used in conjunction with some of the components of the original Bang-Bang control program. This program only highlights the use of PID control.

#### Connector Pane



#### Front Panel



## Controls and Indicators

---



**stop**

**(TF) STOP: Stop acquisition and generation of data, clear the board resources for these, and clear all occurrences.**



**tuning parameters**



**(DBL) P: The Proportional component. This is the amount by which the error gets multiplied. The error is the Set-point minus the Process Variable.**



**(DBL) I: The Integral component. This is how much the PID will depend on what has happened in the past.**



**(DBL) D: The derivative component. This is how much the PID will speculate what will happen in the future.**



**Upper Limit**

**(DBL) Upper Limit: The maximum value the PID can output into the system.**



**Lower Limit**

**(DBL) Lower Limit: The minimum value the PID can output into the system.**



**Set-point**

**(DBL) Set-point: Choose a voltage value for the PID to track.**



**Device**

**(I16) Device: the device number you assigned to the plug-in DAQ board during configuration. This parameter defaults to 1.**



**Rate (S/s)**

**(DBL) Rate (S/s): The rate, in samples per second, at which you wish to acquire and write to the input and output channels. The default is 100 samples/sec.**

## Charts



**(TF) Charts ON/OFF:** Turns the chart updating on or off. Your system will still run with the charts off, but you won't see the current input and output values. The advantage of turning off the charts is that you will be able to achieve a faster Rate.

## Analog Input Channel



**(abc) Channel:** identifies the analog input channel you want to measure. You can choose from virtual channels you have previously created by pressing the menu button. The default input is onboard channel 0.

## Digital Output Channel



**(abc) Output Channel:** The analog output channel to which you want to write your PID Output. You can choose from virtual channels you have previously created by pressing the menu button. The default output is onboard channel 0.

## set-point chart



**(DBL) Set-point Chart:** This is the value you desire the system to achieve, and reflects the value of the Set-point knob.

## process variable chart



**(SGL) Process Variable Chart:** This is the current value of your system which is being acquired by the analog input channel of your DAQ device.

## PID output chart



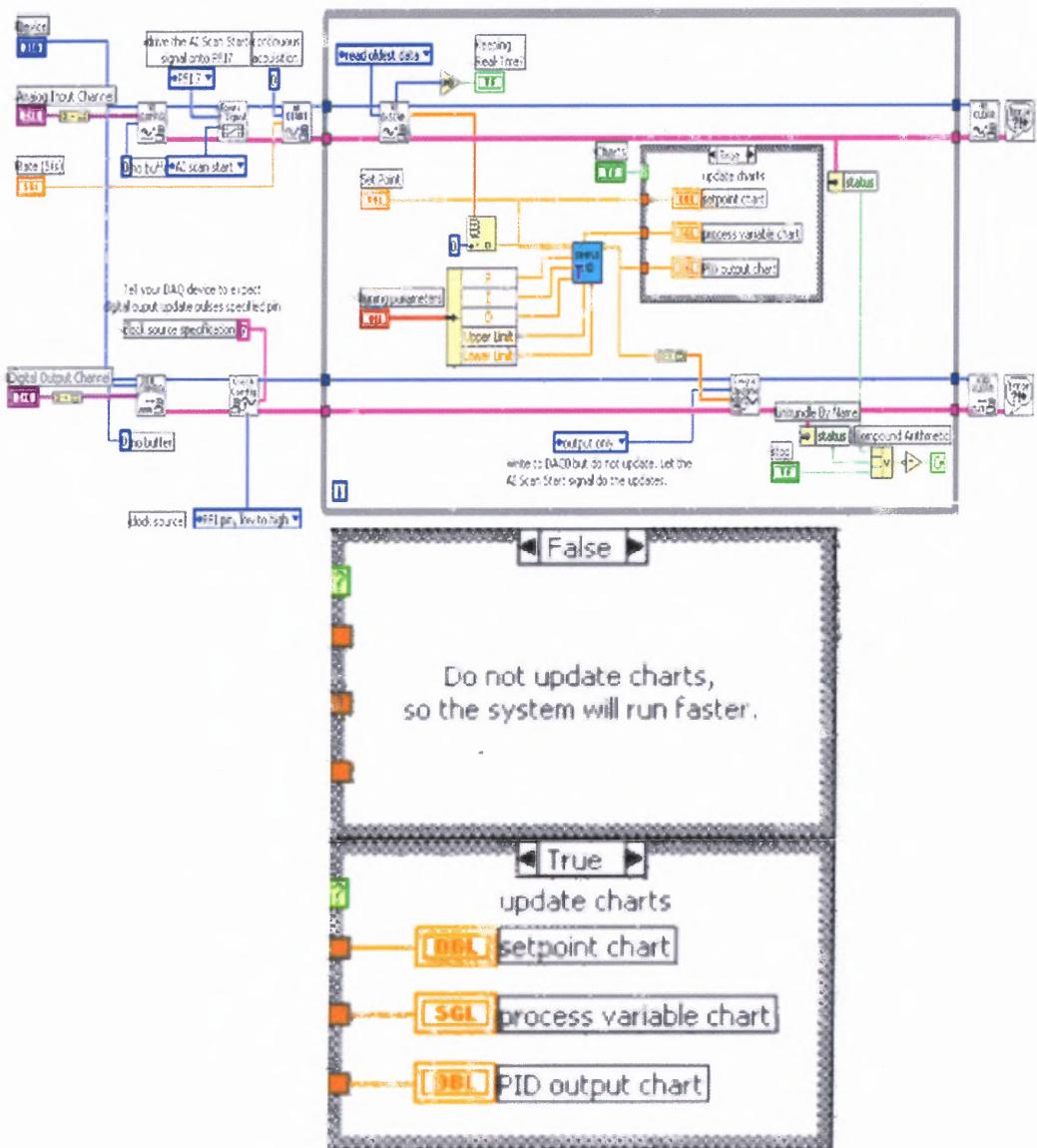
**(DBL) PID Output Chart:** This is the PID output which is being generated at the analog output channel of your DAQ device in an attempt to get the Process Variable equal to the Set-point.

## Keeping Real-Time?



**(TF) Keeping Real-Time?:** Lets you know if your software is able to keep up with the hardware Rate you chose. This is done by checking the "data remaining" output of the AI Single Scan VI to see if the FIFO is empty. You will find you can run faster with the Charts turned off.

### Block Diagram



## List of Sub VIs

**AI ConFigure.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\AI.LLB\AI ConFigure.vi**

**AI Start.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\AI.LLB\AI Start.vi**

**Route Signal.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\MISC.LLB\Route Signal.vi**

**AO Clock ConFigure.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\ZADVD.LLB\AO Clock ConFigure.vi**

**General Error Handler.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\Utility\error.llb\General Error Handler.vi**

**AI Clear.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\AI.LLB\AI Clear.vi**

**simple PID.vit**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\sol\subs.llb\simple PID.vit**

**AO Single Update.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\ZADVD.LLB\AO Single Update.vi**

**AO Single Update (scaled array).vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\ZADVD.LLB\AO Single Update (scaled array).vi**

**AI Single Scan.vi**

**C:\Program Files\National Instruments\LabVIEW 6\vi.lib\DAQ\AI.LLB\AI Single Scan.vi**



**AI Single Scan (scaled array).vi**

**C:\Program Files\National Instruments\LabVIEW  
6\vi.lib\DAQ\ai.llb\AI Single Scan (scaled array).vi**



**DIO ConFigure.vi**

**C:\Program Files\National Instruments\LabVIEW  
6\vi.lib\DAQ\DIGITAL.LLB\DIO ConFigure.vi**

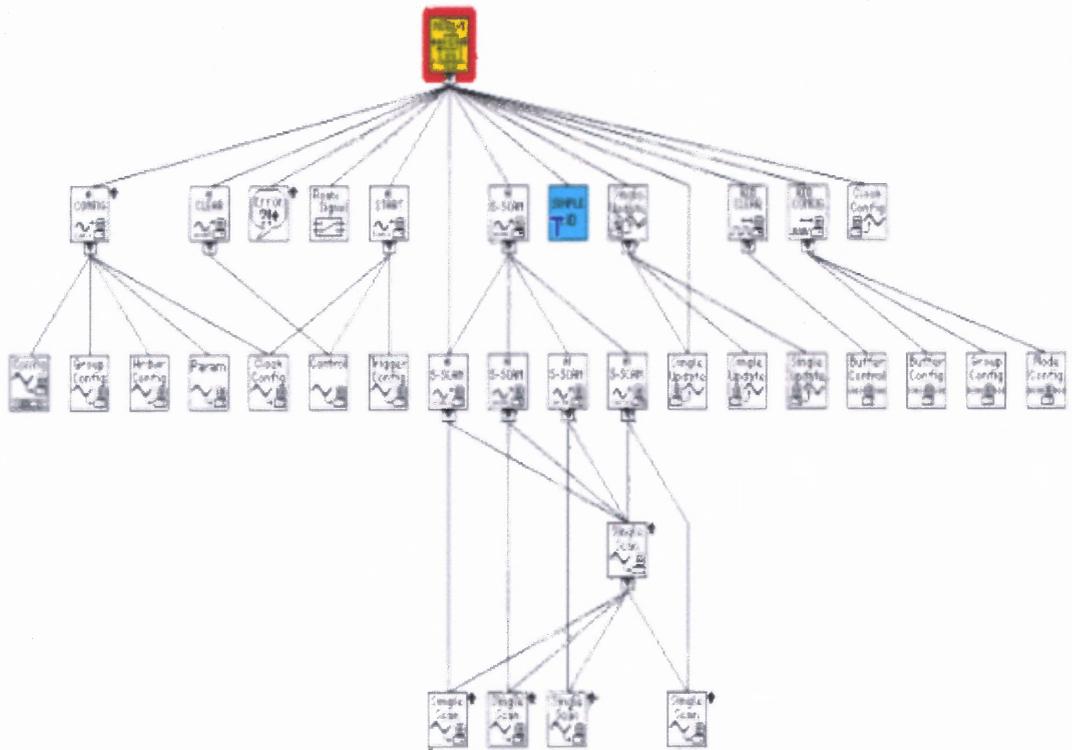


**DIO Clear.vi**

**C:\Program Files\National Instruments\LabVIEW  
6\vi.lib\DAQ\DIGITAL.LLB\DIO Clear.vi**

## Position in Hierarchy

---



## REFERENCES

- [1] Altruis, LLC. (2000). Muscles. <http://www.e-muscles.net/> (3 July 2003).
- [2] Educyclopedia. (2003). Amplifiers and Darlington Pairs. <http://www.ect.mauicc.hawaii.edu/etron/etron112/unit3/index.html> (23 July 2003).
- [3] Feldmen, A. G. (1966). Functional tuning of the nervous system with control of movement of maintenance of a steady posture. Biophysics, 11, 565-578.
- [4] Feldmen, A. G. and Levin, M. F. (1995). The origin and use of positional frames of reference in motor control. Behavioral Brain Science, 18, 723-806.
- [5] GlobelSpec, Inc. (1999). Data Acquisition and Signal Conditioning. <http://data-acquisition.globalspec.com/> (15 July 2003).
- [6] Gottlieb, G. L. (1996). On the voluntary movement of compliant (inertial-viscoelastic) loads by parcellated control mechanisms. Journal of Neurophysiology, 76, 5.
- [7] Hasan, Z. (1983). A model of spindle afferent response to muscle stretch. Journal of Neurophysiology, 49, 989-1006.
- [8] Holmes, K. C. (1998). The limits of Reductionism in Biology. Muscle contraction. <http://www.mpimf-heidelberg.mpg.de/~holmes/muscle/muscle1.html> (8 July 2003).
- [9] HVAC mechanic, Co. (2002). Wheatstone Bridge. [http://www.hvacmechanic.com/wheatstone\\_bridge.htm](http://www.hvacmechanic.com/wheatstone_bridge.htm) (10 June 2003).
- [10] Marconi ECT, Inc. (2001). Understanding the Darlington Driver Circuit. <http://www.marconiect.org/index.php> (10 July 2003).
- [11] Mead Fluid Dynamics, Inc. (2000). Isonic 2 & 3-way Control Valves. <http://www.meadfluiddynamics.com/> (21 May 2002).
- [12] Merlin Systems Corp., Ltd. (2001). Humaniform Robotics:sensors. <http://www.merlinsystemscorp.com/pages/html/stretchspecs.html> (23 Aug 2003).
- [13] Mussa-Ivaldi, F. A., Hogan, N., Bizzi, E. (1985). Neural and geometric factors subversing arm posture. Journal of Neuroscience, 5, 2732-2743.
- [14] National Instruments Co., Inc. (2003). Creating a PID Control Loop Using an E Series DAQ Device and Pulse-Width-Modulated Output. [http://sine.ni.com/apps/we/niepd\\_web\\_display.DISPLAY\\_EPD4](http://sine.ni.com/apps/we/niepd_web_display.DISPLAY_EPD4) (10 August 2003).

- [15] National Instruments Co., Inc. (2003). E Series Multifunction DAQ. <http://sine.ni.com/apps/we/nioc.vp> (22 August 2003).
- [16] National Instruments Co., Inc. (2003). Pulse Train Generation with Changing Pulse Specs (PWM). [http://sine.ni.com/apps/we/niepd\\_web\\_display.DISPLAY\\_EPD4](http://sine.ni.com/apps/we/niepd_web_display.DISPLAY_EPD4) (15 August 2003).
- [17] Otten, B. (1988). Concepts and models of functional architecture in skeletal muscle. Exercise Sport Scientific Review, 16, 89-137.
- [18] Satcure, Inc. (2002). Hobby Electronics Tutorial. <http://www.satcure-focus.com/tutor/index.htm> (23 August 2003).
- [19] Schaafsma, A., Otten, B., and Van Willigen, J. D. (1991). A muscle spindle model for primary afferent firing based on a simulation of intrafusal mechanical events. Journal of Neurophysiology, 65, 1297-1312.
- [20] Shadow Robot Co., Ltd. (1999). Air Muscles. <http://www.shadow.org.uk/products/airmuscles.shtml> (8 June 2002).
- [21] Tondu, B. and Lopez, P. (2000). Modeling and Control of McKibben Artificial Muscle Robot Actuators. IEEE Control Systems Magazine, 1053, 15-38.
- [22] Torrens, R. (2003, January). 4QD-TEC. Electronics Circuits Reference Archive PUT Complimentary Feedback Pair. <http://www.4qdtype.com/putpr.html> (7 August 2003).
- [23] Van der Helm, F. C. T., Veeger, H. E. J., Pronk, G. M., Van der Woude, L. H. V. and Rozendaal, R. H. (1992). Geometry parameters for musculoskeletal modeling of the shoulder mechanism. Journal of Biomechanics, 25, 129-144.
- [24] VanDoren V. J. (1998, May). Basics of Proportional-Integral-Derivative Control. Control Engineering Europe. <http://www.manufacturing.net/ctl/index.asp> (17 June 2003).
- [25] Vishay Intertechnology, Inc. (2003). Shunt Calibration of Strain Gage Instrumentation. [http://www.vishay.com/brands/measurements\\_group/guide/tn/tn514/514jj.htm](http://www.vishay.com/brands/measurements_group/guide/tn/tn514/514jj.htm) (10 June 2003).
- [26] Winters, J. M. (1995). An improved muscle-reflex actuator for use in large-scale neuromusculoskeletal models. Annals of Biomedical Engineering, 23, 359-374.
- [27] Winters, J. M. And Stark, L. (1985). Task-specific second-order movement models are encompassed by a global eighth-order non-linear musculo-Skeletal model. IEEE Engineering Medical Biology, 2253, 1111-1115.

- [28] Zahalak, G. I. (1981). A distribution-moment approximation for kinetic theories of muscular contraction. Mathematical Bioscience, 55, 89-114.