MURDOCH UNIVERSITY

MURDOCH
UNIVERSITY
PERTH, WESTERN AUSTRALIA

Development of a Self-Balancing Robot utilizing FPGA

Engineering Honors Thesis

Ryan Langley

2016

Thesis Document Submitted to the School of Engineering and Information Technology
Murdoch University, in Partial Completion of Engineering Honors Thesis

Author's Declaration

I declare that this thesis is my own account of my research and contains as its main content work which has not previously been submitted for a degree at any tertiary education institution.

_____

Ryan Langley

1 Abstract

The popularity of self-balancing vehicles in modern times is rising as the price of consumer electronics is on a steady decline. Consumers have more access to smarter electronics and mechatronics devices. Self-Balancing vehicles have become an exciting new method of human transport that have the potential to redefine the way we traverse our cities. This document will outline the research, design, construction, implementation and analysis that has been undertaken to develop from the ground up, a scaled down version of a self-balancing vehicle. The two wheeled self-balancing robot discussed in this project will take advantage of National Instruments myRIO-1900 embedded hardware device and will utilize the Field Programmable Gate Array (FPGA) embedded hardware along with PID feedback control loops to achieve stabilization in this inherently unstable system. The final section of this document details the techniques used to achieve this objective and suggests future works for the project that would enable future students to take advantage of this new student engineering asset.

2 Contents

Acknowledgements

Firstly, I would like to thank my parents David and Lisa for supporting me throughout my university studies. I would also like to thank my partner Tiana, for always being there when I needed it most. Shout out to all my university colleagues that have helped me through the dark days, with a special thanks to Michael, aka the big dog of power, who has helped me tremendously throughout the years. I'd also like to thank the engineering technical staff, without their help I would have been surely doomed. Finally, I would like to thank the Murdoch academic staff, my thesis supervisor Gareth Lee and last but not least Graeme Cole who has never failed to spark my interest with a good story.

3 Introduction

In recent years the popularity of self balancing vehicles has increased due to the influence of innovations such as Segway PT [1]. Manufacturers are taking advantage of the affordability and compactness of modern sensor technology, the efficiency of modern electric motors and advancing battery technologies to create some new products that some see as technology that will redefine transportation in our cities [2]. Balancing robots are not a new technology, and the fundamental control techniques that enable them to exist have been taught in linear feedback control engineering theory since the 1950's [3]. Only in recent times with advances in cost, efficiency and size of technologies have balancing vehicles become practical for human transportation.

This project will look at developing a scaled down version of a two-wheeled balancing robot from physical construction, development of software and control techniques using the National Instruments myRIO-1900 as the robot's data acquisition and processing controller. A number of feedback techniques will be used to control the robot's stability and absolute positioning which will communicate through the NI myRIO's FPGA module. Control of the motors will also communicate through the FPGA module with the overall goal of reducing the response time of the system by reducing the time between acquiring, processing information and communicating variations to drive the system. The major goal of this project was to develop a balancing robot using a well-researched and justified approach to hardware selection choices and then have that robot then stabilised using PID control loops so the system will behave appropriately to disturbances and return to its original position. Enabling the robot to then manoeuvre around on commands sent by the operator is considered a secondary objective.

This project was also chosen to demonstrate some of the capabilities of the NI myRIO embedded hardware device and encourage future students to work on projects associated

with the myRIO, including this one. This project is intended to have many opportunities for improvement as it provides good foundations for many other projects in areas such as surveillance, automation, and linear and non-linear feedback control theory.

3.1 Document Structure

This report will follow a chronological style, first discussing some of the research that is relevant for a better understanding in the technical detail of the project. Then the hardware selection methodology will be discussed in detail with reference to reviewed literature and important design decisions. Hardware testing and modifications are then briefly discussed before moving onto the software design methodology and testing of the completed physical assembly with implemented feedback control loops. An analysis of the test results and some software modifications are then carried out before summarizing the works completed within the project and suggesting some future improvements.

4 Literature Review

This section of the report will present the relevant background research for the project. The research will aim to provide the reader with the essential context to the project to ensure that the fundamental concepts of various elements within the project are easily understood. This section will provide detailed understanding of the critical mechanisms required for self-balancing machines along with operating principles of much of the hardware used within this project with the purpose of assisting the reader to better comprehend the project scope.

4.1 Self-balancing Machines

Self-balancing machines refer to the type of machine that when uncontrolled are inherently unstable. The common trait amongst all self-balancing robots or vehicles is the lack of stabilizing supports or wheels which causes the system to pivot around some axis which above lies the machine's center of mass. A visualization of this can be seen in Figure 1, which is a basic diagram of the side profile of a typical wheeled self-balancing machine. It can be seen that the center of mass of the body is located above the pivot point and because of this design the machine is categorized as an unstable system where tiny disturbances will inevitably cause the body to accelerate away from the reference angle θ and towards the ground. The angle θ is the angle that the system theoretically becomes balanced above the pivot point. In practice, balancing the machine at this angle is impossible due to the inability to avoid real world disturbances.

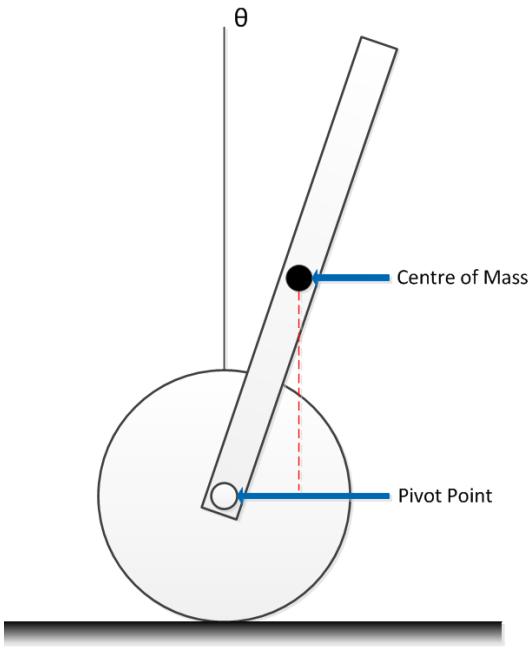


Figure 1: Basic Inverted Pendulum

In control engineering this type of system is called an inverted pendulum, where in the controlled state, the center of mass is positioned directly above the pivoting axis. This type of system is nonlinear, however it still remains useful for demonstrating practices such as the stabilization of inherently unstable systems with linear control concepts. There are two states of equilibrium with this type of system [4]; a stable state in which the body and center of mass are positioned beneath the pivot axis and an unstable state with the pendulum arm now a further 180 degrees, directing upward. In the stable state without control, gravity ensures the system will predictively return to the same natural position with the pendulum arm pointing downwards. The other state refers to when the pendulum arm is pointing directly upward. In this state the system is unstable and will only remain pointing upward with the assistance of a governing force [4]. Achieving stability in this unstable state is the objective of inverted pendulum control. To stabilize an inverted pendulum, the pendulum arm should be balanced by shifting the position of the pivot point to be underneath the center of mass of the body. Depending on the physical setup of the inverted pendulum this can be achieved in various ways. The schematic in Figure 1 shows a wheeled pendulum where the pivot point can be

manipulated by applying torque to the wheels in the direction of the acceleration of the center of mass. To determine the position of the center of mass or the angle of the pendulum arm with respect to the reference angle θ, the system requires some form of feedback measurement. Feedback for an inverted pendulum is not limited to any particular form. However, common mechanisms for feedback control of an inverted pendulum are gyroscopic and accelerometer sensors or encoders [5].

4.1.1 Segway PT

Self-balancing machines have become more popular in recent years due to innovative products such as the Segway Personal Transporter (PT) [6]. The Segway PT was unveiled in December of 2001 on an American talk show before going on sale to the public in November of 2002 [6]. This marked the first publicly available sale of a self-balancing vehicle used for human transportation purposes. Renowned inventor Dean Kamen had a vision to improve the way humans could travel throughout their daily lives, having an efficient self-balancing vehicle to maneuver inconvenient distances was thought to have vast implications such as an improvement to productivity. Before release the product generated an astonishing amount of hype with names like Steve Jobs hailing it as “big a deal as the PC” [2]. Although this hype was short lived, Segway introduced the world to a concept that a self-balancing vehicle could be a practical means of human transportation.

The Segway PT uses a computer controlled gyroscopic based control system. The controller uses information gained from the two accelerometer tilt sensors and five gyroscopic sensors in its Balance Sensor Assembly (BSA) to keep the vehicle stable [1]. The vehicle can be moved in either the forward or reverse direction by simply shifting the weight distribution toward the front or rear. This is usually done by leaning on, or pulling back on the handlebars. To turn the Segway PT, the handle bars can be tilted to the left or right which would communicate to the

dual onboard computers that the user has instructed a turn maneuver. The Segway PT uses two high performance brushless DC motors to stabilize itself by essentially applying torque to the wheels to drive the system in the direction the body is falling. The Segway PT offers high operational safety due to its redundant control systems. Two separate control systems each control half the motor windings at all times to ensure that under the event of a complete failure of either control system the Segway PT can be controlled to safe stop and subsequently be switched off [1].

4.1.2 IO Hawk

A little over a decade after the release of the Segway PT, a self-balancing scooter named the IO Hawk became popular amongst youth becoming a trend as the latest and greatest in portable personal transportation. Many other companies began manufacturing the same type of product and the self-balancing scooter became less formally known as the Hoverboard. The IO Hawk shares most of the same operating principles as the Segway PT which came before it. It requires gyroscopic sensors feedback information to the onboard computer which determines the orientation of the vehicle to ensure that the DC motors provide the precise amount of torque in the right direction to keep the Hoverboard stable. The difference between the IO Hawk and the Segway PT and perhaps the point of innovation comes from the removal of handlebars, which allows complete control of the vehicle with the operator's feet while also enabling the system to be far more compact and easy to transport [7].

4.2 Forms of Feedback

This section will detail a few of the common types of feedback sensor technologies that could be considered appropriate to meet the project objective. Particular focus will be placed on affordable and readily available options that can be implemented simply and will satisfy compatibility with the National Instruments myRIO embedded hardware device.

4.2.1 Quadrature Encoder

Quadrature encoders are a type of rotary encoder that use two output channels to measure both position and direction of rotation. The output channels A and B are situated 90 degrees out of phase of one another. The order in which the rising edge of the output channels is read determines the direction of rotation. The frequency at which the output channels are read also indicates the angular frequency of the shaft [8]. There are various types of rotary technologies that can be employed to achieve similar tasks, such as optical encoders which use slits in a disc made from either metal or glass to shine light into a photodiode. Optical encoders are one of the most common technologies due to their simplistic design. However, optical encoders are sensitive to dust, without an enclosed system it is difficult to determine durability of the sensor. Other types include various magnetic encoders which work on the Hall Effect principle where a voltage is induced over a conductor originating from an electric current induced by a magnetic field perpendicular to the electrical current [9]. Encoders designed around the Hall Effect technology are not affected by dust or small non-conductive particles in the sensor. Simple magnetic rotary encoders can be cheaply manufactured by placing magnetic materials in rubber or plastic wheels, which provides a high level of overall durability in an inexpensive solution [10].

4.2.2 Accelerometer

Although the rotary encoder provides a simple solution to acquiring valuable data concerning the position and direction of motor shafts, additional information relating to the orientation, velocity, acceleration or vibration may also be required. Where the rotary encoder is limited to measuring counts of the encoder shaft and only determining linear acceleration or velocity, the accelerometer can typically measure acceleration in the X, Y and Z planes. The accelerometer can also measure acceleration many times the order of magnitude, with the latest thermal solid state accelerometers capable of measuring in the range of zero to 200,000 g [11]. Although this is an extreme solution used for smart GPS guided artillery shells [11],

commercially available accelerometers can reliably measure hundreds of g's and are relatively inexpensive devices. The use of the accelerometer spans across scientific and industrial and consumer electronics applications like smartphone and tablet orientation or aircraft navigation systems and drone stabilization systems [12]. Accelerometers operate on measuring what is known as proper acceleration which is the acceleration relative to freefall, so an accelerometer at rest will read one g-force upwards due to the influence of that gravity has on objects on earth. An accelerometer can measure these forces in a couple of different ways depending on the technology behind its operation. Mechanical accelerometers like those used in seismic sensors for measuring the magnitude of earthquakes use large masses connected to springs, whereas smaller accelerometers can use capacitive, piezoelectric or thermal gas mechanisms to measure acceleration [12].

4.2.3 Gyroscope Sensor

Perhaps one of the more important sensor types for stabilizing mechatronic applications is the gyroscope sensor. Unlike the accelerometer which measures linear motion, gyro sensors measure angular motion. This type of measurement is particularly good in balancing applications such as the robot described for this project. Gyroscopic sensors come in a variety of types, the largest and most accurate gyros are ring laser gyroscopes used in high performance aircraft and spacecraft [13]. Ring laser gyros are considered to be extremely robust when compared to other types of gyros, this is why they have a significantly low rate of drift [13]. Other types of gyros include Fibre-optic gyros, fluid gyros and the more commonly used vibration gyros which are considerably cheaper [13]. Vibration gyros are much smaller in size than the other technologies mentioned, and because of this they are predominantly used in electronics and mechatronics. Vibration gyro sensors take advantage of the Coriolis Effect by measuring the resulting Coriolis force applied to the sensing arm of a vibrating element. The measured force on the sensing arm is a result of the gyro resisting rotational change on the drive arm which is intentionally vibrated. This follows the basic idea of gyroscope theory where

the momentum of a spinning mass will cause a resisting force to any external angular force [14]. The specific materials and structure of a vibration gyro will depend on the manufacture and the desired level of accuracy.

4.2.4 Inertial Measurement Unit

An inertial measurement unit (IMU) usually contains a triad of accelerometers and gyroscopes to measure linear and angular motion to determine absolute orientation. IMU's can also contain one or three magnetometers to measure the surrounding magnetic fields. However, the main purpose of a magnetometer is to measure the earth's magnetic field for reference so the onboard microcontroller may determine positioning and calibrate against orientation drifting. The three onboard accelerometers measure linear acceleration in all three planes while the three onboard gyroscopes measure angular rotation and represent the information in Euler angles. The classification of object moving in space can be represented in Euler angles with the convention being pitch, yaw and roll [15]. One gyroscope is responsible for measuring each axis. A benefit of using an IMU over using the low level accelerometer and gyro sensors directly comes down to the IMU's onboard data processing capability. The onboard microprocessor acquires and filters the signals measured by the sensors and processes the information through complex algorithms to determine the useful linear and angular vectors or angles while filtering out sensor noise [16]. The calibration of sensors performed by the microprocessor and triad of magnetometers also provides a level of redundancy, avoiding undesirable orientation drifting caused predominately by the gyros sensors.

4.3 DC Motors

In order for the system to respond to feedback information provided by an IMU relating to the orientation of the robot, the controller needs to apply force and create a velocity in the direction of the falling mass. As mentioned in Section 3.1, in order for a balancing system to maintain stability the pivot axis needs to be manipulated to a position underneath the bodies

center of mass. The driving force behind this manipulation comes from torque applied to the wheels by a motor. There are a couple of different DC motor types that could be used to apply the required torque. Servo motors are typically used as rotational actuators. They contain an onboard sensor to determine position feedback meaning additional rotary encoders are not necessary. Servo motors have the advantage of high torque but have a limited rotational range, usually 180 degrees in hobby servo motors [17]. Due to the non-continuous rotational nature of a servo motor means that using one in a balance robot would imply the robot would be unable to move freely and continuously. Stepper motors are not restricted in this way as a stepper motor can rotate continuously and can also provide significant torque at lower speeds or revolutions per minute (RPM). Stepper motors are useful for performing small and precise rotations of the motor shaft making them suitable for a range of applications such as printing, postal sorting machines and positional control for CNC milling. The big limitation to using this type of motor comes from the motor driver. For bi-directional control of a stepper motor a dual H-bridge motor driver is needed [18]. For comparison two standard bi-directional DC motors could be operated using the same motor driver as just one stepper motor. Another common motor used in mechatronics applications is the brushed DC Gearmotor. Typical cylindrical DC motors used in mechatronics have a fast operating speed but do not provide much torque on their own, when attached to a gearbox the deliverable amount of torque available at the output shaft of the gearbox becomes significantly higher. The tradeoff for the increase level of torque comes at a sacrifice to the motor RPM. As DC Gearmotors of various power ratings torque and RPM are inexpensive and readily available in hobby shops [17] this limitation is insignificant to the overall project scope.

4.3.1 DC Motor Torque-Speed Characteristics

A significant limitation with DC motors comes from the deliverable amount of torque over the motor RPM range. At or just after rest the motor can deliver its maximum rated torque, as the speed of the motor steadily increases the available torque decreases linearly as shown in

Figure 2. This is the typical torque speed characteristic of a DC motor. The graph shown in Figure 2 is for a constant motor voltage, if the supply voltage of the DC motor is increased to a new level, the shape of the torque speed curve remains the same but the torque speed curve is shifted up to the new voltage level. This essentially means an increase in voltage causes a linear increase in torque for any given speed along with an increase in both maximum speed and stall torque as shown in Figure 3.

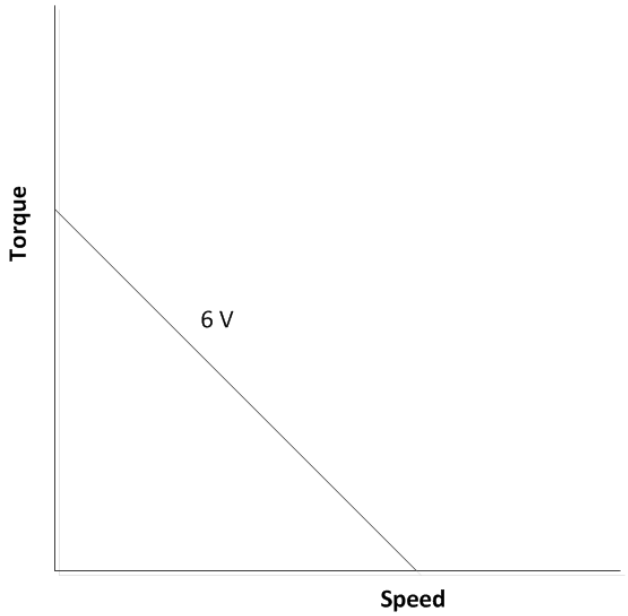


Figure 2: Torque Speed Characteristics of a DC motor

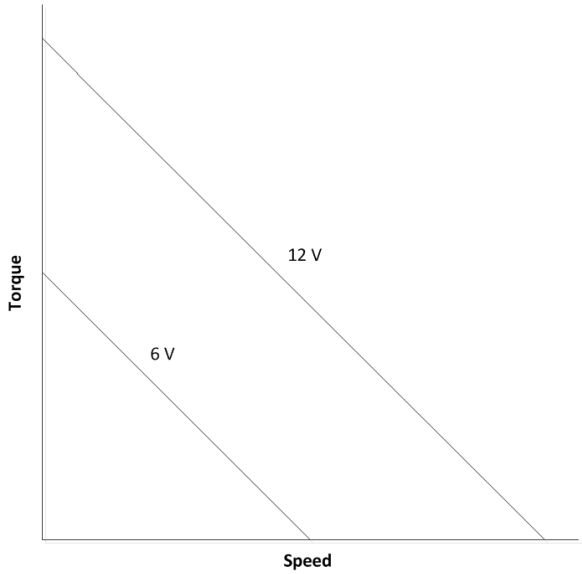


Figure 3: Effect of Voltage on Torque Speed Curve

4.3.2 Pulse Width Modulation Motor Control

Precise control over the speed of a motor can be achieved by adjusting the voltage supplied to the motor terminals. A common way of manipulating the voltage is done with pulse width modulation (PWM) through a motor driver board. Switching devices such as MOSFETs are instructed to switch on at a certain frequency which is known as the PWM switching frequency [19] and stay switched on for a set amount of time, or a percentage of the total switching period, this is known as the duty cycle. The duty cycle is represented as a percentage of the switching period that the MOSFET has remained on. The percentage of the time the MOSFET remains switched on is proportional to the average voltage output delivered to the motor terminals. For example, if the duty cycle is 50% and the supply voltage is 12 volts the average voltage delivered to the motor is 6 volts. PWM is a relatively efficient means of stepping down the average supply voltage as power is not dissipated in resistor circuits to achieve the step down. The power lost in PWM is dissipated in the MOSFET and increases exponentially with MOSFET switching frequency [20]. However at low frequencies very little power is dissipated within the MOSFET.

4.4 National Instruments myRIO

The National Instruments myRIO is an embedded microcontroller device created primarily for students in educational institutions to provide a platform to teach real engineering projects within the space of one semester. NI myRIO is based on National Instruments Remote I/O technology (RIO) that can be found in other NI products such as CompactRIO or Single-Board RIO which are used throughout industry. NI myRIO has a dual-core ARM Cortex-A9 real-time processor along with a Xilinx FPGA [21] with a preloaded FPGA personality designed to immediately aid in mechatronics projects by offering a range of functionality as standard such as I^2C, UART and SPI communications protocols along with encoder and PWM facilitation [21]. The NI myRIO comprises of 40 digital I/O, 10 analog in and 6 analog outputs divided over the three expansion ports along with various communication protocols mentioned earlier, PWM, encoder, 802.11b/g/n wireless and also contains an onboard accelerometer. The physical hardware capability embedded within the myRIO is extensive and will not be covered completely in this document.

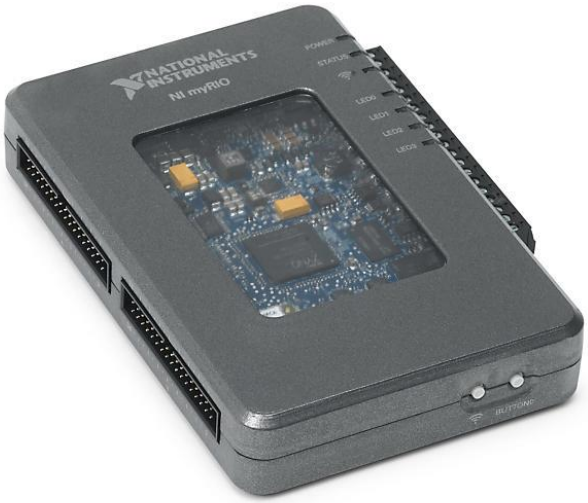


Figure 4 National Instruments myRIO Device [21]

4.4.1 Field Programmable Gate Array (FPGA)

As mentioned in Section 3.4, the NI MyRIO is also equipped with a Xilinx integrated FPGA [22] and allows the user to program a multitude of customized I/O. FPGA is the abbreviation for Field Programmable Gate Array. In short an FPGA is an integrated circuit designed to be configured post sale after manufacturing. The FPGA contains a large array of programmable logic blocks or at a more basic level simple logic gates that usually coincide with various memory elements. When programming the FPGA, physical circuits are created using these logic blocks to represent the desired program. This type of design means that an FPGA is far more robust than a typical microprocessor program. The number of logic gates has grown exponentially since the first commercially available FPGA chips in 1985 with around 8000 logic gates to modern FPGA chips housing millions of gates [23], allowing for far more complex programs within the FPGA. For the balancing robot proposed in this project, an FPGA was utilized for the vehicle stabilization. The justification for this decision comes down to the true, physical circuit architecture that an FPGA offers providing a more reliable and robust hardwired program. Incorporating a PID feedback loop within the FPGA would provide the most robust program, however manipulating the National Instruments FPGA personality which is a premade FPGA template may come with its own challenges that might not be worth the benefits.

4.4.2 Communications Protocols

The NI myRIO offers three common serial communications protocols SPI, UART and I2C. Each achieve similar functionality by providing a means for communication between the master device (myRIO) and any slave devices in the system. These could be feedback sensors, shift registers or other small peripherals.

4.4.2.1 SPI - Serial Peripheral Interface

Serial Peripheral Interface (SPI) is a serial communications bus that was developed by Motorola in the 1980's [24] and has become a standard in serial communications protocols due to full-duplex operating mode using the conventional master/slave architecture. SPI is a synchronous communications protocol with many advantages which help increase its popularity and more, establish it as one of the most commonly used communication protocols for short distance transfer of information. SPI is not limited to any maximum clock speed, and also not limited to standard 8-bit bytes, helping it achieve higher throughput than other protocols such as I^2C [24]. A problem with SPI is the need for more pins on integrated circuit packages, even with the variant of SPI with three wires, other communication protocols such as I^2C are just two pins to provide a solution with less clutter.

4.4.2.2 I2C - Inter-Integrated Circuit

Emphasis here has been placed on the I^2C protocol because of its extensive use in this project. An example of the operation of this protocol is illustrated in this section to further aid the reader with I^2C fundamentals.

I^2C is an abbreviation for Inter-Integrated Circuit, hence the I^2 part. This essentially translates to communications between multiple integrated circuits. The protocol has two signal wires which are serial data (SDA) and serial clock (SCL). It also supports synchronous communication and bi-directional data transfer with multi-master support [25]. Slaves have 7bit addresses within the I^2C protocol with the last bit of the byte reserved for signifying the type of bus transaction (write/read). The data transfer is in byte orientated, that is (8bits) at a time followed by an acknowledge from the receiver whichever that may be at the time depending on the read write operation (master or slave). The two bus speeds supported by the myRIO are standard mode (100kbps) and fast mode (400kbps) [21] [26]. The protocol itself supports

faster bus speeds up to ultra-fast mode 5Mbps (unidirectional), however these are not supported by the myRIO.

All the devices on the bus signal each other by pulling down the SDA and SCL lines. This means that external pullup resistors are required to keep the line at the higher potential. Thankfully the myRIO has 2kΩ internal pullup resistors integrated on its specified I^2C data communications lines [21] so no further pullup resistors are required for the bus. The master, in this case the myRIO, is responsible for initiating and terminating the bus transactions and is also responsible for generating all the clock pulses on the SCL line. There are three different modes available within the I^2C protocol which correspond to write, read and write/read. The write/read mode is the only one of interest for this project as data from specific registers that correspond to various orientations will need to be read from the slave device. To do this the master will first send the slave address and the write bit followed by the data registers to be read from. This is then preceded by a repeated start condition and the address again. However this time the address byte will contain a read bit which will signify to the slave that the master is ready to receive data from the specified register. If the register contains two bytes of data for each data output type, then the most significant byte will be sent first followed by the least significant byte [26]. The data registers will auto increment in the memory from this point on until the master signals a stop condition, so multiple registers can be measured 'essentially simultaneously'.

4.4.3 LabVIEW

LabVIEW is a graphical programming environment used to program the NI myRIO. The real-time processor and FPGA are completely programmable in LabVIEW however they can also be programmed in C or C++ [21]. LabVIEW provides an easy to troubleshoot, visual style approach to programming which is used globally throughout industry and educational institutions. LabVIEW development environment was designed to hasten the productivity of scientists and

engineers by creating a syntax that is easy to visualize and develop code for complex engineering systems. Where LabVIEW excels over traditional text based languages more so than anywhere is its ability to slow and visualize the execution of code when troubleshooting to understand and follow the flow of information within the program and easily identify the cause of program errors.

5 Hardware

This section of the report will describe the justification for certain hardware selection decisions and will provide a brief description for specific components chosen in the design. This section will also detail the steps made in the design and construction phase of the project, and indicates modifications made to the design after hardware testing.

5.1 Hardware: Methodology

5.1.1 Adafruit BNO055 Absolute Orientation Sensor

For the control of any self-balancing vehicle the orientation of its vertical axis must be measured. When the vertical segment of the robot is perpendicular to the ground the robot is in a balanced position. However, as this is a very unstable system small variations will cause the angular velocity to quickly increase resulting in the body quickly accelerating toward the ground around the pivot point due to gravity. In order for the robot to remain upright the motors must provide torque to the wheels driving the robot in the direction in which the center of mass is falling. For this to operate correctly a feedback signal of the robot's position is required. This feedback will be measured through the Adafruit Absolute Orientation Sensor [16] which contains an onboard Bosch BNO055 intelligent 9-axis IMU.

The Adafruit BNO055 was recommended by an academic as a solution to gyroscopic noise and drifting. The reason the Adafruit BNO055 sensor prevails over other IMU's and low level sensors is because Bosch integrated a MEMS accelerometer, magnetometer and gyroscope

into a single die and paired it with an ARM Cortex-M0 high speed processor [26] to filter and arrange the data into a systematic and convenient order. The orientation data can be extracted from data registers in Euler angles, quaternions and vectors. This will reduce the overall complexity and thus amount of time needed to measure the vital input control signals required for the robot stabilization. Because of previously stated reasons this option is more attractive than the gyroscope module suggested by National Instruments in their project essentials guide for the myRIO [27].

5.1.1.1 Data Output

The Bosch BNO055 intelligent 9-axis sensor can output more than just orientation data. The sensor can output the following data [16]:

- Euler Vector data at 100Hz for 3-axis orientation data which is based on a 360-degree sphere.
- Quaternion four-point data at 100Hz for absolute orientation.
- Acceleration Vector at 100Hz, three axis of acceleration (linear motion plus gravity).
- Angular Velocity Vector at 100Hz
- Linear Acceleration Vector at 100Hz
- Gravity Vector at 100Hz
- Magnetic Field Strength Vector at 20Hz
- Temperature at 1Hz

*Units for Acceleration, Angular Velocity, Magnetic Field Strength and Temperature are in $\frac{m}{s^2}$, $\frac{rad}{s}$, μT and °C respectively [16]. These units can be changed by writing to the unit selection register [26].

5.1.1.2 I²C Protocol

The Bosch Intelligent 9-axis sensor is a serial device which can utilize either the I^2C or UART protocol to communicate with the NI myRIO [21] [26]. For this project a decision was made to communicate using the I^2C protocol as the project did not require any of the parallel communication conversion offered by the UART protocol. The myRIO would act as the master in the system and the Adafruit BNO055 sensor would function as a slave.

5.1.2 12 Volt Metal DC Gearmotor

Motor selection was one of the more difficult hardware decisions in the design as many of the other hardware components had to be designed around the motor choice and some system design conventions had to be assigned. For instance, the system voltage had to be determined before motors could be purchased.

5.1.2.1 System Voltage

The system voltage refers to the voltage supplied by the battery to not only pulse the motor and motor driver circuit, but also the power input for the NI myRIO. Motor driver boards are less important in this design decision as it is expected that the voltage range acceptable by the myRIO power input would also lie within the supply voltage range of most motor drivers [28]. The NI myRIO power input specifies an input voltage between the range of 6-16 V so a system voltage of 12 V was chosen as DC gearmotors and batteries were readily available around this voltage level [29] [30]. The block diagram shown in Figure 5 shows the planned power flow for the balancing robot system. As can be seen the battery supplies the motor driver circuit, the motor drivers and thus motors, and this all will be supplied directly with the terminal voltage of the battery. The myRIO will also be supplied with the battery terminal voltage, however an onboard linear voltage regulator [21] will provide the IMU and encoders with the required 5 V power input requirements [16].

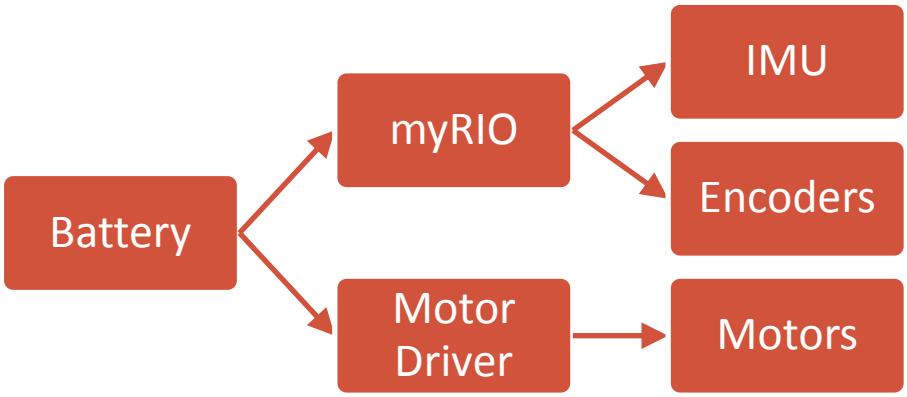


Figure 5: Power Flow Block Diagram

Once the system voltage had been determined a 12 V motor could be selected. However, the next step was to compare the available DC gearmotors based on specified motor RPM and torque. The motors were chosen with more consideration to stall torque than RPM. Motors would be chosen with higher torque then necessary for this application as it was difficult to determine exactly how much torque might have been needed at any one time. Over-designing the system in this way provided a sort of safeguard ensuring the balancing robot had enough torque available when required. A drive motor sizing tool [31] was useful to gain estimates on required torque and RPM that would be necessary for this type of application. Using some estimates of 1.5Kg robot mass, 0.05m radius of drive wheels and 12 V supply, the resulting estimated torque was 0.2 N.m with a motor speed of 100 RPM. Due to the many uncertain variables in this estimation the minimum stall torque was tripled, resulting in a minimum stall torque value around 0.6 N.m for the selection of the motor. The motor speed was left at 100 RPM, which corresponds to 1.667 revolutions per second. With a simple calculation shown in (1), assuming a wheel circumference of 0.314 meter the maximum velocity is determined to be 0.523 m/s which is deemed to be acceptable for a balancing robot. The maximum velocity could easily be increased by replacing the wheels with larger diameter wheels if the system response needed improvement.

$$1.667\ rev/s \times 0.314\ m = 0.523\ m/s \quad (1)$$

The final choice for the motors was the 12V Medium Power 75:1 Metal Gear Motor from Pololu [30]. The gear motors have the following general specifications:

- Stall Current @ 12 V 2100 mA
- Stall torque @ 12 V 0.883 N.m
- Free-run Speed @ 12 V 100 rpm
- Free-run Current @ 12 V 200 mA

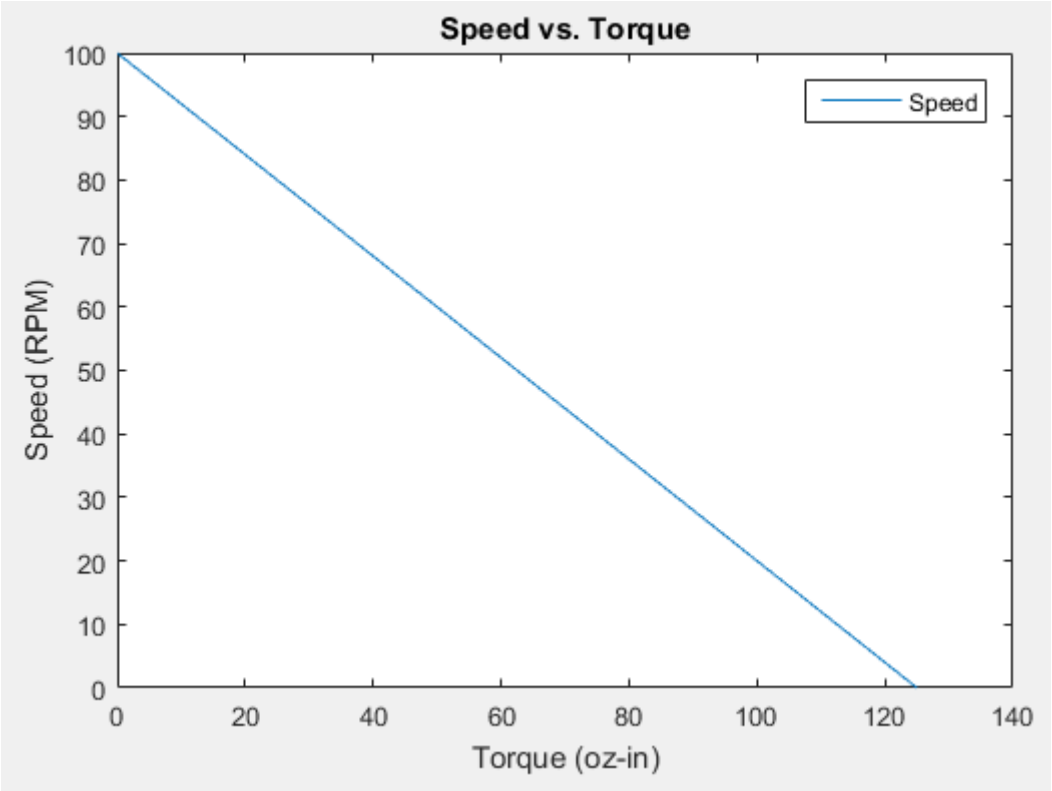


Figure 6: Torque Speed Curve of Pololu Motor

5.1.3 Quadrature Encoder

The motors also come with 48 CPR quadrature encoders on the motors which will provide a total of 3591.94 counts per revolution of the gearbox's output shaft due to the 75:1 ratio [30]. The encoder uses a Hall effect sensor to detect the motor shaft rotation using magnetic disk which is directly connected to the motor shaft. Being a quadrature encoder, two output channels determine the direction and speed of the motor shaft as explained in Section 3.2.1.

5.1.4 11.1 Volt Lithium-ion Polymer Battery

The type of battery chosen to supply power to the balancing robot power systems is a Lithium-ion Polymer Battery (Li-Po). An advantage of choosing this type of battery is a high energy density. When compared to other battery non-lithium based technologies the Li-Po battery has a much higher watt-hour per kilogram rating. This essentially translates to more power for less weight, reducing the charging frequency and also the average charge time per cycle. There are many more advantages when it comes to choosing lithium-ion batteries, some of which include, increased cycle life; high discharge rates and no maintenance requirements [32]. However, the downside to picking this type of battery is that they are typically more expensive and are only available in terminal voltage multiples of 3.7 V as each cell is this voltage.

The final choice of the battery is the Vant Battery LiPo 2250mAh 11.1 V [29]. This battery has ample capacity at 2250mAh and a high discharge rate of 30C, which means that the maximum discharge rate is 30 times the mAh capacity for a duration 30 times less than an hour. This choice means that the original plan to have 12 Volt power system for the robot has also changed to a range of 11.1 nominal Voltage to 12.6 full capacity Voltage due to the battery cell topology. The varying supply voltage from the battery will cause the torque delivered to the wheels to also vary. The battery may need to be kept at a constant voltage or a supply voltage feedback may be necessary to scale the torque appropriately.

5.1.5 L298N Dual Full-Bridge Driver

Once the motors were chosen the required motor drivers could be determined based on the motor stall current. During operation the assumption is made that the motors could possibly draw their stall current whether it be instantaneous or for a very short period of time. To cater for this, the motor driver was chosen so the stall current may never exceed the absolute maximum ratings of the L298N motor driver [33].

Since the battery chosen in Section 4.1.4 is 11.1 V LiPo with a maximum charge voltage of 12.6 V, and the 2100mA stall current is rated at 12 V a new stall current can be calculated (2) that takes into consideration the more realistic maximum stall current the motors will encounter.

$$\frac{12.6\ V}{12\ V} \times 2100\ mA = 2205\ mA \tag{2}$$

Based on the new calculated stall current, the L298N Dual Full-Bridge Driver [33] was chosen as it can provide continuous 2 A DC per channel with a total of two channels and a peak current rating of 3 A [33]. The duinotech Dual/Stepper Motor Controller Module [28] houses the L298N motor driver and contains a 5V regulator that can also be used to power other devices in the project if necessary. The module also features back-EMF and over-temperature protection [28].

Due to the LiPo batteries 30 C rating it can deliver a rated current of 67.5 A. The current that would result from a battery short circuit would be even higher than this, so a 5 A fuse and a switch is placed in series with the battery to ensure the high short circuiting currents do not cause risk of fire or injury and to also protect the motor driver from overcurrent.

5.1.6 3D model

The last pieces of hardware such as mounting brackets for the gear motors were purchased along with solid rubber robotics wheels and aluminum mounting hubs to connect the wheels to the 4mm diameter output driveshaft of the gear motors. These components were recommended by the same source as the Gearmotors [30]. Once all the components were finalized and before the frame was designed, a 3D sketch of the system in AutoDesk Fusion 360 [34] was created to observe what the assembly of the system would look like with some of the components and observe some of the tolerances between components to ensure there would be enough space for wiring. The 3D model also helped design the frame that would house all the components, which was ultimately formed from a single piece of folded steel

sheeting with small folds on the edges for added rigidity. Another benefit of creating the 3D model was the ability to assess the weight distribution of the components. It was desired that the vertical section of the frame be close as possible to perpendicular with the ground, spacers were placed between the myRIO and the frame as an attempt to balance out the system by bringing the center of mass above the pivot point to maintain this perpendicular arrangement. Figure 7 illustrates the 3D models of components created for this section with the complete assembly shown in Figure 8.

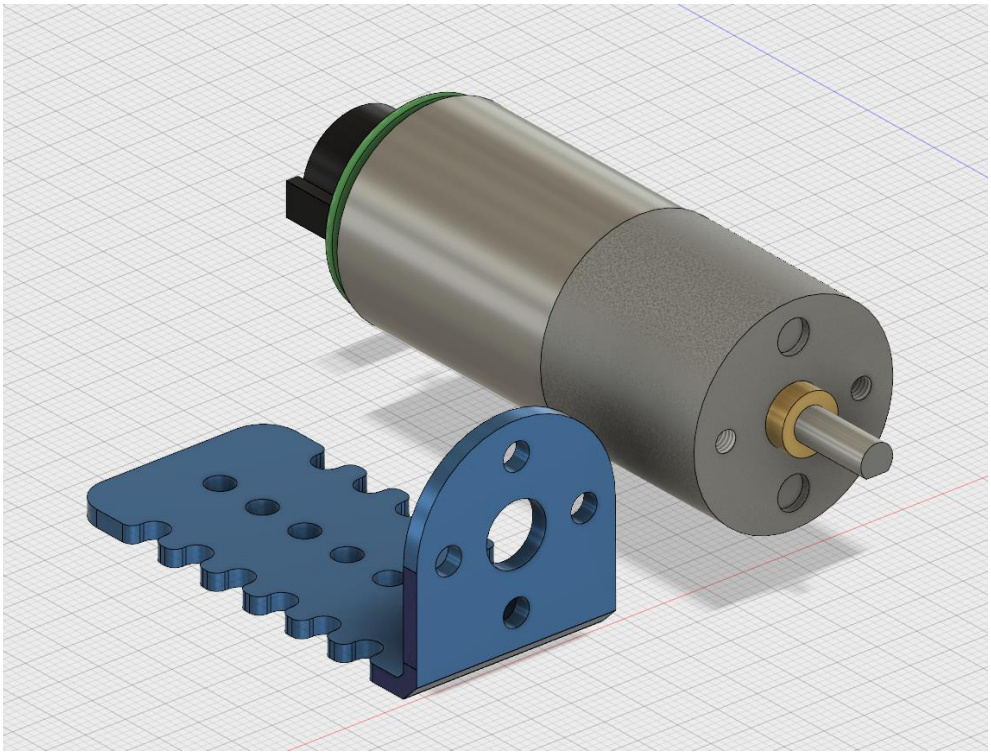


Figure 7: Pololu 25D mm Gearmotor Bracket and 75:1 Metal Gearmotor with 48 CPR Encoder -3D Sketch [22] [16]

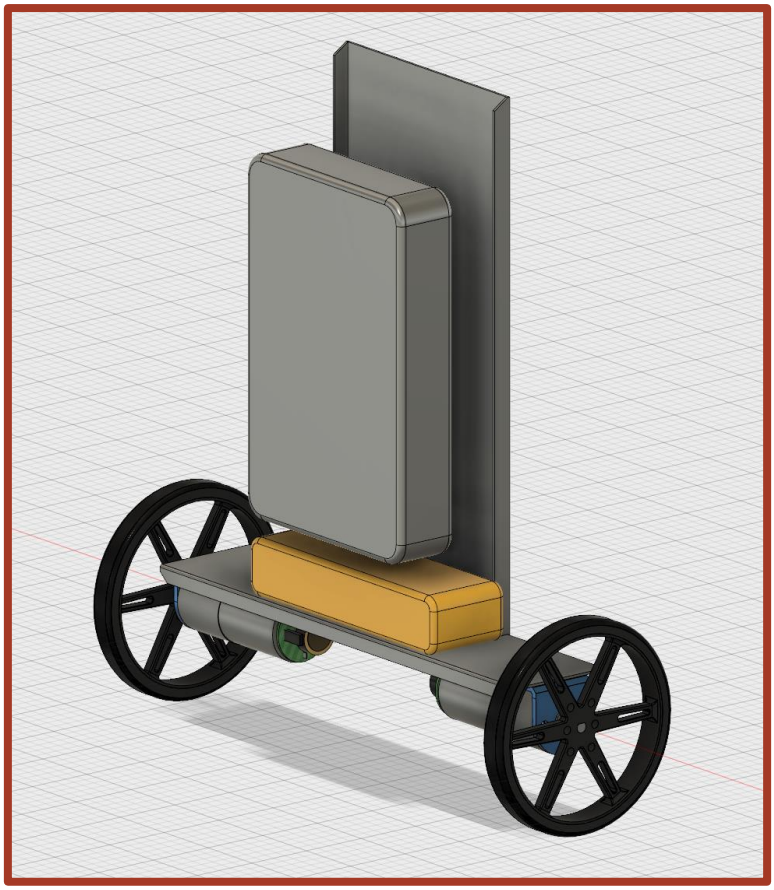


Figure 8: 3D model of complete assembly

5.2 Hardware: Results & Analysis

The frame was constructed and assembled by the technical engineering officer John Boulton according to the given design. The completed assembly can be seen in Figure 9, whereby wiring terminals were added for convenience and the signal and power wires are fed underneath the components in the gaps between the frame to limit the amount of wiring clutter. The weight distribution of components on the frame was very close to being symmetrical over the axis perpendicular with the ground so it was deemed no further adjustments of the NI myRIO positioning was necessary. The communications block diagram shown in Figure 10 details the completed arrangement of sensors and actuators and also illustrates the direction of information flow between peripherals and the controller.

Figure 9: Completed Physical Assembly

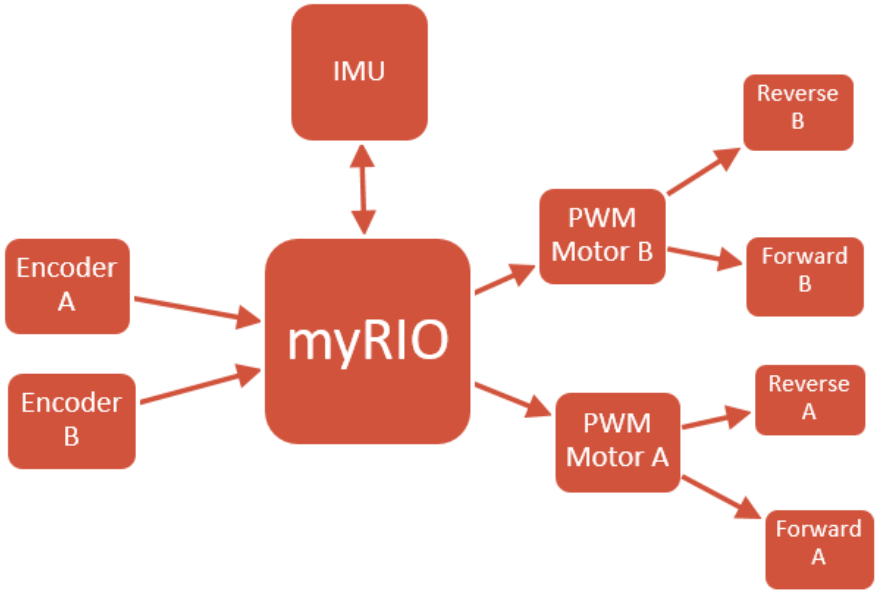


Figure 10: Communication Block Diagram

5.2.1 Hardware Testing

Before designing code for the balancing robot it was essential to test the operation of the hardware components crucial for the stability of the robot. This is carried out to ensure the hardware is not faulty and also to familiarize with the expected signals and overcome any communications issues. Figure 11 shows the generated PWM signals by the NI myRIO through the FPGA data registers, while details on the process for generating the PWM signals is covered in Section 5.1. Figure 12 shows the measured output channels for the quadrature encoders connected to each of the motors. Both PWM generation and measure of encoder channel signals were as expected.

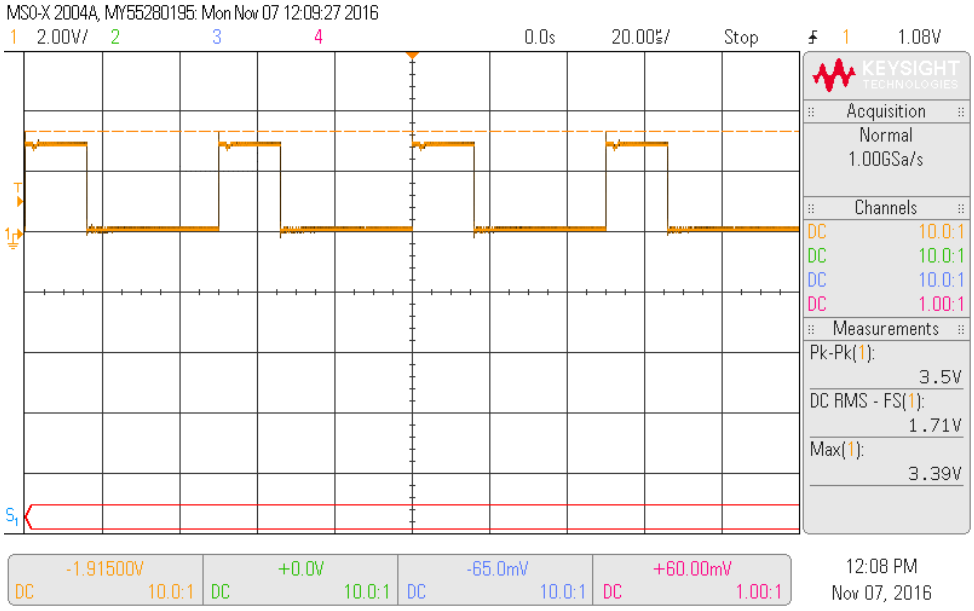


Figure 11: PWM Generation

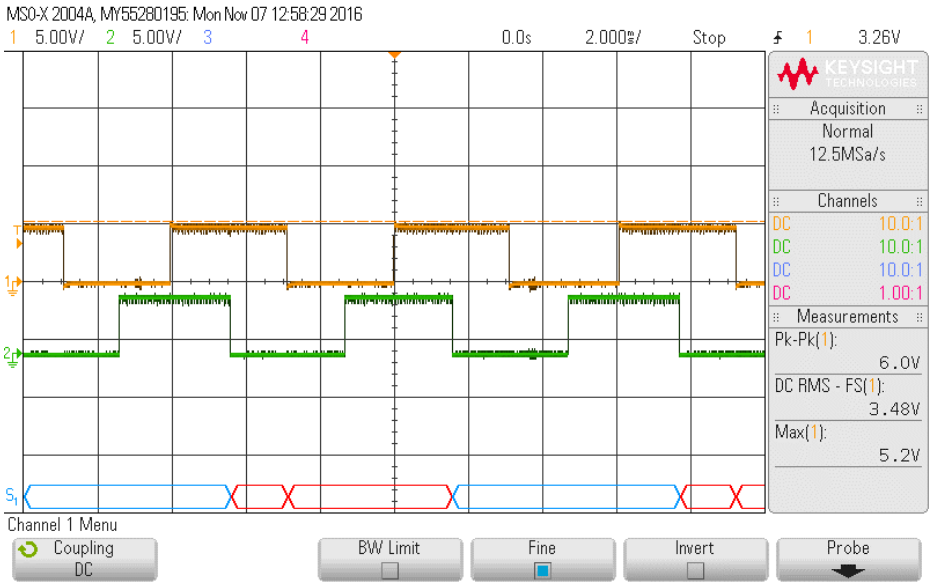


Figure 12: Encoder Output Signals

5.2.1.1 MOSFET Switching Noise

One of the requirements for this project is to have the NI myRIO powered by a battery source to eliminate tethered wires that could interfere with the control of the balancing. This is done as described in Section 4.1.4 with the use of a LiPo battery. However, the battery is also supplying the motor driver board in parallel with the myRIO. The motor driver board contains

MOSFETs that are being switched at high frequencies due to the PWM generation signals. As there is very little shielding on the robot the myRIO power input is susceptible to MOSFET switching noise. Figure 13 shows the magnitude of noise over the battery terminal during a 50% duty cycle PWM generation. The magnitude of the voltage ripple is 4.6 V. Depending on the level of filtering and the quality of the myRIO's onboard linear regulator this could prove to be problematic. Unfortunately, National Instruments do not release any detailed schematics or information surrounding the power input filtering circuits so it was impossible to determine whether this would cause issues without physically operating the device and observing the results. The myRIO did not perform abnormally supplied with the noisy input voltage, there was no noticeable change in the behavior of the controller. The myRIO did not experience any crashing or failures in data acquisition that were noticable.

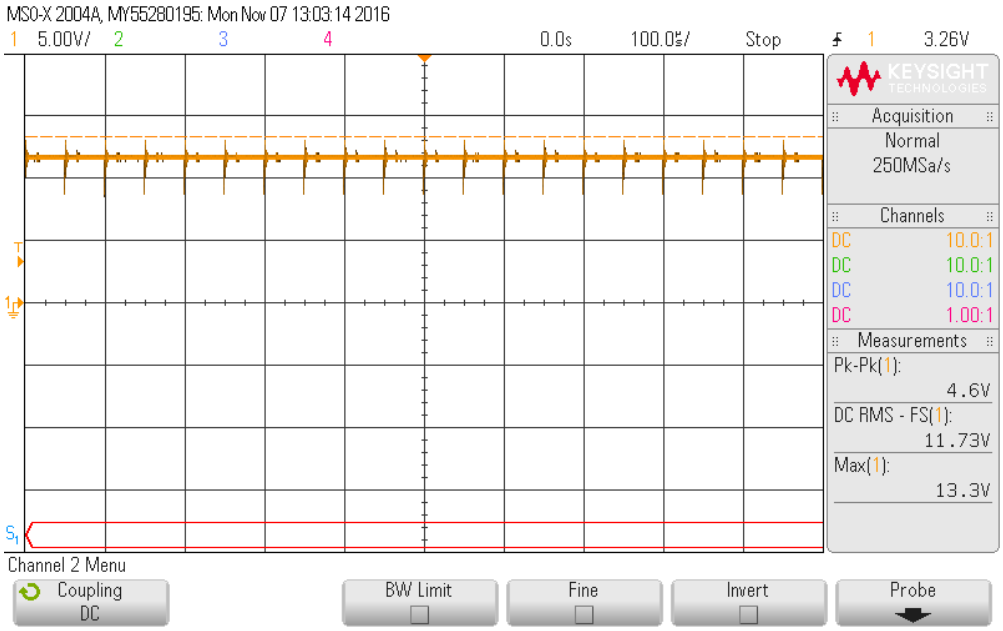


Figure 13: MOSFET Switching Noise on Battery Voltage

5.3 Hardware: Modifications

The initial orientation of the Adafruit BNO055 sensor on the frame of the balancing robot did not allow for measurements to be taken in the range the robot would be operating. With the sensor in its original orientation the angle read from the Euler angle 'roll' register would measure 90 degrees when the system was almost exactly vertical but any angle further than this point would not increase the value read in the register. This was the result of reaching the maximum point in the Euler roll measurable range as described in the device datasheet [26]. This issue could be easily rectified by repositioning the sensor in a different orientation with the required plane facing upward so the device can measure the corresponding 180 degree range in which the robot would be operating.

6 Software

This section of the report will describe steps taken to develop and test software for the robot's stabilization including the implementation of control loops and tuning techniques. This section will also detail the configuration of FPGA registers, data manipulation of feedback signals and the use of a local wireless network for testing, debugging and tuning.

6.1 Methodology

6.1.1 Host VI Operations

The programming in this project was done in LabVIEW using the myRIO FPGA project as a template for the host program or VI on the real-time processor. The FPGA shipping personality 4.0 was used as recommended by National Instruments as the FPGA personality [35]. Using the FPGA project over a standard myRIO projects enables the host VI on to have direct access to the FPGA registers on the FPGA module. All the communication excepting the I^2C protocol communication was done by directly reading and writing to the FPGA registers from the host VI.

The I^2C communications were handled using the I^2C Express VI. The express VI still fundamentally communicates through the FPGA registers in the same manner as the other communication carried out in the host VI program but using the express VI simplifies the necessary code to communicate with the Adafruit BNO055 IMU.

The Host VI also handles all the data manipulation required from the feedback sensors before being processed in the control loops and also handles the controller output data manipulation, transferring control output signals to generated PWM quantities. The Euler roll angle data read from the IMU is first read as two unsigned bytes in little-endian form (least significant byte first). In order to convert this information into useful information corresponding to the angle of orientation the host VI swaps the endian form, then joins the bytes to make a word then

applies the scaling factor stated in the BNO055 datasheet [26] of 1 degree = 16 least significant bits. The resolution of the angle in degrees is 0.0625°.

6.1.2 PWM Configuration

The PWM frequency was chosen based on the maximum human audible range 20kHz [36], so the switching of the MOSFETS would not be audible by the human ear. To achieve this switching frequency the following formula in (3) was taken from the FPGA shipping personality 4.0 [35].

$$f_{PWM} = \frac{f_{clk}}{N(X+1)} \tag{3}$$

Where f_{PWM} is the chosen PWM frequency, f_{clk} is the base 40MHz clock, N is the clock divider and X is the number of counts before the signal is changed. X is the value also written into the PWM.X.MAX FPGA register that sets the PWM frequency.

When a clock divider of 1 is used the count value X is solved for and written into the FPGA PWM.X.MAX count register. For a 20kHz PWM signal a count max value of 1999 is written. The value in the PWM.X.CMP register is the count compare value that corresponds to the duty cycle of the PWM waveform [35]. It is now expressed as a value between 0 and 1999.

6.1.3 Wireless Communications

An important requirement of this project was to set up a local wireless network so that the balancing robot could be operated wirelessly. This was mainly to ensure the control methods behind the robot stabilization was not hindered by the presence of any wires. Ensuring the system is wireless also meant that tuning the controllers would be much easier as the controls, indicators and valuable feedback information could be relayed through shared variables on the myRIO through a tablet with the LabVIEW dashboard app [21]. All wireless communications to and from the myRIO take place through the wireless router connected to the desktop and the onboard WiFi chip embedded within the myRIO.

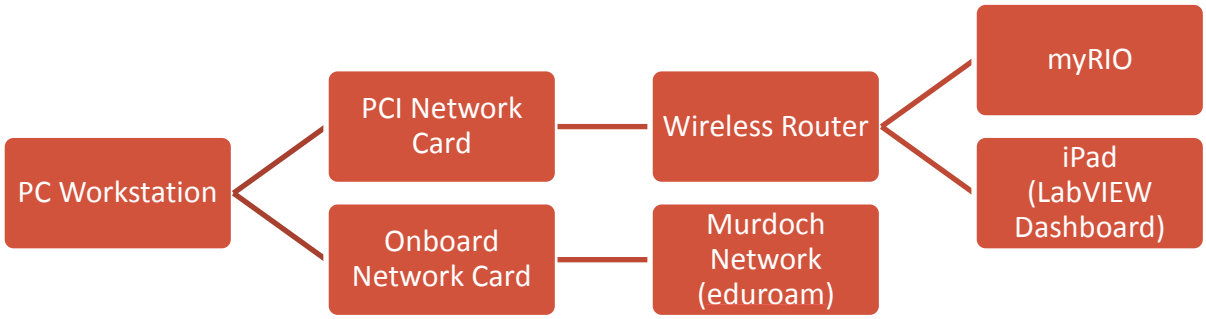


Figure 14: Network Block Diagram

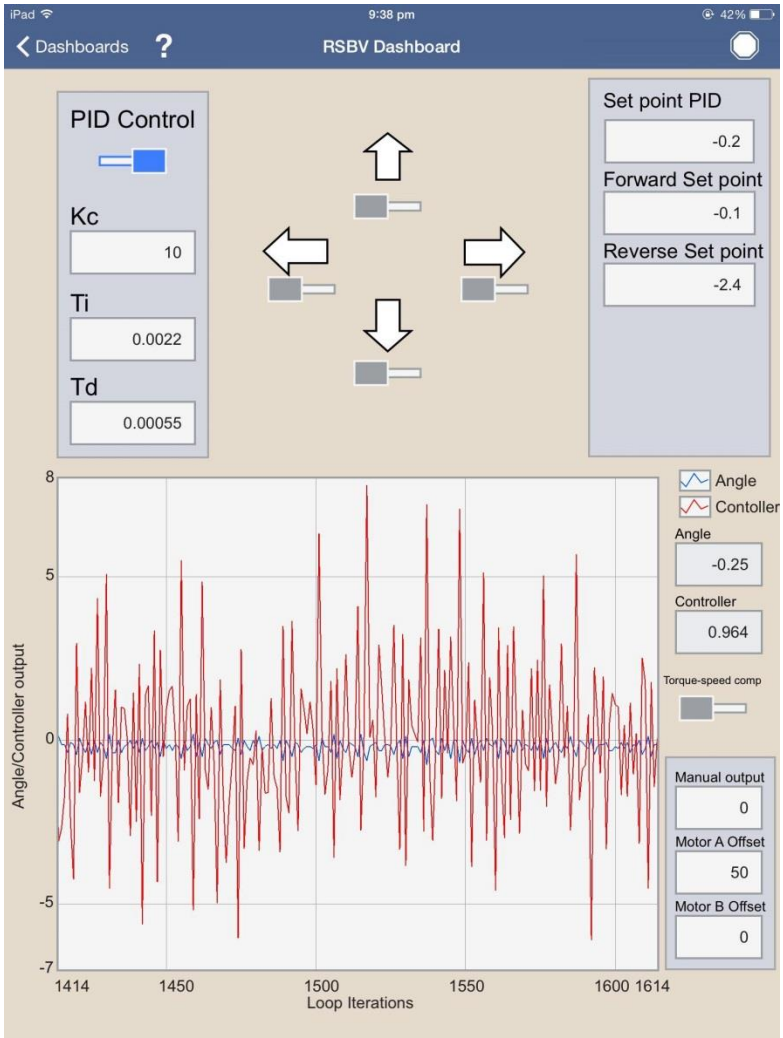


Figure 15: LabVIEW Dashboard of Self-Balancing Robot Parameters

6.1.4 PID Feedback Control

The stabilization of the robot is achieved by implementing a PID feedback control loop around the system. The process variable is the angle of tilt read by the IMU. The controller will

manipulate the motor speed and direction through the duty cycle of PWM generation handled in the host VI. Although this is a non-linear system, linear feedback control can still be used to stabilize the robot.

The PID feedback controller was then tuned using the Ziegler Nichols approach [37]. The gain was increased until the system was able to respond and began to oscillate, then the integral and derivative terms were chosen based on the Ziegler Nichols tuning rules [37]. The integral and derivative terms where fine-tuned a little further past this point, however the relationship of 4:1 (integral: derivative) was kept the same. The PID.VI block was used for the PID feedback control on the LabVIEW host VI as shown in Figure 16.

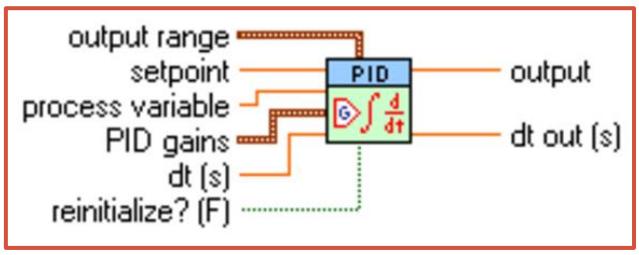


Figure 16: PID Feedback Control VI

6.2 Results & Analysis

With a single PID control loop around the tilting angle and the motor drive, and after some tuning of the PID parameters stability was finally achieved. This can be seen in Figure 17 which shows a histogram of the angles read from the IMU as the control loop operates to keep the robot stable. The histogram confirms that the IMU is not mounted perfectly level, with the average tilt angle around 0.5 degrees. However, this is to be expected due to the imperfect mounting of the IMU on the frame. The Histogram also confirms that the during this test, 100% of the values remained within 2 degrees (inclusive) of each other. The histogram is not normally distributed. Two spikes of high frequency data occurrences happen on either side of the mean. This is a result of the balancing robot taking time to change direction as the

controller works to resist increasing changes in the error between the set point and process variable.

As the battery voltage decreases during operation the torque and speed of the motors suffer the same consequence. This means that initially when the battery is at full charge the controllers act more aggressively. This can be overcome by implementing battery voltage feedback over the battery terminal voltage and scaling the controller outputs proportionally.

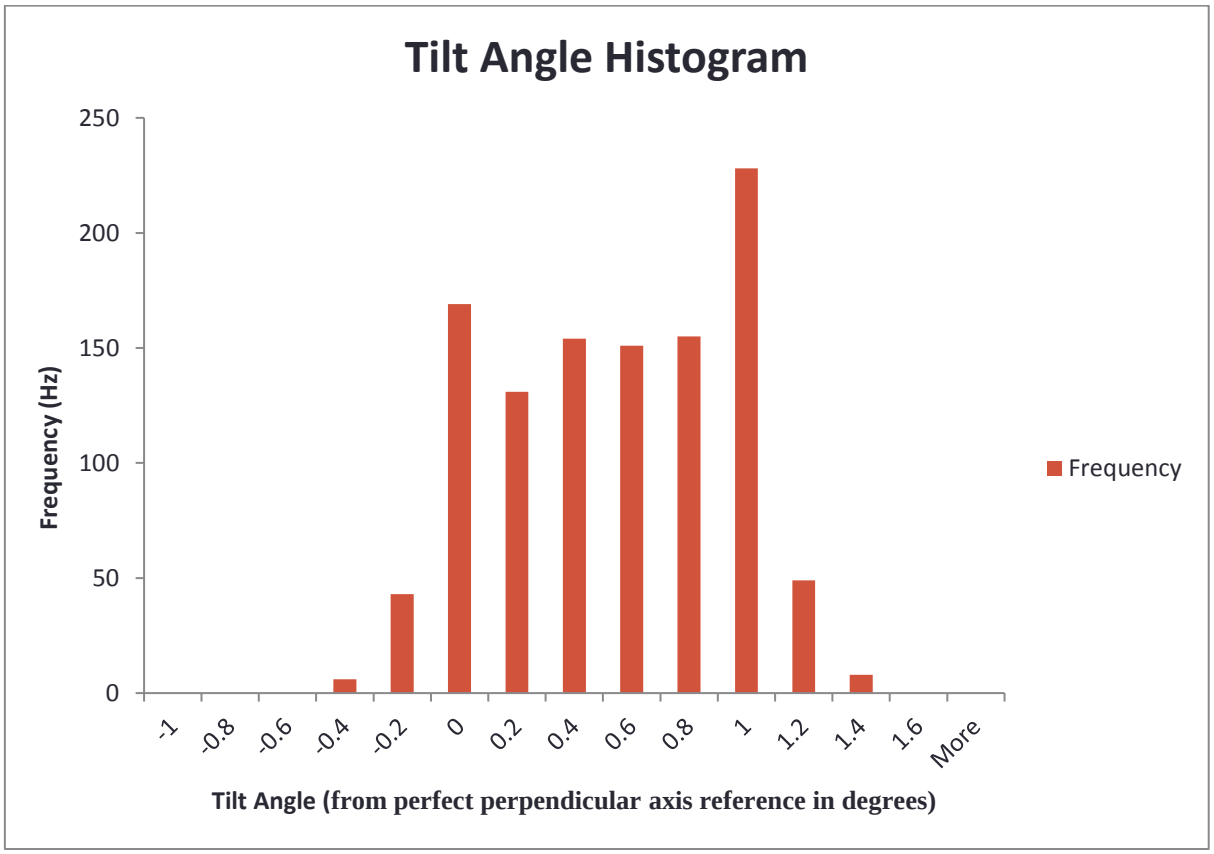


Figure 17: Tilt Angle Histogram

6.3 Modifications

6.3.1 Cascade PID Control

When a large enough disturbance was given to the system the feedback controller began to struggle to keep the process variable at the set point. A drifting behavior became apparent,

whereby the feedback controller could not keep the process variable constant for very long. Sometimes this would lead to complete instability as the drifting increased and the error became too large for the system to respond and the controller output became saturated.

This problem was overcome by introducing a cascade PID feedback loop around the encoder count value and the angle set point of the motor driving controller. The gain of this PID feedback loop was set 100 times smaller than the main PID feedback loop. This was done to ensure that the second cascade PID loop reacts much slower than the loop directly responsible for stabilization. Besides fixing the drifting issue, a major benefit of this new cascade PID loop is absolute position control. The balancing robot would now return to its initial position in which it began before the introduction of the disturbance. Hence in Figure 18 as the system returns to its initial position after a large disturbance.

6.3.2 Maneuvering

With the set point of the cascade PID loop the count value of the encoder, maneuvering the robot forward and back became an easy task. One revolution of encoder counts (~3600 counts) added to or subtracted from the set point would cause the robot to either move forward or backward successfully. Rotating the robot was done by releasing control and driving the motors in reverse directions for a set number of program loop iterations before returning control to stabilize any errors developed in the rotation. For larger rotations this process can just be repeated.

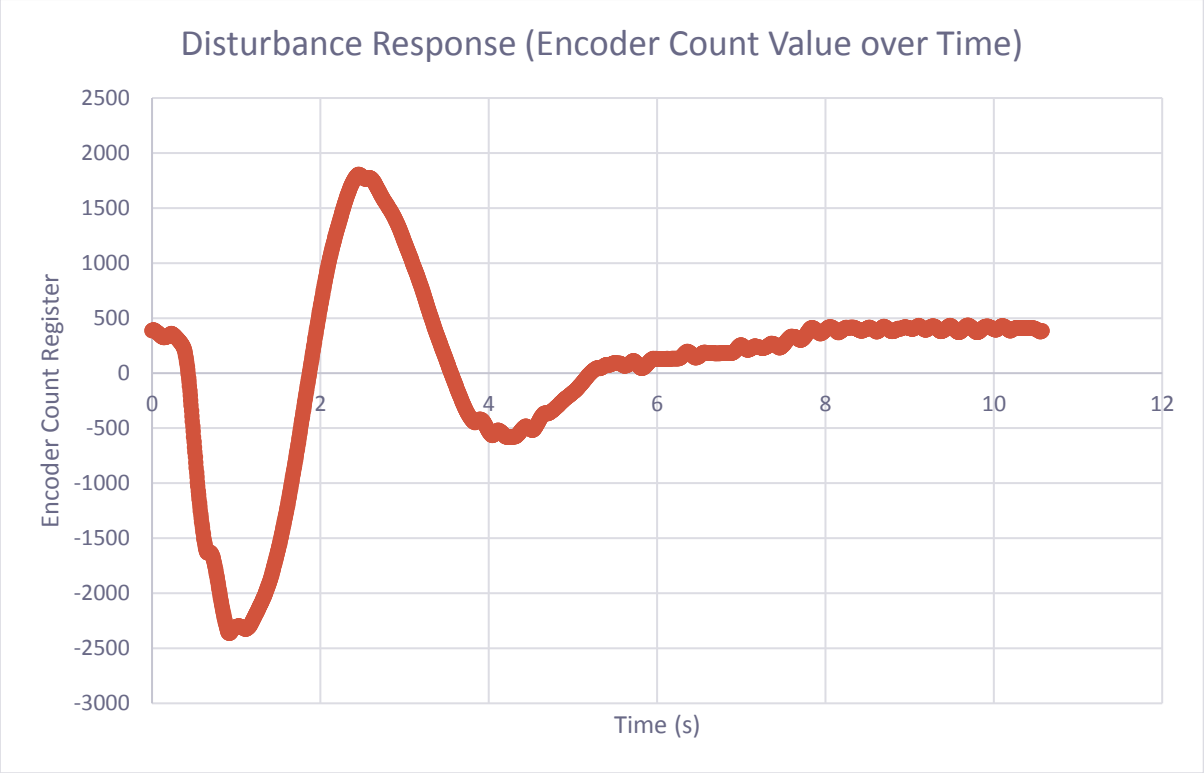


Figure 18: Disturbance Response (Encoder Count vs Time)

7 Summary

The objective of this project was to develop a self-balancing robot utilizing the National Instruments myRIO embedded device, it proved to be a challenging research and development project that ultimately prevailed and met the primary objective. The outcome of producing a battery powered self-balancing robot that can be controlled and tuned wirelessly, and can recover from fairly large disturbances introduced to the system. The robot has demonstrated the ability to return back to its absolute position when recovering from disturbances due to the cascade PID feedback controller's encoder set point. The stability of the system can be considered as quite good, under normal operation the full range of angular motion is within 2 degrees.

8 Future Works

8.1 Hardware Additions

A Microsoft LifeCam HD-3000 Webcam was purchased for this project as an additional piece of hardware that could provide the robot with surveillance capabilities. The robot was constructed with a second, currently unused motor driver board mounted to the frame. Combining the Webcam with a stepper motor could provide this project with some interesting possibilities such as image stabilization with a control loop around the tilt angle and stepper motor to ensure stable streaming of video imaging.

Another important addition to this project would be the addition of battery voltage feedback. This would ensure that the behavior of the controllers is not effected by the battery terminal voltage. To add this functionality a resistor divider circuit could be used to step the battery voltage down to the analog input range (+/- 10 V).

8.2 Hardware Modifications

The system was over designed around the required motor torque, so increasing the diameter of the wheels used to drive the robot would increase the system's ability to respond to larger disturbances and also increase its maximum speed.

8.3 Modelling and Model Based Control

An area that was not focused on in this project was modelling of the system. Modelling a system like this and implementing a model based control strategy would require a substantial amount of extra time. Although that was the original goal, it was soon realized that one semester was not enough time to design, construct and model the system.

9 Bibliography

[1] Segway, “Technology,” Segway, INC, 18 November 2015. [Online]. Available: http://www.segway.ch/en/infos/technologie.php. [Accessed 25 October 2016].

[2] TIME, “Reinventing the Wheel,” 2 December 2001.

[3] T. Kailath, “Linear Systems,” Prentice-Hall, Englewood Cliffs, 1980.

[4] J. Lam, “Control of an Inverted Pendulum,” Institut fur Automatik- Automatic Control Laboratory, Zurich, 2011.

[5] K. Sultan, “Inverted Pendulum Analysis, Design and Implementation,” Institute of Industrial Electronics Engineering, Karachi, 2003.

[6] Segway, “Segway- Our story so far,” Segway, INC, 18 November 2015. [Online]. Available: http://www.segway.com/about/our-story. [Accessed 25 October 2016].

[7] IO Hawk, “A New way to Move,” IO Hawk, 2016. [Online]. Available: http://iohawk.com/. [Accessed 25 October 2016].

[8] National Instruments, “Quadrature Encoder Fundamentals,” National Instruments, 14 March 2016. [Online]. Available: http://www.ni.com/white-paper/4763/en/. [Accessed 25 October 2016].

[9] Y. Qiu, “Basics of Hall Effect,” 27 April 1997. [Online]. Available: http://www.pha.jhu.edu/~qiuym/qhe/node2.html. [Accessed 25 October 2016].

[10] E. Eitel, “Basics of Rotary Encoders: Overview and New Technologies,” Machine Design, 7 May 2014. [Online]. Available: http://machinedesign.com/sensors/basics-rotary-encoders-overview-and-new-technologies-0. [Accessed 25 October 2016].

[11] J. Keller, “Army chooses rugged accelerometers from Meggitt for testing the Excalibur smart artillery shell,” Military Aerospace Electronics, 19 October 2016. [Online]. Available: http://www.militaryaerospace.com/articles/2016/10/accelerometers-smart-artillery-testing.html. [Accessed 26 October 2016].

[12] C. Woodford, “Accelerometers,” Explainthatstuff!, 8 June 2016. [Online]. Available: http://www.explainthatstuff.com/accelerometers.html. [Accessed 24 September 2016].

[13] Epson , “Gyro sensors- How they work and what's ahead,” Epson, 2016. [Online]. Available: http://www5.epsondevice.com/en/information/technical_info/gyro/. [Accessed 20 October 2016].

[14] A. Ronzo, “What is a Gyroscope,” Sparkfun, 2015. [Online]. Available:

https://learn.sparkfun.com/tutorials/gyroscope. [Accessed 25 October 2016].

[15] E. W. Weisstein, “Euler Angles,” Mathworld-A Wolfram Web Resource, 2016. [Online]. Available: http://mathworld.wolfram.com/EulerAngles.html. [Accessed 1 November 2016].

[16] K. Townsend, “Adafruit BNO055 Absolute Orientation Sensor,” Adafruit, 23 11 2015. [Online]. Available: https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor/overview. [Accessed 10 09 2016].

[17] Pololu, “Continous-rational servos and multi-turn servos,” Pololu, 11 July 2011. [Online]. Available: https://www.pololu.com/blog/24/continuous-rotation-servos-and-multi-turn-servos. [Accessed 20 October 2016].

[18] B. Earl, “Driving a Stepper,” Adafruit, 2016. [Online]. Available: https://learn.adafruit.com/all-about-stepper-motors/driving-a-stepper. [Accessed 20 September 2016].

[19] M. Barr, “Introduction to Pulse Width Modulation (PWM),” Barr group, 1 September 2001. [Online]. Available: http://www.barrgroup.com/Embedded-Systems/How-To/PWM-Pulse-Width-Modulation. [Accessed 5 Novermber 2016].

[20] G. Lakkas, “MOSFET power Losses and how they affect power-supply efficieny,” *Analog Applications Journal,* pp. 22-26, 2016.

[21] N. Instruments, “NI Product Manuals,” [Online]. Available: http://www.ni.com/manuals/. [Accessed 10 August 2016].

[22] Nation Instruments Corporation, “LabVIEW FPGA Module User Manual,” Mar 2004. [Online]. Available: http://www.ni.com/pdf/manuals/370690b.pdf. [Accessed 20 09 2016].

[23] James Press, “Xilinx, Inc. History,” Funding Universe, 1997. [Online]. Available: http://www.fundinguniverse.com/company-histories/xilinx-inc-history/. [Accessed 28 October 2016].

[24] Cirelis, “SPI Interface,” EWA, 2016. [Online]. Available: http://www.corelis.com/education/SPI_Tutorial.htm. [Accessed 22 October 2016].

[25] N. Semiconductors, “I2C-bus specification and user manual,” NXP Semiconductors, 2014.

[26] Bosch Sensortec GmbH, “Data sheet- BNO055 Intelligent 9-axis absolute orientation sensor,” Bosch Sensortec GmbH, Reutlingen/ Germany, 2015.

[27] E. Doering, “NI myRIO Project Essentials Guide,” National Instruments, 2011.

[28] TechBrands by Electus Distribution Pty. Ltd - Jaycar, “duinotech Dual/Stepper Motor Controller Module,” [Online]. Available: http://www.jaycar.com.au/medias/sys_master/images/h1a/h90/8872261091358/XC4492-dataSheetMain.pdf. [Accessed 10 09 2016].

[29] All Things RC, “Vant Battery Lipo 2250mAh 11.1V 3S 30C (21x35x105 @ 170g),” [Online]. Available: http://www.allthingsrc.com.au/max-power-lipo-2250mah-30c-3s-11.1v-21x35x105. [Accessed 20 09 2016].

[30] Robot Gear Australia, “75:1 Metal Gearmotor 25Dx54L mm MP 12V with 48 CPR Encoder,” Pololu, [Online]. Available: https://www.robotgear.com.au/Product.aspx/Details/4664-75-1-Metal-Gearmotor-25Dx54L-mm-MP-12V-with-48-CPR-Encoder. [Accessed 10 09 2016].

[31] Robot Gear Australia, “Drive Motor Sizing Tool,” Pololu, 7 March 2013. [Online]. Available: http://www.robotshop.com/blog/en/drive-motor-sizing-tool-9698. [Accessed 10 September 2016].

[32] Battery University, “What is the best battery?,” Battery University, 1 11 2010. [Online]. Available: http://batteryuniversity.com/learn/archive/whats_the_best_battery. [Accessed 20 09 2016].

[33] STMicroelectronics, “L298 DUAL FULL-BRIDGE DRIVER,” Jan 2000. [Online]. Available: http://www.st.com/content/ccc/resource/technical/document/datasheet/82/cc/3f/39/0a/29/4d/f0/CD00000240.pdf/files/CD00000240.pdf/jcr:content/translations/en.CD00000240.pdf. [Accessed 1 09 2016].

[34] Autodesk, *Fusion 360,* Autodesk.

[35] National Instruments, “myRIO Shipping Personality 4.0,” 2016. [Online]. Available: ftp://ftp.ni.com/pub/devzone/tut/myrio_personality_40.pdf. [Accessed 15 September 2016].

[36] G. Elert, “Frequency Range of Human Hearing,” 2004. [Online]. Available: http://hypertextbook.com/facts/2003/ChrisDAmbrose.shtml. [Accessed 25 October 2016].

[37] W. H. R. Babatunde A. Ogunnaike, Process Dynamics, Modeling and Control, New York: Oxford University Press, Inc., 1994.

[38] Microsoft, “LifeCam HD-3000,” 2016. [Online]. Available: https://www.microsoft.com/accessories/en-au/products/webcams/lifecam-hd-3000/t3h-00014. [Accessed 1 09 2016].

[39] S. M. I. Shui-Chun Lin and Ching-Chih Tsai, “Development of a Self-Balancing Human Transportation Vehicle for the Teaching of Feedback Control,” IEEE TRANSACTIONS ON

EDUCATION, 2009.

[40] National Instruments Corporation, “NI LabVIEW High-Performance FPGA Developer’s Guide,” National Insturments, 2014.

10 Appendices

10.1 L298 Motor Driver Datasheet

L298

DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

Multiwatt15

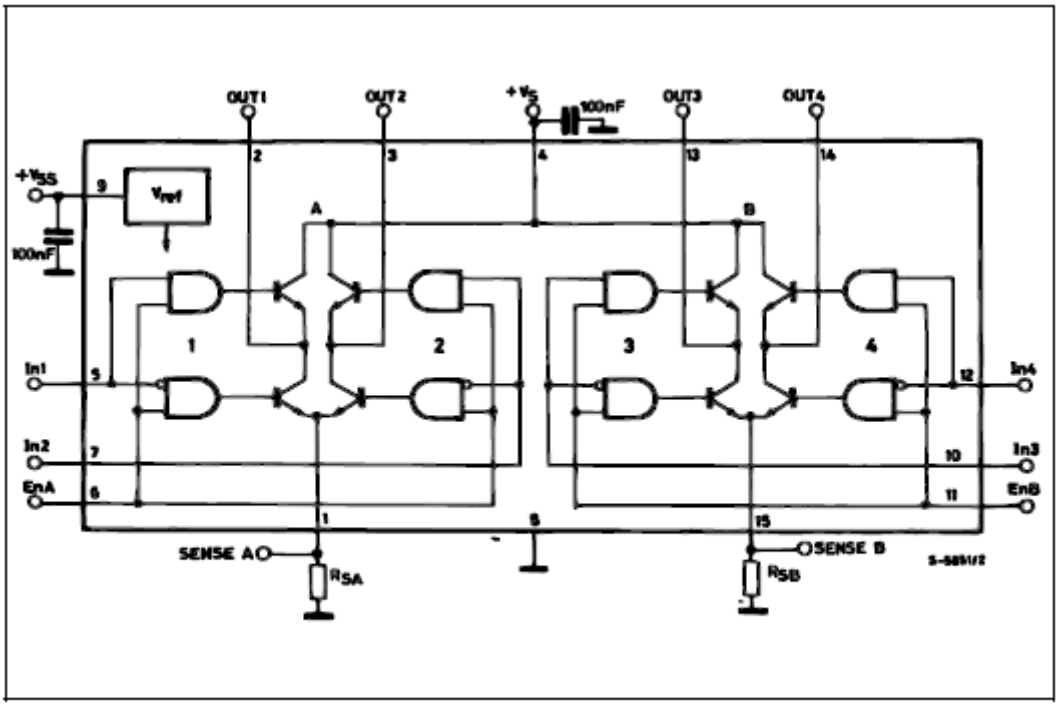
PowerSO20

ORDERING NUMBERS : L298N (Multiwatt Vert.)
L298HN (Multiwatt Horiz.)
L298P (PowerSO20)

DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

BLOCK DIAGRAM



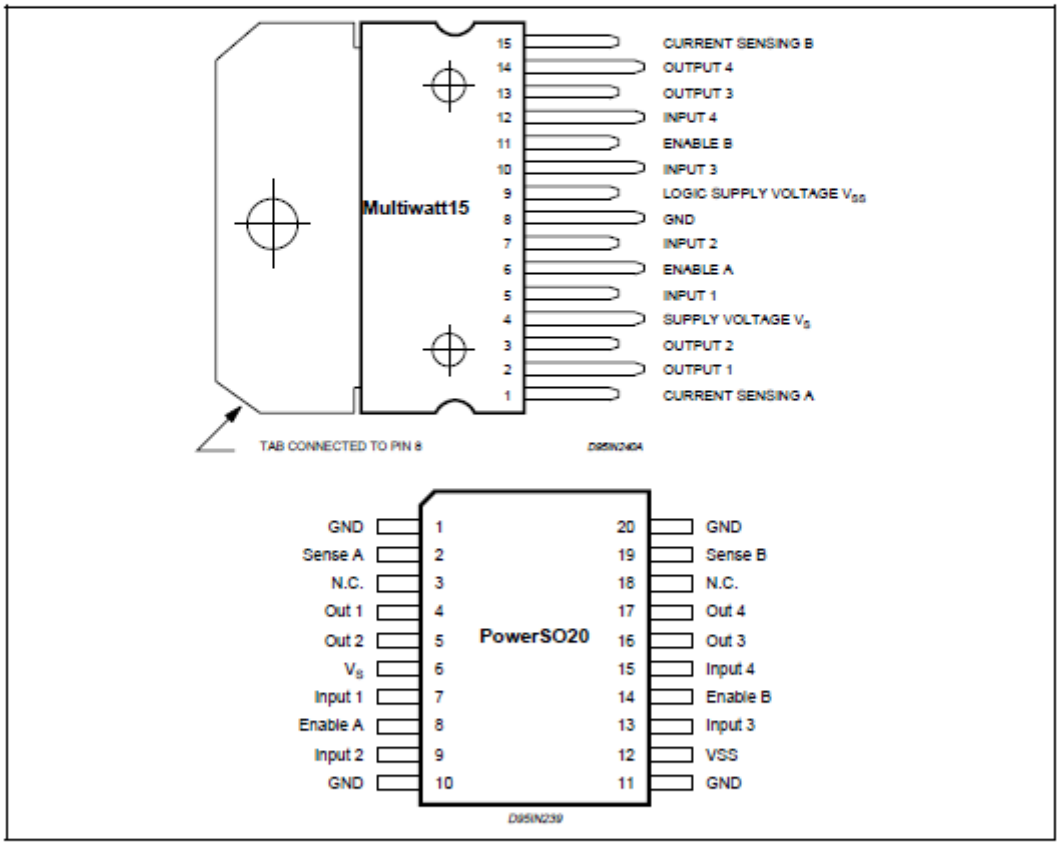
January 2000 1/13

L298

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_S	Power Supply	50	V
V_{SS}	Logic Supply Voltage	7	V
V_i,V_{en}	Input and Enable Voltage	–0.3 to 7	V
I_O	Peak Output Current (each Channel) – Non Repetitive (t = 100µs) –Repetitive (80% on –20% off; t_{on} = 10ms) –DC Operation	 3 2.5 2	 A A A
V_{sens}	Sensing Voltage	–1 to 2.3	V
P_{tot}	Total Power Dissipation (T_{case} = 75°C)	25	W
T_{op}	Junction Operating Temperature	–25 to 130	°C
T_{stg}, T_j	Storage and Junction Temperature	–40 to 150	°C

PIN CONNECTIONS (top view)



THERMAL DATA

Symbol	Parameter		PowerSO20	Multiwatt15	Unit
$R_{th\ j\text{-}case}$	Thermal Resistance Junction-case	Max.	–	3	°C/W
$R_{th\ j\text{-}amb}$	Thermal Resistance Junction-ambient	Max.	13 (*)	35	°C/W

(*) Mounted on aluminum substrate

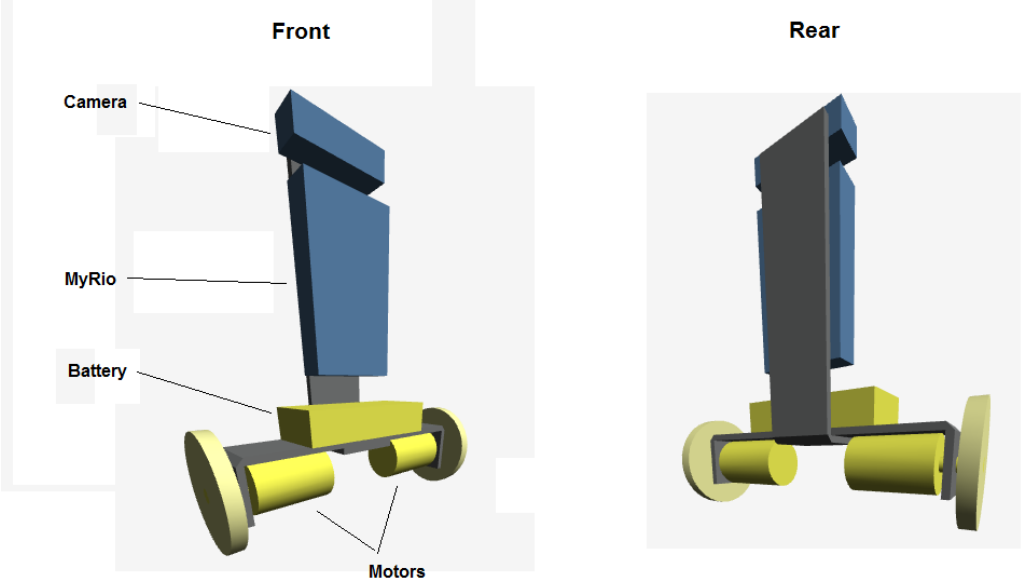
PIN FUNCTIONS (refer to the block diagram)

MW.15	PowerSO	Name	Function
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V_S	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1,10,11,20	GND	Ground.
9	12	VSS	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground.
10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
–	3;18	N.C.	Not Connected

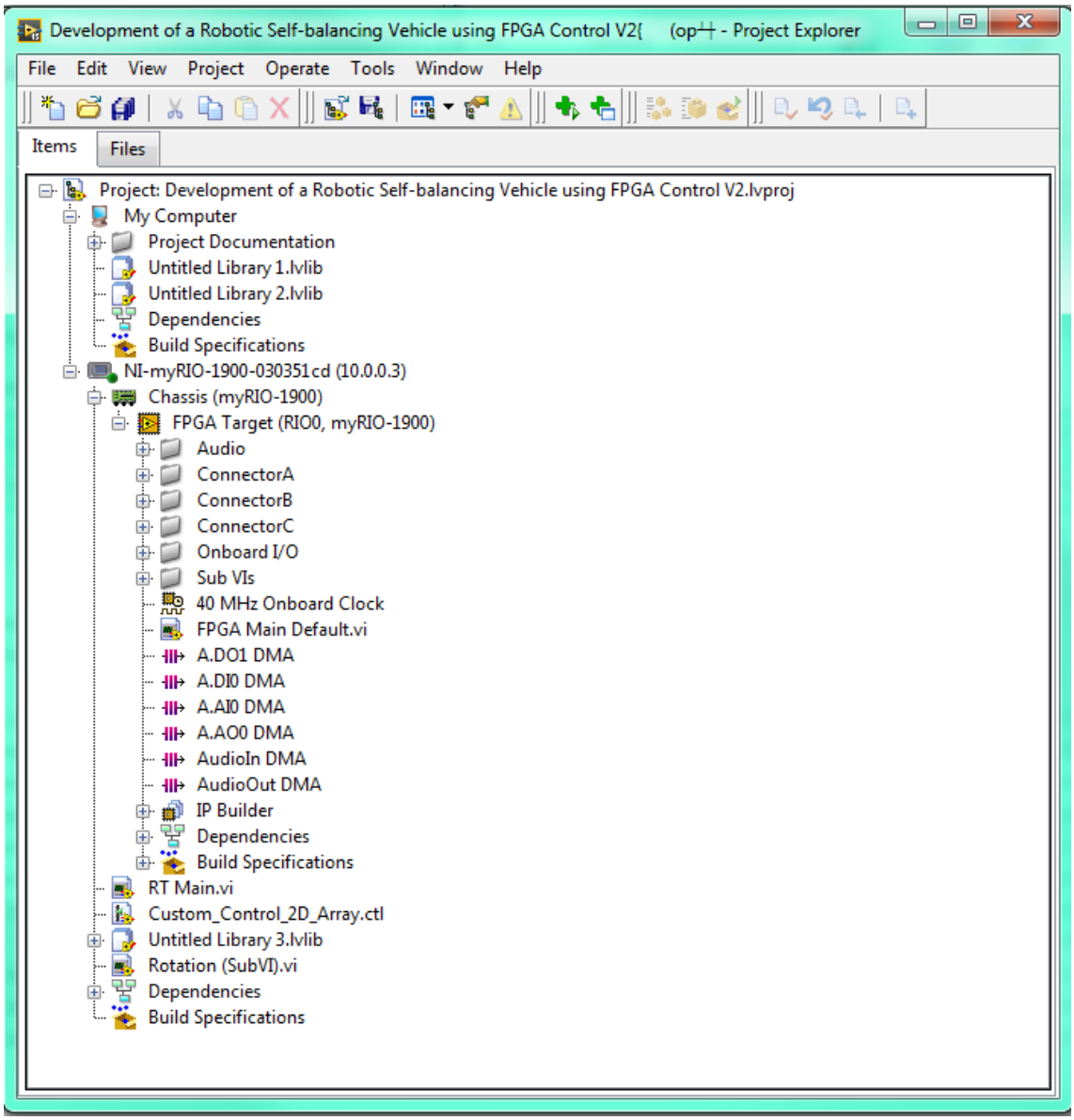
ELECTRICAL CHARACTERISTICS (V_S = 42V; V_{SS} = 5V, T_j = 25°C; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_S	Supply Voltage (pin 4)	Operative Condition	V_{IH} +2.5		46	V
V_{SS}	Logic Supply Voltage (pin 9)		4.5	5	7	V
I_S	Quiescent Supply Current (pin 4)	V_{en} = H; I_L = 0 V_i = L V_i = H		13 50	22 70	mA mA
		V_{en} = L V_i = X			4	mA
I_{SS}	Quiescent Current from V_{SS} (pin 9)	V_{en} = H; I_L = 0 V_i = L V_i = H		24 7	36 12	mA mA
		V_{en} = L V_i = X			6	mA
V_{IL}	Input Low Voltage (pins 5, 7, 10, 12)		–0.3		1.5	V
V_{IH}	Input High Voltage (pins 5, 7, 10, 12)		2.3		VSS	V
I_{IL}	Low Voltage Input Current (pins 5, 7, 10, 12)	V_i = L			–10	µA
I_{IH}	High Voltage Input Current (pins 5, 7, 10, 12)	Vi = H ≤ V_{SS} –0.6V		30	100	µA
V_{en} = L	Enable Low Voltage (pins 6, 11)		–0.3		1.5	V
V_{en} = H	Enable High Voltage (pins 6, 11)		2.3		V_{SS}	V
I_{en} = L	Low Voltage Enable Current (pins 6, 11)	V_{en} = L			–10	µA
I_{en} = H	High Voltage Enable Current (pins 6, 11)	V_{en} = H ≤ V_{SS} –0.6V		30	100	µA
$V_{CEsat\ (H)}$	Source Saturation Voltage	I_L = 1A I_L = 2A	0.95	1.35 2	1.7 2.7	V V
$V_{CEsat\ (L)}$	Sink Saturation Voltage	I_L = 1A (5) I_L = 2A (5)	0.85	1.2 1.7	1.6 2.3	V V
V_{CEsat}	Total Drop	I_L = 1A (5) I_L = 2A (5)	1.80		3.2 4.9	V V
V_{sens}	Sensing Voltage (pins 1, 15)		–1 (1)		2	V

10.2 First Concept/Proposal



10.3 LabVIEW FPGA Project



10.3.1 Additional Information

All LabVIEW Code, datasheets, related documents and information regarding this project will be placed on a USB and left plugged directly into the USB port on the myRIO. The LabVIEW block diagram is too big to be placed within the appendices, for access to this code, refer to the project USB.