Automated Foosball Table
Final Report

Sponsored by

YASKAWA

Supported by

Warrior Table Soccer

COGNEX
vision

Prepared by:

Jim Stefani	jstefani@calpoly.edu
Alex Herpy	aherpy@calpoly.edu
Brett Jaeger	bjaeger@calpoly.edu
Kevin Haydon	khaydon@calpoly.edu
Derek Hamel	Dhamel@calpoly.edu

Advisor
Professor Lee McFarland mlmcfarl@calpoly.edu

California Polytechnic State University
San Luis Obispo, CA
2013

Executive Summary

This project is the second iteration of an automated foosball table for Yaskawa America as a tradeshow display. The table is meant to provide an interactive experience which highlights the speed and precision of the Yaskawa hardware. The first iteration of the project was mainly focused on creating the physical hardware for the system and to begin the basic programming for the system. This phase of the project was focused on finalizing the physical hardware of the system, implementing the vision system and to continue the basic programing of the system AI. A third team will be assigned to bring the project to completion by fully implementing the AI system and making any required changes to the physical hardware which are required.

The automated Foosball system is comprised of two major system elements. The first element is the motor cabinet, which houses the PLC, motors and amplifiers used to actuate the system. It also acts as a display case for the motors system. The other major element is the foosball table itself, which is comprised of several subsystem components. The foosball table system contains a vision arch which houses the vision system, a playfield cover which prevents users from injury, and a roof which blocks direct lighting on the table.

Several hardware components were created or modified during this phase of the project. The roof structure was designed and built complete this quarter, as were brackets which connected the motor cabinet and foosball table. A gap cover was also designed and built to cover an exposed portion of the motor cabinet. While not fully completed, the hardware used in the safety system has been begun and should be completed by the future team. The scoring system for the table was also approached during this phase of the project, and it was concluded that the current scoreboard should be redesigned.

The original vision system started by the first team was found to be insufficient to meet the requirements of the foosball system. To simplify the process of creating the vision system, a Cognex Insight 7400 camera system was donate d to the project by Cognex. This camera system was found to be sufficient to meet the minimum requirements of the project with relatively little work. Future teams should focus on improving the frame rate of the vision system.

The AI program developed during this phase is working and playable, though it is relatively crude. Future iterations of the AI program should use sequential function charts to organize the program and predictive play should be implemented. More sophisticated play strategies can also be implemented to improve the playability of the system.

Contents

Table of Figures

Table of Tables

Chapter 1: Introductory Material

Project Motivation

Yaskawa America wants to attract people to their products at trade shows by using an interactive display that demonstrates the capabilities of Yaskawa servo motors and controllers. Yaskawa has requested Cal Poly students to build an automated foosball table to act as this display. This foosball table will use Yaskawa motors and servo controllers to automate one side of a foosball table so a person can play against a computer. Yaskawa will benefit from the completion of this project by having a display that is both exciting and informative because it will show potential customers the speed and accuracy of its servo motors and controllers while entertaining them at the same time. The people who we expect to interact with the table include: customers at the tradeshows who will be playing with the table, the technicians in charge of putting the table together and transporting the table, as well as people at Cal Poly Open Campus events where the table (the one left at Cal Poly) will be displayed.

Problem Definition

The final goal for this project is to produce two tables (one for Yaskawa, one for Cal Poly) that have an automated side with two degrees of freedom per rod (sliding and rotating). Yaskawa motors and controllers will be used in the automation of the table. An algorithm will be created using IEC 61131-3 programming languages working with the vision system that was created by the previous group working on this project to compete against a basic foosball player. This algorithm will attempt to block the opponent’s shots and also kick the ball towards the opposing goal.

Objectives

The goal of this project is to finish the automated foosball table begun by team Foos-Ro-Dah and to develop an AI for the table. To generate a list of objectives for the project, a quality function deployment (QFD) chart was generated, which can be found in Appendix A. The QFD compares customers’ requirements with engineering requirements which will be tested upon completion to determine if the project was successful. The customer requirements were given weights based on the overall importance of the requirement to the customer and the end user. The engineering requirements were then related to the customer requirements to determine which engineering requirements were critical to the success of the project. Tables created by other universities will be used as a benchmark to compare how they met both the customers’ requirements and the engineering requirements based on publications of the designs. This information was also used to generate engineering requirement targets for the project. Page two of Appendix A contains a comparison of the engineering requirements with themselves to determine if a correlation exists between them. Table 2 contains a list of engineering requirements and following it is a detailed description of the requirements.

Table 1: List of Engineering Requirements

Spec #	Parameter Description	Requirement or Target	Tolerance	Risk	Compliance
1	Goal Sensor	yes	Min	M	
2	LCD Menu	yes	Min	L	
3	% of Inner workings visible	80%	Max	L	I, S
4	System response time	25ms	±5ms	M	A, T, S
5	Power delivered to ball (or motor torque)	15N-m	±2N-m	M	A, T
6	Time to assemble from base components	180min	Max	L	I
7	Vibrations experienced during operation	low	Max	M	A, T
8	System sensing of the ball in motion	10m/s	±1m/s	H	A, T
9	No direct contact between player and moving parts	yes	Min	M	I
10	Reliability of the mechanical system	99.99%	Max	L	A
11	Measurements of the total space required for operation	1.5x1.5 m^2	±.25 x .25 m^2	L	I, T
12	Smooth, Variable Movement Speed	2-10 m/s	±1m/s	H	T, I
13	New player or user learning time	30sec	±10sec	L	I, T
14	Aesthetic Assessment Scale 1-10 (Sponsor)	7	Min	M	I, S
15	Aiming (hitting ball in direction of goal)	±30 degrees	Max	H	T
16	Cost analysis (our target does not include donations)	$5,000	Max	L	A
17	Fatigue Analysis (Unable to test)	100 hours	Min	M	A
18	Weight < 250lb for cabinet	< 250 Lbs	Max	M	T
19	Tune the motors	Yes	Min	M	T, A
20	Confirm motor size	Yes	Min	M	T
21	Create Modular Function Library Using IEC 61131-3 Languages	Yes	Min	H	T, I
22	Basic AI difficulty	Yes	Min	H	T, I
23	Normal AI difficulty	Yes	Min	L	T, I
24	Advanced AI difficulty	Yes	Min	L	T, I
25	Transportable by 2 people	Yes	Max	L	I

Risk: H = High
M = Medium
L = Low

Compliance: A = Analysis
T = Testing
I = Inspection
S = Similarity to Existing Designs

Below is a more in depth explanation of each requirement:

- A sensor used to detect if a goal has been scored and to send a signal to the display.
- An LCD display used to allow users to select difficulty and for technicians to run diagnostics from.
- Percentage of moving parts visible is a requirement for this project because the purpose of the end product will be to demonstrate the performance of the motors and how well they work with the system.
- The response time of the system is directly to demonstrating high performance of the motors and how well they work with the sensing equipment.
- The power delivered to the ball is important to select the appropriate equipment that will match and exceed human capabilities.
- The time to assemble is a requirement due to the end product needing to be transported.
- Vibrations on the product should be low in order to increase system life and overall quality to the player. Also effects the accuracy of the vision system.
- The system must be capable of sensing the ball at a high velocity in order to demonstrate high system performance. This requirement is critical as it will be one of the more difficult to setup and develop and the end product will be completely inoperable without it. The goal was set to the high end of testing performed by othergroups with similar projects, specifically the Kiro [9] and Eindhoven University projects [7].
- Due to safety of those using and near the product being of the utmost importance there must not be any contact between the users and the moving parts of the product.
- The mechanical system must be reliable in order to be of quality and have consistent performance.
- The total size of the project is important as transportation and storage as well as space allocated at trade shows may be limited. The requirement listed is our current best guess and may be subject to change as the project requires.
- The motors must be able to move smoothly and consistently for aesthetic value and to reduce the vibrations in the table.
- New player learning time should be as low as it takes for a player to learn to use a normal foosball table.
- Aesthetic assessment of the product should be done by Yaskawa to ensure trade show quality presentation.
- Aiming the ball is important to ensure the machine will correctly engage and impress the player. This will be a very hard requirement to meet because it requires the motors and the

sensing equipment to work together with a small error margin. This will likely require a lot of efficient programming. Our goal is to obtain a working prototype that can be improved on and fine-tuned at a later date, thus our requirement has a large range of ±30 degrees.

- The cost of the product should be as low as possible but quality is the most important. The $5000 listed does not include the donated motors, actuators, and controllers from Yaskawa and currently does not reflect the cost of a vision system.
- Fatigue of the system is not expected to be a very significant problem but we will still perform analysis to ensure that the table will at the very least remain intact for 100 hours of operation. Since testing this would require running to failure we will only be performing theoretical analysis not actual tests.
- The weight of the components should not be excessive to make transportation easier.
- The motors must be tuned to ensure they perform at their maximum potential.
- The sizing of the motors must be confirmed to ensure the best performance of the table.
- The beginner difficulty level should offer challenge to novice players. Characterized by lower speeds, reduced accuracy and limited numbers of operations. Should only block and kick the ball.
- The normal difficulty level should offer challenge to average foosball players, and should mimic their abilities. Moves more quickly than the beginner speed, has moderate accuracy and a standard range of functions. Should be able to block, kick and pass the ball.
- The advanced difficulty level should offer challenge to expert foosball players, and should mimic their abilities. Moves at maximum system velocities, has high accuracy and a large range of functions. Should be able to block, kick and pass the ball and perform “trick shots”.
- The table should be easily transported by two individuals of average strength and with some equipment.

Background

Existing Solutions

Research into existing automated foosball tables has revealed several different tables which have been built by different universities from around the world. It also showed that the previous team's research into existing designs was thorough and so much of the information they gathered is used in this background section. All of the automated tables use non-automated tables as their base, and add motors and controllers to automate them. Figure 1 contains a parts diagram of a typical non-automated foosball table. All of these tables have at least one side which is automated and can at least perform the basic motions required to play foosball. These elementary motions are the rotation of the rods, which hold the foosmen, and the lateral translation of the rods. This, coupled with a means of detecting the foosball on the field, allow the tables to block shots made by human opponents and to attempt to make shots of its own. The tables use a combination of linear and rotary motors to actuate the rods containing the foosmen [5] [8] [7]. Some of the tables are better able to control the foosmen and have varying levels of skill at which the computer can play [5]. There are several different ways of detecting the position of the foosball and of the foosmen [2] [5]. One method of detection is the use of a grid of lasers, and while this method could be extremely accurate, but there concerns that if the spacing is to tightly the lasers might illuminate several optical sensors. Another method used in existing tables is employment of a high speed camera system, which is suspended above the table. Because the camera is above the table, the foosball can be lost under the foosmen or the rods, which is a concern. Image processing is also extremely memory intensive, and the camera may lose track of the ball if it moves to quickly.

http://www.foosballsoccer.com/_/rsrc/1312136452159

Figure 1: Foosball table parts

The table which performs the best was developed by students at the University of Adelaide, Australia. The table uses a grid of lasers to detect the ball, and the rods are driven by linear and rotary motors which have been tuned to quickly and accurately move the foosmen. The rods used in the table are telescopic, which keeps the human player from coming into contact with the rods. The table and motors are covered by a plexiglass housing, which prevents the ball from leaving the field and prevents the motors from being touched during operation [5]. The plexiglass allows for safe observation of play without obstructing human players view of the field or the view of the motors while they operate. There are added parts on the rods which are believed to be an accelerometer to calculate the position of the players. The table also uses metal gears and a large amount of CNC machining was required to fabricate the table. The table has two difficulty settings, a slow moving beginner mode and a rapidly moving advanced mode [5].

Figure 2: Robotic Foosball Table From the University of Adelaide, Australia.
http://sites.mecheng.adelaide.edu.au/robotics/db_pics/projgalimg_337.jpg

Another example of an automated table is the one created by the students in the University of Akron. The table has the motors mounted to a side table, which can be seen in Figure 3. The table uses the stock rods and has no safety features. It also moves more slowly than the table developed by the University of Adelaide [8]. The detection system used by the Akron table is an infrared vision system, which includes infrared lights and phototransistors. Another table was developed by the Eindhoven University of Technology in Netherlands which used a vision system as its form of detection. The camera was mounted on a structure which suspends it over the center of the table, and an algorithm was developed to track the ball using the cameras images. Yellow foosmen were used as the computers foosmen because it is easier for the camera to track the bright yellow [8]. Telescopic rods are used in this design, and the motors are housed in a plastic housing.

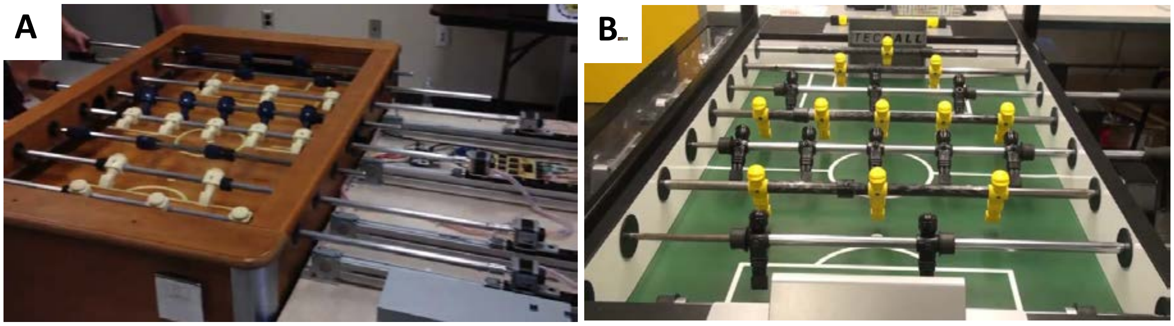
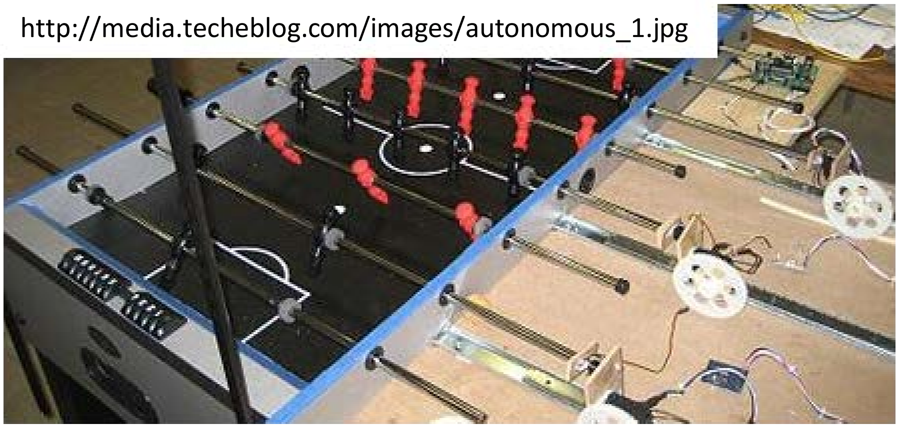


Figure 3: A. Foosball table from the University of Akron, Ohio (www.youtube.com/watch?v=XwONuoe3BB0). Notice motors and actuators on adjacent table. **B.** Foosball table form the Eindhoven University of Technology, Netherlands [7]. Notice the motors on the

There are also several tables which are not as sophisticated as the previous three existing tables. A table made in Denmark used telescopic rods driven by linear actuators and rotary motors, but was slow when compared to other tables [2]. A table made by the Georgia Institute of Technology was a low budget proof of concept. The table was relatively slow, which was attributed to gearing problems [3]. Rotary motors were used to both rotate the rods and to drive a rack and pinion which moved the rods linearly [3]. These motors were mounted on an adjacent table, and were left completely exposed. A low resolution camera, suspended over the table, was used to track the ball.



http://media.techeblog.com/images/autonomous_1.jpg

Figure 4: The foosball table made be the Georgia Institute of Technology. The gearing used to drive the rods can be seen.

The software for the tables shown was developed using programs such as MATLAB©. The software controls the motors and uses information from the detection system to locate the ball. If the ball is near one of the computers foosmen, it will attempt to push the ball forward [8]. Some table will also attempt to intercept the foosball as it is traveling.

The table developed by universities from around the world can be used to gain insight into how this project can best be completed. The table developed by the University of Adelaide is extremely complex and has many of the features which should be incorporated into the final version of this project. Table 1 compares the three best tables found during research.

Table 2: Comparison of top three existing automated tables.

	University of Adelaide	University of Akron	Eindhoven Int. of Tech.
Vision	Laser grid	Infrared light + phototransistor	Camera on top of table + image processing
Safety	Housing covers table	No features	Housing covers motors; telescopic rods
Motor selection	Rotary + gears	Linear actuators and rotary motors	Rotary motors
Motor mounting	On adjacent structure	On adjacent table	On table
Motion level	Complex and accurate	Slow and few errors	Complex and accurate
Machining required	CNC	No	Little
Aesthetics	High	Low	Medium

Current Project Progress

Team Foos-Ro-Dah has successfully completed their portion of the automated foosball system. Figure 5 contains an image of the system when it was displayed at the Cal Poly senior expo. The table has basic motion functionality, all of the motors and amplifiers are connected to the PLC and the system has been run several times using a program to generate virtual ball positions, which the table reacts to. The motor cabinet, vision arch and foosball table are all largely completed, though some small modifications need to be made before the system is finalized. A very basic AI system has been developed, and performs well when given ball positions.

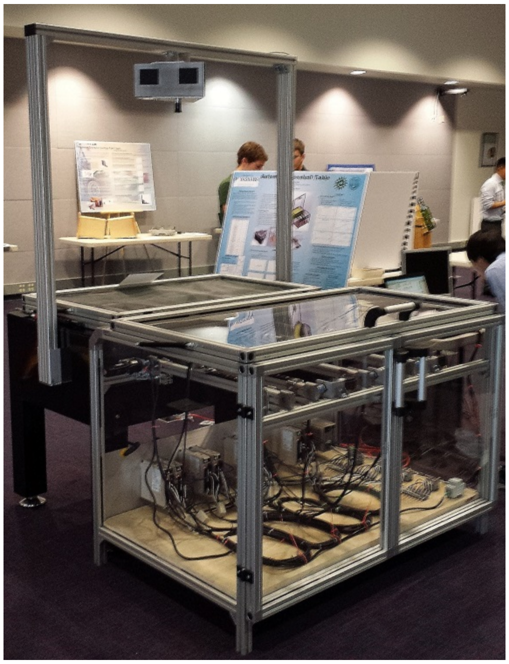


Figure 5: The table at Cal Poly's senior project expo.

There are several systems which are still uncompleted. The vision system is currently operational, but does not satisfy the requirements laid out by team Foos-Roh-Dah. The playfield cover proposed by team Foos-Roh-Dah is almost complete, but a material which does not interfere with the vision system still needs to be selected to act as a barrier between the human player and the playfield. The scoreboard system also needs to be completed. A frame has been constructed which will house the camera and the electronics for the scoreboard, but only the camera has been inserted and none of the electronics have been assembled.

Chapter 2: Design Development

Preliminary Physical Concepts

This section is a discussion of the major physical components of the system which still need to be designed for the basic functionality of the foosball system to be complete. Four major areas will be discussed. These are the lighting of the table, the attachment of the table to the motor cabinet, the alignment of the motors and the rods, and the playfield cover.

Lighting

Multiple methods of lighting the table were brainstormed and discussed. The methods ranged from mounting the lights up high to mounting the lighting below and using a translucent playing field. The goal of the lighting system is to improve the cameras ability to track the ball and to illuminate the field of play for the human player.

Lights Mounted on Score Arch:

Our first idea was to mount lights on the score arch. The lights would be slightly above the camera, so the image would be back-lit. This would help reduce shadows.

Lights Mounted on Score Arch with a Diffusor:

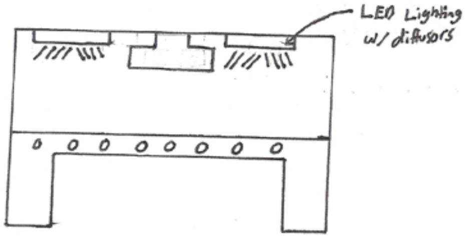


Figure 6: Sketch showing the ligths with diffusers concept.

This design is essentially the same as above, except it would include a diffuser. The diffuser would further help reduce shadows because it would make the light "softer." Figure 6 is a sketch of this concept.

Lights Mounted Below the Playfield:

This design would place the lights below the playfield. This would ultimately include replacing the green playfield with a translucent one in order to have the light shine through. While this design would help reduce reflection by having a more powerful light override the reflective light, it ended up scoring poorly because replacing the field would be incredibly difficult to do in a way that made the table still look professional. Also, we had concerns that the light from below could actually make it more difficult for the player to see what was happening on the field because it could be blinding. Figure 7 shows a sketch of this concept.

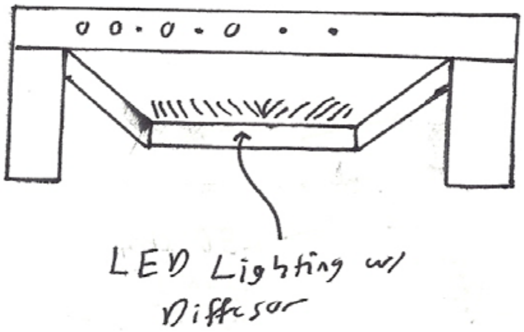


Figure 7: Sketch of the lights bellow the table concept.

LED Strip Lights Mounted at the Playfield Level:

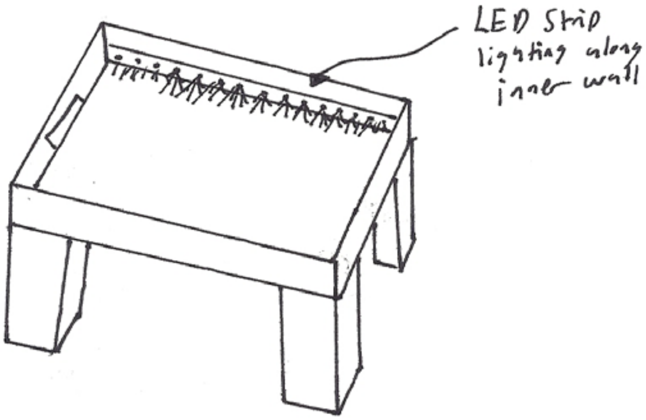


Figure 8: Sketch of the LED strip lights on the playing field concept.

This design would place LED strip lighting along the inside bottom edge of the foosball playfield. This design scored poorly because we thought that it could interfere with the movement of the ball and the foosmen. Also, since the lighting would come from the side, the foosmen would cast, long shadows on the field which would not be beneficial for the user or the camera system. Figure 8 shows a sketch of this concept.

LED Strip Lights Mounted inside the Playfield Cover:

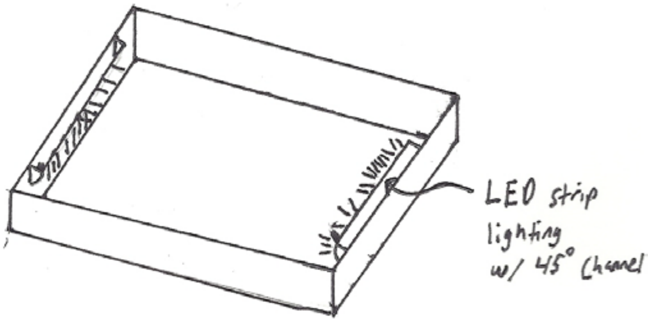


Figure 9: Sketch of the LED strip lights on the play field cover concept.

This design is similar to the one above, except the lights would be moved upwards so they are just inside the cover that protects the user from the spinning foosmen. This design would not interfere with the movement of the foosmen or the ball, though it would still cast long shadows because the light would be mainly from the side. Figure 9 shows a sketch of this concept.

LED Strip Lights Mounted above the Playfield Cover:

This design would move the strip lights further up, so they would be about a foot above the playfield height. This would mimic stadium lighting. This design did not score well because it would cast shadows and would also potentially create reflections in a plexiglass playfield cover. Figure 10 shows a sketch of this concept.

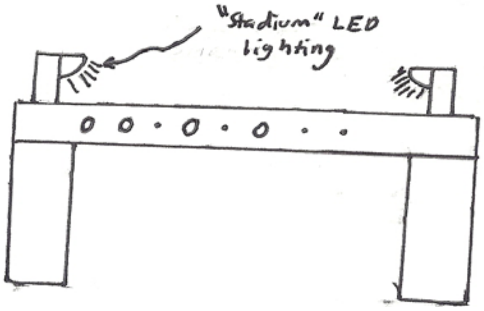


Figure 10: Sketch of the LED strip lights above the playing field concept.

Foosball Table and Motor Cabinet Attachment System

To ensure that the system functions consistently and safely, and to reduce wear on the motors, rods and bearings, an attachment system between the Foosball Table and Motor Cabinet should be included in the final design. The primary goal of this system will be to hold the table and cabinet against one another, and prevent them from becoming separated during play. It should also reduce vibrations transmitted between the table and the cabinet, and will also aid in the alignment of the motors and the foosmen rods. There are several different concepts which have been generated, and the options which have been found to best meet the requirements for the attachment system will be discussed in the following section.

Bolts through Legs

This concept involves placing two or three bolts through the legs of the foosball table on the computer driven side. These bolts would be inserted into receivers mounted on vertical 8020 struts. These struts would need to be added to the cabinet because the existing 8020 struts are not positioned properly and cannot be moved without a redesign of the cabinet as a whole. Rubber washers would be used to dampen vibrations and to reduce wear on the table and the cabinet. Figure 11 shows a sketch of this concept.

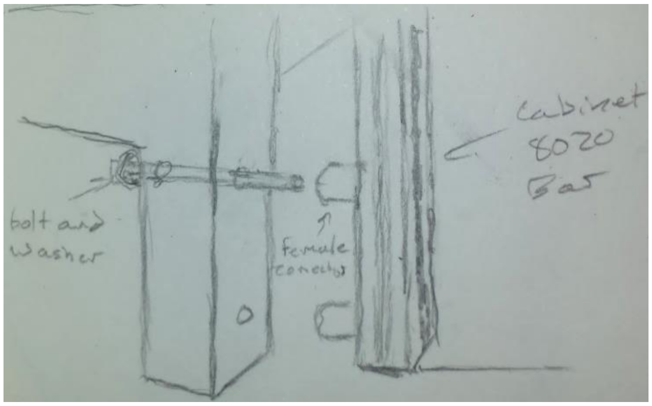


Figure 11: A sketch showing the leg bolt attachment concept.

There are two major concerns with implementing this concept. Firstly, designing and manufacturing the receivers which would mount on the 8020 struts would be difficult and time consuming. Additionally, adding the second vertical strut to the cabinet would require the modification of the aluminum mounting for the amplifiers and the PLC.

8020 Strut added to the side of the Table

This concept involves mounting an 8020 strut horizontally beneath the rods on the computer driven side of the table. This strut could then be attached to a corresponding strut on the cabinet using Boch's quick connectors. Rubber padding could be placed between the two struts to damp out vibrations and to prevent the two struts from wearing each other out. Figure 12 shows a sketch of this concept.

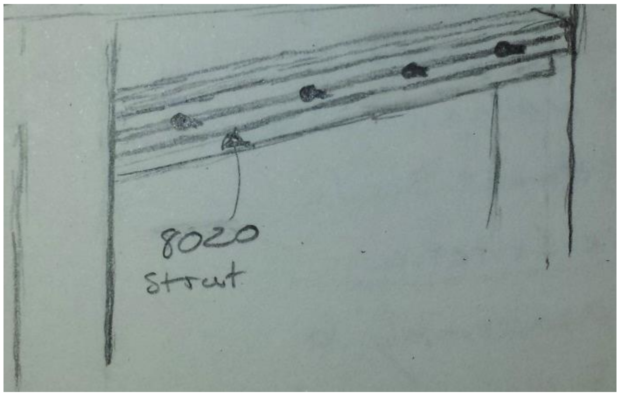


Figure 12: Sketch showing the 8020 strut attached to the side of the table concept.

Several concerns have been identified in the implementation of this concept. Because the struts would need to be placed relatively high on the table and the cabinet, attaching and disassembling the system would be awkward, as most points would only be accessible through the cabinet. Alignment also becomes an issue because if the struts on the table and the cabinet were not parallel, aligning the motors and the rods would be almost impossible. Finally, because the attachment point on the table is the relatively unsupported side of the table, there is a chance that table could be damaged over time.

Brackets

This concept involves mounting brackets on the side of the table using bolts place through the legs of the table. The bracket would have a flange through which t-bolts could be inserted. These t-bolts would then be used to attach the motor cabinet to the bracket. Rubber washers would be used to reduce vibration transmission and reduce wear on the table, the brackets and the motor cabinet. Figure 13 shows a sketch of this concept.

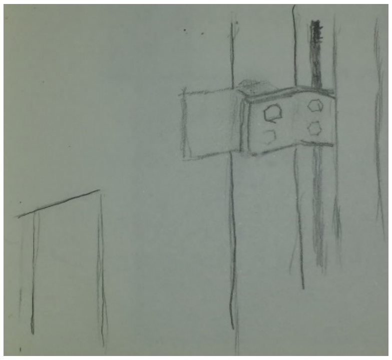


Figure 13: Sketch showing the bracket concept.

There are several possible difficulties associated with this concept. Each bracket used in the system would need to be machined by hand, which would be time consuming, though relatively strait forward. Additionally, because the brackets would remain attached to the table, transportation may be made more difficult.

Foosball Table and Motor Cabinet Alignment

Alignment refers to how to set up the motors in the correct positioning with the rods on the foosball table. Incorrect alignment will damage the motors and cause significant loss of performance when compared to a properly aligned motor. Alignment of the system also goes hand-in-hand with attachment of both tables in the system. The motors must be aligned and maintain that alignment through motor vibrations, uneven ground, and assembly.

Leg Leveled Mount

The idea of this design was to incorporate a ladder system into each of the legs on the motor casing table. Combined with a level to determine proper balance of the casing table and the foosball table it would provide a way to individually move each leg vertically such that if the table was on a slight slant the legs could be moved to compensate for it. The problem with this design is that it would be expensive, tedious, and require redoing the currently built casing table which would take time. The design is riskier than other designs from a safety perspective as it opens the door for a possible failure on one of the leg mountings and potentially causes the casing table to fall over and hurt someone.

Beam Constrainers

Beams, most likely 8020 will be attached to the legs of the foosball table in such a way that the casing table has to be aligned with the foosball table in order to fit into the beams. In this way the beams act as limiters which prevent the tables from being misaligned and are easy to setup. Additionally they cost little compared to the alternatives due to the small amount of materials needed to fully constrict movement. This method would look less appealing but it should be done such that the beams constrict from inside as the casing is the larger table, so it would not be too much of an issue.

Brackets

Brackets would be attached on the foosball table's legs and machined so that the cabinet legs fit inside of them. This insures that the motors and rods are parallel with each other by constraining the position of the cabinet. This Also maintains structural integrity between the two systems and also has the added benefit if transferring some of the vibration through both tables fairly well compared to the beam constrainers. It is easy to setup but it has the problem of not being aesthetically pleasing. This could be fixed by making it somewhat hidden, but as the foosball field is the main attraction of this device that might be an acceptable sacrifice to make in exchange for the benefits this method brings. Another downside to this concept is that it does not align the rods and the motors vertical direction, but this could be solved by adding another alignment concept in addition to the brackets.

Infrared Slot Sensors

Using an infrared beam on one side and a sensor on the other would allow for easy alignment without the need for additionally beams or brackets to be mounted to the legs of either table. The downside to this method would be that the cost would be much more than any other method and would still not keep perfectly aligned as the sensor would have a range of space where it accepts the laser, which might not be accurate enough for our motor alignment. Additionally, because there would not be any physical constraints the tables would move while the motors are active, possibly misaligning and causing damage.

Clamps

Similar to the brackets but instead using clamps and no bolts. This would be easier to setup and could be used on different places as needed depending on the slope of the floor or any arbitrary variable that would make a static constraint like a beam or bracket unusable. The problem is that they do not look good for a professional product and can easily not work if there is a sudden jolt in either table which could render the clamps useless if they fell off. For this reason, it is not a safe device to use and should not be considered as a serious option for our project.

Playfield Cover

To provide a barrier between the playfield and the user, a playfield cover should be attached over the foosball table. The primary function of the playfield cover is safety. A barrier will prevent both flyaway balls reaching the user and prevent the user from reaching into the playfield while the machine is still running. A secondary, but essential property of the cover is that it must allow both the user and the vision system to view the playfield. This attribute is the true deciding factor for the proposed design. Two main concept structures have been produced, each with variations that have been considered in the following section.

Top Door Design

The first of the two proposed structures is the current design implemented by the first generation on this project. The cover is mounted directly on top of the table opening upwards to give access to the playfield. The advantages of this design are that it is easily assembled, and may be left permanently attached to the table making it extremely portable. The variations of this concept relate to the chosen barrier material. Proposed materials include a screen mesh, or some variety of anti-glare glass or plexiglass. Either of these materials fulfills the safety requirement with a solid barrier separating the user and the moving parts of the machine.

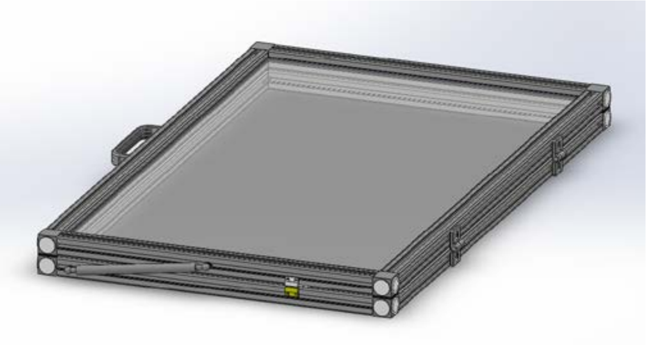


Figure 14: A solid model of the Flip Top Cover

Some concerns with this design are that the camera used for the vision system has difficulty viewing the playfield through the chosen material. Some testing would have to be done to decide upon an appropriate material that promotes visibility.

Window Cage

The second proposed structure is a new design that would mount on top of the table encompassing the vision system, creating a windowed box over the table. By encompassing the vision system within the cover, the camera will have an unobstructed view of the playfield. Acrylic panels similar to the ones included in the current system would provide sufficient visibility to the user. This structure would equally fulfill the safety requirement.

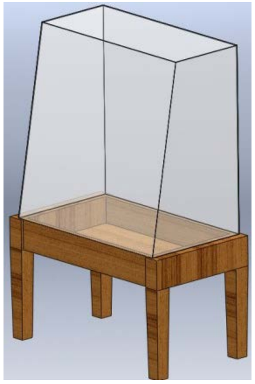


Figure 15: Solid model of the encompassing window concept.

A disadvantage of this concept is that it will be much larger than the current version of the playfield cover. This size could be reduced by designing the structure to be collapsible into individual panels for transportability; alternatively, it too could be left affixed to the table during transport, providing protection to the vision system.

Physical Concept Analysis and Selection

The following section contains a description of each concept which was chosen and the rational used to make that decision. Appendix C contains the decision matrices used to evaluate and compare the different ideas for each concept.

Lighting

After some ideas had been thought up, we researched lighting and its use with cameras. The first thing we researched was how lighting can be used to reduce shadows. After looking at multiple sources, it was clear that lighting from behind would result in the fewest shadows because shadows are created in photography when light is coming from the sides or back.

Another major factor that we had to consider when rating our possible lighting designs was how the lighting could hinder or hurt the reflection problem of the current playfield cover design. It was clear that any lights that would be mounted above the acrylic playfield cover would contribute towards glare and reflection, though light pointed up through the acrylic could potentially reduce the glare.

After researching these two topics, we decided to make a decision matrix (see above) that ranked the potential designs based on these two categories along with many more such as cost, ease of manufacturing, portability, and how well we thought each design could light up the field completely. After the decision matrix was completed, two results seemed like they would be the best all around: mounting the lighting on the scoreboard arch with diffusers, and mounting the lights on the inside edge of the playfield cover.

We were only able to narrow the concepts down to two because the lighting design ultimately depends on the playfield cover design. If we stay with the current, flip-cover playfield cover design, mounting the lighting below the cover will be more effective because it will not cause reflections, but it may create shadows. If we switch to the plexiglass box idea for the playfield cover, reflections will not be an issue because the camera will be inside the plexiglass; therefore, reducing shadows will be more important, and mounting the lights above the camera will be the optimal design. The decision matrix was not a waste of time though because it narrowed down the designs to two solid choices.

Attachment System

Brackets were chosen as the concept which would best meet the requirements for the attachment system between the foosball table and the motor cabinet. There were several reasons that brackets where identified as the most desirable concept. It will require the least amount of effort to assemble and disassemble because the fasteners would all be on the outside of the system, making them easy to access. All of the other top concepts require some fastening to be done inside the motor cabinet or under the table when the system is being assembled for use. Because t-bolts easily slide in the 8020 slot before they are fully tightened and are easily accessed on the bracket, the motors and the rods can be aligned while the table and the cabinet is attached, making alignment much simpler. Brackets also allow for the table and the cabinet to be flush with one another, reducing the amount of space which needs to be covered between the table and the motor cabinet. Figure 16 shows a concept model of a bracket, and how it would be place in the system.

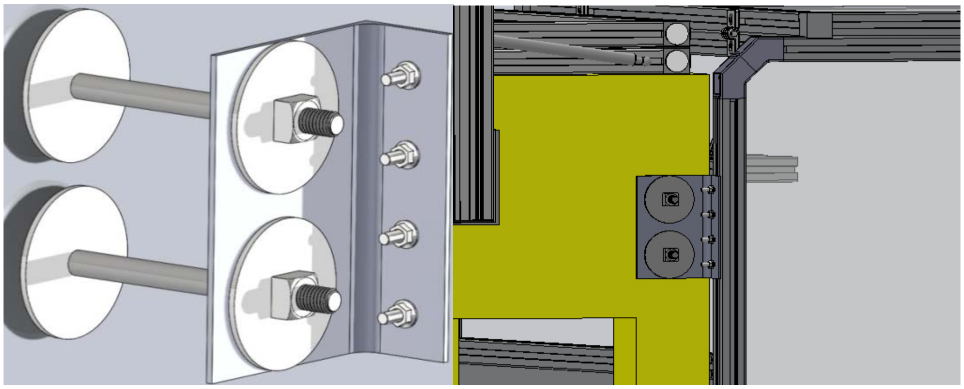


Figure 16: Solid models of the bracket concept.

Alignment System

The bracket concept and the infrared slot sensor were selected as the best choice for alignment for several reasons. First, it allows us to use the brackets for both alignment and attachment between the two tables, knocking out two problems with one solution. The addition of the slot sensor allows vertical alignment of the system to be easily confirmed. Another reason is that brackets allow for discrete placement and does not distract from the overall design of the foosball table project. Brackets also provide sturdy and steady support, meaning vibrations from the motor will be more damped due to the higher mass of both tables combined. Brackets are also easy to install and replace which is a plus compared to more complicated solutions. The slot sensor would also be relatively easy to replace and maintain.

Playfield Cover

A decision between the two proposed covers has not yet been finalized, and is largely up to the preferences of the sponsor. Both designs will be safe and allow access to the playfield. The differences to base a decision upon include visibility for the camera as well as players, portability and ease of assembly, as well as aesthetic preference. Both designs will have some additional cost. The first, though already constructed, will require samples of various barrier material to be purchased and tested before a final sheet/screen can be ordered. Some of the proposed materials are relatively expensive($45.87/sqft) in comparison to the acrylic currently used on the motor table ($3.09/sqft). The second design could utilize some of the leftover 8020 aluminum extrusions, but will likely have a larger overall cost due to the greater amount of panels required.

Motor Testing and Tuning

The following section is a discussion of the methods which will be used to test and tune the motor system. It also contains a discussion of the characterization of the vibrations of the system during the motors operation.

Verifying Max Ball Velocity

The goal of this test is to determine if the 100 Watt rotary motors used to spin the rods can accelerate the foosball to a velocity of 8 m/s, and that during this motion the motors torque output falls within the intermittent operating curve for the motor. Table 3 contains a list of specific parameters which will be used during the test. Each of the motors will be tested to insure that the different rod configurations do not have an effect on the motor performance. The motors performance will be monitored using functions built into Motionwork. The balls velocity will be determined using photo gates and a DAQ. A foosmen will be used to kick the balls through the photo gate, and the DAQ will process the information and determine the balls velocity. This method has been used by team Foos-Roh-Dah during their final system testing, and Figure 17 contains an image of the setup they used. The tests performed by team Foos-Roh-Dah achieved a velocity of 10.6 m/s, but only velocity data was gathered. Once a ball velocity of 8 m/s is achieved and characterized, greater ball velocities will be tested until either the motors maximum torque output becomes too great or the camera can no longer capture the ball while it is traveling. After the initial testing is complete, the system will be tuned, and then another round of testing will commence.

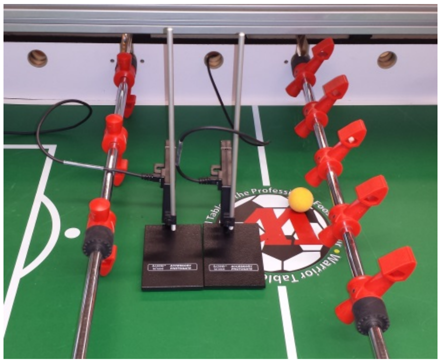


Figure 17: Test setup used by team Foos-Roh-Dah

Table 3: Table of testing parameters used to verify maximum ball velocity.

Testing Parameter	Target Value
Foosball minimum velocity	8 m/s
Angle of foosman rotation	90 degrees
Time to perform angle change	200 ms
Position accuracy	.05 degrees
Settling time	100 ms

Verifying Lateral Rod Velocity

The goal of this test is to determine if the 150 Watt rotary motors used to drive the linear motion of the rod can achieve a velocity of 2 m/s, and that during this motion the motors torque output falls within the intermittent operating curve for the motor. Table 4 contains a list of specific parameters which will be used during the test. Each of the motors will be tested to insure that the different rod configurations do not have an effect on the motor performance. The motors performance will be monitored using functions built into Motionworks. The performance of the motor will be recorded using the function in MotionWorks. Once the minimum goal of 2 m/s is achieved, the linear velocity will be increased until the motors torque output becomes too great or until the linear belt system cannot operate at the tested velocity. After the initial testing is complete, the system will be tuned, and then another round of testing will commence.

Table 4: Table of testing parameters used to verify the lateral rod velocity.

Testing Parameter	Target Value
Minimum linear rod velocity	2 m/s
Distance travelled during test	.25 m
Time to perform angle change	125 ms
Position accuracy	.0005 m
Settling time	100 ms

Motor Tuning

To tune the motors, two different approaches will be used. First, the motors will be run in tune-less mode, which will give a baseline performance for each of the motor pairs. We will then use the auto-tuning function in MotionWorks to tune the motors. The performance of the motors after the auto-tuning will be recorded and saved. The motors will then be set back to their default state, and the motors will be re-tuned manually. The results of this manual tuning will be saved and the overshoot, rise time and settling time will be compared to the results of the auto-tuning to determine which of the two methods worked the best. The method which provides the best results will be used for the final system tuning.

Vibration Analysis

To determine the amplitude and the frequency of vibrations experienced by the motor cabinet and the foosball table during operation, accelerometers will be attached to different points of the system. The motors will then be run near their maximum operating conditions, and the accelerations of the table will be measured in each of the three Cartesian directions. Acceleration data for each direction will be collected using an oscilloscope. These accelerations will then be converted into force data and bode plots will be used to characterize the vibrations. Once the vibrations of the table are characterized, dampers can then be selected to minimize the effect of the vibrations on the table and cabinet, and to minimize vibration translation between the table and the cabinet.

AI Development and Logic

Overview

The logic that dictates the AI functionality must be straightforward and easy to interpret. For our project, the AI responses will be a function of vision data, extrapolated data, and difficulty level. The vision data is read through a camera that can see the entire playfield and feed through a computer. The computer then uses this data to find numerical values for current ball position and velocity. Variable data calculated will then be used to find where the ball will be in the future. This data is sent to the PLC controlling the motors, which then decides on the proper response to current playfield situation. Certain playfield conditions will be used as flags that change the behavior of the system. For instance, if a flag exists for ball ownership, and it returns that the ball belongs to the opposing player, then the PLC will respond with defensive positioning and react accordingly.

There is expected to be 3 difficulty levels, beginner, intermediate, and expert. Currently, the beginner mode will be played from a defensive-only strategy, randomly hitting it back up field. Each rod will be actively trying to block the ball from passing behind them. This allows the player, who is a beginner, to learn the mechanics of the game and how to shoot the ball. The intermediate difficulty will be more advanced and include offensive schemes in the design. However, it is expected that the goal shooting from the AI side will not be accurate and goals will be scored by luck. This difficulty makes the player focus on defense and offense, but in a less intensive setting. The expert difficulty is when accuracy matters for the AI. In addition to it playing defense and shooting the ball forward, shots will be done with calculations based on where the ball will be when the ball is hit, and adjusted to make as many goals as accurately as possible.

Rules

The system's behavior will follow these basic rules to transition from defense to offense, moving the ball upfield to score. To implement these rules, an AI task run cyclically will contain four rod control POUs. This will enable easy communication between rod controllers, and a centralized location for all other tasks (such as the vision system, and the UI) to communicate with the AI of the system as a whole. POUs will be created of increasing levels of complexity for types of movements rods can execute. An example of such a structure would be a kick. A kick POU could be executed by any rod, and would need inputs such as desired direction and speed. This kick POU could be called by a passing POU, or any of the multiple shot POUs.

- Every rod not in control of the ball, and 'behind' the current ball position should be continuously moving to position a foosman between the ball and the goal. This state will be considered the defense mode.
- Bars should move to intercept balls that are within reach.
- Bars not in control of the ball and 'ahead' of the current ball position, should be flipped up, out of the way of a shot on goal or passes from the rods behind.
- Bars other than the forward-most bar, the 'forwards,' in control of the ball will execute a passing routine, involving clearing it from the defense to the midfield and then to the forwards.
- Clearing could involve passing back and forth between the two defending rods to 'mix-up' the opposing team's player configuration looking for an opening.
- Passing will involve two adjacent rods communicating to move the ball up field.
- Only the 'forwards' will take shots on goal, and will have a set of shot types to choose from depending on the defender's player positions.

Additional System Improvements

This section describes additional improvements which need to be made to the system. These improvements take the form of replacing existing parts of the system, adding components or generating relevant diagrams.

Vision System

Current Status

Currently, the vision system does not work well enough to be an effective way of measuring the foosball's position and/or velocity. It currently only collects data at around 11 frames per second and often mistakes the yellow ball for the white lines on the table. 11 fps is not nearly fast enough to accurately track the ball. If the ball is moving at 8 m/s (the fastest we were able to manually hit the ball), and the software is running at 11 fps, the ball can move 73 cm in one frame. This means that the ball will move more than two thirds down the field without the AI knowing that anything has happened. Because of John Inlow's (the original programmer) unknown status on continuation with the project Dr. Macedo is working to find another computer science major to collaborate with us on building a camera system that runs quickly enough.

MotoSight 2D:

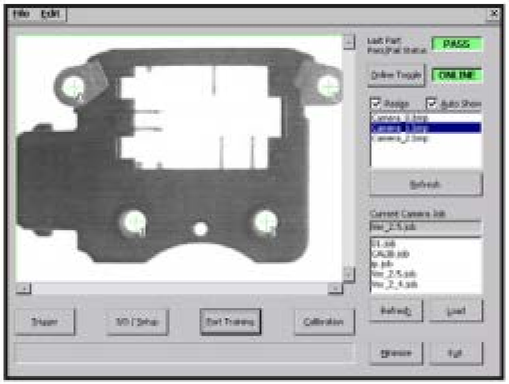


Figure 18: MotoSight 2D in use.
http://www.motoman.com/datasheets/MotoSight%202D.pdf

The MotoSight 2D could be used as a vision system for the foosball table. Using a Motoman product would be beneficial because it has the added benefit of advertising for Motoman, another division Yaskawa. The software is designed for high-speed picking, and is rated for 60 fps, so it should work for our application and using the MotoSight 2D setup would be make interfacing with the PLC easier because it is meant to work in similar applications. A datasheet for the software is included in Appendix D.

Using our current camera has the potential to be more effective in the long run because the camera is rated for 160 fps. Also, since the software would be built for the single purpose of tracking foosballs, if it is designed correctly it could end up being much lighter-weight and therefore faster than MotoSight2D.

Human Machine Interface

Currently, there is no way for the user to interface with the machine besides going into the code and changing variables in real time. Obviously, this will not be acceptable for the final product because the foosball table will be used in a trade show environment and most of the users will be inexperienced with programming PLCs. Allowing an inexperienced user to change code could be potentially dangerous to the machine and, more importantly, dangerous to that user and/or other users.

In order to make the machine safe for users and spectators we will include a human to machine interface (HMI). The inputs and outputs of this interface are detailed below.

Table 5: Contains the expected inputs and outputs of the HMI

Variable	Input/Output	Physical Control
Player Speed	Input	Slider/Knob
Kicking Power	Input	Slider/Knob
AI Difficulty	Input	Buttons/Switches
Emergency Stop	Input	Button
Reset	Input	Button/Switch
Playfield Cover Open	Output	Light
Ball Stuck	Output	Light
System Status	Output	Light
Power to Axes	Output	Lights
MotionWorks Error	Output	Light

There are two different ways to implement an HMI for the foosball table. The first would be to build a panel with physical switches, buttons, knobs, and lights that would be wired to an I/O module. Another would be to use a tablet or computer to mimic these inputs and outputs with software. Both options have their advantages and disadvantages.

The advantage of the physical HMI is that it would be hard-wired to the system, so the only way it could fail is if a wire became disconnected. A software HMI could fail if the software crashes, or if the tablet or computer loses power. Another advantage of the physical HMI is that it would give the user tactile feedback.

The advantage of the software HMI is that multiple interfaces could be created easily. There could be an HMI with minimal information for the user, and HMI with much more detailed debugging information for the technician responsible for keeping the table in working order. Also, a software HMI would make the whole machine as a whole more technologically advanced.

Scoreboard

The score board is currently unfinished. A frame has been constructed which will house the camera and the electronics for the scoring system, but these electronics have not been assembled or inserted into the frame. Figure 19 shows the current state of the scoreboard system. To complete this system, the electronics for the board itself will be assembled and inserted into the frame and the electronic switches in the goals will also be installed. All of these systems will then be connected to the PLC using an I/O module and a function to track and display the score will be developed.

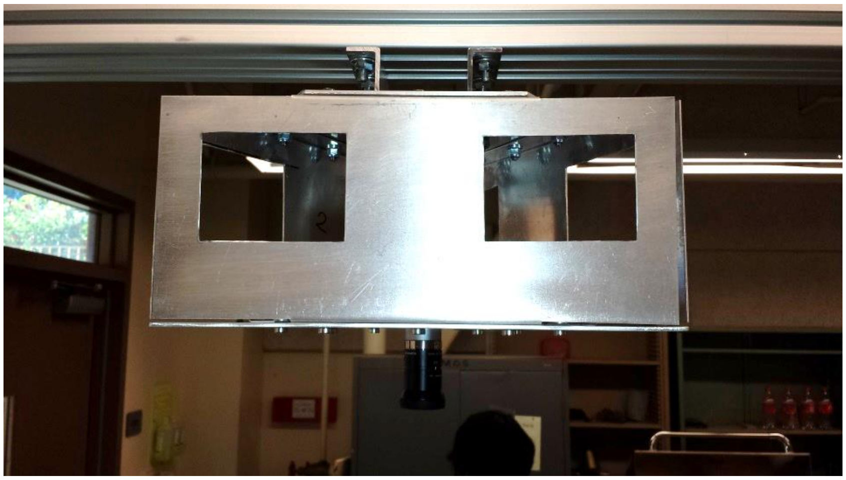


Figure 19: The frame of the scoreboard with the camera in place.

Couplings

Because the current couplings attaching the rods and the motor are rigid, and there are two bearings in the motor and a third on the table, the rod/motor system is over constrained. This makes alignment difficult and could cause accelerated wear in the components. To solve this issue, the couplings attached to the motors are going to be replaced with flexible couplings from Heli-Cal. These couplings will allow remove the extra constraint from the system, and will allow two bearings to be used in the table, which will reduce vibrations in the rods during actuation .We are currently in talks with a representative from Heli-Cal to pick a proper coupling, and are leaning toward using the DS Series, shown in Figure 20. It allows for a 3 degree angular offset, can withstand torque up to 234lbf-in, and can operate up to 10000rpm.

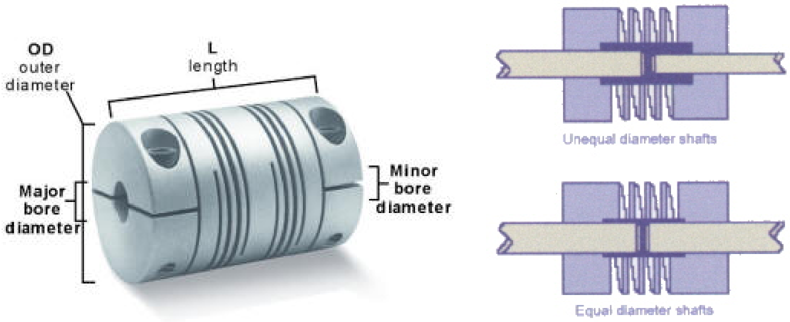


Figure 20: Coupling DSAC 30 -14 -12 (all mm)

Table 6: Parameters being used to select new couplings.

Parameter	Numerical Value
Bore Diameters	14 mm and 12 mm
Projected Duty Cycle	50%
Service Life	5 years
Outer Diameter Envelope	30 mm
Torque Transmission	0.9 Nm
Max Axial Load	120 N

Cover for Exposed Section of Rods

In the current design there is an exposed section of rods between the motor cabinet and the foosball table. Figure 21 contains an image of the exposed section of the rods. This is a safety issue and the exposed section requires a cover to prevent the machine from harming users, operators or spectators. The proposed cover will be constructed by joining pieces of acrylic or polycarbonate sheet, and will cover the entirety of the open section. The cover would be separate from the table and the cabinet, and would be set into place after the rest of the system is place.

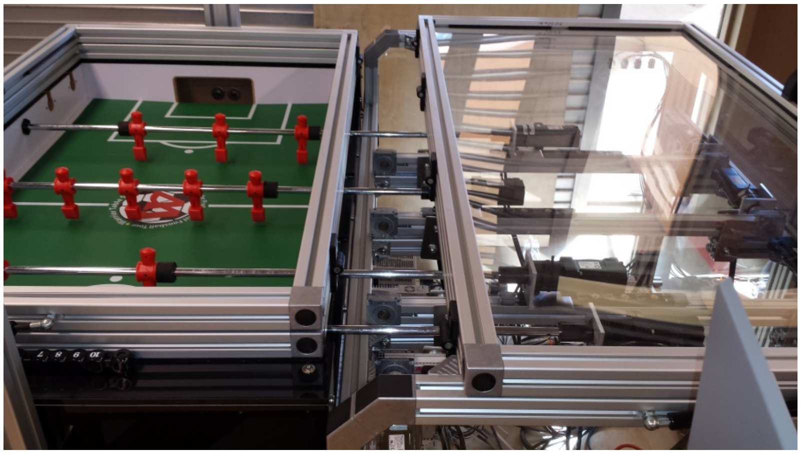


Figure 21: The exposed section of rods to be covered.

Adding Feet to the Cabinet

Currently, the vertical 40mm x 40mm struts of the cabinet are acting like feet, an example of the current feet is shown in Figure 22. This is not ideal, as the extruded aluminum provides little friction to prevent the table from sliding during operation and could potentially damage the floor it is placed on. To solve this problem, rubber feet or casters should be added to the bottom of the table. This reduces vibration transmission between the cabinet and the floor, reduces the risk of the cabinet sliding during operation, and protects the tradeshow floor.

Figure 22: Current feet on the motor cabinet.

Improvements to Linear Actuator Mountings

There are several improvements which can be made to the linear actuator mountings. The first improvement is the replacement of the small 8020 struts at either end of the cabinet, seen in Figure 23, with solid 8020 struts. This improves the overall stability of the cabinet during operation.

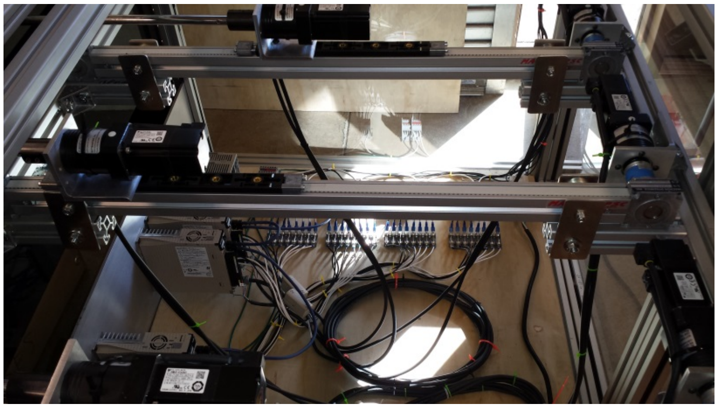


Figure 23: Image showing the two small 8020 struts used to support the actuators.

The current brackets used to attach the linear actuators to the 8020 struts, shown in Figure 24, allow the actuators to shift during operation. This shifting lowers the lifetime of the actuators and of the motors used to rotate the rods.

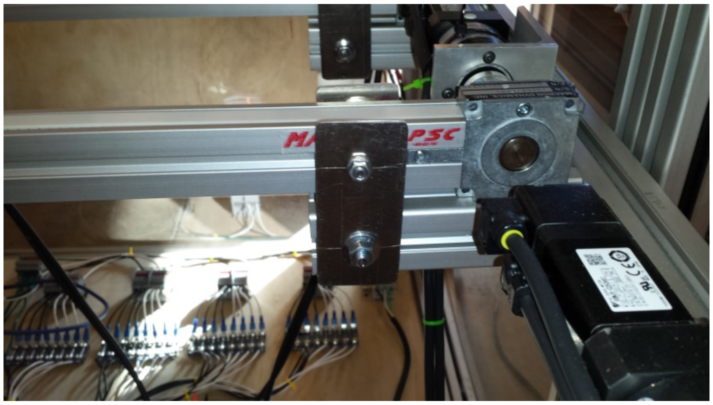


Figure 24: The current brackets used to attach the linear actuators to the motor cabinet.

The final improvement which could be made to the linear actuator mountings would be the addition of cable tracks to the sides of the 8020 struts. These cable tracks will help to organize the wires to the rod motors, reduce the wear on the wires and will improve the aesthetics of the cabinet.

Chapter 3: Final Design

Hardware Design

Table to Cabinet Brackets

Description

The purpose of the table to cabinet brackets is to attach the foosball table and motor cabinet together. This is to prevent the two components from moving during operation which could be dangerous. It also insures that the foosball table rods and the motors do not become misaligned. To insure that the attachment between the table and the cabinet is secure, four brackets will be used and will be placed at the top and bottom of each side of the system. Because the cabinet cannot be placed symmetrically with the table the design for the brackets on the left and right sides of the table are different. Figure 25 and Figure 26 show the Solidworks models of both the left and the right side brackets. Drawings for each of the brackets can be found in Appendix D.

Figure 25: A solid works model of the bracket which will be mounted to the left side of the motor cabinet.

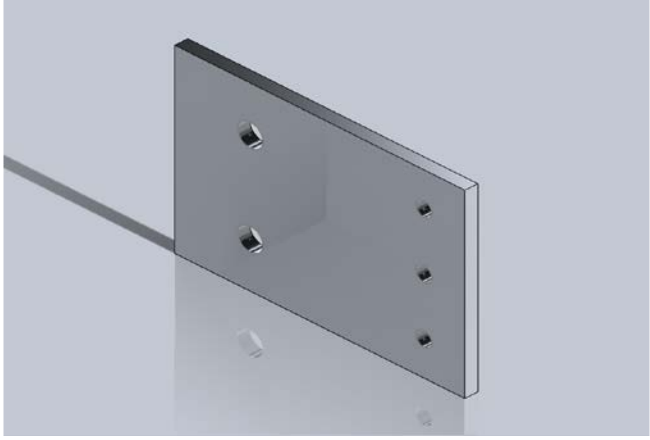


Figure 26: A solid works model of the bracket which will be mounted to the left side of the motor cabinet.

Material Selection

Quarter inch thick aluminum stock, both 90 degree L and plate, was chosen for the material used to construct the cabinets. The aluminum fits with the aesthetics of the overall system, and is capable of withstanding the relatively low loads which will be applied to the brackets. Aluminum is also easily machined, which will decrease the time the team spends producing the brackets.

Analysis and results

Because of the difference in the geometries of the brackets on the left and right side of the system, the analysis of the brackets was performed separately. A force of 25 pounds is assumed to act on each of the brackets. Analysis of the brackets can be found in Appendix E.

Left Side Brackets

The brackets which will be used to attach the left side of the table to the left side of the cabinet will be L-shaped to accommodate for the distance between the cabinets leg and the tables leg. To compensate for the slight slope of the table legs, the holes used for the M10 bolts are at a slight angle, which will allow the brackets to rest flat against the legs of the table.

Static analysis was used to show that the brackets will not fail when loaded by the cabinet. Table 7 contains the results of this analysis. Fatigue analysis was used to assess the lifetime of the bracket, and Table 8 contains the results of this analysis. All of the analysis on the bracket shows that they will be able to withstand the loads they will be subjected to.

Table 7: Results from the static analysis of the left brackets

Maximum Allowable Load	363.6 lbf
Force Applied	25 lbf
Factor of Safety	14.8

Table 8: Results from the fatigue analysis of the left brackets

Fatigue Strength	14,000 psi
Cycles for Fatigue Strength	5.0 x 10^9 cycles
Fully Reversed Stress	1851 psi
Factor of Safety	7.56

Right Side Brackets

The brackets on the right side of the system will be manufactured from a flat plate of aluminum because the legs of cabinet and the legs of the table are relatively flush. Because the gap between the legs increases as near the bottom of the legs, a rubber plate will be placed between the lower bracket and the tables' leg, which will allow the bracket to grip the table.

Static analysis was used to show that the brackets will not fail when loaded by the cabinet. Table 9 contains the results of this analysis. Fatigue analysis was used to assess the lifetime of the bracket, and Table 10 contains the results of this analysis. All of the analysis on the bracket shows that they will be able to withstand the loads they will be subjected to.

Table 9: Results from the static analysis of the left brackets

Maximum Allowable Load	24,665 lbf
Force Applied	25 lbf
Factor of Safety	986.6

Table 10: Results from the fatigue analysis of the left brackets

Fatigue Strength	14,000 psi
Cycles for Fatigue Strength	5.0 x 10^9 cycles
Fully Reversed Stress	33.33 psi
Factor of Safety	420

Solidworks Analysis

Finite element analysis was used in Solidworks to confirm that the brackets will not fail under the assumed conditions. The results of this analysis show that the each of the bracket types will be able to withstand the required loads. The details and results of the analysis performed using Solidworks can be found in Appendix E.

Lighting/Roof

Description

The purpose of the lighting system combined with the roof is to attempt to control the amount of light going into the camera. Because the playfield cover is made from acrylic, any direct light from above the camera will reflect off of the acrylic and create glare. Our solution to this problem is to build a roof that covers the table from direct light and then to light the table from below the playfield cover.

Testing

In order to prove that a roof design would work to eliminate glare and LED lights would provide the necessary lighting, we first had to do some preliminary testing. The photo below shows the camera's view at full ambient light with no playfield cover.

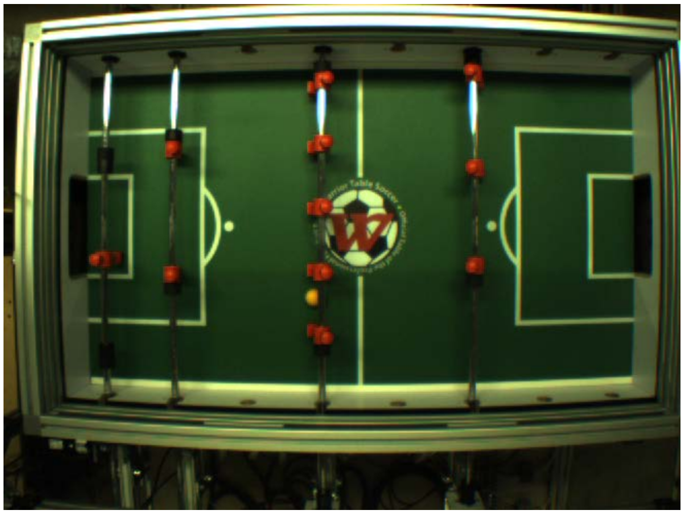


Figure 27: Camera view with no playfield cover

The next photo shows the camera's view with the acrylic playfield cover.

Figure 28: Camera view with acrylic cover

The glare from the fluorescent lights was very strong and completely obscured the ball from the camera's view. In the next photo, we used mounted cardboard above the vision arch in order to block out the direct light from the fluorescent lights.

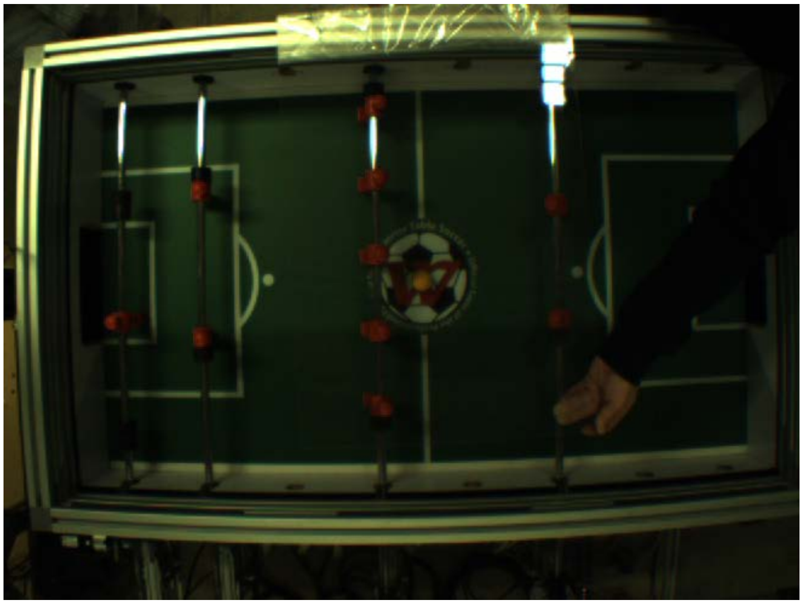


Figure 29: Camera view with cardboard blocking most direct light

In this photo, the table looks identical to the photo with no acrylic cover, so it is clear that blocking direct light to the acrylic does a sufficient job of reducing glare. The small amount of glare in this photo is a result of using a strip of cardboard which did not completely block the fluorescent light. We also tried testing with a screen cover, without blocking the fluorescent lights, as shown in the photo below.

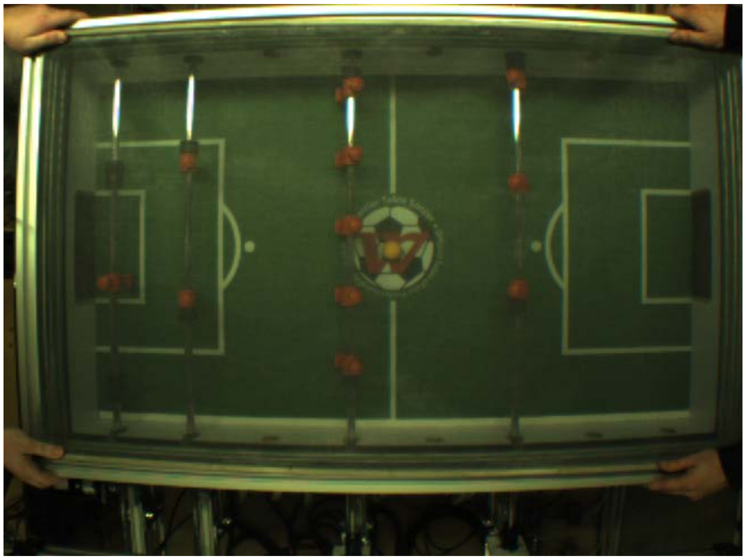


Figure 30: Camera view with screen-door playfield cover

Although the screen does not block the ball from the camera's view, it is definitely not as clear as the acrylic. Also, the screen does pick up a little bit of glare as well. Because of the lack of clarity and the fact that it still has some glare, we decided that the roof and acrylic combination had the best results.

Finally, we had to make sure that we could light the table in a situation where there is not a lot of indirect light to the table, especially because the roof would block all of the direct light. To test the concept of using LEDs to light below the playfield cover, we ordered some cheap LEDs (approximately $25) and tested how well they could light the table with the rest of the room's lights off. The photo below is the view from the camera with the lights turned off, and the one below that is the view when the LEDs are turned on.

Figure 31: Camera view in a dark room

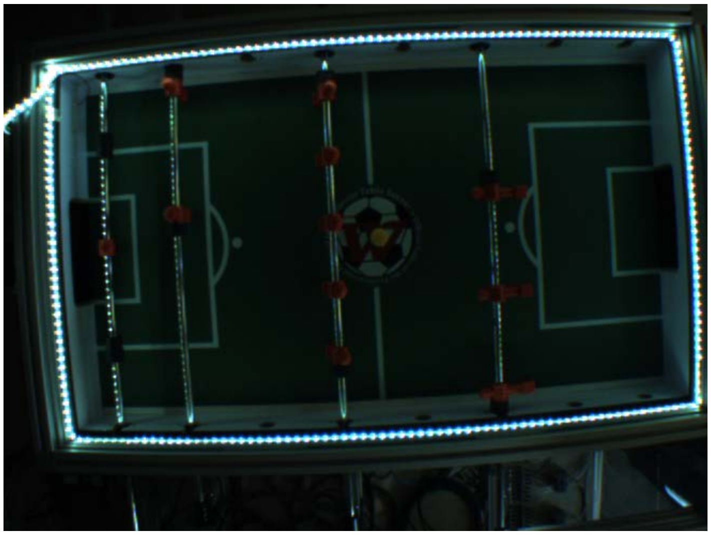


Figure 32: Camera view in a dark room with LEDs

Although the table is not completely bright, it is clear that LEDs are a viable option for lighting. These are incredibly cheap LEDs, so we are confident that more powerful ones will light the table better. Also, a diffusing layer of plastic can be added to reduce the brightness of the individual LEDs so they do not cause eye strain for the human player.

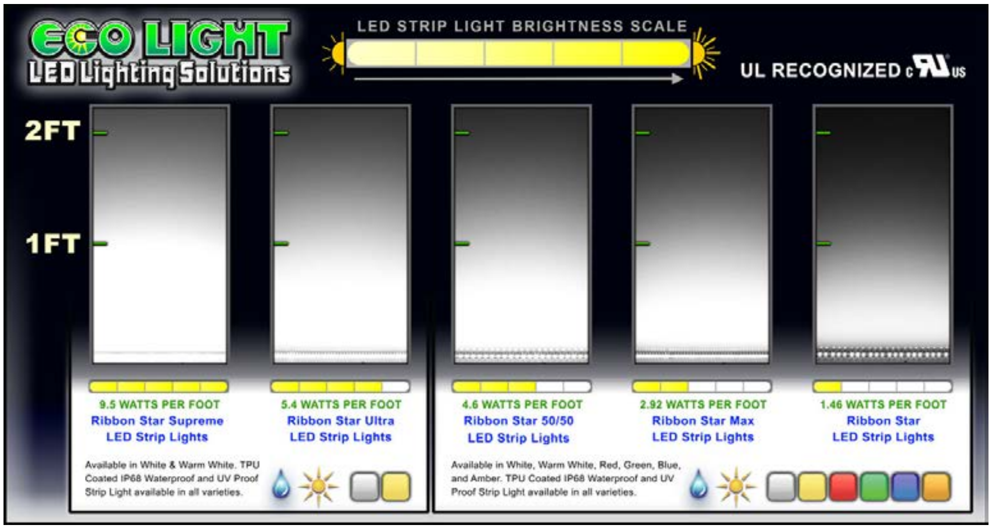


Figure 33: Practical LED strip light brightness ratings

Roof Design

Description

The main criteria for the roof was that it would be light-weight and easy to assemble/disassemble. The design needed to be light because it will need to be lifted up above the vision arch and it needed to be easy to assemble and disassemble because the whole system needs to be broken down every time the table is moved to a new tradeshow.

Design

Using simple geometry, we were able to calculate the size of the roof needed to block all direct light reflected into the camera. (See calculations In Appendix E). After the size was calculated, it was clear that the roof would need to be disassembled into multiple parts in order to reduce spaced needed in shipping the system. After researching many building material options including pvc pipe, t-slot extrusions, and many others, we decided on using Bosch Rexroth Ecoshape tubing. The Ecoshape tubing has the benefits of being relatively inexpensive, lightweight (it is made from aluminum), easy to assemble (its connectors only require an allen key), and easy to interface with the vision arch (it has a profile with a 10mm t-slot). The photo below shows the roof frame design by itself and the photo below that shows it installed into the vision arch.

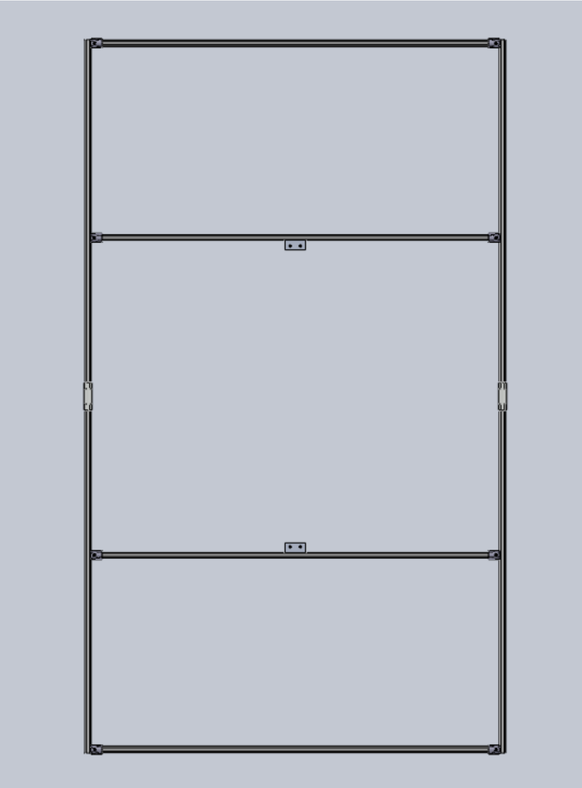


Figure 34: Top view of roof frame

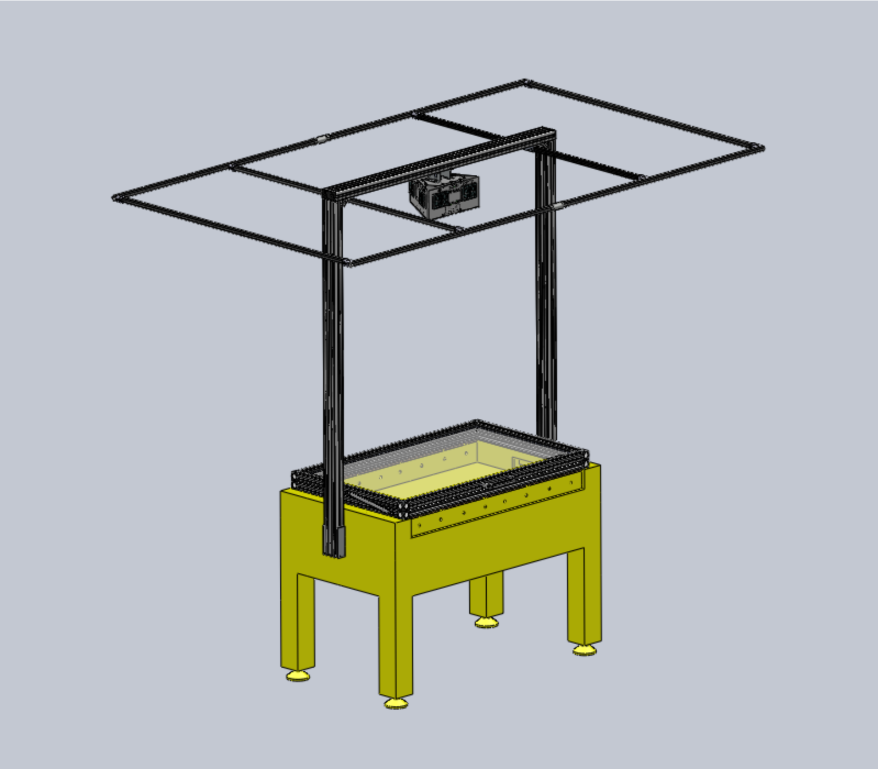


Figure 35: Roof frame installed in table arch

These solid models do not show that there will be fabric that sits on top of the frame and vision arch. This fabric will be secured to the frame using Velcro.

Approximate beam deflection calculations were done to make sure that the frame would not sag and look unprofessional (see calculations in Appendix E) and the worst result was 0.9mm, so we can be confident that the structure will not sag under its own weight, or under the weight of the fabric.

Gusset Plate

Description

Currently, the vision arch does not vibrate when the foosmen move. After some of our design changes though, this will probably not be the case. First, because we need to keep the table and the cabinet aligned, the brackets are needed to fix the two together, which means that the load from the motors moving and stopping can transfer into the table. With this load in the table, the vision arch will most likely shake. Adding the roof to the top of the vision arch will then amplify this vibration because it is adding mass to the end of a cantilever beam.

One way to counteract this vibration, besides damping, is to make the vision arch more rigid. The simplest way to do so is to design a gusset plate. This plate would bolt partway up the vision arch and then downwards into the table.

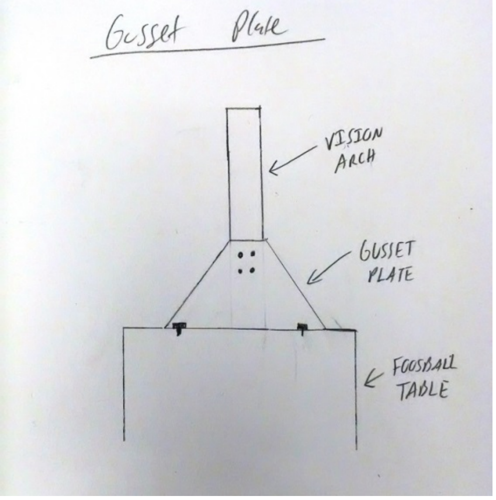


Figure 36: Gusset plate sketch

Calculations

Because the gusset is designed to make the vision arch more rigid, we wanted to see how it would affect the natural frequency of the vision arch. We modeled the arch as a cantilever beam with a mass attached to the end. An example calculation for the natural frequency of the vision arch without the gusset can be found in appendix E . We then assumed that everything below the height of the gusset plate would be rigid, so in essence, the gusset plate increases the natural frequency by shortening the cantilever beam. The results of our calculations back this claim up (see table and chart below).

Table 11: Natural frequencies of the vision arch with different gusset heights

Gusset Height(m)	Effective Beam Length(m)	Natural Frequency(rad/s)	Natural Frequency (Hz)	Period(ms)
0	1.827	54.5	8.67	115.34
0.1	1.727	59.5	9.47	105.61
0.2	1.627	65.3	10.39	96.21
0.3	1.527	72.1	11.47	87.15
0.4	1.427	80.1	12.75	78.43
0.5	1.327	89.7	14.27	70.07
0.6	1.227	101.2	16.11	62.06
0.7	1.127	115.5	18.38	54.42
0.8	1.027	133.2	21.21	47.15
0.9	0.927	156.0	24.83	40.28
1	0.827	185.9	29.58	33.81
1.1	0.727	226.4	36.03	27.75
1.2	0.627	283.8	45.17	22.14
1.3	0.527	369.8	58.86	16.99
1.4	0.427	509.1	81.03	12.34
1.5	0.327	762.8	121.41	8.24

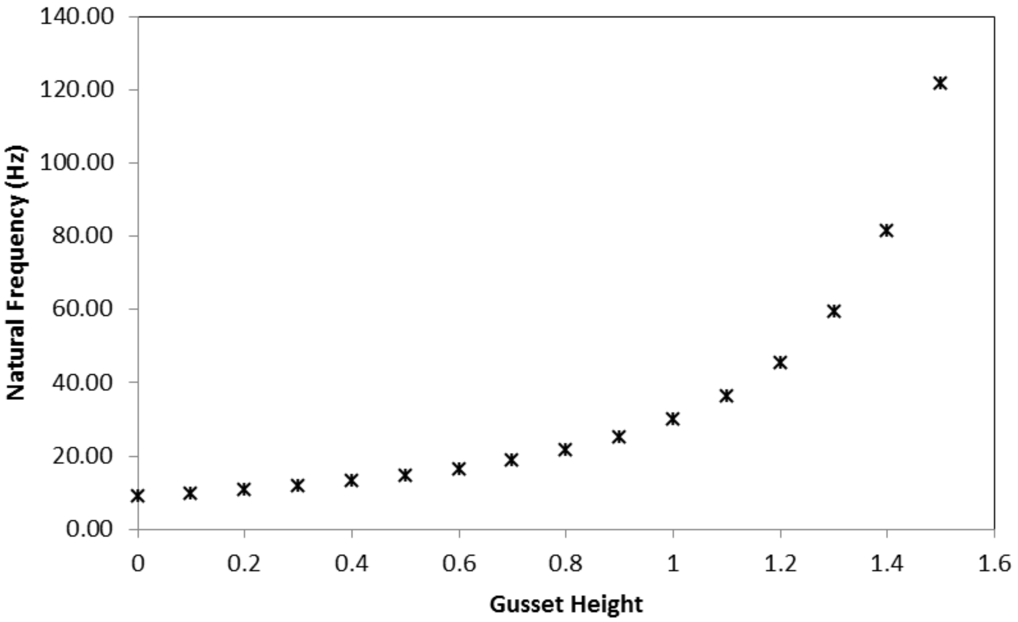


Figure 37: Effect of gusset height on vision arch natural frequency

Still, because the players do not move side to side in a periodic manner, it is difficult to use this data to know the optimal gusset height to reduce vibration. We will use this data when programming in order to create optimal move profiles that will not excite the vision arch at its natural frequency.

Plan of Action

Because we cannot be sure that the gusset plate will help prevent the vision arch from shaking, or is even necessary, we will hold off on building a gusset plate. We will first try to fix the issue by using rubber in the brackets to damp out the vibrations before they reach the vision arch. Then, if necessary we will manufacture the gusset plates help stiffen the arch and raise its natural frequency outside the range of the movement of the players.

Laser Alignment Tool

Description

Although the brackets should keep the table aligned with the cabinet, there is the possibility that the motor brackets could be shifted during transport, or that the table and the cabinet or on uneven surfaces. In this case, the technicians who set up the table will need to realign it. We decided that we would design a tool that could make the alignment process easier and more accurate than by eye. The design that we settled upon is a cross-hair laser that sits in the shaft coupler and a target that sits in bushing hole on the far side of the table.

Design

There are two main components for this design, the target and the laser carrier. We decided to make both parts out of Delrin because is both soft compared to metal and extremely easy to machine. Below is a 3D model of the target.

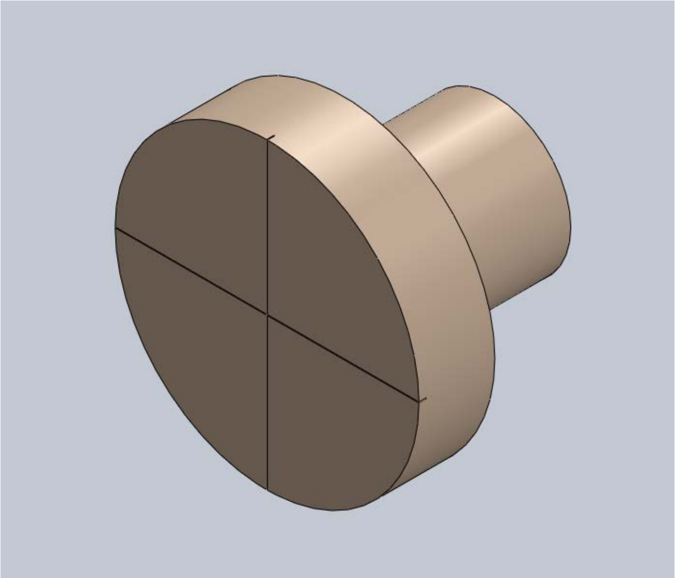


Figure 38: Laser target CAD model

This target will sit in the 1inch hole that is on the far side of the table. Two perpendicular lines will be scored into the face of the target in order to allow the technician to see the center of the target easily. This process will be completed on a mill so the crosshair is aligned perfectly.

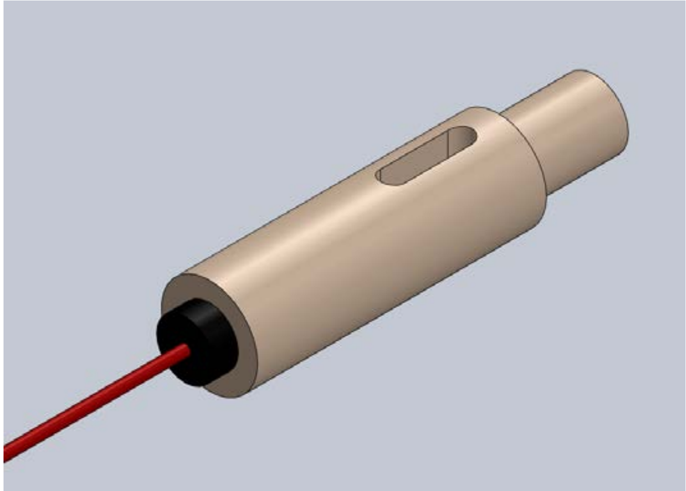


Figure 39: Laser Carrier CAD model

The laser carrier is the other component in the alignment tool and will be inserted in the coupler that fastens the shaft to the motor. The laser will have a crosshair pattern that will shine towards the target on the other side of the table. When aligning the motors, the technician will first align both the laser and target crosshairs and then slowly spin the shaft coupler. If the centers of the crosshairs still match, then the shafts are aligned perfectly. The slot at the top of the laser carrier is to allow for the laser diode's wires to be connected to an external battery pack.

<u>Table to Cabinet Gap Safety Cover</u>

Description

There is a narrow gap between the foosball table and the motor cabinet that is not covered by either the playfield cover or the main motor cabinet top door. This space is required to install and couple the foosmen rods. In operation, this gap should be covered to promote safety of users and prevent damage to the equipment housed in the motor cabinet. The cover will be made of two narrow sheets of clear plastic sheet, from the left over material for the main playfield cover. The narrow sheets will be permanently fixed together at an angle with the appropriate acrylic adhesive. This safety cover will be easy to install and remove between set-up and operation while maintaining the overall aesthetic of the motor cabinet. Figure 40 shows Solidworks models of the cover alone, and Figure 41 displays where it would be attached to a section of the motor cabinet. Drawings for the cover can be found in Appendix D.

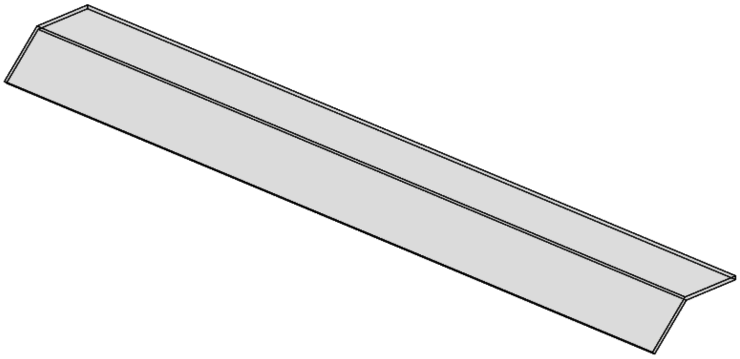


Figure 40: A Solidworks model of the gap cover which will be mounted above the rods between the motor cabinet and table.

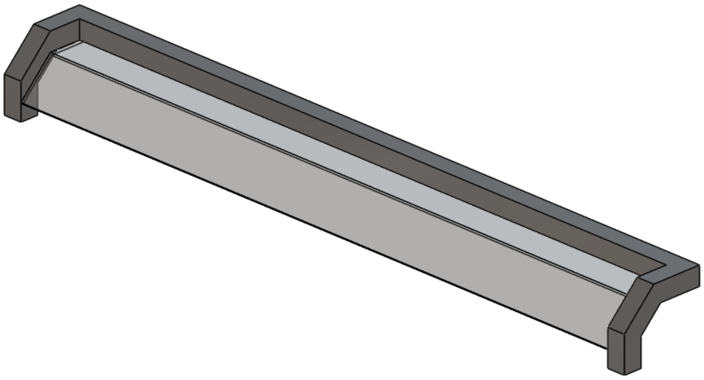


Figure 41: A Solidworks model of the gap cover shown installed in a section of the motor cabinet.

Material Selection

Eighth inch thick PGET sheet from McMaster-Carr was chosen to be consistent with the other clear panels on both the motor cabinet and the playfield cover. Left over material from the playfield cover will be sufficient to cover the gap.

Programing

This section is a discussion of the current programming plan for the system. It will detail a general outline for the program, which will include major tasks and subtasks. This plan will be used to generate a prototype for the program and will be used as guide in verifying all requirements for the prototype are met.

Major Tasks and Subtasks

The main program can be broken into several major, largely independent tasks. Each of these tasks will handle one of the major functions which the system must be capable of performing. The following section will detail each of the major task and the function it seeks to fulfill. The section also includes a description of the subtasks each major task will include. Appendix H contains state transitions diagrams for each of the major tasks, with the subtasks acting a states.

- **Rod Control** – The main function of the Rod Control task will be to control the motors which drive the foosball rods. It will use the position and trajectory of the ball to determine what each of the rods should be doing at any given time. It may also be used to generate strategies for the rods, though this function may be handled in a separate task.
    - **Defense** – This subtask will control the defensive portion of the AI.
    - **Offensive** – Controls the offensive portion of the AI.
    - **Strategy** – Controls the best course of action for the AI system.
    - **Home Move** – Performs a homing move.
    - **Shutdown** – The motors are stopped and depowered
    - **Test Move** – Performs a test move
    - **Idle** – Motors are powered, but not moving
- **HMI** – This task will control the HMI used by both the player and the operator of the table to select settings and gather information about the table. It must be able to interact with the display used to make the physical portion of the HMI system.
    - **User Interface** – deciphers user inputs. Used for difficulty selection, start, stop and rest commands.
    - **Operator Interface** – Used by the operator to set up the machine for play and to insure the system is operating properly. Used to: set home, perform a home move, perform a test move, test capture.
- **Ball Tracking/Prediction** – This task will interface with the vision system to retrieve ball kinematic data. This data will then be used to plot the trajectory of the ball, which will be sent to the Rod Control task for processing. The complexity of this task is largely based on abilities of the vision system.
    - **Retrieving Ball Data** – Collect data from the vision system.
    - **Predictions** – perform any required predictions based on ball data
    - **Send Data** – Send data to the Rod Control Task
    - **Idle** – Waits for game to start
- **Safety System** – The Safety System task is the most important of the tasks. It will monitor the safety switches mounted on the playfield cover and motor cabinet, and if the switches are tripped, it will shut the motor system down. It will also be responsible for re-starting the motors once the switches are re-engaged.
    - **Monitoring Safety Switches** – Detects switch states.
    - **Shutdown** – Shuts system down if a switch is tripped.
    - **Restart** – Restarting after the switch is reengaged
- **Initialization** – This task will handle any initializations of the system which is required.
- **Score Keeping** – The Score Keeping task will control all aspects of the score process. This includes detecting when a goal is made, tracking the score, displaying the current score and resetting the score when the game is finished.
    - **Score Tracking** – Keeps track of score of the current game.

  - **Display Score** – Displays the score of the game on the scoreboard
  - **Reset** – Resets the score and scoreboard.
  - **Goal Detection** – Detects when a goal is made.
  - **Idle** – Waits for game to start

Additional Planned Improvements

Safety Switches

Description

The emergency stop sensors will be installed on each of the three doors of the motor cabinet, as well as the playfield cover. These safety sensors will signal the PLC to stop moving when the doors are opened. The previous group chose magnetically actuated switches available from McMaster-Carr. These contactless switches will not wear over time as a mechanical switch eventually would. Information about the chosen switch (65985K11) can be found in Appendix J.

HMI

Description

A human machine interface (HMI) will be included in the system for two modes of use. The first user interface (UI) will be designed for the human player. It will include controls for difficulty setting, start/pause game and reset. The second UI mode will be for debugging the system during set-up or trouble-shooting. It will include information about the state of the system as well as controls such as home axes, or individual rod control. Currently the system is controlled through the PC, a touch screen HMI has been requested from Yaskawa and should be available before the end of this quarter.

User's Manual

Description

An operating manual will be created containing information on system capabilities, how-to for setup procedures and a trouble-shooting section. There will also be a section with a bill of materials and assembly instructions to assist in producing more tables. Within the bill of materials, vendors and other contacts will be included.

Goal Sensor

Description

Ball sensors will be installed in both goals to keep track of the score automatically. The sensors will be monitored by an Arduino Mega microcontroller, which will also update the 7-segment displays installed in the overhead score board. Using the Arduino microcontroller will free up IO ports for the PLC and reduce the complexity in the programming on the PLC by shifting the score keeping to a separate system. A communication line could be used between the PLC and the microcontroller to control the display for various messages or score resets.

Material Selection

The sensors will be constructed with momentary contact switches with a custom flap to intercept the ball as it is funneled towards the ball retrieval port at the front of the table. This physical contact sensing will prevent false or missed goals that may have been a problem with noncontact sensors such as PIR sensors.

Vision System

Description

Currently, the table's vision system consists of the Basler acA640-120gc camera feeding images to be processed by the PC. This system uses hue to track the ball, but it does not process images at an acceptable speed, or frames per second (fps). To improve the performance of the vision system, a new system has been requested from Cognex. Cognex vision systems consist of both a camera and an integrated image processing box. Cognex has donated two In-Sight camera systems, the 7400C and 7400. The first is a color camera with an improved fps which will allow for hue tracking as the current system does. The second camera runs at an even greater fps though it is capable of only gray-scale image capture. This camera would utilize pattern recognition to track the shape of the ball. Information about these systems can be found in Appendix J.

Chapter 4: Product Realization

This section contains descriptions of the final Foosball system. This includes the hardware produced for the project, the code produced to operate the system and other goals which have been completed.

Hardware

Table to Cabinet Brackets

Fabrication

The L-shape and plate brackets, shown in used to connect the table to the cabinet were manufactured out of 6061 Aluminum. They were cut to their rough dimensions and then precision machined using a mill to achieve final dimensions and to place the required bolt holes. In the case of the plate brackets, shims were manufactured to account for the slope in the tables legs. Table 13 and Table 14 contain the detail procedure used to fabricate L-shaped and plate brackets respectively.

Table 12: The procedure used to fabricate the L-shaped brackets.

Step	Description
Rough Cut	**Two section of 6061 aluminum 90 degree stock (4 in. leg length) were cut to roughly 4 in. in width using a horizontal band saw.**
Square and Final Width Dimensioning	**The cut ends of the brackets were squared using a mill. The width of each bracket was then made 4 in. using a mill.**
M8 Holes	**The three 8 mm holes were drilled using an 8 mm drill bit on a mill. These holes were difficult to place due to the angle of the table legs and great care was taken during this step.**
½ inch Holes	**The two ½ in. holes were drilled using a ½ in drill bit on a mill.**

Table 13: the procedure used to fabricate the plate brackets and the shims.

Plate Bracket	
Step	**Description**
Rough Cut	**Two plates, roughly 5.6x4 in., were cut from a of 6061 aluminum plate using a vertical band saw.**
Square and Final Width Dimensioning	**The edges of each plate were squared using a mill. A mil was then used to produce the final dimensions of the plate.**
M8 Holes	**The three 8 mm holes were drilled using an 8 mm drill bit on a mill.**
½ inch Holes	**The two ½ in. holes were drilled using a ½ in drill bit on a mill.**
Rubber Shim	
Step	**Description**
Cutting	**A razor blade was used to cut a 3x4 in. section of 70A fiber reinforced neoprene.**
½ inch Holes	**Two ½ in. holes were drilled using a ½ in drill bit and a hand drill, using the plate brackets as a guide.**
Wooden Shim	

Step	Description
Cutting	**A XxX section was cut from a ¼ in. thick piece of wood using a vertical band saw.**
Angling	**A belt sander was used to shape the angle of the shim. The angle was first approximated and then refined by placing the shim in its place. When the shim fit between the table and the bracket, angling was complete.**
Painting	**The shim was painted with black glossy paint.**
½ inch Holes	**Two ½ in. holes were drilled using a ½ in drill bit and a hand drill, using the plate brackets as a guide.**

Instillation

The brackets were position on and attached to the motor cabinet using the M8 t-slot bolts. A hand drill and ½ in. drill bit were then used to drill holes in the table through which the ½ in. bolts were to be placed into. 2 in. long ½ in. bolts were then placed in the top brackets and 6 in. long bolts were placed in the bottom brackets. Nuts were the applied and tightened.

Roof

Because we tested the effectiveness of a roof in eliminating glare before we built the roof, its purpose did not change when implemented, though the overall change was quite large. When we first mounted the roof to the vision arch, it was clear that due to the flexibility of the couplers and the length of the cantilever that it would vibrate far too much to appear safe to the user. In order to make the roof more rigid, we extended poles upwards and then used wire rope to attach the corners to these poles. The roof is now rigid enough so that its vibrations are due to the movement of the table and not its own flexibility.

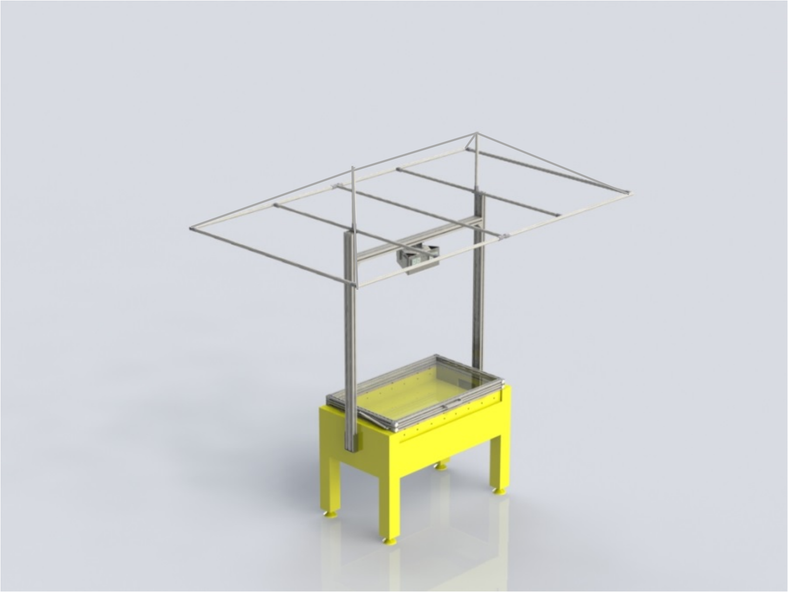


Figure 42. Final Roof, table, and playfield cover design

Because we are not experts at sewing, we decided to attach a tarp to the roof to act as the main method of blocking light. Currently, the tarp is silver and does not match the frame of the roof perfectly, but we have ordered a blue one (to match Yaskawa's logo) that fits the frame more correctly. The drawings for the new roof can be found in Appendix D.

Gusset Plate

Because we thought that the extra weight of the roof could create large vibrations in the vision arch, we designed a gusset plate to help stiffen the arch. After we noticed that the vibrations in the vision arch were small enough to have little effect on the camera performance, we decided to spend our time on more important issues like programming and calibrating the camera.

LEDs

After installing the LEDs, it became clear that they were bright enough that they could be abrasive to the player's vision if they looked at them for an extended period of time. To combat this issue, we inserted sills made from 90 degree aluminum angle iron into the playfield cover. These sills are large enough to block the light from hurting the player's eyes and small enough to not block the camera's view of the edges of the table. Below are two photos that show how the sills block the light from the LEDs.

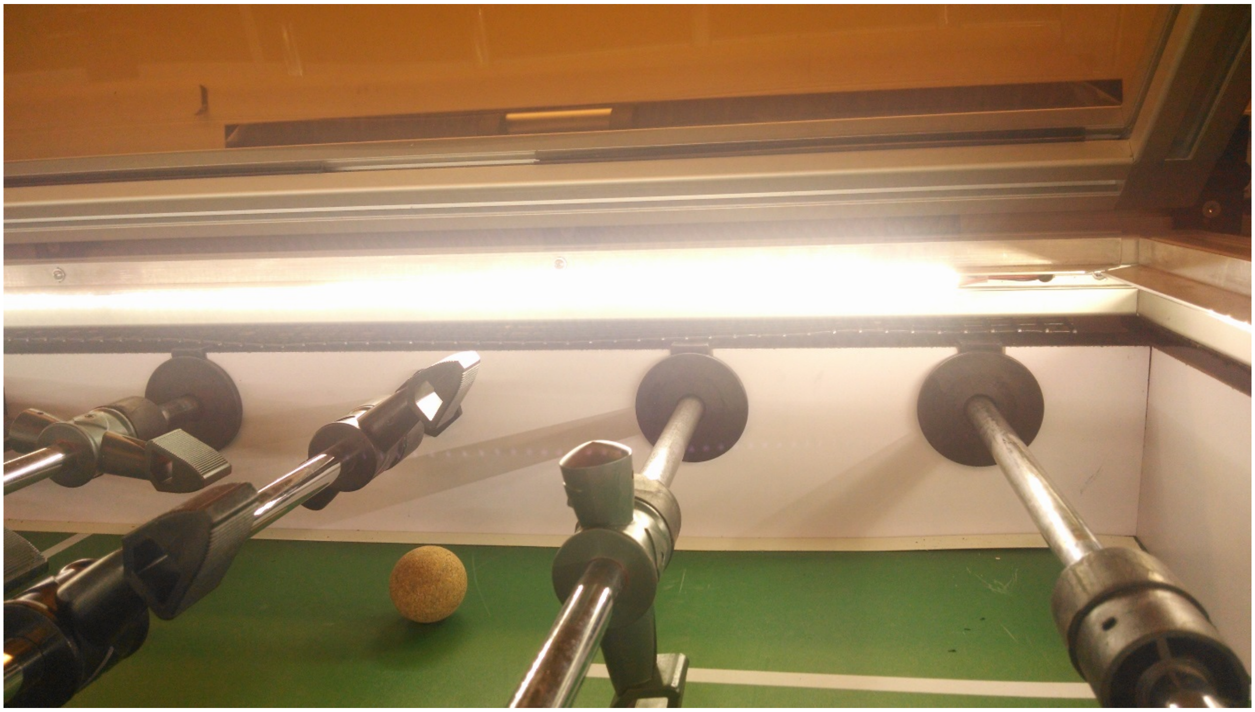


Figure 43. Below the sills, the LEDs are blindingly bright

Figure 44. The aluminum sills block the blinding LEDs when the view is from above

Gap Cover

The gap cover was made as a safety feature close the space between the motor cabinet and the foosball table. It is made of a steel mesh that are small enough to prevent fingers from reaching in and encloses the gap completely. It is bent at one end of the area to close the open vertical gap that was caused by the different lengths of the table and the cabinet. The gap cover is held in place by T-slots in three separate areas and when fully fastened is rigid and does not move. The cover sits in the 80-20 grooves along the beams, with the T-slots holding it in place where the grooves are above it in the cabinet. The mesh is easy to install and only takes one or two minutes to fully put in place. A metal mesh material was chosen over plastic or aluminum plating due to the flexibility needed in the design and the ease of installation a mesh provides. The black coating on the mesh also makes it blend in fairly well with the table and prevents it from being a large distraction.

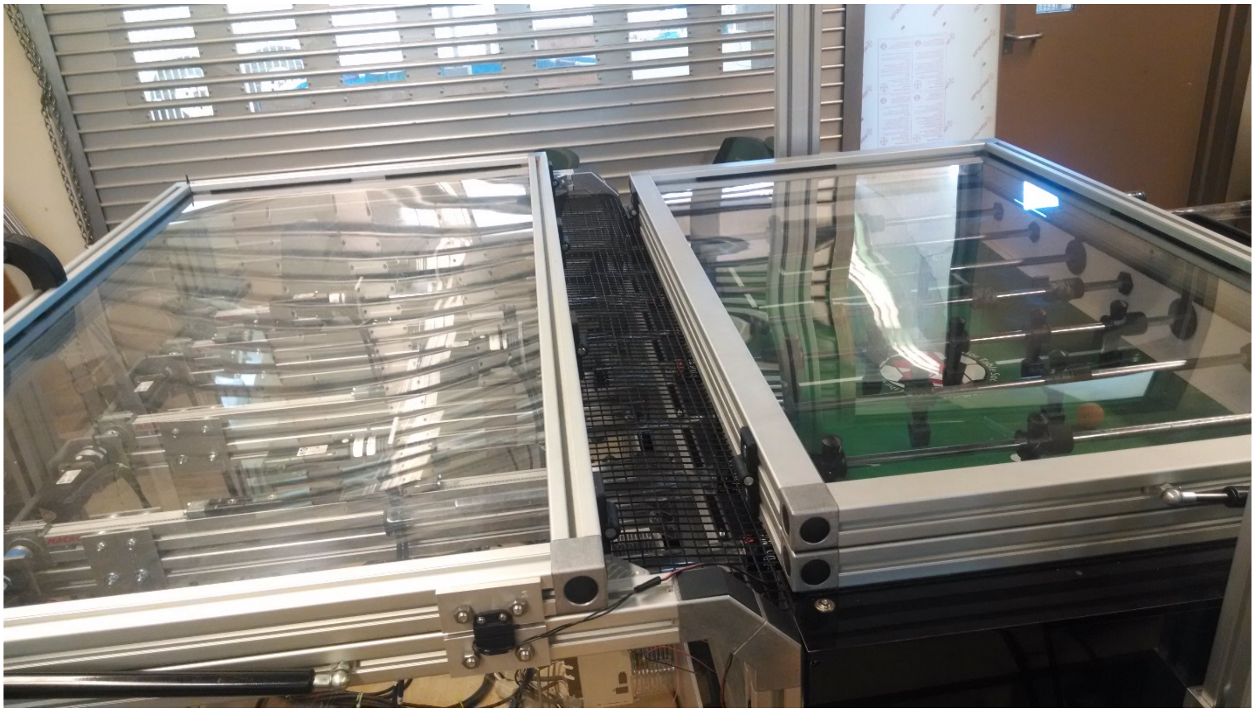


Figure 45. Gap Cover that protects the user from the moving rods

Replacing Actuator Beams

One of the improvements which was identified for early completion was the replacement of the two cantilevered supports, Figure 42, with a single 8020 strut, Figure 43. This improvement is meant to increase the rigidity of the table and to reduce the load carried by the actuators. This was completed relatively easily because the previous team left sufficient 8020 stock to cut four supports to the proper length.

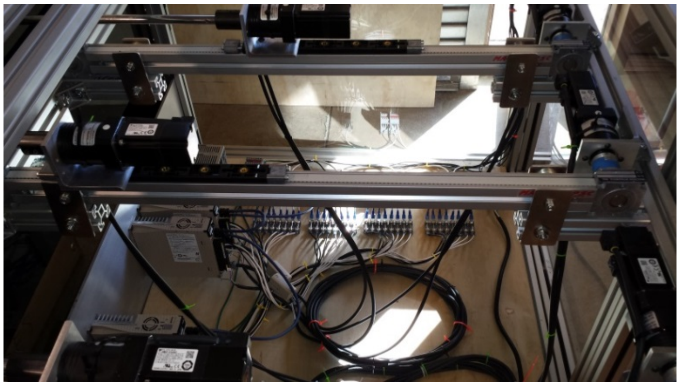


Figure 46: Some of the cantilevered struts which were replaced

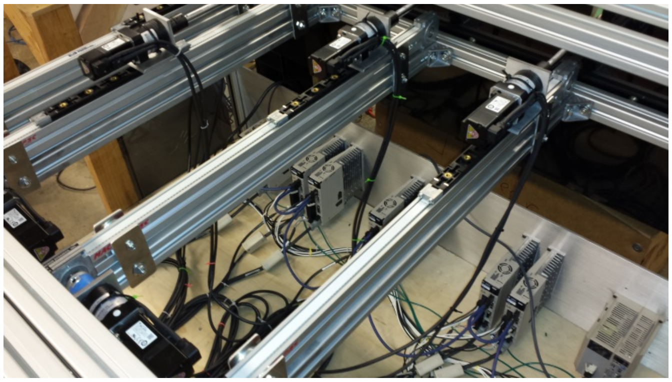


Figure 47: The actuators after the cantilevered beams were replaced

Reconfiguring the Goalie Linear Actuator

The goalie actuator was originally configured in such a way as to cause the actuator carriage to crash into the end of the actuator before the rubber stopper on the foosball rod hit the table. This could cause significant wear on the actuator, and the impact was extremely noisy. With the help of a Macron dynamics representative, the actuator was reconfigured to prevent a crash from occurring. Figure 44 shows the actuator before it was reconfigured, and Figure 45 shows it after it was reconfigured.

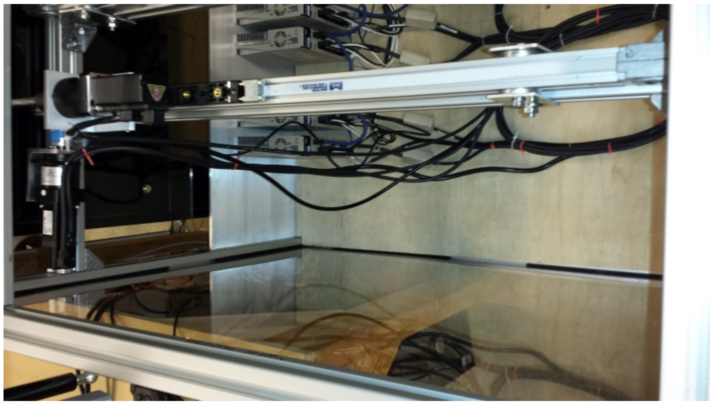


Figure 48: The configuration of the Goalie rod before it was changed

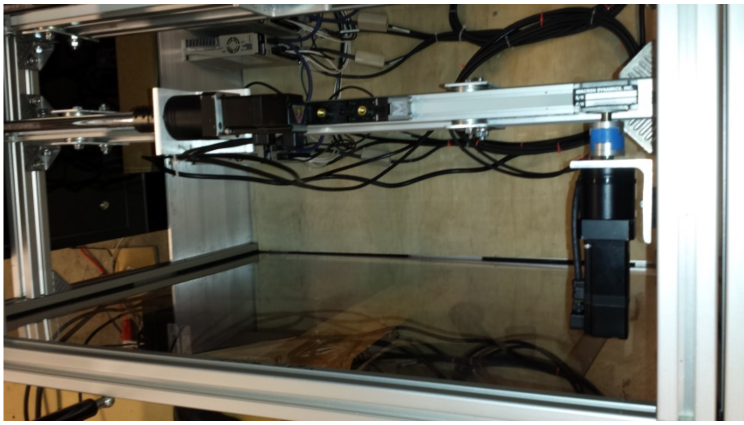


Figure 49: The current configuration of the Goalie actuator

Repositioning the Motors

To attach the brackets to the motor cabinet and table successfully, the motor cabinet's position relative to the table needed to be modified. The right side of the table was made flush with the right side of the motor cabinet. Because of this shift, the motors needed to be moved approximately 3cm to the right of their previous position. This modification was made with little difficulty.

Safety Switches and I/O Module

The safety switches for the top of the motor cabinet and the playfield cover have been installed on brackets and their electrical leads are ready to plug into the PLC's I/O module. Though, without a power source for the I/O module, they have not been fully implemented. The switches for the front doors of the motor cabinet were not installed.

Figure 50. Installed I/O module (right)

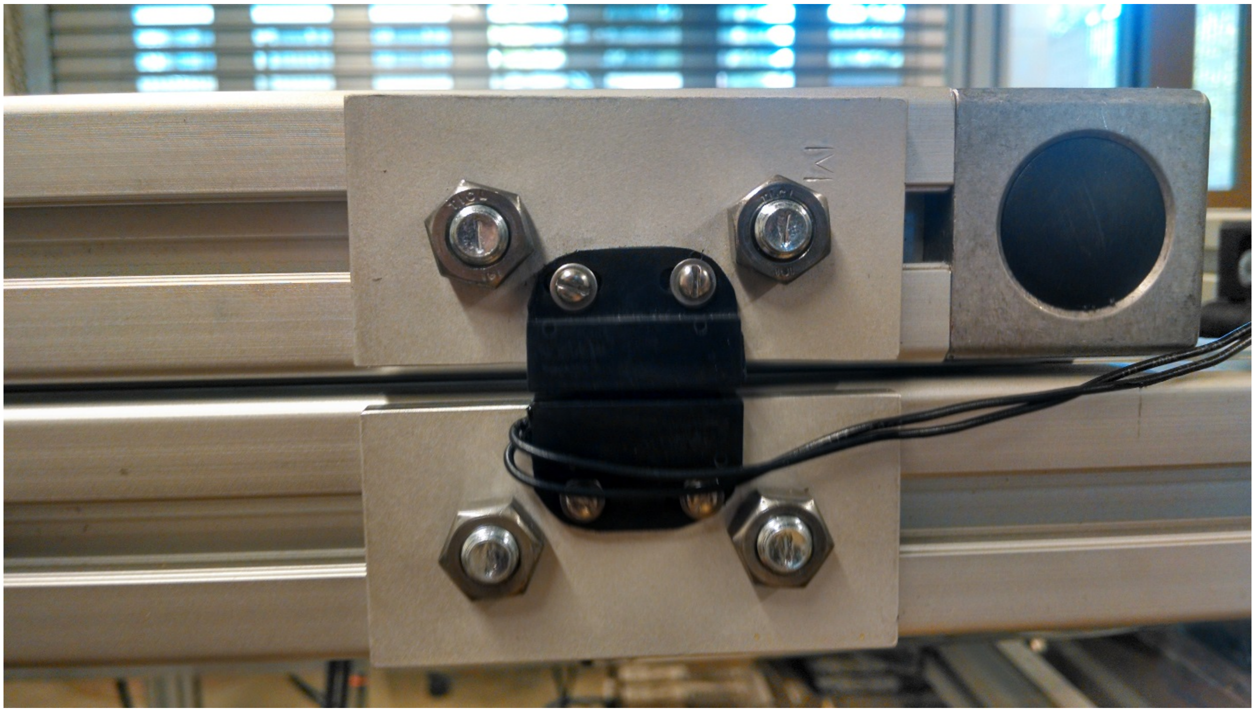


Figure 51. Installed magnetic safety switch

Programming

Overall Goal

Because setting up the camera to communicate with the PLC took longer than expected, we did not have much time to code before our sponsor came to visit on the 22nd of May. Instead of building up towards a complicated AI that has many different states and possible actions, we wanted to see if we could get the code so it would block the ball and kick it forwards. With this amount of code, we would at least be able to test the performance of the camera and make sure that the camera is an acceptable method of tracking the ball. Also this basic program would demonstrate the effectiveness of the motors in translation as well as rotation. Additionally, we were able to ensure that the code we produced was modular and could be used in later versions of the project as a library.

Function and Program Description

This section contains descriptions of the different functions and programs developed during the project. Appendix K contains screen captures of each item described in the following section.

Rod Enable

This function block enables both motors on the given rod. It allows the rod to be enabled or disabled, and for the rod to be reset. It also informs the user if the rod was successfully enabled and if the motors have encountered an error during operation.

Zero Single Rod

The purpose of this function block is to set the zero position for both the translational and rotational motors on the given rod. As there are no limit switches on motors, it is necessary to place the rods in the zero positions by hand. The rods should be zeroed after an alarm, an unexpected power down or the system, or if an error in position is noticed by the operator.

Single Rod Translation

This function block is used to translate the given rod to a desired position within the physical limits of the system. The block accepts an input in the form of a real position. It then determines if the given position is within the minimum and maximum movement range and if it is not the result is saturated at the appropriate extreme. After the position is accepted or saturated, the move command is issued to the motor to translate to the desired location.

Set Rod Angle

This function is used to set the angle of the rotational motor of a given rod. The desired angle is input into the function block and the move is executed. The direction taken by the motor is shortest route from its current position to the desired position. The block includes torque monitoring, which will prevent the rotary motor from overloading if the ball is caught beneath the foosman.

Rotational Torque Monitor

This function block prevents the rotational motor from overloading during a move. The block is constantly monitoring the torque of the given motor, and in the event that the torque reaches the set limit, the block rotates the motor in the direction opposite to the increasing torque. It outputs a signal which can be used to prevent any other actions to be taken on the rod until the unjamming move is complete.

Rod Information

This function block reads and outputs the current torques, positions and velocities of the translational and rotational motors of a given axis.

Kick Function

This function block is used to execute a kick with the rotary motor of the given axis. The kick consists of three distinct moves. The first is the windup which moves the foosmen back in preparation for the kick. The next move is the kicking move, which sweeps the foosmen quickly forward through an arc which terminates near the foosmen’s maximum reach. The final move is the return to zero move, which moves the rod back into the zero position. This block includes a torque monitoring block to prevent the rotary rod from overtorquing.

Home Single Rod

This function moves the rotary and linear motors of a given axis back to their zero positions. The motors should be homed before operation of the system to ensure that zeroes are properly set.

Rod Position Logic

The purpose of the rod position logic function blocks is to take the desired y position of the rode and tell the translation motor how much to move in order to place a foosman at that position.

These function blocks work using zones. For example, on the three man rod, there are three zones. Each zone is one third of the width of the table (excluding the width of the bumpers). If the desired y position is in the 1st zone, then the rod will move that distance minus the width of the bumper. If it is in the second zone, the rod will move that distance minus the width of the bumper and the distance between the first and second foosmen. This means that the second foosman will be at the desired y position. This pattern repeats for all of the different rods; the rods with more players just have more zones.

Rotation Logic

The purpose of the rotation function block is to allow the rod to rotate three different ways. First, if the ball is further up the field than the rod, the foosmen should be pointed down to block the ball. If the ball is near the rod, then it should kick. Finally, if the ball is behind the rod, it should flip up to avoid blocking kicks from the rods behind it.

This function block works by toggling variables which are attached to three different action blocks which are described above.

Defense Logic

The purpose of the defense logic is to make the last two rods work in tandem in order to block more of the goal.

The first goal of this function block is to make sure that the last two foosmen stay within the goal as long as the ball is in front of the rods. If the ball is outside of the goal, the last two foosmen guard the post closest to the ball. Once the ball is in front of the goal, the foosman on the second rod stays slightly to the inside of the ball and the goalie staggers slightly to the outside of the ball. This essentially creates a double-wide defender.

Ball Position Logic

This is a standalone program, executed before main, which reads the incoming ball position data from the vision system and passes that information onto main via global variables. When the ball is lost by the camera, zeros are sent to the PLC, and so zeros are rejected by this function and the previous valid position is maintained by the system.

Main

Main is an amalgamation of each of the function blocks described above into a working and playable, if simple, AI program. It also includes the necessary blocks to perform set up before play begins. It is broken into three main sections. The first contains the rod enables for the rods, the zero position functions for the rods and the homing functions for the rods. The next section contains the translational logic for the rods and the translation functions required to move each rod. It also contains the defensive logic used on the last two rods. The final section contains the rotational logic for the rods and the set angle and kick functions required to move the rods.

Vision System

Overall Setup

The vision system for the project ultimately utilized a Cognex 'insight 7400' (7400) gray-scale camera, for the vision and tracking functions of the automated foosball table. Using the 7400 allowed us to achieve 20-25 fps data update rates while the camera tracked the ball and communicated position coordinates to the Yaskawa PLC.

The camera interface and job creation is accomplished through Cognex's 'In-sight' software package. The software package has multiple programming modes; initially we utilized the basic easy-build method, and for the final implementation, switched the job construction to the spreadsheet-based method in order to eliminate the unnecessary aspects of processing from the camera's job file.

Programming Methodology

The focus for camera implementation was to balance the speed of the job with the ability to find the object, in this case, the foosball. Due to the densely packed nature of the foosball table, in which the ball is often blocked or obscured from the camera this presented a challenge. For ease of implementation and to time constraints, we chose to utilize a pattern match over other types of possibly faster object finding methods such as Blob. The Blob detection method was our initial choice based on its quick run-time. The Blob method was unfortunately limited in that any time the ball touched another object such as a foosmen, or any number of the graphics stenciled on the playing field, the blob would fail to make a correct match. This was something of a critical issue considering the amount of time the camera was unable to identify the ball under normal play conditions.

The detection method that we chose to implement was the 'Pattern Match' approach, which, though slower than Blob detection, gave more overall detection under empirical testing. Conditions in which the ball was visible to the camera, but still touching other objects would still result in a successful detection of the ball and allow the camera to provide accurate data to the PLC. For blob detection the process time could be reduced to approximately 25 milliseconds over all, while for the final implementation of the pattern detection we were seeing process times of about 40-46 milliseconds, which translate into a range of about 20-25 fps.

Ball Tracking Challenges

Since we were using the gray-scale camera as opposed to the color camera, we encountered several detection obstacles resulting from the colors involved in the search area. The main issues tend to be interrelated lighting and color problems. In order to minimize the color problems we decided to paint the player's foosmen, which were default red, to a dark green that blended with the table's green. We did this because we found that the camera would match parts of the foosmen as the ball, especially when the ball was hidden. This became more of an issue because we had reduced the constraints for ball matching to try to increase the amount of times the camera would properly identify the ball in conditions where the ball was partially obscured, such as when a player was moving a ball side to side or masked by lighting variations on the table. The main issue with this condition was that we would have been forced to implement some method of positional sanity check within the PLC to catch conditions that did not make sense. That option would have been troublesome for a variety of reasons, not the least of which is the potential rate of speed of the ball in play combined with the large number of missing positions due to field pieces hiding the ball from detection.

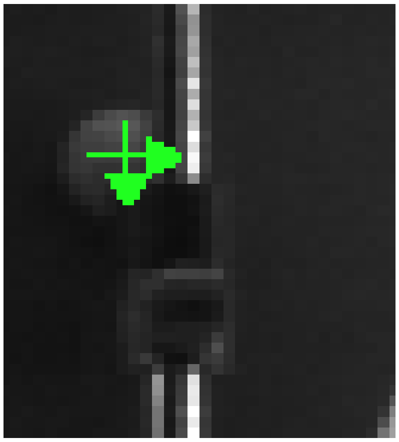


Figure 52. An example of a problem spot. The ball is partially obscured by a player and is only intermittently identified.

By painting the players, we managed to eliminate all objects that were not the ball from being identified as the ball. This scenario was ideal because it resulted in positional data of (0,0) for x and y respectively when the ball was not found. For all other conditions the camera would return a calibrated to millimeter x and y position for the PLC. The other color change we made was to switch from a yellow ball to a lighter colored, cork ball. The cork was much closer to white, and had the added advantage of a less mass and a lower friction coefficient. The increased contrast between the light brown of the cork and the light color spaces on the field, under consistent lighting, increased the time the ball is successfully detected.

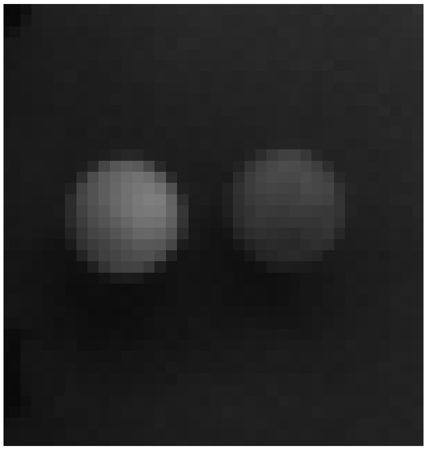


Figure 53. Yellow ball (left) vs cork ball (right) in grayscale.

Communication with the PLC

The PLC and the camera both supported a number of different communication protocols, such as PROFINET or TCP/IP Modbus. We opted to select the Ethernet over IP protocol (EIP) for its simplicity and low overhead of processing. The implementation of EIP allows us to configure the communication between PLC and camera so that no commands to the camera, from the PLC are required in the PLC program. EIP is a broadcast protocol, in which the PLC shouts a request for update (RPI) on the camera's 'input channel', at which point the camera sends the data in its register. The is7000 has built in status bits located in the EIP instance memory (registers). Monitoring these allows for comm verification after the hardware profile for the device is created within the PLC program. The backup method for communication was going to be to use the TCP/Modbus protocol, which is more cumbersome and requires more computing overhead and command coding. Since minimizing the job time of the camera was the primary objective and the MP3200iec PLC we were using has a free 10/100 Ethernet port available, the primary choice was clearly EIP.

The two hardest parts about getting the vision system up and running were the configuration of the communication protocols within the PLC and the camera. Both devices use their own respective proprietary interface software. Cognex used in-sight (v 4.9), and Yaskawa used MotionWorks IEC 2 pro. The Cognex in-sight software suite was ideal for rapid integration since it has different interface formats available. The initial hardware setup is fully accessible within its default Easybuilder mode, which is how we started, later switching to the Spreadsheet mode to refine the job times and reduce the overhead placed on the camera by using the default builder. The major hurdle for integration was ultimately a subtle detail that was never properly covered in the help files or documentation of either manufacturer, (about EIP). EIP as mentioned above is based around input and output instances (registers), the camera has fixed size instances, and the available literature indicates that for EIP to operate properly, the instances defines in the PLC need to match the byte size of their respective camera instances. As an automated consequence of the hardware configuration within MotionWorks, global variable(s) memory is allocated, and at the same time a communication status variable is created within the PLC code. We configured the devices according to the respective instructions found within associated software help-files and manuals, but encountered a conflicted status for the camera communication while in the Debug mode of the running PLC program. The status variable indicated alternating connected/reconnecting. We created several other global variables in our PLC code to monitor bits we knew would change while the camera was running to monitor the connection, but were not seeing any data. After numerous sessions with both Cognex and Yaskawa tech support, and no solution in sight were about to try to implement a different communication method and give up on EIP. As a final attempt before reassessing the interface approach, we stumbled upon a figure within the help file in MotionWorks, showing an older Yaskawa PLC hardware configuration image for an older Cognex camera. We shrunk the instance byte size definitions in the PLC hardware configuration setup to match those in the image to see what would happen, and miraculously our connect/reconnect error had resolved itself. After doing some more investigating into what the problem ultimately was we discovered two Things that would have saved us a large investment of time. The most important thing we learned was that, apparently matching instance byte sizes is not necessarily critical. When we reduced the byte size for the PLC hardware configuration, we defined our input instance to be 32 bytes. This size more than accommodated the EIP required registers and the 4 bytes of data we had the camera outputting for ball position. The second thing we learned later was that Yaskawa has a hard limit on EIP instance size at 498 bytes, so when we were trying to define the instance to be 500 bytes, the software allowed us to without errors, but the PLC was unable to interpret the instance because of the size limit. This was something we were unable to find any documentation on, but what we were told when we were relating our integration challenge to our project's corporate representative.

Image Calibration

Once the camera and the PLC were communicating properly, the focus shifted to optimizing the visual tracking. This included writing our pattern match job in spreadsheet mode to reduce superfluous code within the camera, also calibrating the field of view to reduce the effects of lens distortion on the positional data tracking. We used the dot matrix style calibration routine supported by the In-sight software called calibgrid. This routine uses a large sheet a paper with regularly spaced dots of a known distance, placed within the field of view to calibrate pixel references to known real world axis. This was the most effective calibration of the possible choices we attempted from the many options available within the software.

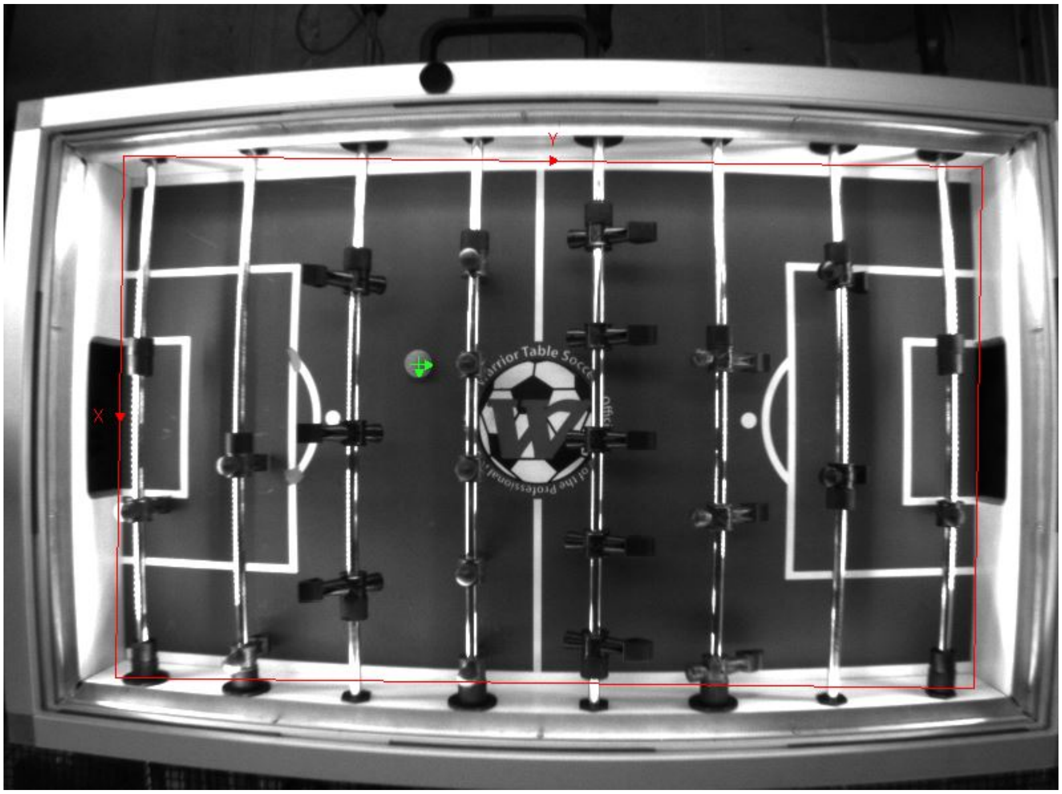


Figure 54. Camera image before calibration

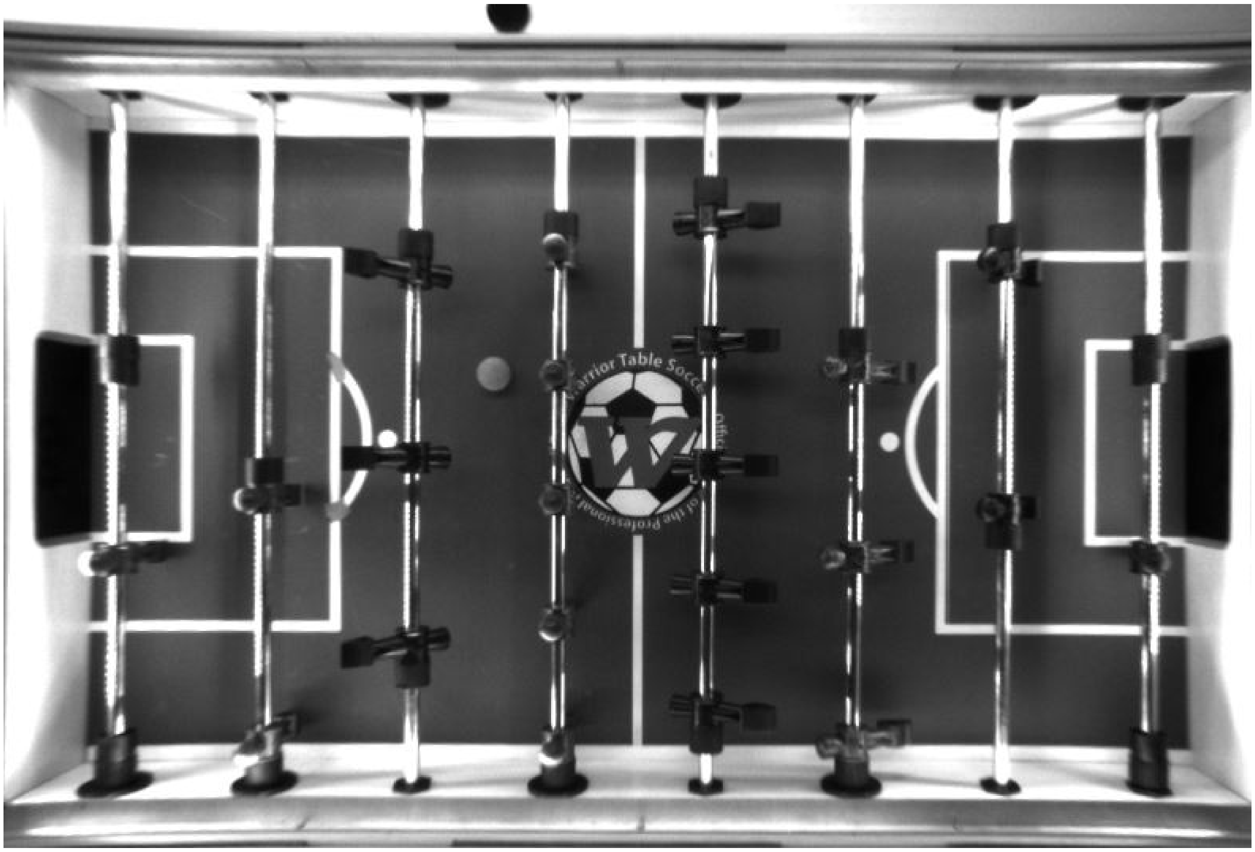


Figure 55. Image after calibration

Image Buffering

Due to the unique challenge of the foosball environment, as mentioned earlier we were only able to empirically test the performance of our algorithms by watching the Livestream of the camera in operation, while concurrently watching the ball position variables update in the PLC program debug mode. In order test and tune more effectively, a network switch was installed, so that the computer could communicate with both the PLC and the camera simultaneously. Before the network switch was installed, we had to manually move the camera's Ethernet cable to the PLC for testing and back to the computer for tuning. Once our network was setup for optimal testing and monitoring we began to notice an unexplainable delay between the real work start of ball motion and the data update in PLC variables. We initially thought the delay was either a network, or software effect, but once the program code for the rods was implemented we observed the delay affecting the PLC play response. The issue we discovered is that the camera has a default internal image buffer that can is adjustable. The default setting is 15 images, which at approximate 40 milliseconds per job, added up to about the half-second or so delay we were noticing. We were able to decrease the image buffer to 3 images, noticeably increasing the real world response of our system.

Wiring Diagram

The wire layout described in this overview (see Appendix I for full diagram) combines an overview of all specific modules and motors, their orientation, and the wires connected to each motor. Standard wires are the green, black, and white wires. The power wire is a large insulated wire that splits into one green, black, and white wire which connects to the power module. A wire number and color specified on a specific module connects to another module if the other module has the same wire color and number. Images where multiple wires are labeled as a specific wire number indicate that all wires are going to attach to the same module and carry the same signal. The PLC connects the amplifiers through the Mechatrolink-IIIbus cables that are daisy-chained together. Attached to the PLC is an added I/O module that allows the camera to communicate with the PLC directly. Each amplifier connects to a specific motor which powers the motors and feeds information back and forth. The amplifiers are buffered by a series of fuses routed through circuit breaker modules which direct current through them. Green grounding wires are used on the amplifiers and attached towards the bottom by the metal grasps. When adjusting wires, ensure the main power cord is unplugged to avoid causing damage to oneself and the equipment. Each switch controls the power supply to a set of amplifiers and the motors attached to those amplifiers.

There are small battery packs that are attached to the motor wires which connect to the amplifiers. Currently there is no way to know if a battery is dead or not, so if there is trouble getting a motor to start, checking the battery is a good starting point. While wiring is being done, always ensure the main power is detached from the power module to ensure safety while handling the wires and modules. Make sure the motor wires are firmly in place, as due to the rapid movement the wires will undergo more stress and strain that any of the other wires.

The wires are labeled with a two digit number as shown in the wiring diagram.

The purpose of each module type is explained below:

- The power module is responsible for feeding power from an electrical source and into the system. Includes switch lines, breaker lines, and grounding lines
- The breaker module feeds power or stops electric flow when the switch assigned to it flips on or off, which in turn goes through a series of fuses before going off to a set of two amplifiers and two motors.
- The PLC module feeds information to the amplifiers
- The fuses act as a dead switch in case of a power surge, these control current flow to the amplifiers
- The amplifier controls motor movements, relays information to and from the PLC, and has grounding wires to prevent static buildup
- The motors control the movement of the rods in the table in linear motion and a radial motion.
- The switches act as switches, either allowing electricity to flow through the specific breaker module, amplifiers, and motors, or it restricts it

Chapter 5: Design Verification

Testing

This section discusses the testing and verification of the Foosball system. Some initial tests have been performed, and the results of these tests are discussed.

AI Testing

Because the purpose of the table is to play against people, we decided that the best way to test our code is to have lots of people play against it. We used the senior project expo as a great testing bed for our program because people of all skill levels played our table. Some people, who clearly had experience playing foosball were quick enough to beat the AI by moving faster than the camera could track, while many beginners struggled against the AI. We are satisfied that our first attempt at creating an AI was comparable to a beginning player.

Vibration Analysis

Description

This series of test was used to determine the magnitude and frequency of vibrations experienced by the system during the operation of one of the linear motors. This data will help to better quantify the forces experienced by the system during play and is integral in the design of the roof system.

The program JOG function in Sigmawin+ was used to perform the test moves of the motor. Two types of moves were used during the test, a single long move at high speed and a series of short moves at high speed, and the parameters of the moves can be found in Table 12. The acceleration data was collected using a K330 3-axis Accelerometer in a Samsung Galaxy S4 using the Accelerometer Monitor application. Three different test conditions were evaluated, a single move with the table unattached to the cabinet, multiple moves with the table unattached to the cabinet, and multiple moves with the table attached to the cabinet using clamps. Four different locations were included vibrational study, the Plexiglas cabinet top, an 8020 strut on the top of the cabinet, the top of the table, and at the top of the vision arch The results of each test can be found in Table 13 and the testing details can be found in Appendix F.

Table 14: Contains the parameters used for the JOG function during the vibration testing

Move Type	Move Distance (mm)	Motor Speed (min-1)	Actuator Speed (m/s)	Acceleration Time (ms)	Cycles
Long	400	6000	1.87	150	1
Short	40	6000	0.59	150	5

Table 15: Results of the vibrational testing. The maximum amplitude and average frequency of the vibrations at each location are given.

	Long Move Unattached		Short Move Unattached		Short Move Attached	
Location	Max Amp. (m/s)	Frequency (Hz)	Max Amp. (m/s)	Frequency (Hz)	Max Amp. (m/s)	Frequency (Hz)
Plexiglas on Cabinet	-2.434	34.39	3.114	25.58	-1.746	19.04
8020 on Cabinet	2.470	33.64	3.775	25.62	-1.719	18.34
Top of Table	-0.539	29.26	0.643	18.31	-1.958	19.29
Vision Arch	0.640	-22.03	0.230	16.79	1.943	14.76

Discussion

The results of the test show that, for a single motor operating near maximum capacity, the vibrations in the cabinet and table are relatively low. Short quick moves seem to cause vibrations with greater amplitude and frequency than those caused by the single move. This should not present a problem though as the rods will not be moved in this manner often. Additionally, dampers will be used if necessary to prevent any oscillations which might become dangerous.

The difference between the attached and unattached system are quite apparent. In the unattached case, the vibrations experienced by the table and vision arch are relatively low, while the vibrations in the cabinet are high. In the attached system, the table and cabinet experience the same magnitude of vibration, which is lower than the vibrations in the cabinet from the unattached test. This is most likely a result of the increased mass of the system, and shows that while vibrations will be transmitted through the brackets, the overall effect on the system will be relatively small.

Once we had all of the hardware components installed, including the roof, it became clear that vibrations were not going to be a large issue. As we played against the machine, we watched the live video stream and saw that the vibration had a negligible effect. We are satisfied with the vibrations of the system.

Inertia Ratio

Description

This test was meant to determine the inertia ratios of the goal rods motors. The collected data helps to characterize the system and improve motor tuning results. The data can also be compared to the calculated inertia ratios from the previous team.

The assessment of the motor inertia ratios were performed by using the Moment of Inertia Identification function in Sigmawin+. The parameters of the test moves can be found in Table 14 and are the default parameters of the function. The function was run several times and each of the calculated inertia ratios was recorded. Table 15 contains the collected data and the average of inertia ratios found for each of the motors.

Table 16: The parameters used in the Inertia Identification Function.

Parameter	Value
Acceleration (min-1/s)	20000
Speed (Min-1)	1000
Moving Distance (Rotation)	2.5
Pn100:SPEED LOOP GAIN (0.1Hz)	400

Table 17: Results of the Inertia Identification test

Run	Linear Motor – Axis 3	Rotary Motor – Axis 4
1	645%	185%
2	625%	178%
3	632%	179%
4	631%	172%
5	633%	172%
6	635%	176%
Average	633%	177%

Discussion

The results of the inertia ratio tests are promising. The averages for both of the motors are below those expected by the other team. This should indicate that the motors will perform better than expected, once properly tuned. The test also indicates that the inertia ratios are well below the maximum value for use with the auto tuning function. This will help save valuable time later in the project.

Motor Tuning

Goalie Rotary Motor

The autotuning function of Sigmawin+ was used to tune the rotary motor for the goalie rod. The function worked extremely well, and the response of the motor was greatly improved. To assess the effect the autotuning function had, the motor was first run using the JOG function in the tuneless mode, and the trace function was used to gather performance data. The autotuning function was then run. Once it was complete, the same JOG move was made and the data was collected by the trace function. Table 16 contains the settling time associated with both moves, and Appendix G contains additional information about the tuning process.

Table 18: Comparison of the settling times before and after tuning

Tuneless Settling Time	Tuned Settling Time	Percent Improvement
189.6	4.62	97.6%

Goalie Linear Drive Motor

The tuning of the linear drive motor was more involved than that of the rotary motor because the autotuning function of Sigmawin+ would not run on the motors. As an alternative, the custom tuning function was used. The JOG function was used during this tuning, and the parameters of the move can be found in Table 17. The motor was first run in tuneless mode to determine its initial response to the move. The motor was then set back to tuning mode, and a series of changes were made to the feed forward and feedback gains of the system. The trace function was used to record the response of the motor each time a change was made to the gains. The details of the tuning can be found in Appendix G.

Table 19: The JOG move parameters used to tune the linear drive motor

Move Type	Move Distance (mm)	Motor Speed (min-1)	Actuator Speed (m/s)	Acceleration Time (ms)	Cycles
Long	400	6000	1.87	150	1

A significant improvement in motor response and settling time were made after the tuning was complete. Table 18 contains the final results of the tuning process and a comparison to the results of the tuneless move. It is expected that when the autotuning function again works with the linear motors, another improvement in motor performance will be made.

Table 20: Comparison of the settling times before and after tuning

Tuneless Settling Time	Tuned Settling Time	Percent Improvement
612.72	17.34	97.1%

Chapter 6: Bill of Materials

Cabinet/Table Bracket

Description	Source	Part Number	QTY.	Unit Price	Total Cost
Extruded Structural Aluminum Bare Angle 6061 T6 4"x4" 2ft	Online Metals	NA	1	33.36	33.36
Multipurpose Aluminum (Alloy 6061) Rectangular Bars—Unpolished (Mill) Finish 1/2" 4"x1'	Mcmaster Carr	8975k428	2	13.7	27.4
1/2" ID Rubber Washers Neoprene	Mcmaster Carr	90133A425	1	8.87	8.87
Type 316 Stainless Steel Square Nuts 1/2"-13	Mcmaster Carr	92891A400	8	2.63	21.04
Square-Head Steel Bolts 2" 1/2"-13	Mcmaster Carr	92327A304	8	3.12	24.96
10MM, M8 T-bolt fastening kit, L=14	Bosch Rexroth	8981021342	12	0.79	9.48
				Total	125.11

Actuator Brackets

Description	Source	Part Number	QTY.	Unit Price	Total Cost
1/8'' 12''x24'' Al sheet	Mcmaster-Carr	8973K79	1	25.03	25.03
10MM, M8 T-bolt fastening kit, L=14	Bosch Rexroth	8981021342	24	0.79	18.96
8MM, M6 T-bolt fastening kit, L=18	Bosch Rexroth	8981019578	40	0.79	31.60
				Total	50.50

Safety System

Description	Source	Part Number	QTY.	Unit Price	Total Cost
DC Rated-SPST-NO-Magnetic Switch	McMaster Carr	65985K11	4	12.14	12.14
I/O module 16 input/output	Yaskawa	JAPMC-IO2301	1	Donated	Donated
				Total	48.56

Roof System

Description	Source	Part Number	QTY	Unit Cost	Total Cost
D28L L=5600MM	Bosch Rexroth	3842541212	2	24.78	49.56
D28L, N10 L=5600MM	Bosch Rexroth	3842541214	1	37.87	37.87
Cap Cover, D28L, BLACK ESD	Bosch Rexroth	3842541195	4	0.92	3.68
0 - 90 D28 Connector	Bosch Rexroth	3842543480	2	6.25	12.50
Connector, 90°	Bosch Rexroth	3842541173	14	2.06	28.84
Double, 4-Hole	McMaster-Carr	5537T186	2	5.8	11.60
Silver 6’x10’ Tarp	Tarp Surplus	TS06X10C	1	7.80	7.80
				Total	143.55

Lighting

Description	Source	Part Number	QTY.	Unit Price	Total Cost
Ribbon Star Ultra LED	Ecocity Light LED	RL-SC-RSSU24-W-10	1	114.99	114.99
Mean Well 24VDC, 60W PS	Ecocity Light LED	PS-MW-60-24	1	42.99	42.99
				Total	157.98

Laser Alignment

Description	Source	Part Number	QTY	Unit Cost	Total Cost
1" Diameter Polypropelene Rod	McMaster-Carr	8658K55	1	3.54	3.54
2" Diameter Polypropelen Rod	McMaster-Carr	8658K59	1	12.12	12.12
Adjustable Cross-Hair Red Laser Module	Apinex	YCHG-650C	1	19.95	19.95
				Total	35.61

Total Cost

Description	Total Cost
Cabinet/Table Bracket	125.11
Actuator Brackets	45.62
Safety System	48.56
Roof System	151.35
Lighting	157.98
Laser Alignment	35.61
	564.23

Chapter 7: Conclusion & Recommendations

Recommendations

Vision System

Future evolution of the vision system has several potential aspects. Based on the fact that the PLC task priorities are vastly underutilized in the current configuration, for example, only fractions of each run cycle of the 'Fast task' in each fast task cycle is being utilized by code. This means that if a fast task is 2 milliseconds wide maybe 40 microseconds of each fast task clock allocation is being taken advantage of. This leaves plenty of processor to incorporate a multiple camera vision system. This could significantly decrease the number of positions in which ball position is not known. An increase in known ball position will increase the effectiveness and accuracy of any predictive play code that might be programmed into future iterations, such as ball velocity, future position predictions, and opponent 'learning' AI functionality.

The camera supports the ability to store multiple jobs available for loading and running by sending command instructions to the camera from the PLC. The current configuration to setup so that only the selected startup job runs without going into the camera software and manually loading a different job. This could be useful if alternate tracking methods are desired for future versions.

Integration of lighting or other sensors that would allow the PLC to determine configuration settings to the Camera. This would allow variables such as the contrast to be adjusted on the fly by the PLC as ambient lighting changes affect the tracking quality of the camera job.

Future inclusion of a video monitor system to display the camera view to a crowd might necessitate the use of some of the High-speed outputs rather than EIP to communicate with the PLC, freeing up the Ethernet outputs for live streaming to monitor screens for spectator viewing.

The camera is not currently outputting process data to the PLC, only ball data. System data could be incorporated into future iterations for monitoring performance. This would be a simple improvement, which could allow for better system-wide analysis of trends and performance.

Goal Sensing

There are two options for goal sensing: physical sensors installed on both goals, or implementing logic on the ball tracking to determine when a goal is scored through the vision system. The latter option will require less hardware and is recommended.

Safety Switches

A power source for the I/O module is required to complete the implementation of the safety switches for the motor cabinet and playfield cover. As for the front doors of the motor cabinet, we recommend installing a locking mechanism to keep the doors closed during operation instead of installing the remaining two safety switches. Once the table is running, we do not foresee any reason to open the front doors of the motor cabinet.

Score board

The current design of the scoreboard is overly complicated. With three sides of the scoreboard to display four digits, and each seven-segment display wired up there are over 120 wires to plug into the breadboards to fit into the scoreboard box that also houses the camera. The triangular shape was also difficult to manufacture and as a result has imperfect seams. It is also difficult to install the camera so that it is properly aligned with the playfield. For these reasons we recommend redesigning the scoreboard, prioritizing the ease of camera installation and so that there are a reduced number of displays. A simple cube with one front-facing score display will probably prove the simplest to manufacture and provide ample room to make installing the camera easy.

HMI

The Human Machine Interface was not implemented during this phase of the project because a physical HMI unit was not procured from Yaskawa due to technical difficulties. An HMI unit should be obtained and integrated with the PLC. A HMI program can then be developed using Visu+. This program should contain a player interface which allows users to select difficulty, start the game, reset the score and end the game. It should also contain an operator interface which allows the operators of the system to perform basic diagnostic and setup functions on the system.

Lighting

Roof

The silver tarp that is mounted on the roof frame right now is not expected to be the final solution for the roof. We ordered a custom blue canvas tarp but its dimensions were found to be out of tolerance when it was received. We recommend that the next group should look into

Currently, the tarp is mounted to the roof using zip ties. Because the zip ties must be thrown out every time the tarp is reattached to the roof, we recommend finding a new method to attach the cloth to the top of the roof that is more sustainable. Velcro straps is probably a good place to start brainstorming for a tarp attachment system.

Gusset Plate

We did not have enough time to complete the manufacturing of the gusset plate. We cut out the geometry of the plates, but the plates still need to be welded and holes need to be drilled. We do not feel that the vibration of the vision arch in its current state affects the performance of the camera, so we only recommend spending the time to finish the gusset plate if new additions increase the vibrations.

Programming Improvements

Because we had very little time to develop the logic for the PLC, we are sure that many improvements could be added to the code. Right now, the code executes on the level of microseconds, so many more advanced calculations could be made without sacrificing performance. This section contains the suggested additions or improvement which could be made to the program in relation to performance or playability. Additions such as the safety program and HMI program are discussed in the related hardware recommendations section.

Velocity Tracking

Currently, the rod logic only takes the position of the ball into account and not its velocity. This is a problem because if the ball is kicked at an angle, the translation of the rods will always lag behind the movement of the ball in the translational direction (the y-direction) since the code does not compensate for the refresh rate of the camera.

If the velocity is calculated, then the ball's trajectory could be predicted (neglecting ball spin). With the trajectory predicted, the foosmen would be able to move where the ball is going to be instead of where it is now. This would help counteract the time offset created by the camera.

The easiest way to calculate the velocity would be to continuously store the previous ball position and subtract the previous position from the current position. Since the trajectory of the ball only matters when the human player hits the ball up the field, the trajectory of the ball may need to be calculated only when there is a large spike in the x-direction velocity. It will be easy to predict how the ball will bounce of the walls by treating the wall as a mirror, but ricochets off of players could be difficult to predict or react to. Obviously a good trajectory algorithm will be difficult to create, but it will allow the table AI to reach another level of competition.

Offensive Capabilities

Right now, the AI's offense is very "dumb." All it does is try to kick the ball forwards if it is in front of one of the rods. It does not care what angle it will kick it or if it is trying to pass or shoot; it just kicks.

One addition to the code that could become incredibly effective is a method for trapping and then passing the ball. Through our testing, we found that passing the ball forward is incredibly easy because the foosmen are designed to "catch" the ball with the back of their "foot" if held at a 45 degree angle. The difficult action will be initially gaining possession of the ball in a controlled manner. Numerous amounts of trial and error will be required to master gaining control of the ball, but if it can be achieved, it will open the door for much more sophisticated offensive strategies. Control of the ball will allow the AI to have a decision tree because it will allow for both passing and shooting.

Improved Motor Condition Monitoring

Currently, the only motor monitoring which is done by the system is torque monitoring on the rotary motors of each axis. While this does prevent many alarms when a foosman traps the ball, some still do occur due to positioning and velocity errors. The cause of these errors is unknown and further investigation needs to be performed. Once the cause of the alarms is discovered, functions should be written to prevent the errors from occurring and to allow for the system to self-correct as errors occur. Additionally, monitoring for the translational motors should be considered, though no alarms have occurred in these motors using the current code.

Implementation of Sequential Function Charts

Sequential function charts are one of the five programing languages supported by motion works. It lends itself well to programming the upper layers of a given program as a state machine or series of state machines. Each SFC contains a series of states and conditional transitions between those states. The SFC will execute the state it is in until a transition condition is met, which causes the SFC to activate the next state. SFCs are easy to follow while still allowing extremely intricate and complex operations to be executed.

SFCs were not used in this iteration of the foosball program, but should be used in future versions of the program. A basic template for the SFCs which could be used in the final program can be found in Appendix H.

Conclusion

This document presents the designs and decisions which we feel satisfy the requirements laid out in our proposal report. We have gone over background research, design requirements, concepts and concept selection, and our management plan for the project. If the concepts and plans developed in the report are acceptable, we request that Yaskawa America grant us permission to proceed with project development. We request that if Yaskawa America agrees to go ahead with the project that we require an activation key for the OPC Server, I/O Module, and HMI.

Acknowledgments

We would like to thank Warrior Table Soccer for greatly discounting the foosball table used in the project. We would also like to thank team Foos-Roh-Dah for being extremely helpful in getting us up to speed on the project and giving us a good idea of what needs to be done to successfully complete the project. We would also like to thank Yaskawa America for providing support to the team. We would also like to thank Cognex and T-Slots for donating components for our project.

Appendices

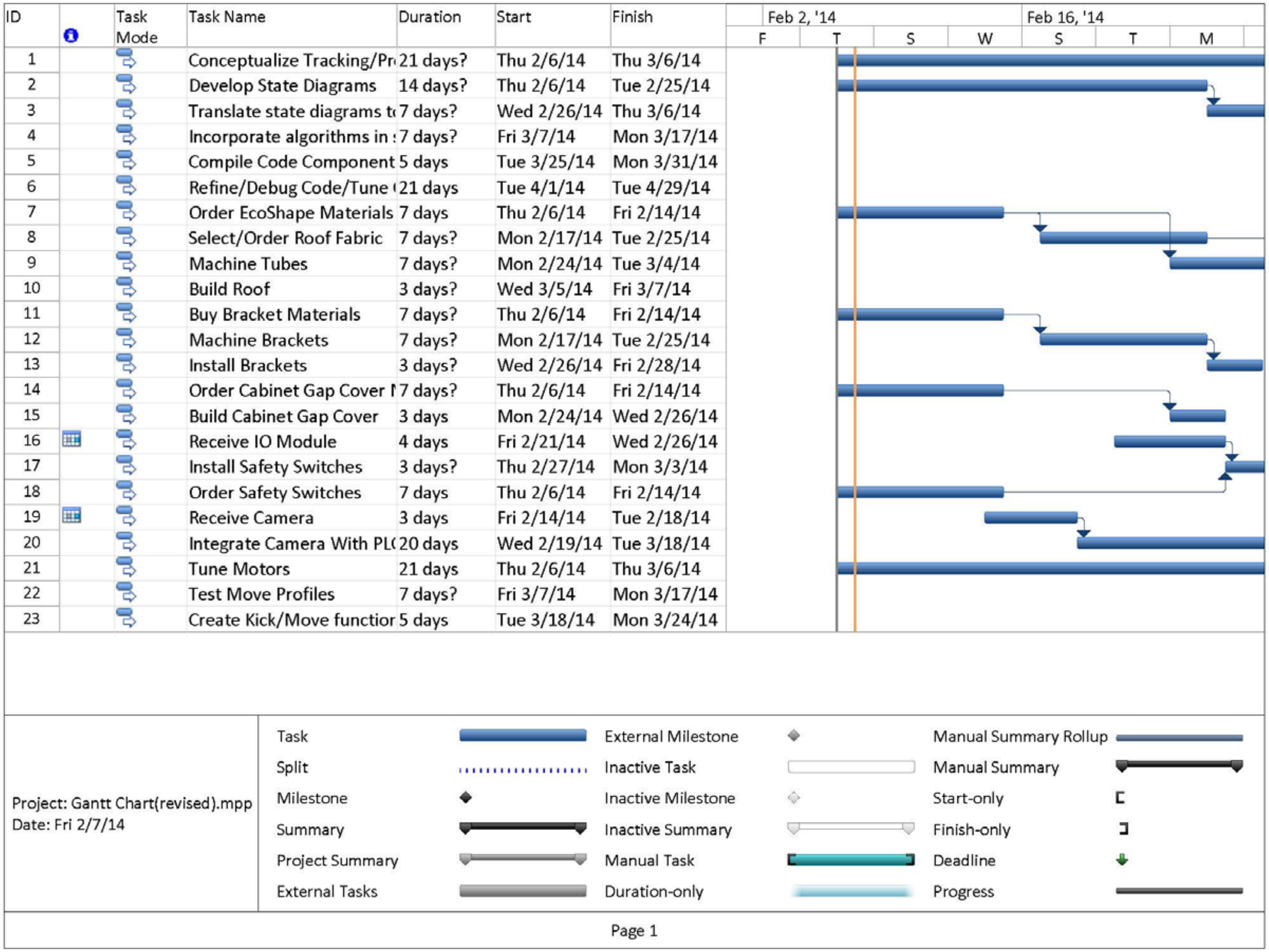
Appendix A: Quality Function Deployment

Customer Requirements

The A Team - Automated Foosball		Engineering Requirements																									Benchmarks (1 to 5)				
Customer (Step #1) Requirements (Whats)	Weighting (1 to 5)	Goal Sensor	LCD Menu	% of Inner Workings Visible	System Response Time	Power delievered to ball	Time to assemble from base components	Vibrations experienced during operation	System sensing of the ball in motion	No direct contact between player and moving parts	Reliability of the Mechanical Ststem	Measurements of the total space required for operation	Smooth variable Movement Speed	new player or user learning time	Aesthetic Assesment Scale 1-10 (Sponsor)	Aiming (hitting ball in direction of goal)	Cost analysis (our target does not include donations)	Fatigue Analysis (Unable to test)	Weight < 250lb for cabinet	Tune the motors	Confirm motor size	Create Modular Function Library Using IEC 61131-3 Languages	Basic AI difficulty	Normal AI difficulty	Advanced AI difficulty	Transportable by 2 people	The Design of a Semi-Automated Football Table - IEEE	Single Player Foosball Table with an Autonomous Opponent - Georgia Institute of Technology	KiRo - University of Freiburg	Automatic Foosball Table "Foosbot" - University of Akron	Designing a foosball table acuator -University of Eindhoven
Automated Scoring System	2	9	9																								1	2	5	1	2
Demonstrate high speed integration of vision system with offensive and deffensive strategies	5			1	9	1		1	9				1			3				3			3	9	9		4	1	4	1	5
Must not stop working during trade show	5			1		1	1	1		1	9							9		9							3	1	4	3	3
Software must be easy to understand and improve	3																					9					3	5	3	4	2
Ease of assembly/disassembly	4			1			9												3							3	3	3	2	3	2
Once assembled, minimal instruction required to setup and play	4			1			1				3			9													5	3	5	5	5
Trade show quality aestetics	4	3	3	9								3			9												3	1	4	3	4
Ball must be unable to leave field of play	5									9																	2	1	3	1	2
Rods must be unable to strike players	5									9																	2	2	3	2	3
Should last 100 hours without maintenance	3					1		3			9							9		3							3	2	4	3	3
Table must be stable during play	5						3	3			1	3															3	4	3	4	4
Must fit inside the allocated trade show space	5											9															3	2	2	3	4
Demonstrate high precision and high speed integration of motors and actuators	5				9	9		1	3		9		3			9		3		9	9		3	9	9		2	1	3	1	5
Cost kept reasonable for high quality product	2																9										3	5	2	2	4
Life of the product should be reasonable given the application	4							3			9							9									3	2	4	3	3
Portable	4											1							9		3					9	1	2	3	4	1
Challenge players of varying ability	3		1		9	1			1				9	3		9						3	9	9	9		1	1	2	4	1
Create PLC function block library for different basic moves	5																					9					0	0	0	0	0
Powered by single outlet (200V)	2					1															3						3	3	3	3	3
Units				% Visible	ms	N-m	min		m/s		%	m^2	m/s	sec		Degrees	$	Hours	kg												
Targets		Yes	Yes	80%	25	15	180	low	10	yes	99%	1.5x1.5	2-10	30	7	±30	5,000	100	250	yes	yes	yes	yes	yes	yes	yes					
The Design of a Semi-Automated Football Table - IEEE					45																										
Single Player Foosball Table with an Autonomous Opponent - Georgia Institute of				50%	100				0.46	No		1.52x2.44					710														
KiRo - University of Freiburg				10%	40				6	Yes				40			27,000														
Automatic Foosball Table "Foosbot" - University of Akron				50%	35	0.33			6.7	No							5,780														
Designing a foosball table acuator -University of Eindhoven7Automated				70%	30	13.2			10	Yes		1.43x0.74		30			25,812														
Importance Scoring		30	33	54	117	63	60	51	63	95	170	76	47	45	36	87	18	123	48	114	63	36	57	117	117	48					
Importance Rating (%)		18	19	32	69	37	35	30	37	56	100	45	28	26	21	51	11	72	28	67	37	21	34	69	69	28					

	Transportable by 2 people	Advanced AI difficulty	Normal AI difficulty	Basic AI difficulty	Create Modular Function Library Using IEC 61131-3 Languages	Confirm motor size	Tune the motors	Weight < 250lb for cabinet	Fatigue Analysis (Unable to test)	Cost analysis (our target does not include donations)	Aiming (hitting ball in direction of goal)	Aesthetic Assesment Scale 1-10 (Sponsor)	new player or user learning time	Smooth variable Movement Speed	Measurements of the total space required for operation	Reliability of the Mechanical Ststem	No direct contact between player and moving parts	System sensing of the ball in motion	Low Vibrations experienced during operation	Time to assemble from base components	Power delievered to ball	System Response Time	% of Inner Workings Visible	LCD Menu	Goal Sensor
Goal Sensor																								+	
LCD Menu		+	+	+								+	+												
% of Inner Workings Visible												+					-								
System Response Time		+	+	+			+				+			+				+	+						
Power delievered to ball									-		+								+						
Time to assemble from base components	+																								
Low Vibrations experienced during operation									+			+		+		+									
System sensing of the ball in motion		+	+	+	+																				
No direct contact between player and moving parts												+			+										
Reliability of the Mechanical Ststem						+	+		+																
Measurements of the total space required for operation	+							+																	
Smooth variable Movement Speed						+	+		+		+	+													
new player or user learning time		+	+	+																					
Aesthetic Assesment Scale 1-10 (Sponsor)																									
Aiming (hitting ball in direction of goal)		+	+	+	+		+																		
Cost analysis (our target does not include donations)																									
Fatigue Analysis (Unable to test)						+	+																		
Weight < 250lb for cabinet	+																								
Tune the motors		+	+	+		+																			
Confirm motor size																									
Create Modular Function Library Using IEC 61131-3 Languages		+	+	+																					
Basic AI difficulty																									
Normal AI difficulty																									
Advanced AI difficulty																									
Transportable by 2 people																									

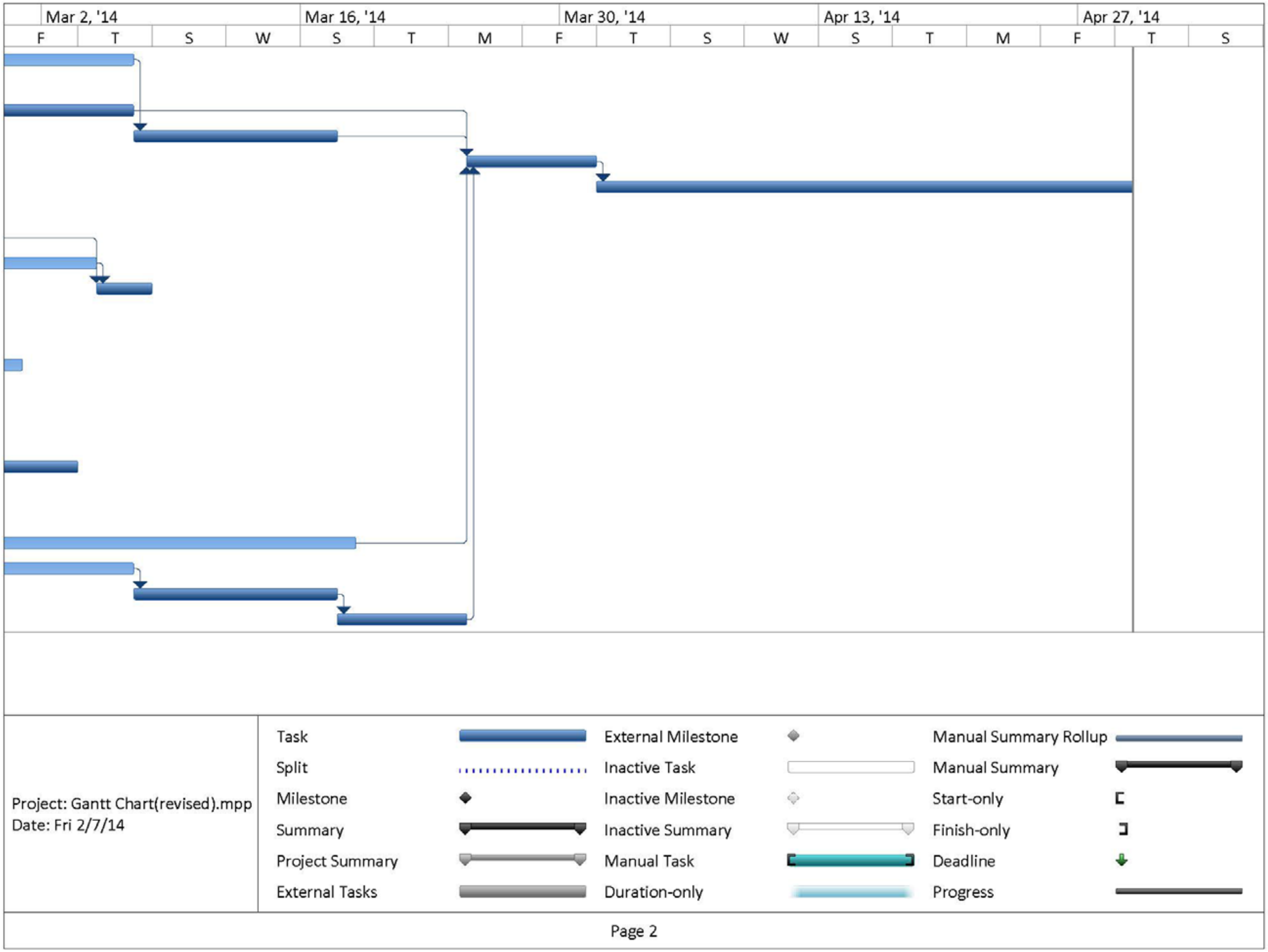
Appendix B: Project Management



ID	ℹ	Task Mode	Task Name	Duration	Start	Finish
1			Conceptualize Tracking/Pr	21 days?	Thu 2/6/14	Thu 3/6/14
2			Develop State Diagrams	14 days?	Thu 2/6/14	Tue 2/25/14
3			Translate state diagrams t	7 days?	Wed 2/26/14	Thu 3/6/14
4			Incorporate algorithms in :	7 days?	Fri 3/7/14	Mon 3/17/14
5			Compile Code Component	5 days	Tue 3/25/14	Mon 3/31/14
6			Refine/Debug Code/Tune (	21 days	Tue 4/1/14	Tue 4/29/14
7			Order EcoShape Materials	7 days	Thu 2/6/14	Fri 2/14/14
8			Select/Order Roof Fabric	7 days?	Mon 2/17/14	Tue 2/25/14
9			Machine Tubes	7 days?	Mon 2/24/14	Tue 3/4/14
10			Build Roof	3 days?	Wed 3/5/14	Fri 3/7/14
11			Buy Bracket Materials	7 days?	Thu 2/6/14	Fri 2/14/14
12			Machine Brackets	7 days?	Mon 2/17/14	Tue 2/25/14
13			Install Brackets	3 days?	Wed 2/26/14	Fri 2/28/14
14			Order Cabinet Gap Cover I	7 days?	Thu 2/6/14	Fri 2/14/14
15			Build Cabinet Gap Cover	3 days	Mon 2/24/14	Wed 2/26/14
16	▦		Receive IO Module	4 days	Fri 2/21/14	Wed 2/26/14
17			Install Safety Switches	3 days?	Thu 2/27/14	Mon 3/3/14
18			Order Safety Switches	7 days	Thu 2/6/14	Fri 2/14/14
19	▦		Receive Camera	3 days	Fri 2/14/14	Tue 2/18/14
20			Integrate Camera With PL(	20 days	Wed 2/19/14	Tue 3/18/14
21			Tune Motors	21 days	Thu 2/6/14	Thu 3/6/14
22			Test Move Profiles	7 days?	Fri 3/7/14	Mon 3/17/14
23			Create Kick/Move functior	5 days	Tue 3/18/14	Mon 3/24/14

Project: Gantt Chart(revised).mpp
Date: Fri 2/7/14

Page 1


Mar 2, '14
Mar 16, '14
Mar 30, '14
Apr 13, '14
Apr 27, '14
F
T
S
W
S
T
M
F
T
S
W
S
T
M
F
T
S
Project: Gantt Chart(revised).mpp
Date: Fri 2/7/14
Task
Split
Milestone
Summary
Project Summary
External Tasks
External Milestone
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
Deadline
Progress
Page 2

Appendix C: Concept Evaluation

Table 21: Playfield cover Decision Matrix

Requirement	**Weight**	**Concept**			
		Top Door Cover (Current form)		Large Cover Encompassing Camera Arch	
		Screen Mesh	Clear Sheet (plastic/glass)	Preassembled	Panels
Ease of Assembly	4	4	4	3	2
Camera Visibility	8	1	2	4	4
Player Visibility	8	2	3	4	4
Safety	7	3	3	3	3
Portability	3	4	4	2	3
Aesthetic Quality	5	2	4	3	3
Ease of Playfield Access	6	4	4	3	3
Total		107	133	136	135

Table 22: Alignment Decision Matrix

Requirement	**Weight**	**Concept**				
		Leg Mount Levels	Beam Constrainers	Brackets	Infrared Slot Sensor	Clamps
Ease of Assembly	6	2	5	4	1	4
Ease of Access	4	3	4	4	3	3
Safety	8	3	4	4	5	1
Portability	2	1	4	3	3	3
Aesthetics	6	5	2	1	5	1
Cost	4	4	3	3	1	4
Total		96	110	96	98	72

Table 23: Attachment Decision Matrix

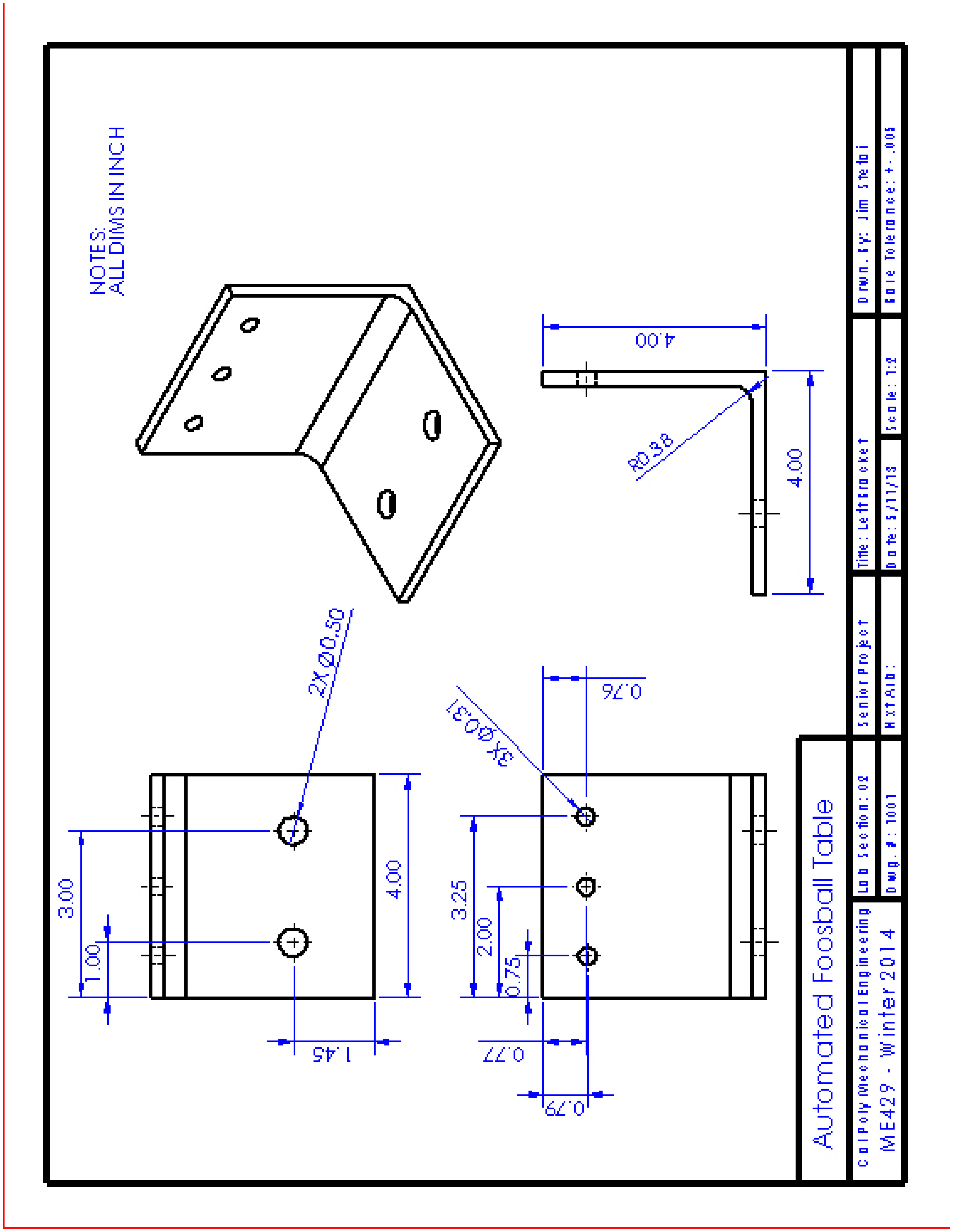
Requirement	Weight	Concept						
		Bolts Trough Legs	Straps	Bolts Through Lower Portion of the Table	8020 Struts Table Side	8020 Struts on Top of Table	Clamps Between Legs and Cabinet	Brackets
Ease of Assembly	6	1	4	2	2	3	4	4
Facilitation of Alignment	6	3	1	3	2	2	1	4
Vibration Reduction	3	4	1	4	2	1	1	3
Aesthetic Value	5	4	1	4	4	2	1	2
Attachment Quality	8	3	1	3	3	1	1	4
Safety	7	4	1	3	4	1	1	4
Maintenance	4	1	3	1	1	2	2	3
Reliability	4	3	1	3	3	2	1	4
Total		124	69	123	118	74	65	155

Table 24: Alignment Decision Matrix

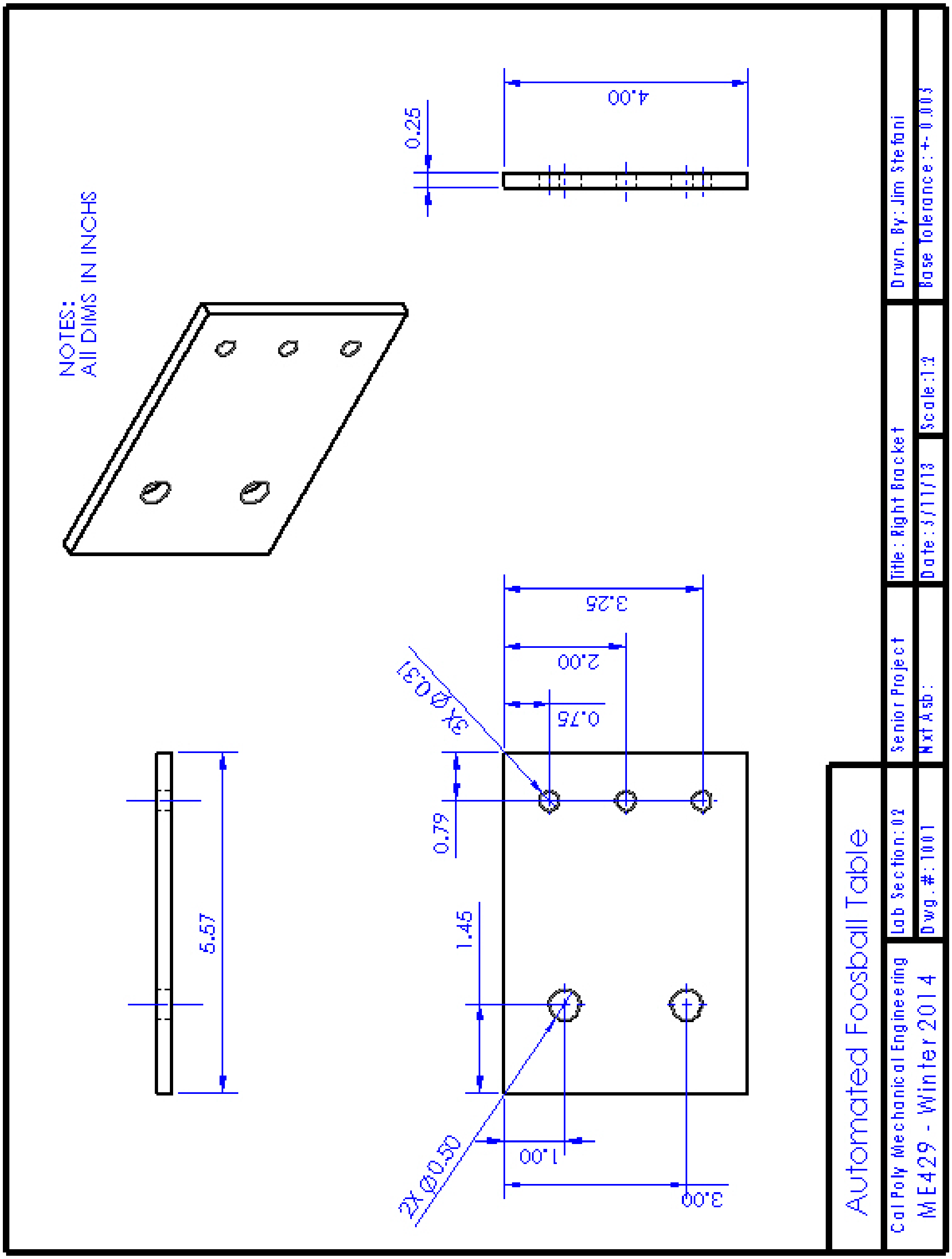
Requirement	Weight	Concept						
		Lights on score arch	Lights with diffusors on camera arch	Mount Lights below playfield (with translucent playfield installed)	LED strip lights mounted on side of playfield	LED strip lights mounted inside playfield cover	LED strip lights mounted above playfield cover	No lighting
Aesthetics	3	3	3	2	4	4	5	3
Does not interfere with foosball play	4	5	5	2	1	5	4	5
Resists Reflection	4	1	3	5	4	4	3	3
Does not create shadows	4	1	4	5	1	2	3	5
Brightens Table	5	3	4	2	3	3	3	0
Manufacturability	2	4	4	1	3	3	3	5
Transportability	3	3	3	1	3	3	2	5
Cost	2	4	4	3	3	3	3	5
Total		77	102	75	72	92	88	96

Appendix D: Drawing Packet

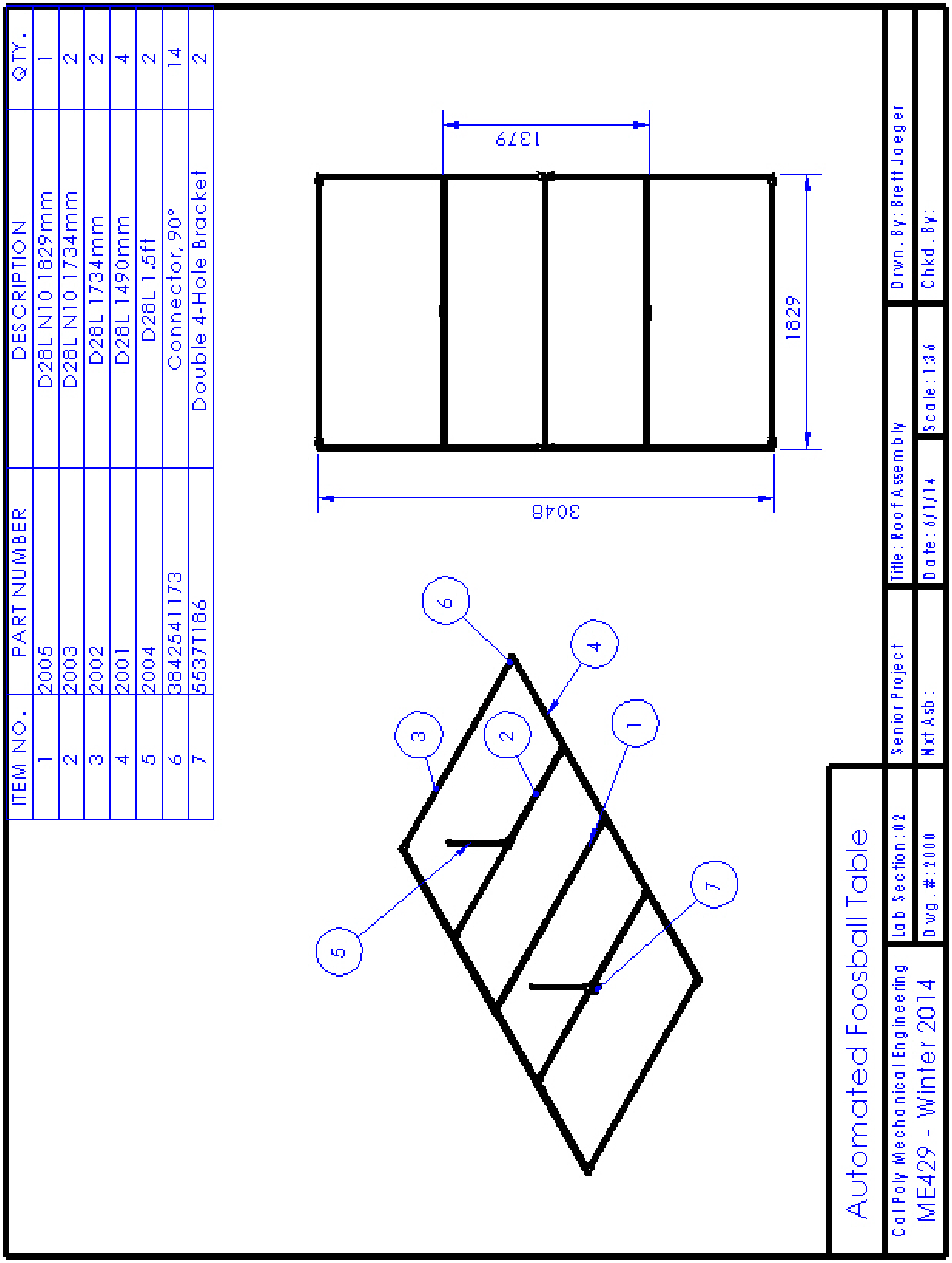
Left Bracket



Right Bracket

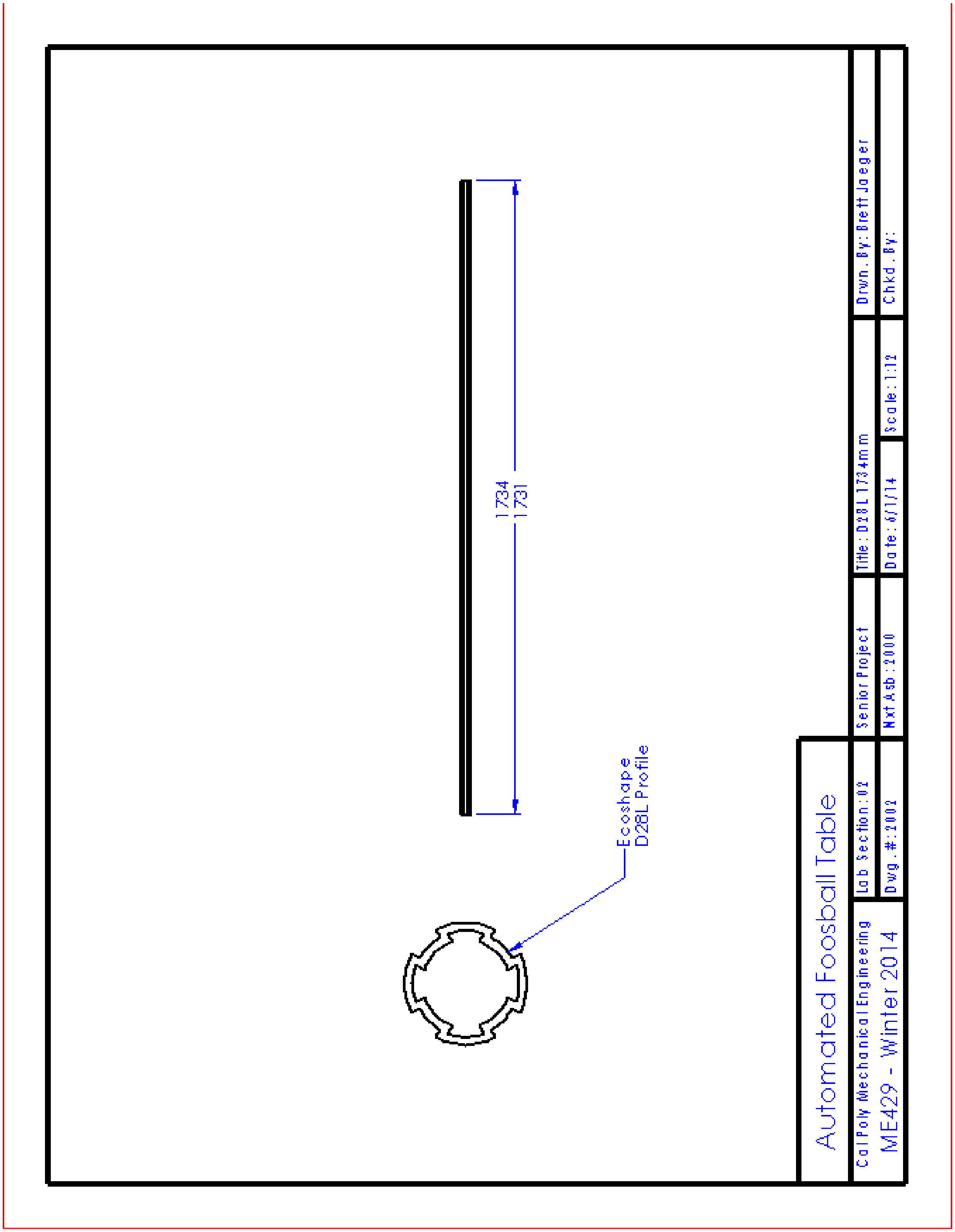


Roof Assembly

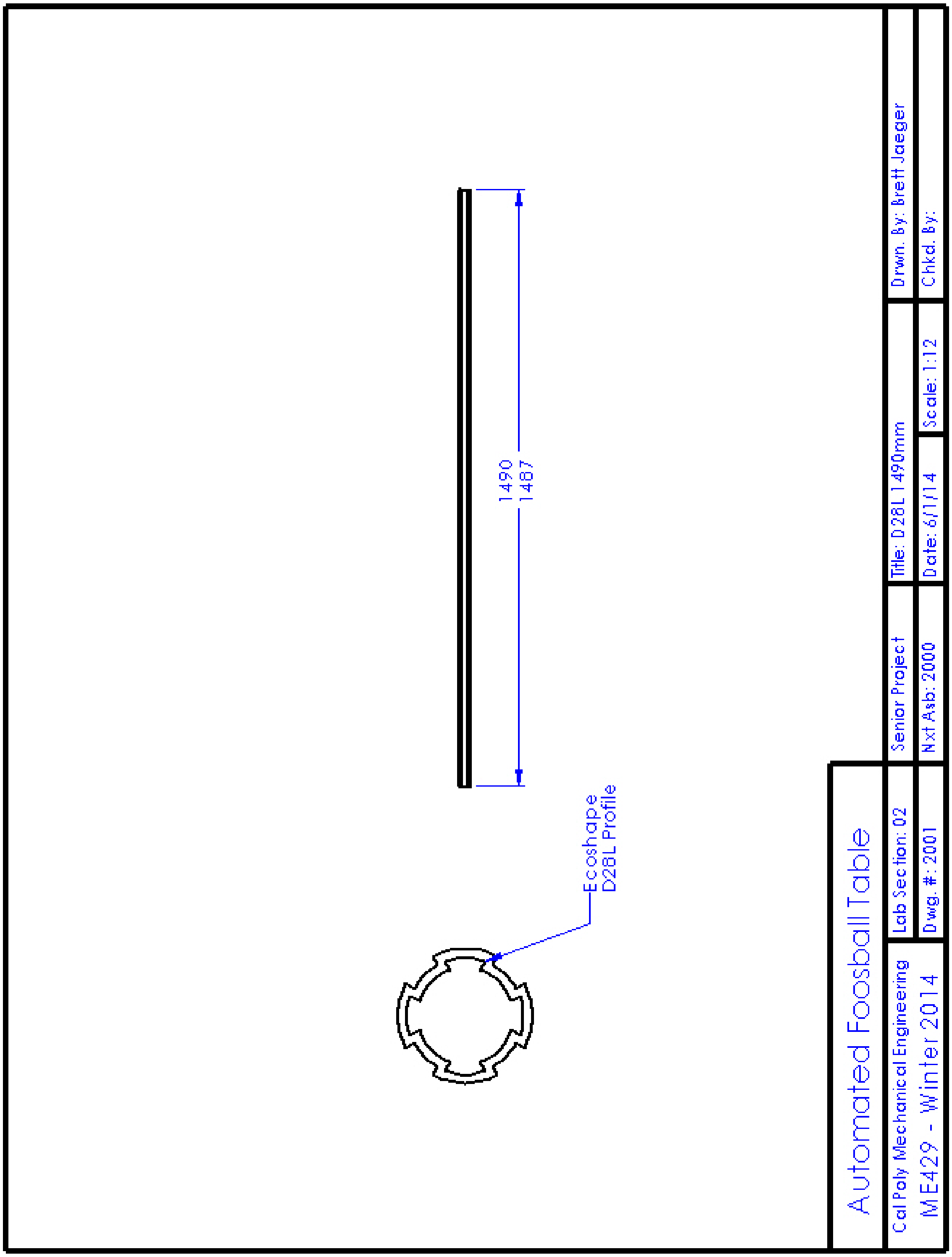


ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	2005	D28L N10 1829mm	1
2	2003	D28L N10 1734mm	2
3	2002	D28L 1734mm	2
4	2001	D28L 1490mm	4
5	2004	D28L 1.5ft	2
6	3842541173	Connector, 90°	14
7	5537T186	Double 4-Hole Bracket	2

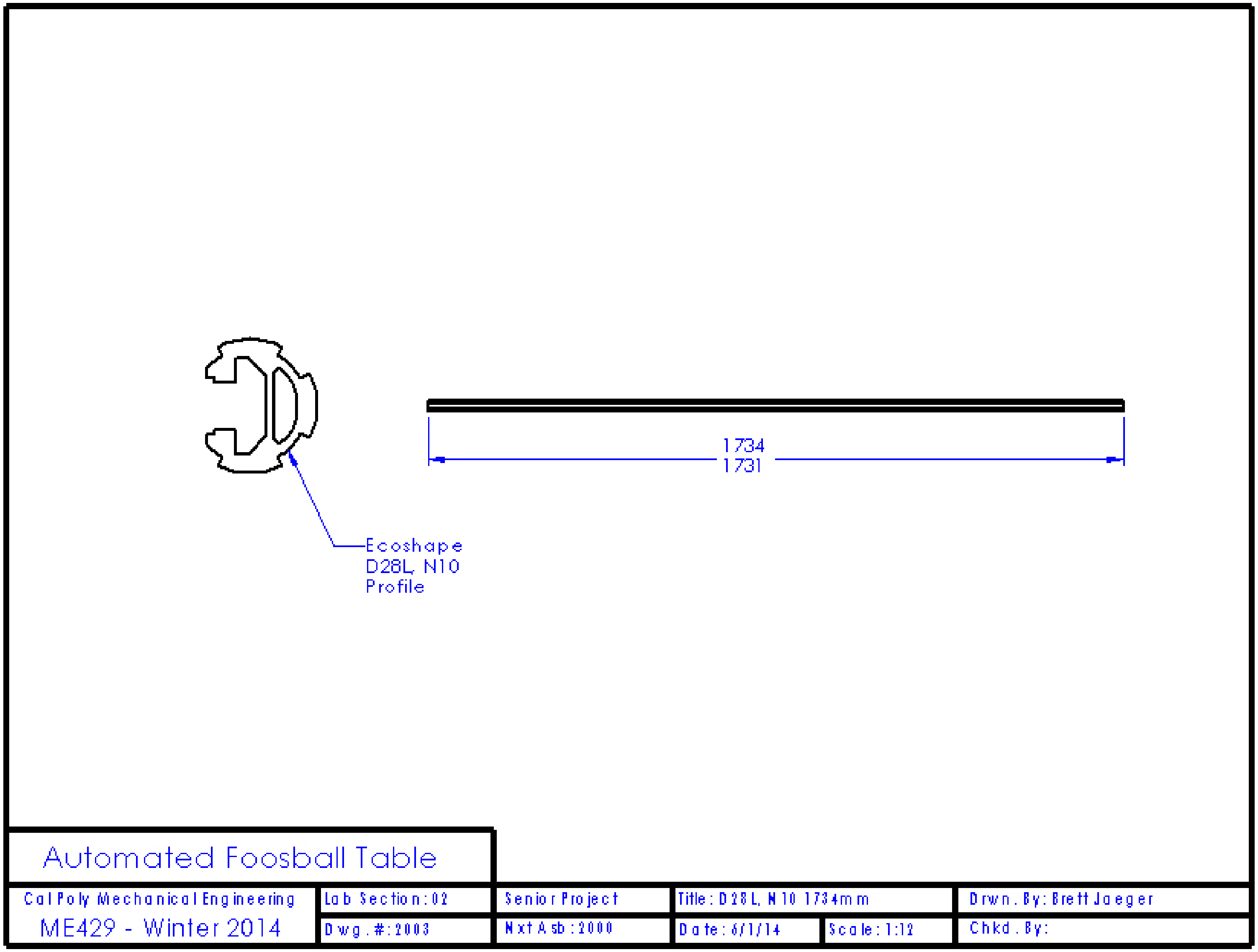
Eco-1734


1734
1731
Ecoshape
D28L Profile
Automated Foosball Table
Cal Poly Mechanical Engineering
ME429 - Winter 2014
Lab Section: 02
Dwg. #: 2002
Senior Project
Nxt Asb: 2000
Title: D28L 1734mm
Date: 6/1/14
Scale: 1:12
Drwn. By: Brett Jaeger
Chkd. By:

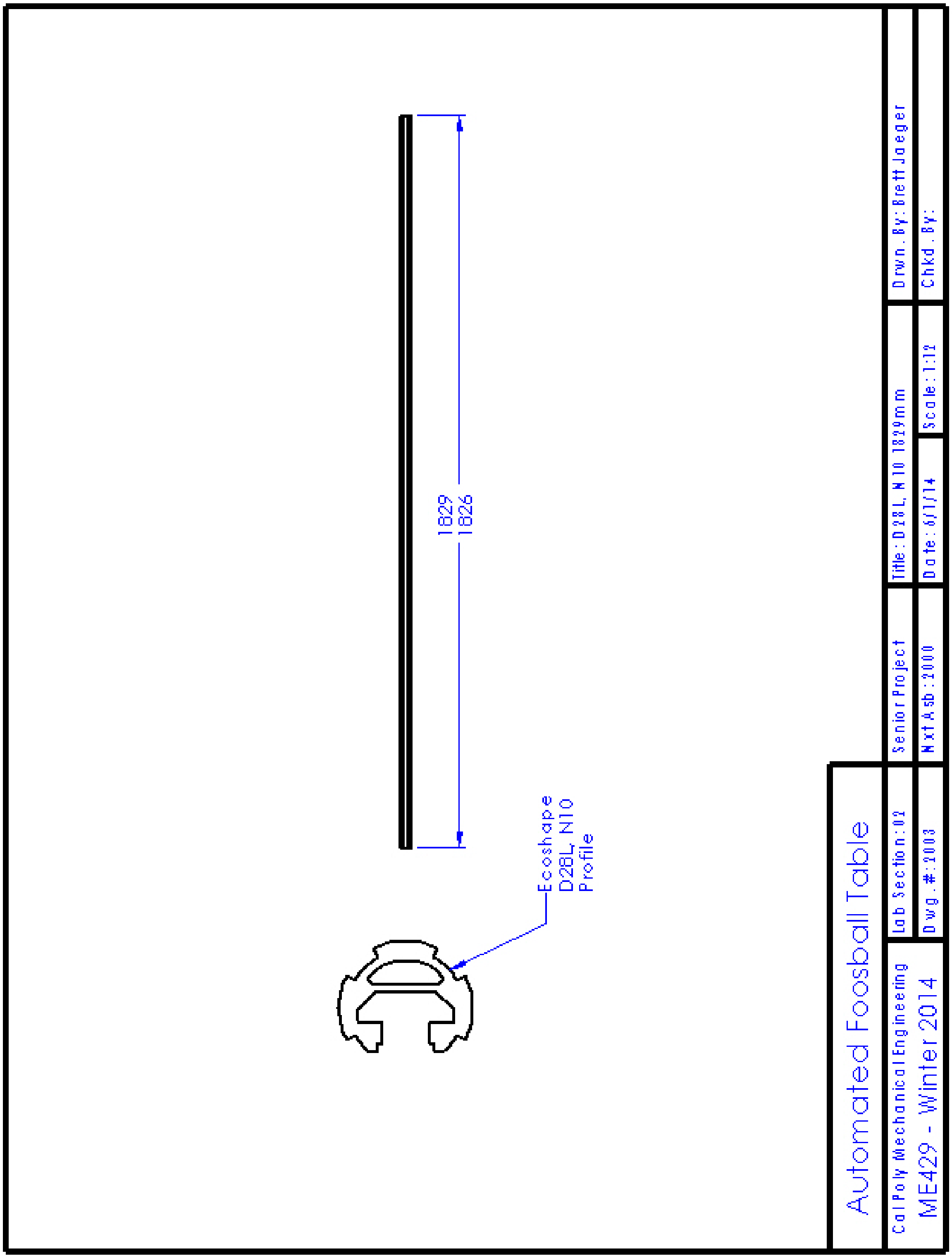
Eco-1490


1490
1487
Ecoshape
D28L Profile
Automated Foosball Table
Cal Poly Mechanical Engineering
ME429 - Winter 2014
Lab Section: 02
Dwg. #: 2001
Senior Project
Nxt Asb: 2000
Title: D28L 1490mm
Date: 6/1/14
Scale: 1:12
Drwn. By: Brett Jaeger
Chkd. By:

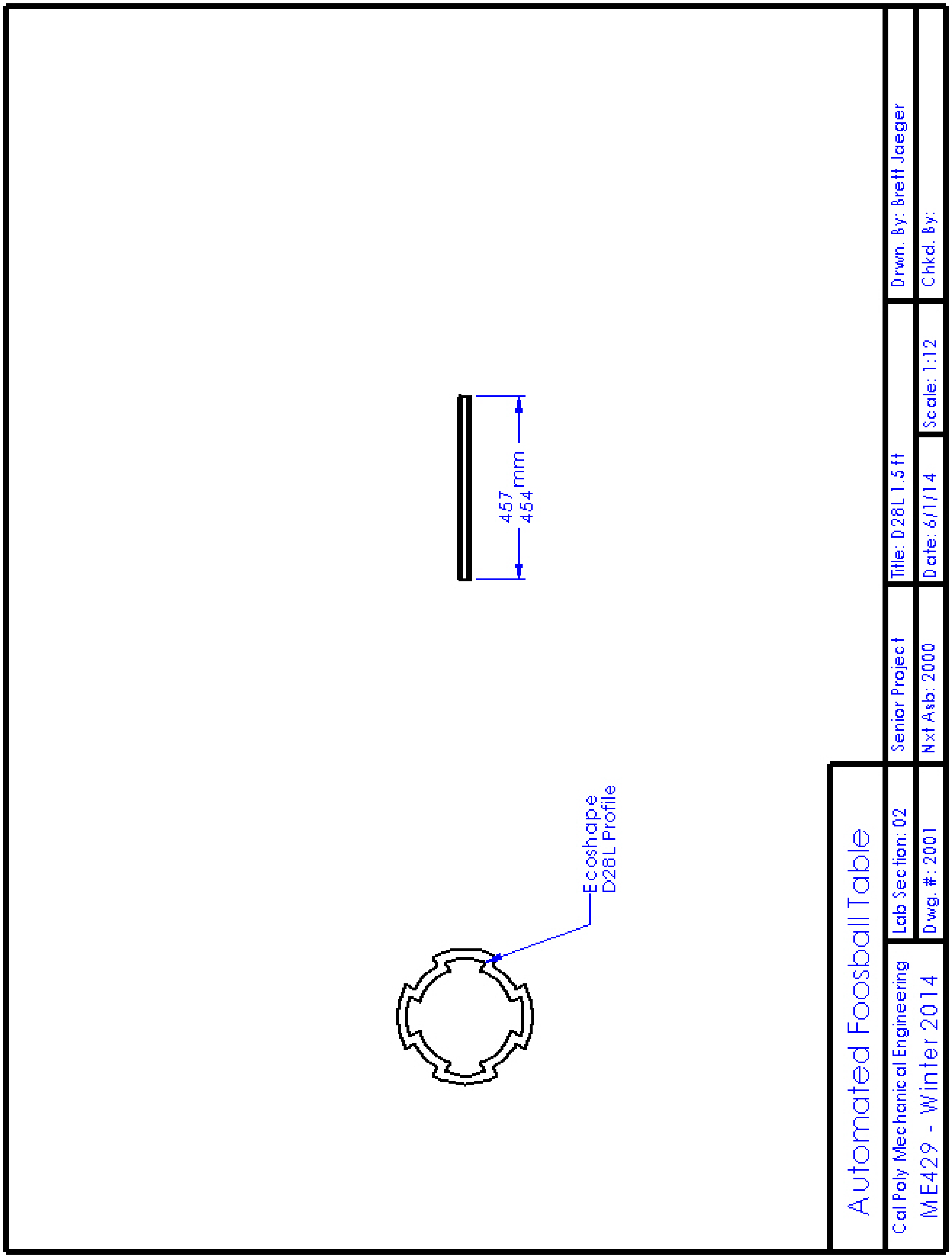
Eco-10-1734


Ecoshape
D28L, N10
Profile
1734
1731
Automated Foosball Table
Cal Poly Mechanical Engineering
ME429 - Winter 2014
Lab Section: 02
Dwg. #: 2003
Senior Project
Nxt Asb: 2000
Title: D28L, N10 1734mm
Date: 6/1/14
Scale: 1:12
Drwn. By: Brett Jaeger
Chkd. By:

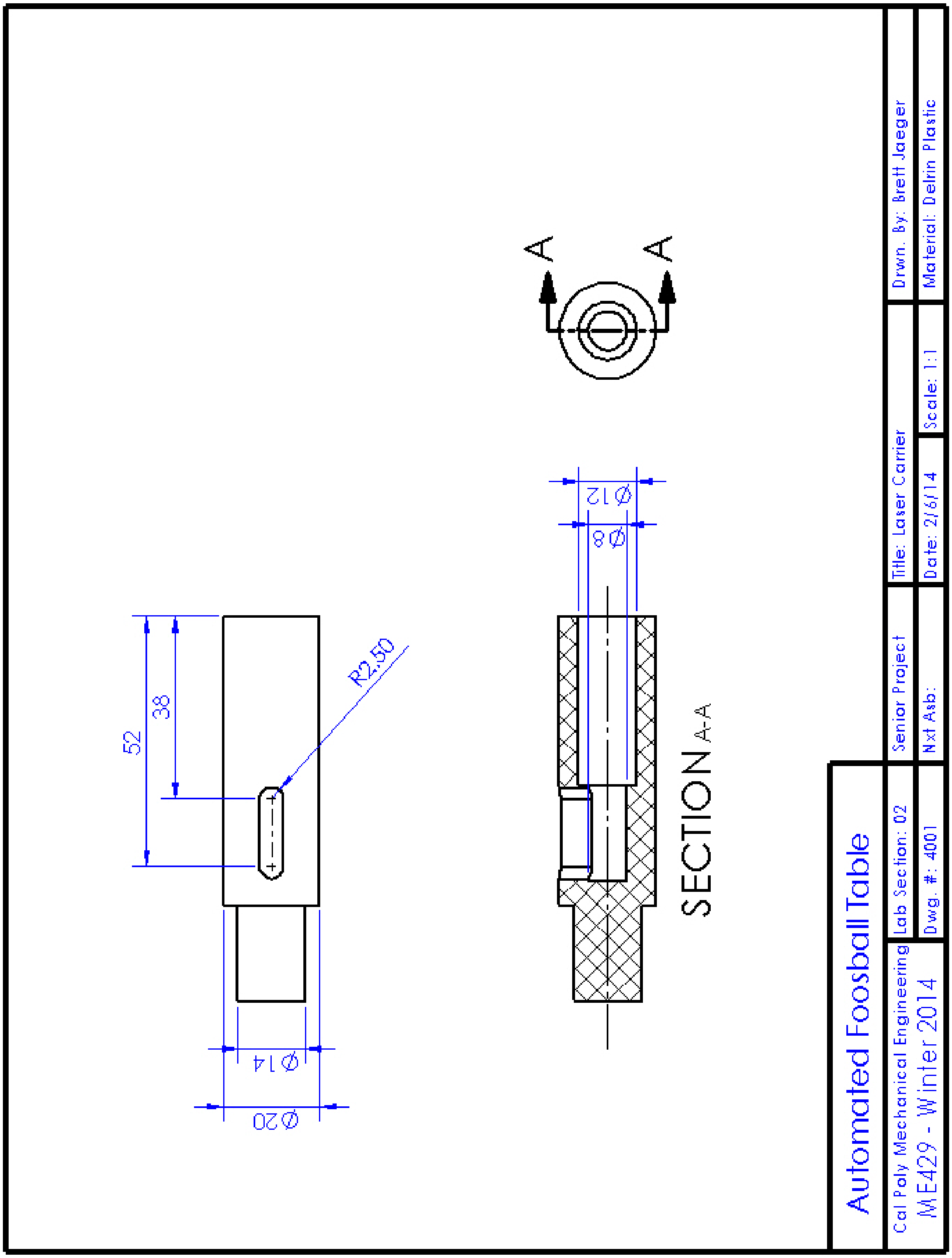
Eco 10 1829


1829
1826
Ecoshape
D28L, N10
Profile
Automated Foosball Table
Cal Poly Mechanical Engineering
ME429 - Winter 2014
Lab Section: 02
Dwg. #: 2003
Senior Project
Nxt Asb: 2000
Title: D28L, N10 1829mm
Date: 6/1/14
Scale: 1:12
Drwn. By: Brett Jaeger
Chkd. By:

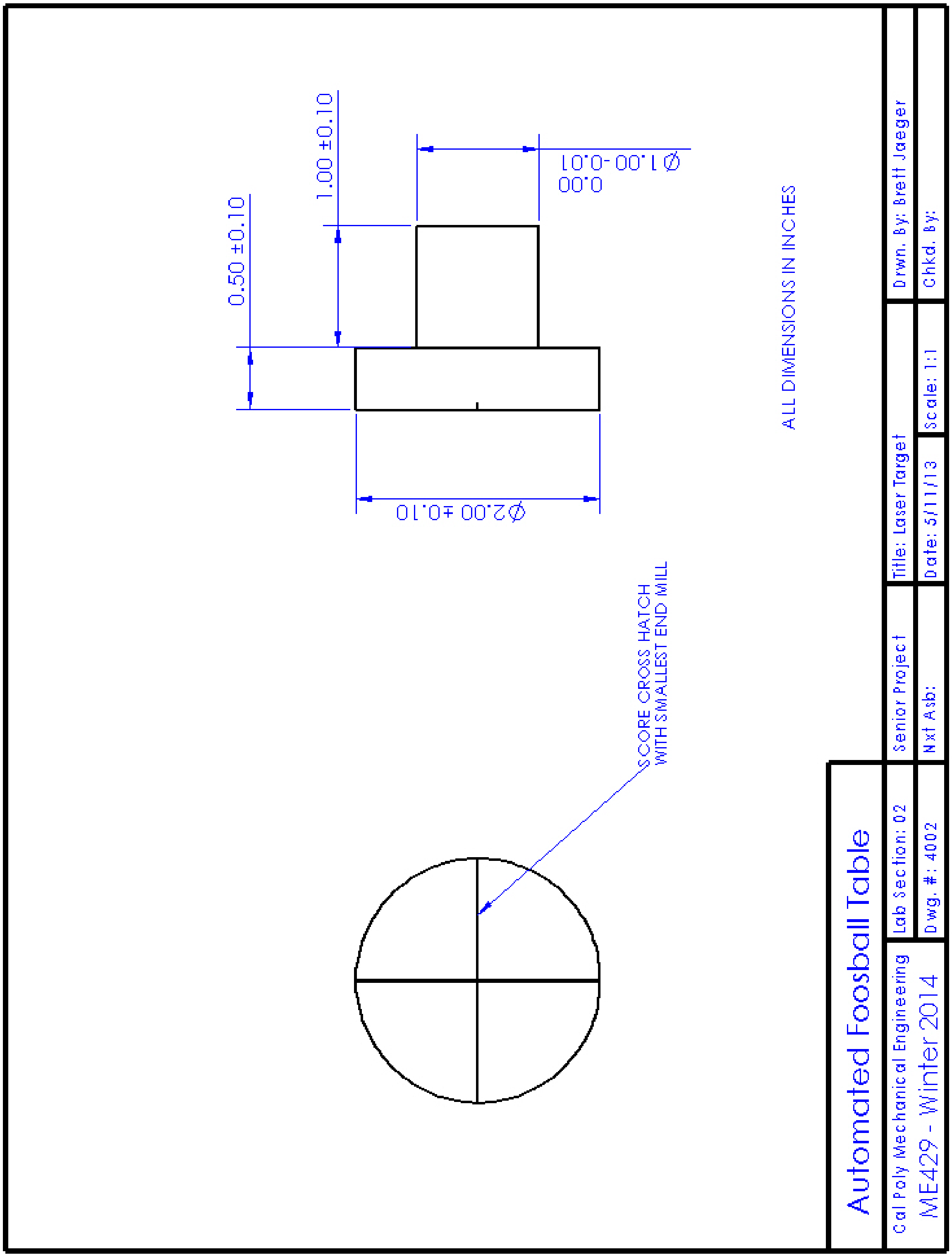
Eco 1.5ft


457
454 mm
Ecoshape
D28L Profile
Automated Foosball Table
Cal Poly Mechanical Engineering
ME429 - Winter 2014
Lab Section: 02
Dwg. #: 2001
Senior Project
Nxt Asb: 2000
Title: D28L 1.5 ft
Date: 6/1/14
Scale: 1:12
Drwn. By: Brett Jaeger
Chkd. By:

Laser Carriage



Laser Target

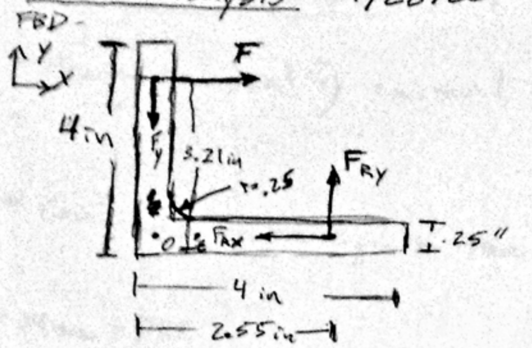


Appendix E: Detailed Analysis

Bracket Analysis

Bracket analysis - 1/28/2014

FBD -



Yield = 35,000 psi

Solution:

$$\sum F_y = 0$$

$$F_y = F_{Ry}$$

$$\sum F_x = 0$$

$$F = F_{RX}$$

$$\sum M_o = 0$$

$$F(3.085\text{in}) = F_{R_y}(2.425\text{in})$$

$$F_{R_y} = 1.272F.$$

Calculate maximum allowable stress assuming point Ƶ has max stress concentration.

$$K_t = \frac{\sigma_{max}}{\sigma_o}$$

$$\sigma_{max} = 35{,}000\text{psi}$$

From Figure A-16 solve K_t

$$\frac{r}{d} = \frac{\cancel{.25}\ .375\text{in}}{.25\text{ in}} = 1.$$

Because $D >> d$, assume $K_t = 1.3$

$$\sigma_o = \frac{\sigma_{max}}{K_t}$$

$$= \frac{35{,}000\text{psi}}{1.3}$$

$\sigma_o = 26{,}923$ psi, allowable

$\sigma_0 = \sigma_{bend} + \sigma_{axial}$

$\sigma_{bend} = \frac{Mc}{I}$

$M = F(3.085 in)$

$c = \frac{.25}{2} = .125 in$

$I = \frac{t\delta^3}{12} = \frac{4in(.25)^3}{12} = 5.2083 \times 10^{-3} in^4$

$\sigma_{bend} = \frac{(3.085 in)(.125 in)F}{5.2083 \times 10^{-3} in^4}$

$\sigma_{bend} = 74.04F \, in^{-2}$

$\sigma_{axial} = \frac{F}{A}$

$= \frac{F_y}{(4in)(.25in)}$

$\sigma_{axial} = F_y = .786F$

∴ bending dominates

Max allowable force

$26,923 psi = 74.04 . F \, in^{-2}$

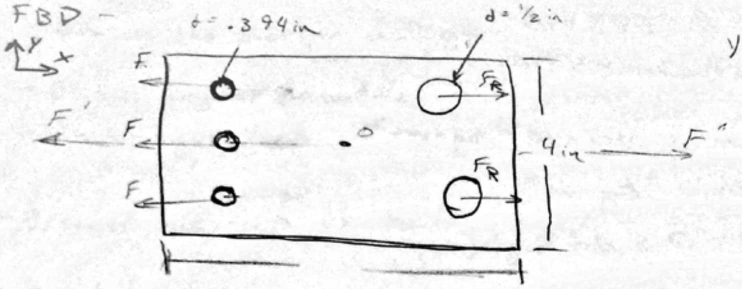
$F_{max} = 363.6 \, lb_f$

Expected load is 24.5 lbf

$\eta = 14.8$

Bracket analysis Continued 1/30/2014

FBD –



Yield = 35,000 psi

t = .25

Solution

$$\sum F_x = 0$$

$$3F = 2F_R = F'$$

$$\sum M_o = 0$$

use A-15-1 in Shigley's for k_t

$k_t = 2.35$, for the 3 m10 holes

$k_t = 2.45$, for the 2 ½ in holes

$$\sigma_0 = \frac{F}{A} = \frac{F'}{(4-3(.394in))(.25in)}$$

$\sigma_0 = 1{,}419\ F'\ in^{-2}$ 10mm holes

$$\sigma_0 = \frac{F}{A} = \frac{F'}{(4-2(.5))(.25)}$$

$\sigma_0 = 1{,}333\ F'\ in^{-2}$, ½ in holes

∴ 10mm holes critical case

$$F'_{max} = 35{,}000\ psi \times \frac{1}{1.419}\ in^2$$

$$F'_{max} = 24{,}665\ lbf$$

$$\eta = 986.6$$

60

Fatigue analysis For the left brackets 2/2/2014

assume -

$F = 25\ \text{lbf}$, Fully reversed

$S_y = 35{,}000\ \text{psi}$

Bending dominates

$\sigma_{bend} = 74.04F$

Solve for the factor of safety, using the mod-Goodman criterion

mod-Goodman $\dfrac{\sigma_a}{S_e} + \dfrac{\sigma_m}{S_{ut}} = \dfrac{1}{n}$

$\sigma_m = \dfrac{\sigma_{max} + \sigma_{min}}{2}$

$\sigma_a = \left| \dfrac{\sigma_{max} - \sigma_{min}}{2} \right|$

$\sigma_{max} = 74.04\,(25\text{lbf}) = 1851\ \text{psi}$

$\sigma_{min} = 74.04(-25\text{lbf}) = -1851\ \text{psi}$

$\sigma_m = \dfrac{1851\text{psi} - 1851\text{psi}}{2} = 0$

$\sigma_a = \left| \dfrac{1851 + 1851}{2} \right| = 1851\ \text{psi}$

For aluminium, there is no endurance limit. therefore a comparison between σ_a and Al-6061 Fatigue strength will be made.

$S_F = 14{,}000\ \text{psi}$ for 5×10^8 cycles

$n = \dfrac{S_F}{\sigma_a} = \dfrac{14{,}000\ \text{psi}}{1851\ \text{psi}}$

$n = 7.56$ Acceptable

For right side

$\sigma_a = 33.33\ \text{psi}$

$S_F = 14{,}000\ \text{psi}$

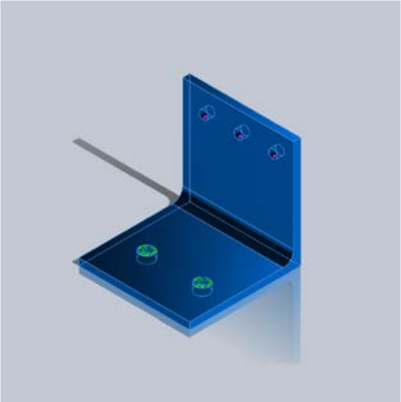
$n = \dfrac{S_F}{\sigma_a}$

$n = 420$

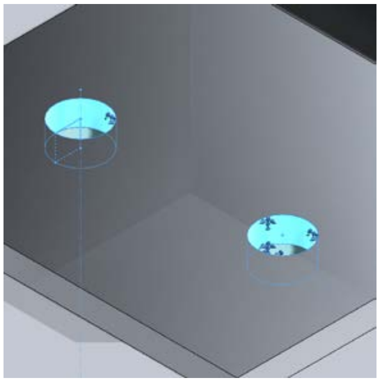
Units

Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m^2

Material Properties

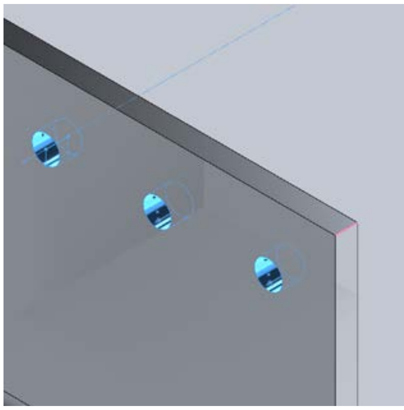
Model Reference	Properties	Components
	Name: **6061 Alloy** Model type: **Linear Elastic Isotropic** Default failure criterion: **Max von Mises Stress** Yield strength: **5.51485e+007 N/m^2** Tensile strength: **1.24084e+008 N/m^2** Elastic modulus: **6.9e+010 N/m^2** Poisson's ratio: **0.33** Mass density: **2700 kg/m^3** Shear modulus: **2.6e+010 N/m^2** Thermal expansion coefficient: **2.4e-005 /Kelvin**	**SolidBody 1(1/2 (0.5) Diameter Hole1)(Leftsidebracket)**
Curve Data:N/A		

Loads and Fixtures

Fixture name	Fixture Image	Fixture Details
Fixed-1		**Entities:** **2 face(s)** **Type:** **Fixed Geometry**

Resultant Forces

Components	**X**	**Y**	**Z**	**Resultant**
Reaction force(N)	**-0.000137806**	**0.00205231**	**-109.017**	**109.017**
Reaction Moment(N·m)	**0**	**0**	**0**	**0**

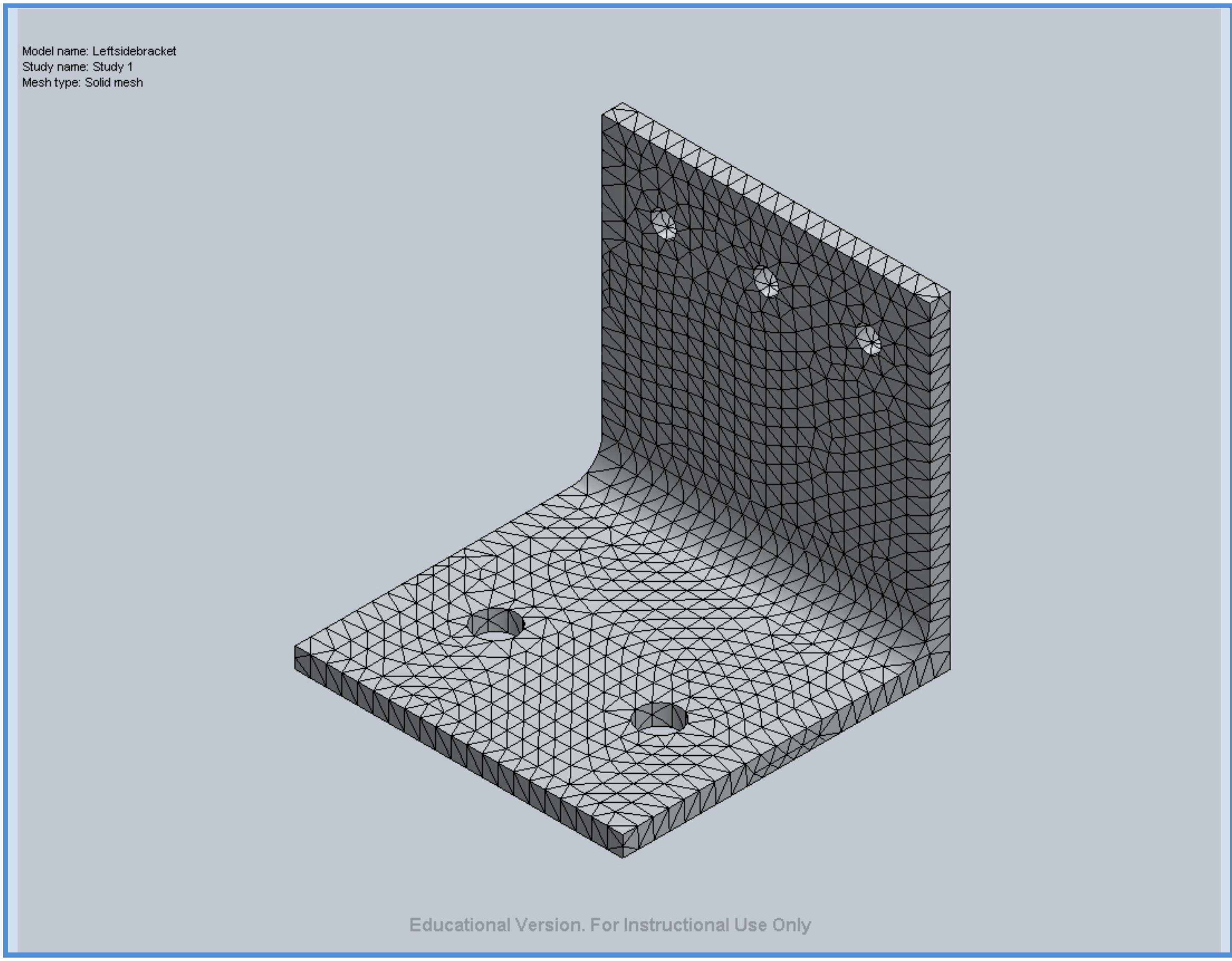
Load name	Load Image	Load Details
Force-1		**Entities:** **3 face(s)** **Reference:** **Edge< 1 >** **Type:** **Apply force** **Values:** **---, ---, 109 N**

Mesh Information

Mesh type	Solid Mesh
Mesher Used:	Standard mesh
Automatic Transition:	Off
Include Mesh Auto Loops:	Off
Jacobian points	4 Points
Element Size	0.197648 in
Tolerance	0.00988239 in
Mesh Quality	High

Mesh Information - Details

Total Nodes	15484
Total Elements	8886
Maximum Aspect Ratio	3.6542
% of elements with Aspect Ratio < 3	99.8
% of elements with Aspect Ratio > 10	0
% of distorted elements(Jacobian)	0
Time to complete mesh(hh;mm;ss):	00:00:01
Computer name:	ME-192-134-15


Model name: Leftsidebracket
Study name: Study 1
Mesh type: Solid mesh
Educational Version. For Instructional Use Only

Resultant Forces

Reaction Forces

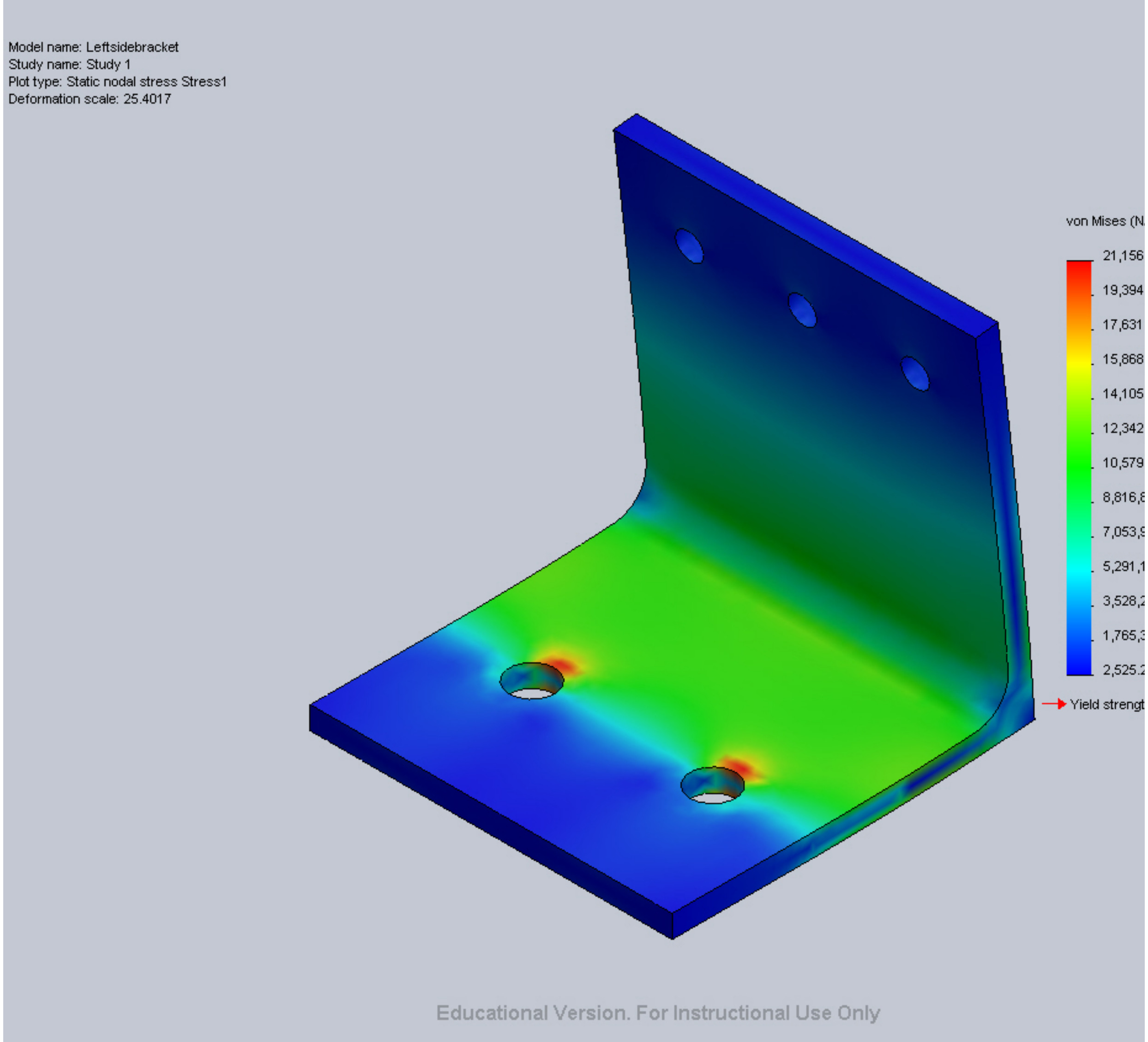
Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	-0.000137806	0.00205231	-109.017	109.017

Reaction Moments

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N·m	0	0	0	0

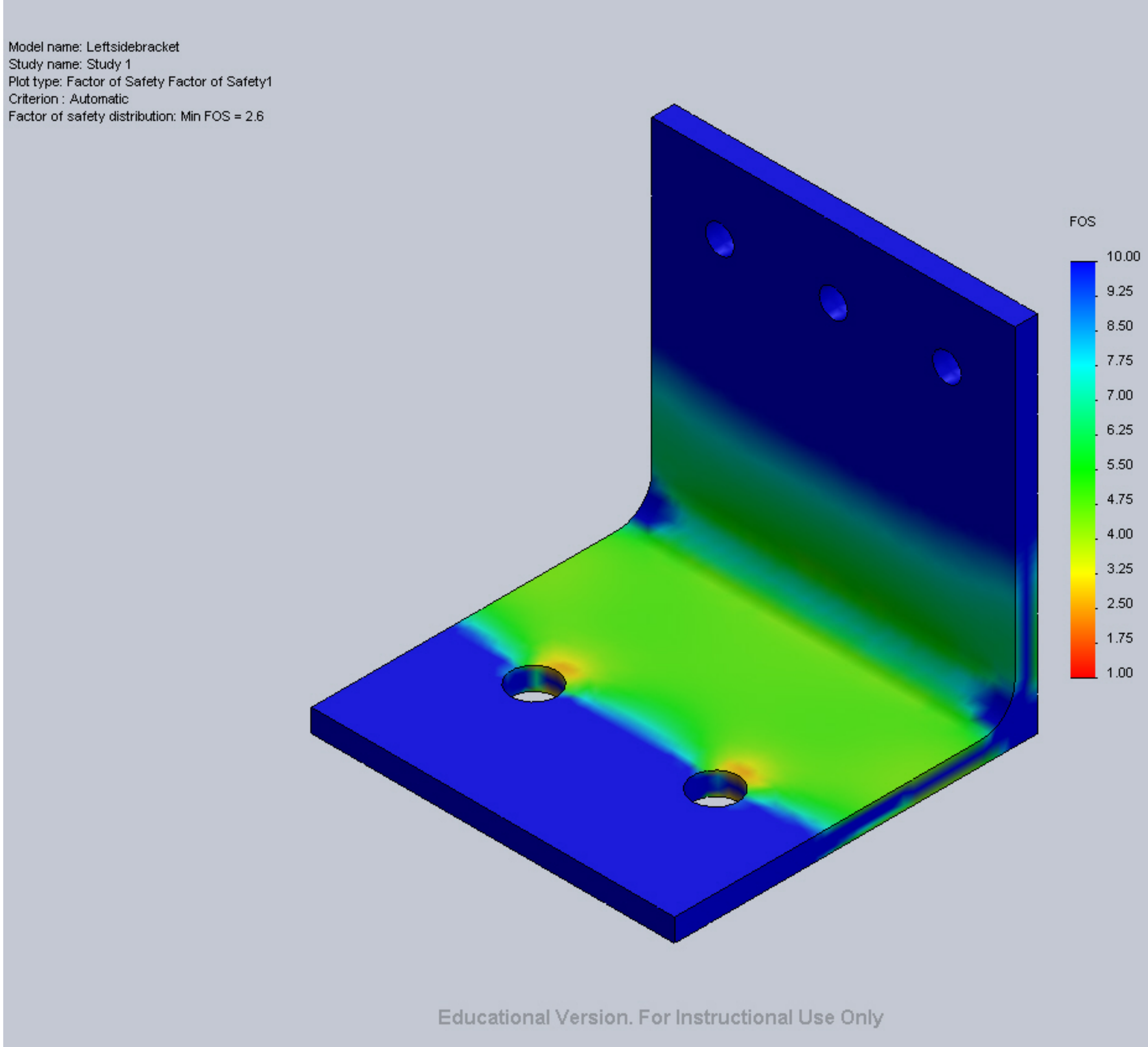
Study Results

Name	Type	Min	Max
Stress1	VON: von Mises Stress	2525.18 N/m^2 Node: 464	2.11569e+007 N/m^2 Node: 15261



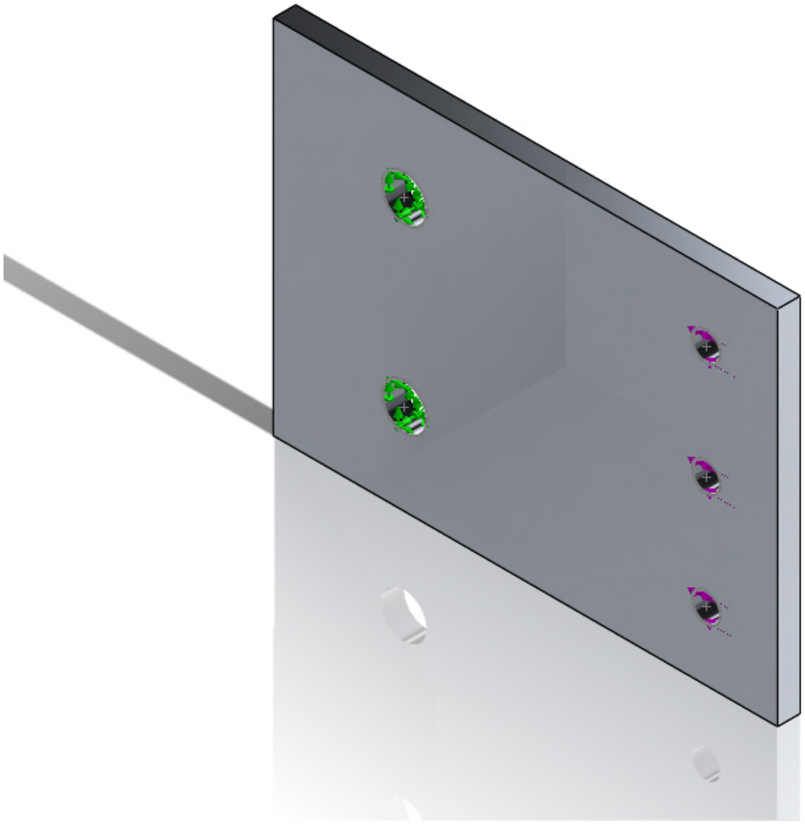
Leftsidebracket-Study 1-Stress-Stress1

Name	**Type**	**Min**	**Max**
Factor of Safety1	Automatic	2.60664 Node: 15261	21839.5 Node: 464



Leftsidebracket-Study 1-Factor of Safety-Factor of Safety1

Model Information



Model name: Rightsidebracket
Current Configuration: Default

Solid Bodies			
Document Name and Reference	Treated As	Volumetric Properties	Document Path/Date Modified
1/2 (0.5) Diameter Hole1	Solid Body	Mass:0.239728 kg Volume:8.87883e-005 m^3 Density:2700 kg/m^3 Weight:2.34934 N	C:\Users\melab\Downloads \Rightsidebracket.SLDPRT Feb 01 13:23:31 2014

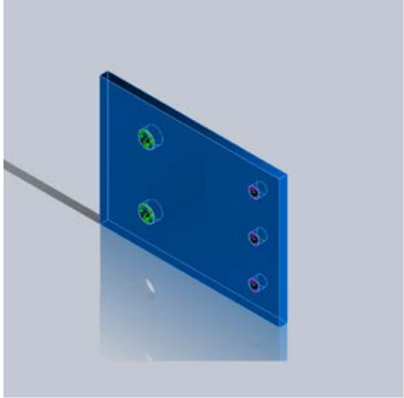
Study Properties

Study name	Study 1
Analysis type	Static
Mesh type	Solid Mesh
Thermal Effect:	On
Thermal option	Include temperature loads
Zero strain temperature	298 Kelvin
Include fluid pressure effects from SolidWorks Flow Simulation	Off
Solver type	FFEPlus
Inplane Effect:	Off
Soft Spring:	Off
Inertial Relief:	Off
Incompatible bonding options	Automatic
Large displacement	Off
Compute free body forces	On
Friction	Off
Use Adaptive Method:	Off
Result folder	SolidWorks document (C:\Users\melab\Downloads)

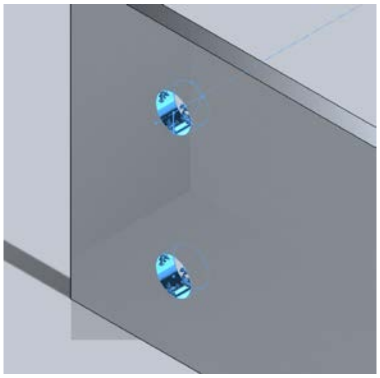
Units

Unit system:	SI (MKS)
Length/Displacement	mm
Temperature	Kelvin
Angular velocity	Rad/sec
Pressure/Stress	N/m^2

Material Properties

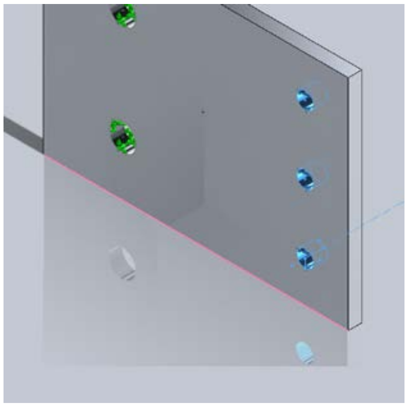
Model Reference	Properties	Components
	Name: **6061 Alloy** Model type: **Linear Elastic Isotropic** Default failure criterion: **Max von Mises Stress** Yield strength: **5.51485e+007 N/m^2** Tensile strength: **1.24084e+008 N/m^2** Elastic modulus: **6.9e+010 N/m^2** Poisson's ratio: **0.33** Mass density: **2700 kg/m^3** Shear modulus: **2.6e+010 N/m^2** Thermal expansion coefficient: **2.4e-005 /Kelvin**	**SolidBody 1(1/2 (0.5) Diameter Hole1)(Rightsidebracket)**
Curve Data:N/A		

Loads and Fixtures

Fixture name	Fixture Image	Fixture Details
Fixed-1		**Entities:** 2 face(s) **Type:** Fixed Geometry

Resultant Forces

Components	**X**	**Y**	**Z**	**Resultant**
Reaction force(N)	**108.997**	**-0.00358787**	**-0.00190064**	**108.997**
Reaction Moment(N·m)	**0**	**0**	**0**	**0**

Load name	Load Image	Load Details
Force-1		**Entities:** 3 face(s) **Reference:** Edge< 1 > **Type:** Apply force **Values:** ---, ---, -109 N

Connector Definitions

No Data

Contact Information

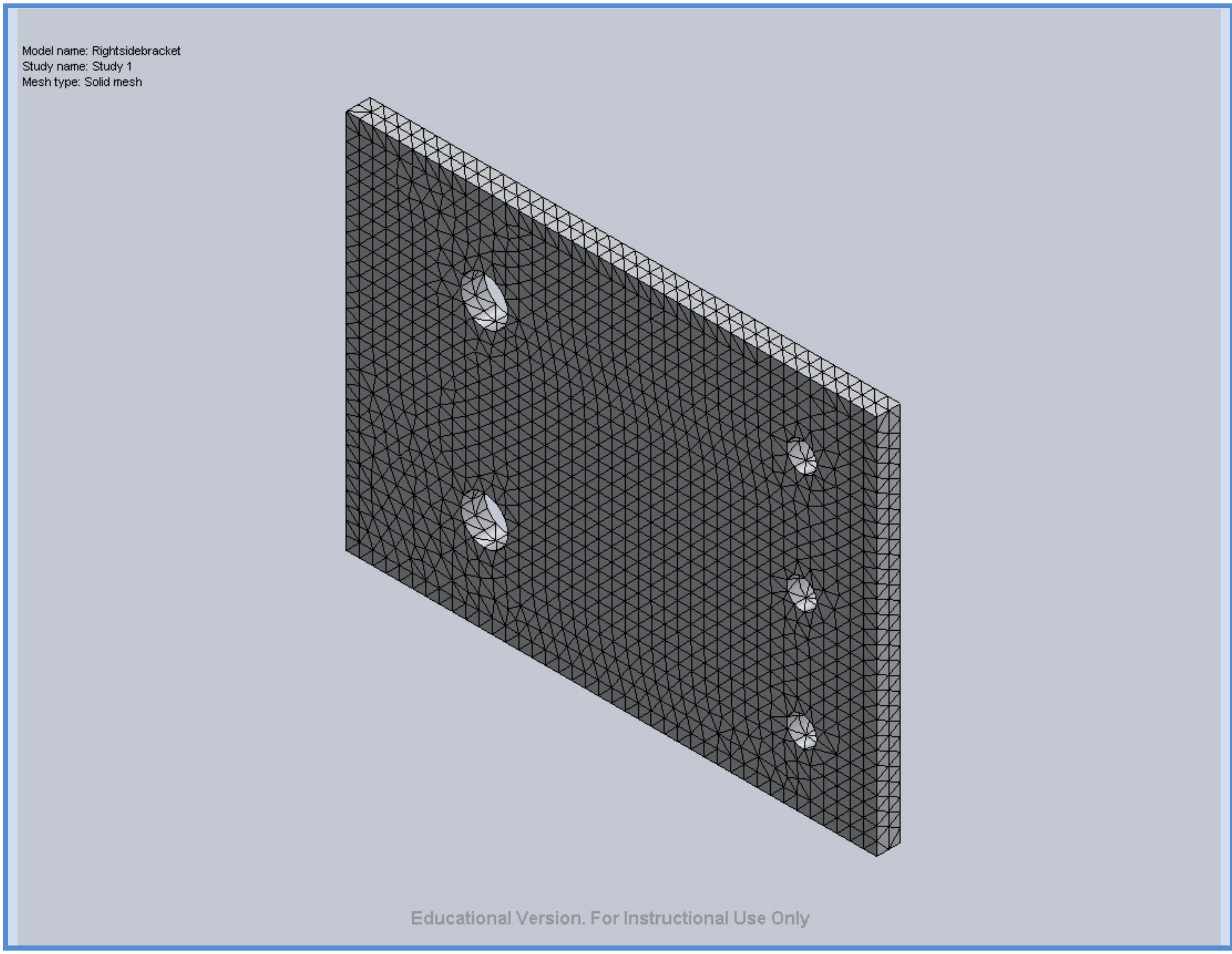
No Data

Mesh Information

Mesh type	Solid Mesh
Mesher Used:	Standard mesh
Automatic Transition:	Off
Include Mesh Auto Loops:	Off
Jacobian points	4 Points
Element Size	0.140554 in
Tolerance	0.0070277 in
Mesh Quality	High

Mesh Information - Details

Total Nodes	24049
Total Elements	14081
Maximum Aspect Ratio	3.0805
% of elements with Aspect Ratio < 3	100
% of elements with Aspect Ratio > 10	0
% of distorted elements(Jacobian)	0
Time to complete mesh(hh;mm;ss):	00:00:02
Computer name:	ME-192-134-15



Sensor Details

No Data

Resultant Forces

Reaction Forces

Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N	108.997	-0.00358787	-0.00190064	108.997

Reaction Moments

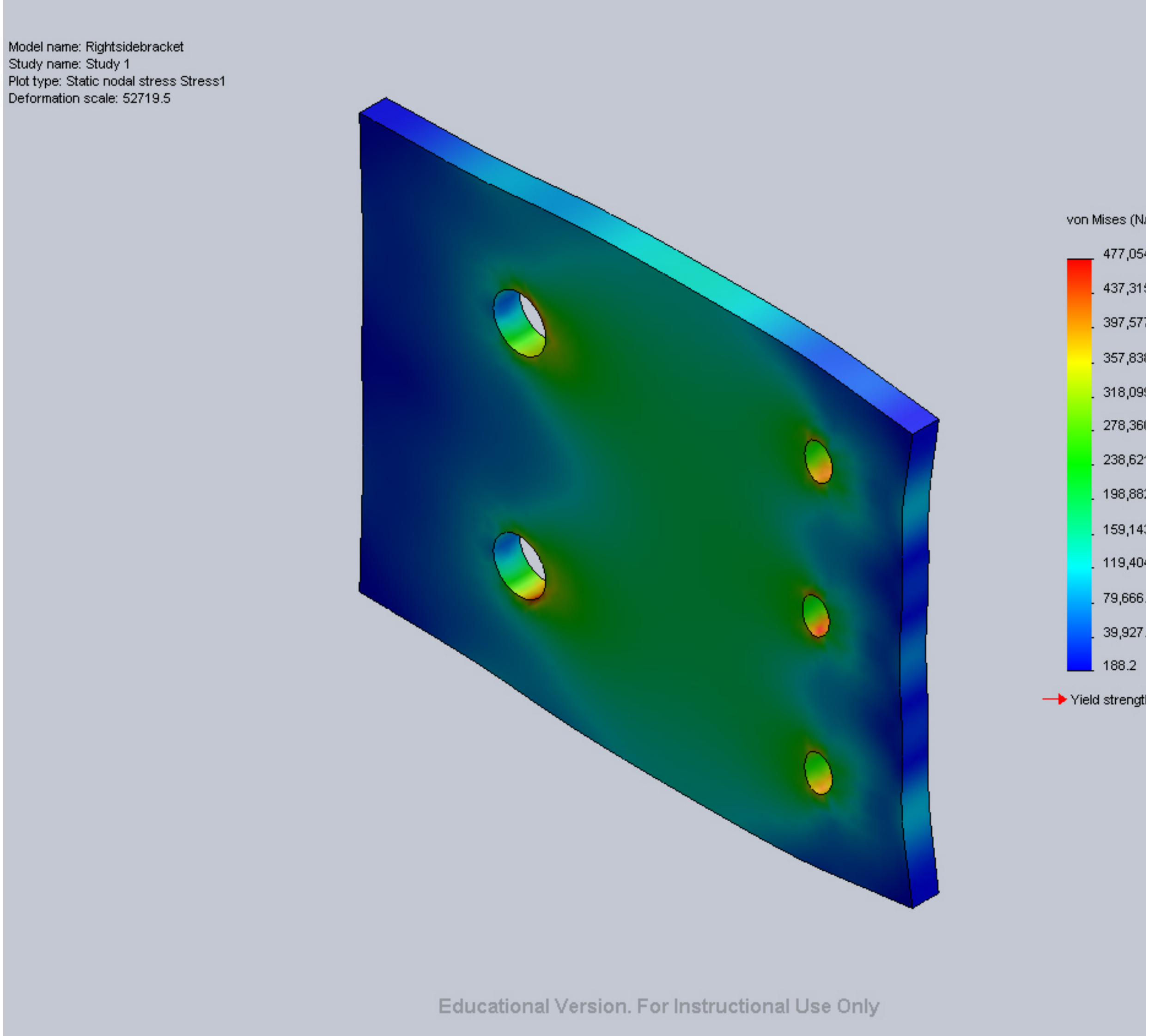
Selection set	Units	Sum X	Sum Y	Sum Z	Resultant
Entire Model	N·m	0	0	0	0

Beams

No Data

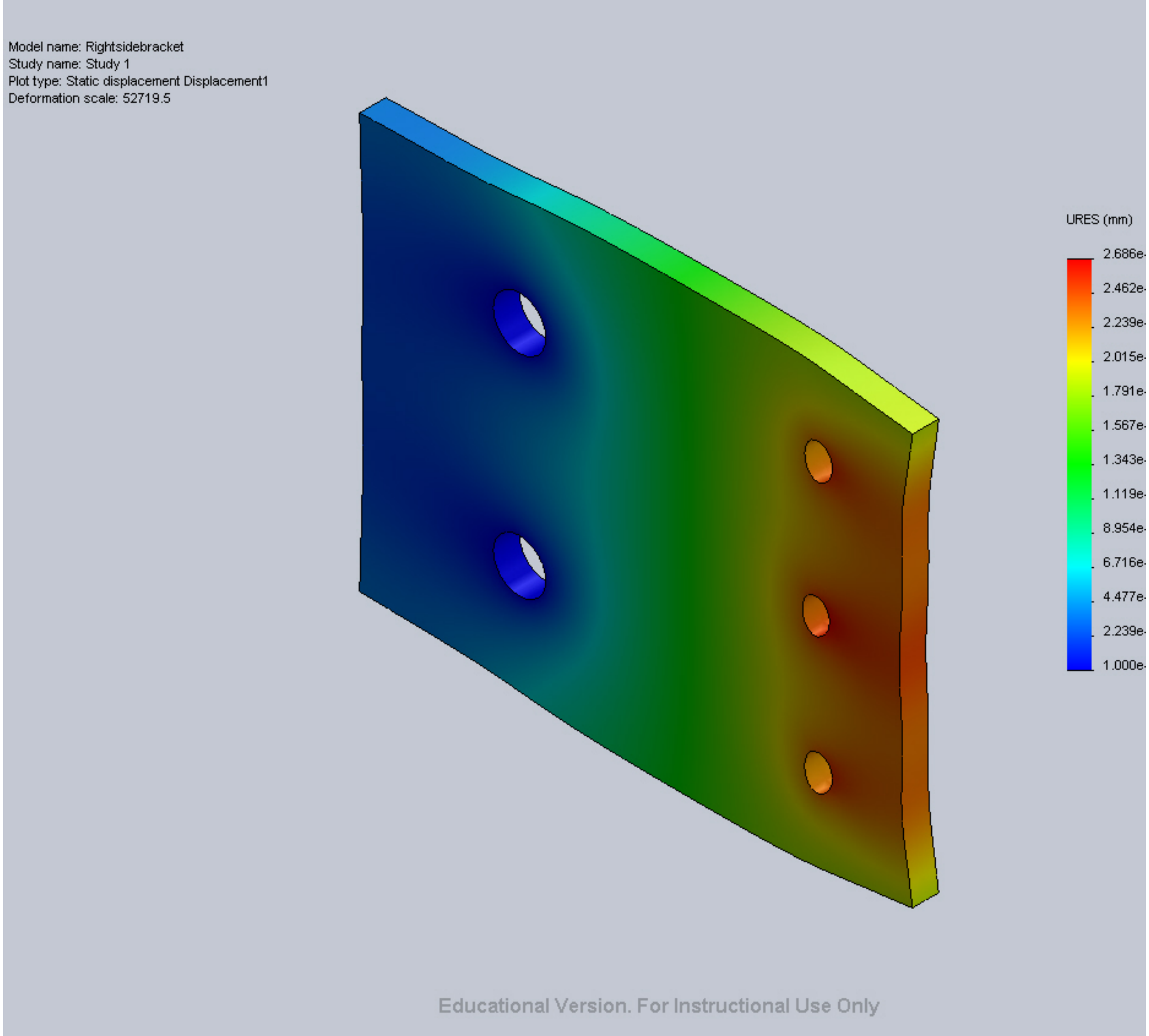
Study Results

Name	Type	Min	Max
Stress1	VON: von Mises Stress	188.227 N/m^2 Node: 21873	477055 N/m^2 Node: 23520



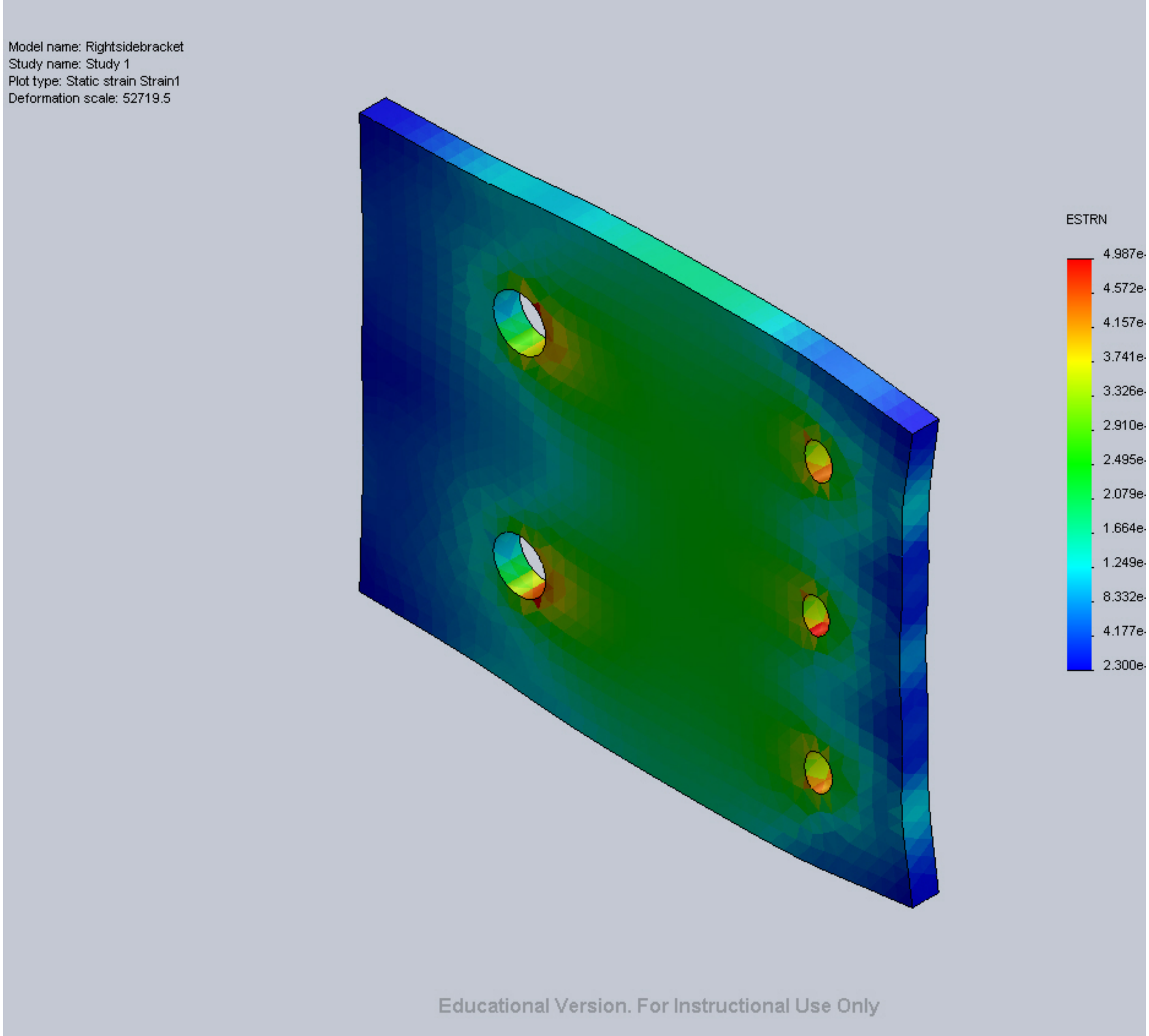
Rightsidebracket-Study 1-Stress-Stress1

Name	Type	Min	Max
Displacement1	URES: Resultant Displacement	0 mm Node: 1	0.000268627 mm Node: 74



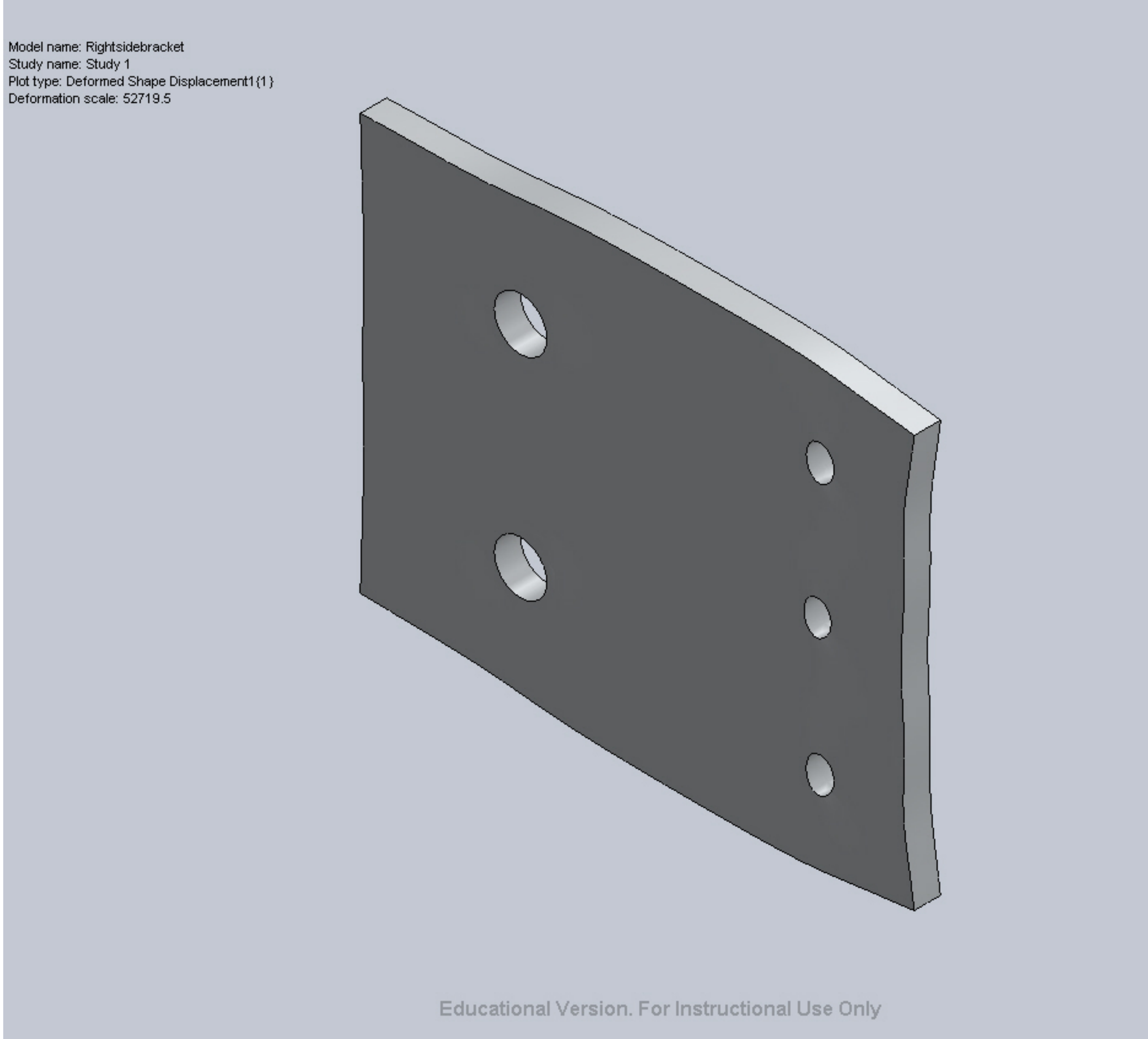
Rightsidebracket-Study 1-Displacement-Displacement1

Name	**Type**	**Min**	**Max**
Strain1	ESTRN: Equivalent Strain	2.29962e-009 Element: 11095	4.9875e-006 Element: 448



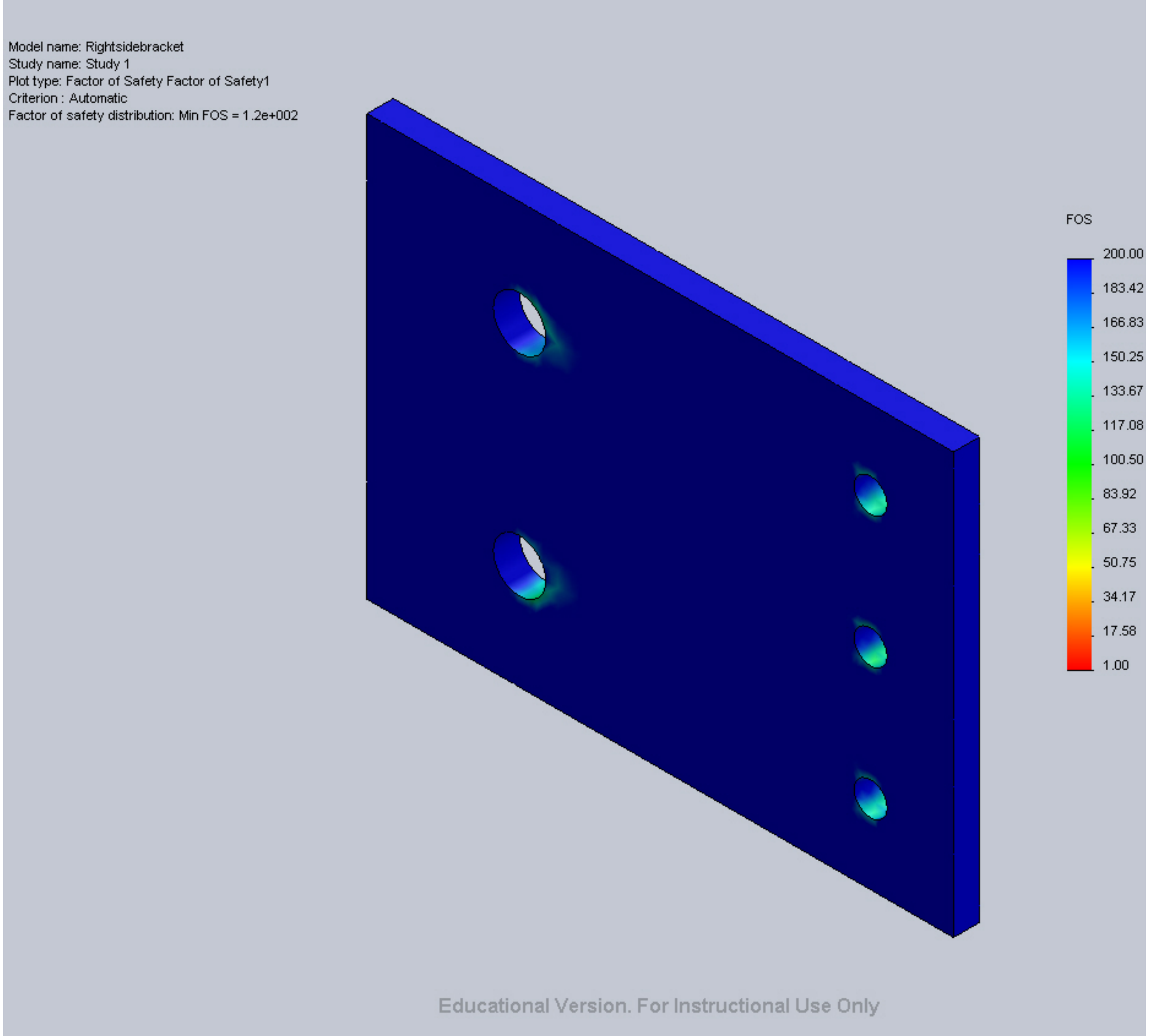
Rightsidebracket-Study 1-Strain-Strain1

Name	**Type**
Displacement1{1}	Deformed Shape



Rightsidebracket-Study 1-Displacement-Displacement1{1}

Name	**Type**	**Min**	**Max**
Factor of Safety1	Automatic	115.602 Node: 23520	292989 Node: 21873



Model name: Rightsidebracket
Study name: Study 1
Plot type: Factor of Safety Factor of Safety1
Criterion : Automatic
Factor of safety distribution: Min FOS = 1.2e+002

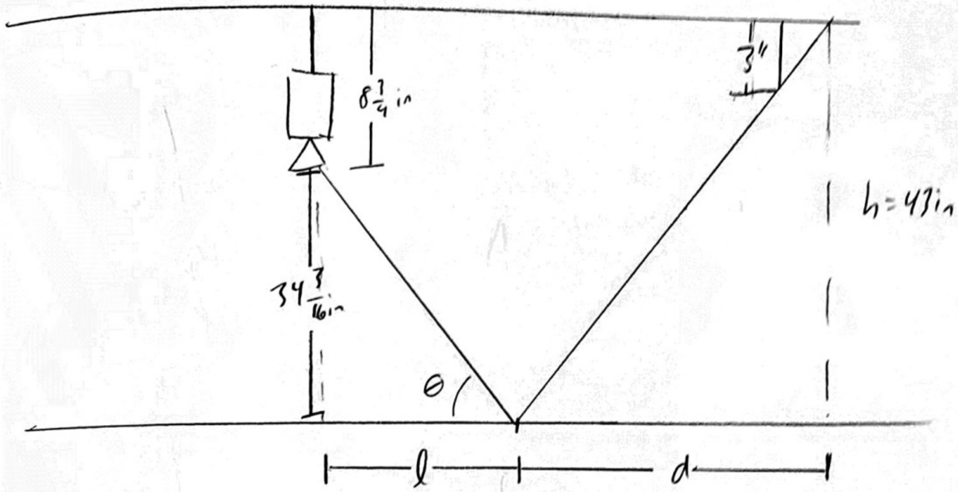
Rightsidebracket-Study 1-Factor of Safety-Factor of Safety1

Roof Analysis

Camera Shelter Calcs

1-16-14

⓸⓻



X: $l = 23\frac{1}{8}$in , $h = 43$in ← long side of table

$$\frac{34\frac{3}{16}}{23\frac{1}{8}} = \frac{43}{d}$$

$d = 29.1$in

Y: $l = 13.5$in ← short side of table

$$\frac{13.5}{34\frac{3}{16}} = \frac{d}{43}$$

$d = 17.0$ in

If we have a 3" overhang:

X: $$\frac{23.125}{34.1875} = \frac{d}{40}$$

$d = 27.1$ in

Y: $$\frac{13.5}{34.1875} = \frac{d}{40}$$

$d = 13.8$ in

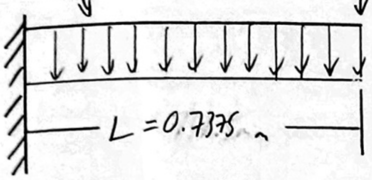
Ecoshape Bending Calculations

weight per unit length of ecoshape

$\omega = (0.48\ \mathrm{kg/m})(9.81\ \mathrm{m/s^2})$ $P = \frac{(2.79\ \mathrm{m})(0.48\ \mathrm{kg/m})(9.81\ \mathrm{m/s^2})}{2} = 6.57\ \mathrm{N}$

length of long bar supported by 4 beams

Supported by 2 beams



$$\delta_\omega = \frac{\omega L^4}{8EI}$$ ← Deflection due to weight

$$\delta_\omega = \frac{(0.48\ \mathrm{kg/m})(9.81\ \mathrm{m/s^2})(0.7375\ \mathrm{m})^4}{8(69\times10^9\ \mathrm{N/m^2})(1.33\ \mathrm{cm^4})\left(\frac{1\ \mathrm{m}}{100\ \mathrm{cm}}\right)^4}\left(\frac{1000\ \mathrm{mm}}{1\ \mathrm{m}}\right)$$

$$\underline{\delta_\omega = 0.19\ \mathrm{mm}}$$

$$\delta_P = \frac{PL^3}{3EI}$$ ← Deflection due to beams on end

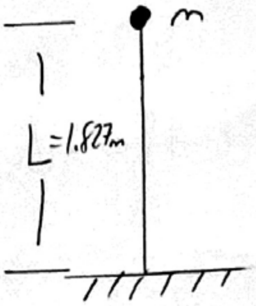
$$\delta_P = \frac{(6.57\ \mathrm{N})(0.7375)^4}{3(69\times10^9\ \mathrm{N/m^2})(1.33\ \mathrm{cm^4})\left(\frac{1\ \mathrm{m}}{100\ \mathrm{cm}}\right)^4}$$

$$\underline{\delta_P = 0.71\ \mathrm{mm}}$$

$$\boxed{\delta_{TOT} = 0.90\ \mathrm{mm}}$$ ← This value is certainly acceptable

1. ⓓ Vibration of Vision Arch

* Model as a beam with point mass on end
- point mass is half the mass of the cross beam, the roof, and the scoreboard combined



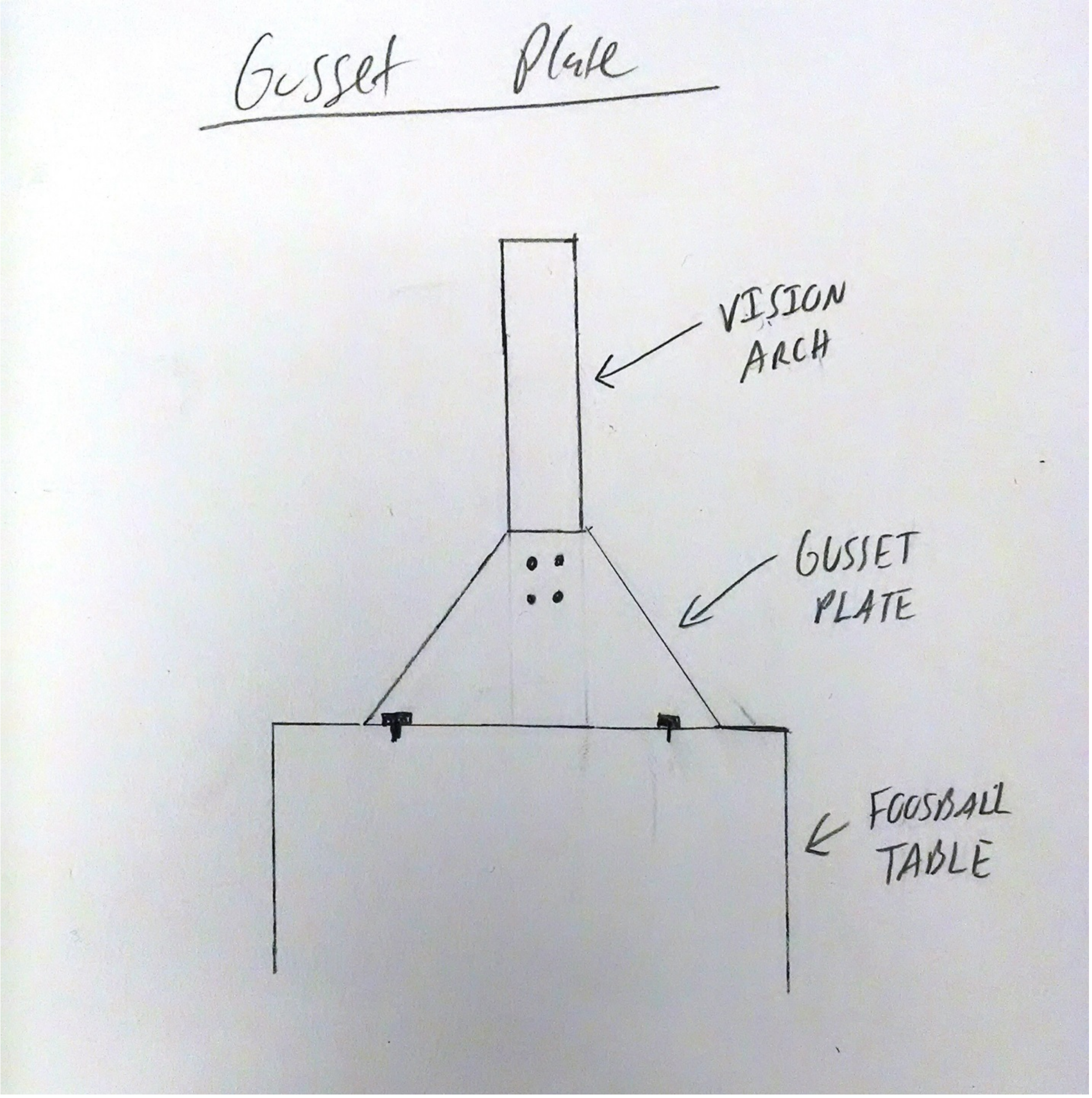
$$\omega_n = \sqrt{\frac{3EI}{(m + 0.23\, m_{beam})L^3}}$$

$$\omega_n = \sqrt{\frac{3(69\times10^9 \frac{N}{m^2})(8.184\times10^{-7} m^4)}{(8.1kg + 0.23(3 kg/m)(1.827m))(1.827m)^3}}$$

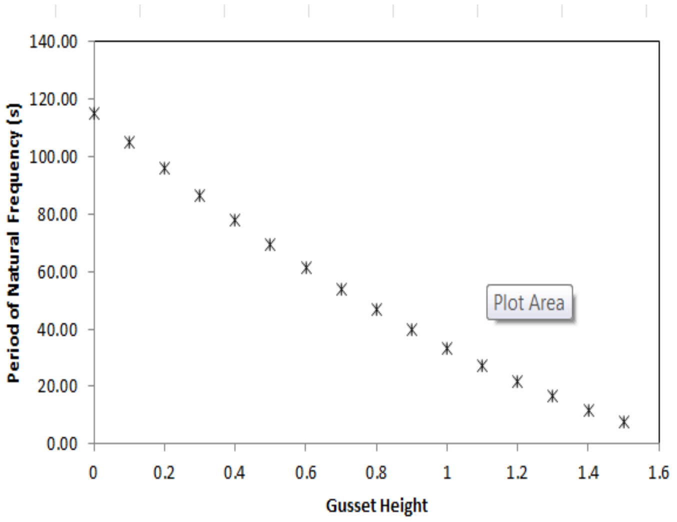
ω_n = 54.5 rad/s or 8.67 Hz

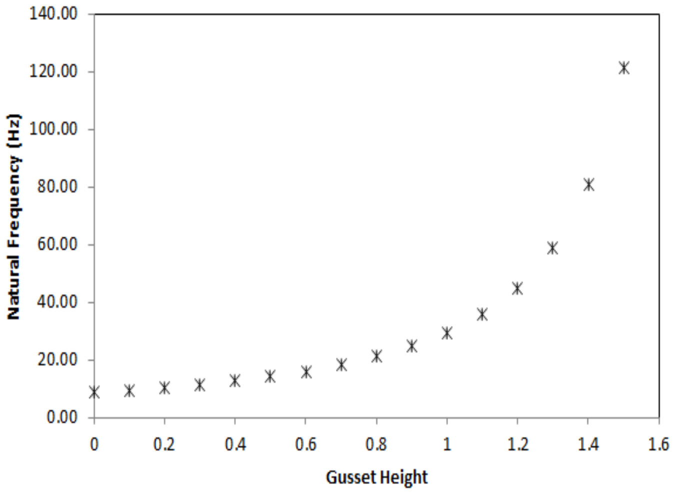
*(Adding a gusset will decrease the length (L) and therefore increase the natural frequency

Gusset Plate



Constants					Variables				
E (N/m^2)	I (cm^4)	I (m^4)	mu (kg/m)	m(kg)	h(m)	L(m)	omega(rad/s)	frequency (Hz)	Period(ms)
6.90E+10	81.84	8.18E-07	3	8.1	0	1.827	5.45E+01	8.67	115.34
					0.1	1.727	5.95E+01	9.47	105.61
					0.2	1.627	6.53E+01	10.39	96.21
					0.3	1.527	7.21E+01	11.47	87.15
					0.4	1.427	8.01E+01	12.75	78.43
					0.5	1.327	8.97E+01	14.27	70.07
					0.6	1.227	1.01E+02	16.11	62.06
					0.7	1.127	1.15E+02	18.38	54.42
					0.8	1.027	1.33E+02	21.21	47.15
					0.9	0.927	1.56E+02	24.83	40.28
					1	0.827	1.86E+02	29.58	33.81
					1.1	0.727	2.26E+02	36.03	27.75
					1.2	0.627	2.84E+02	45.17	22.14
					1.3	0.527	3.70E+02	58.86	16.99
					1.4	0.427	5.09E+02	81.03	12.34
					1.5	0.327	7.63E+02	121.41	8.24





Appendix F: Testing

Vibration Testing

This section details the procedure used to test the vibrations in the table and cabinet during the operation of one of the linear motors. Data was collected from four different locations, the top of the Plexiglas on the cabinet, a horizontal 8020 strut on the top of the cabinet, the top of the table, and the top of the vision arch. Three separate move scenarios were observed, a long move with the cabinet and table unattached, a short move with the cabinet and table unattached, a short move with the cabinet and table attached. Table 24 contains the parameters used in the JOG function to perform the test moves. Figures 46 - 57 are vibration plots of each test at each of the locations and Table 25 contains the results of the vibration analysis.

Table 25: Contains the parameters used for the JOG function during the vibration testing

Move Type	Move Distance (mm)	Motor Speed (min-1)	Actuator Speed (m/s)	Acceleration Time (ms)	Cycles
Long	400	6000	1.87	150	1
Short	40	6000	0.59	150	5

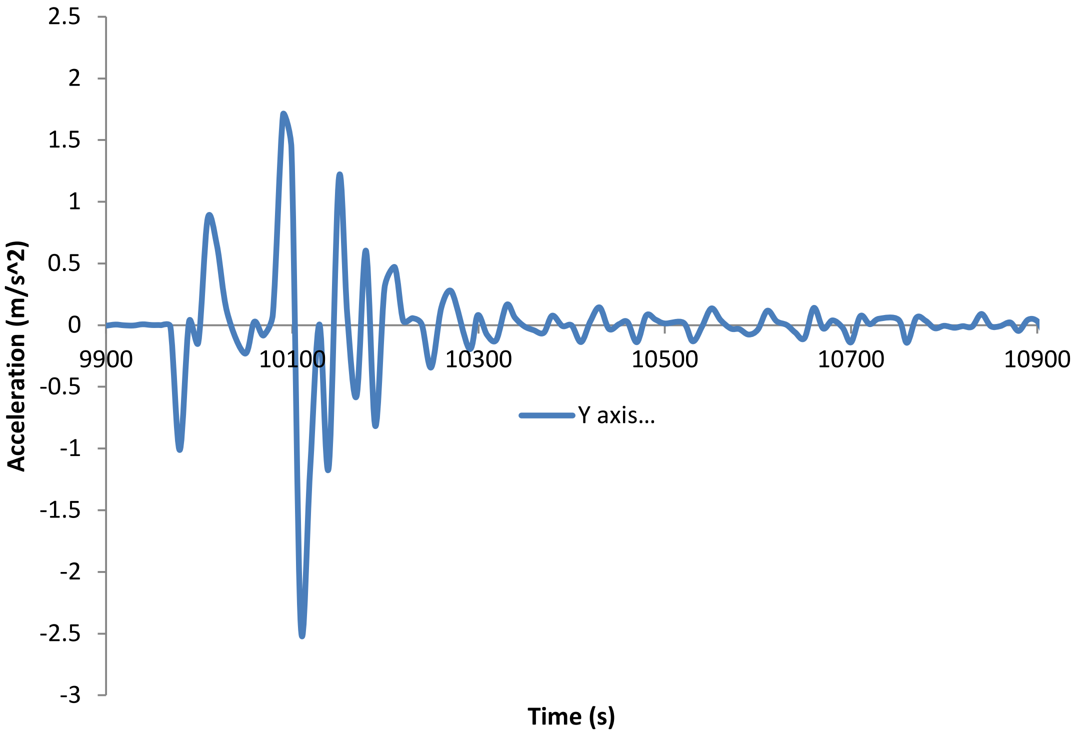


Figure 56: Vibration data for a single move taken on the Plexiglas

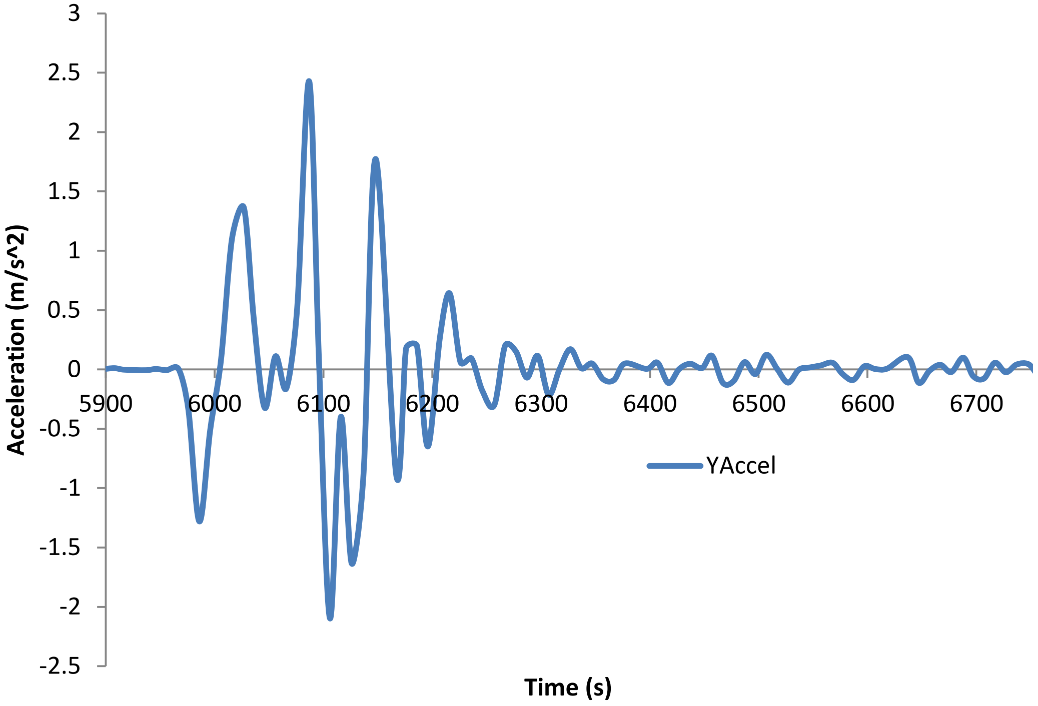


Figure 57: Vibration data for a single move taken at the Horizontal 8020 strut

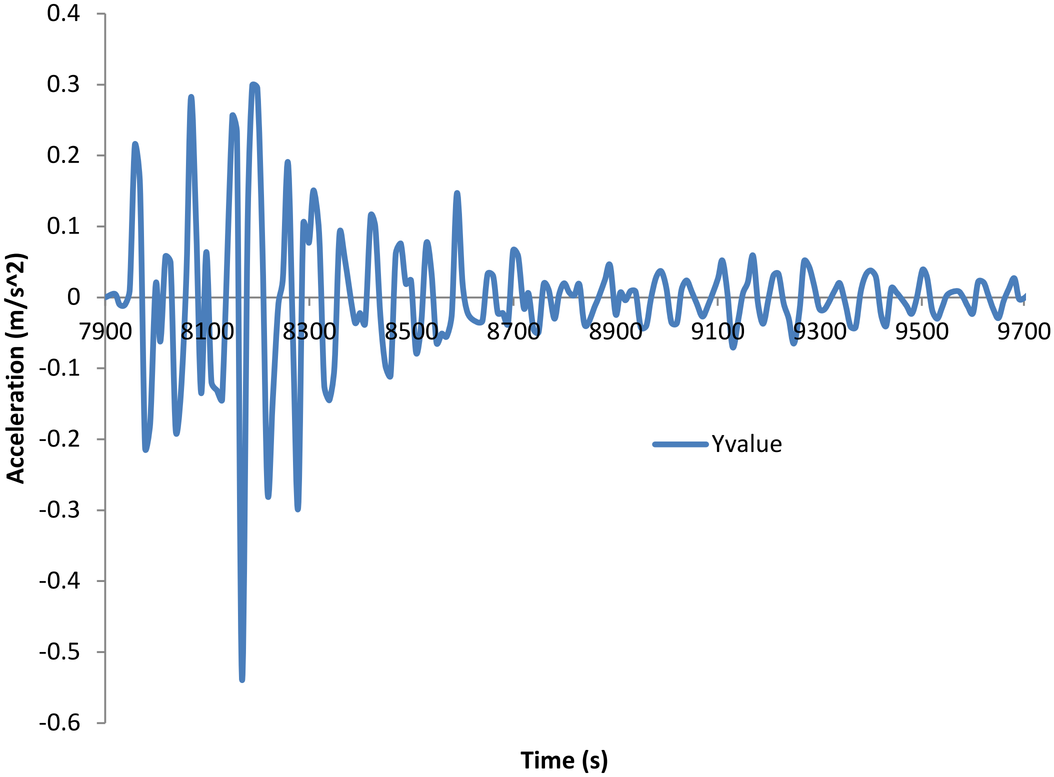


Figure 58: Vibration data for a single move taken on the table

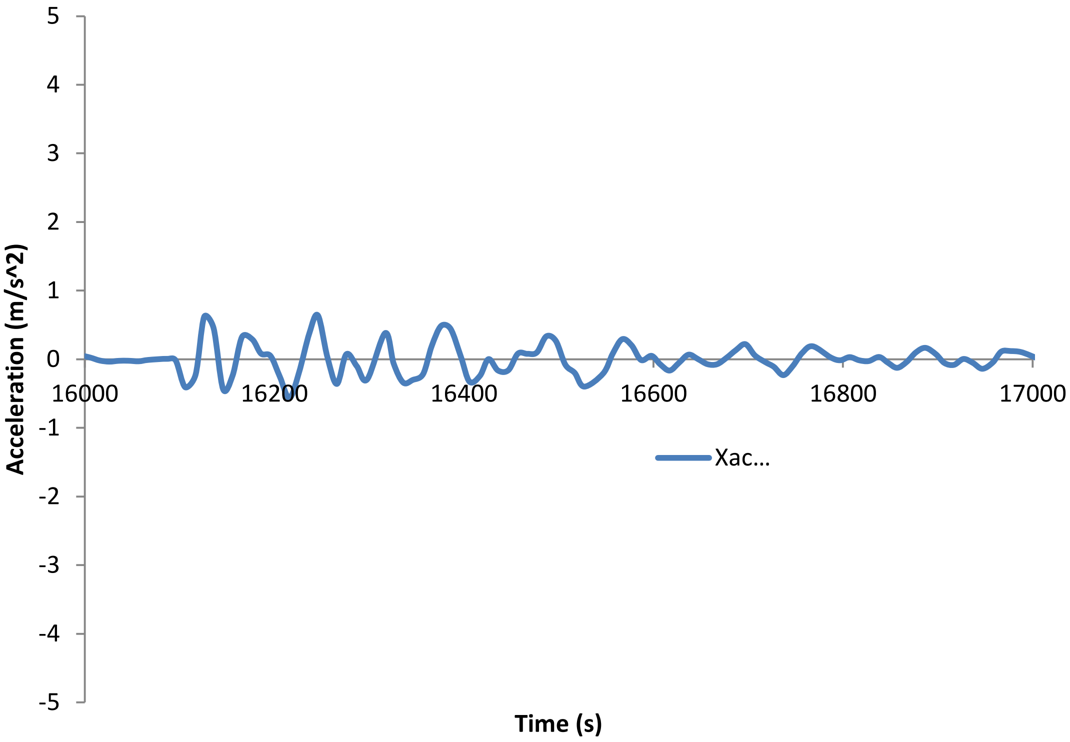


Figure 59: Vibration data for a single move taken at the top of the Vision Arch

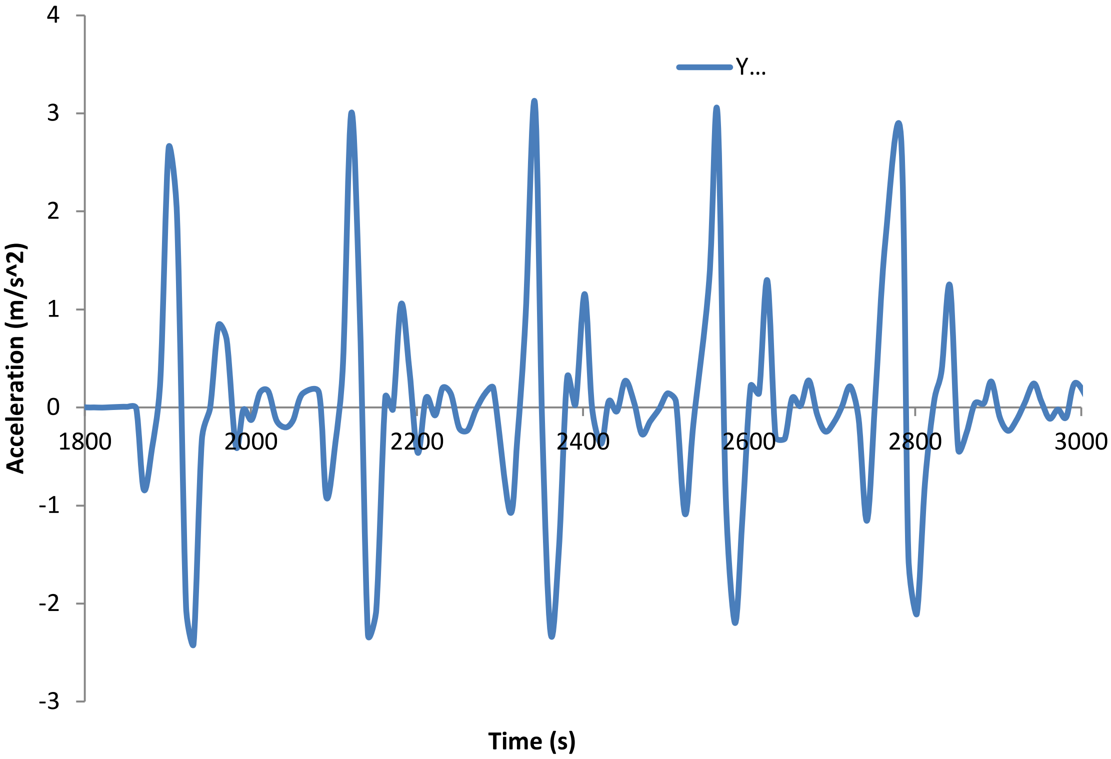


Figure 60: Vibration data for a series of moves taken on the Plexiglas

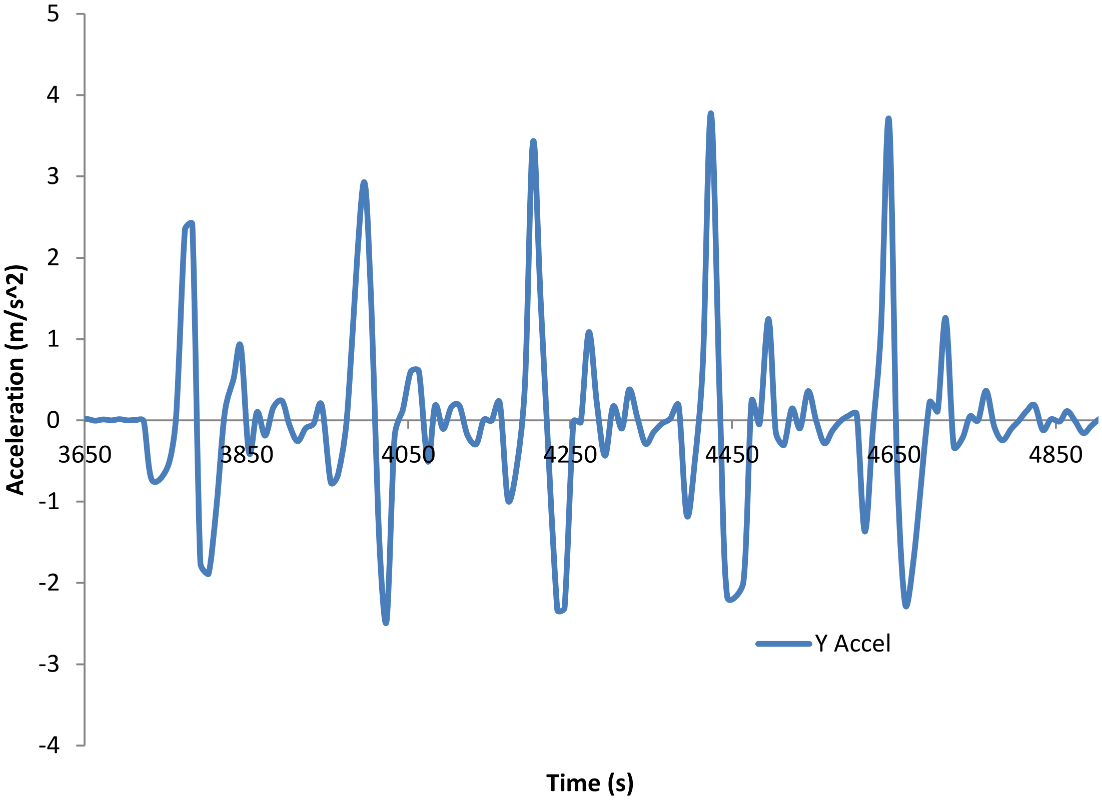


Figure 61: Vibration data for a series of moves taken at the Horizontal 8020 strut

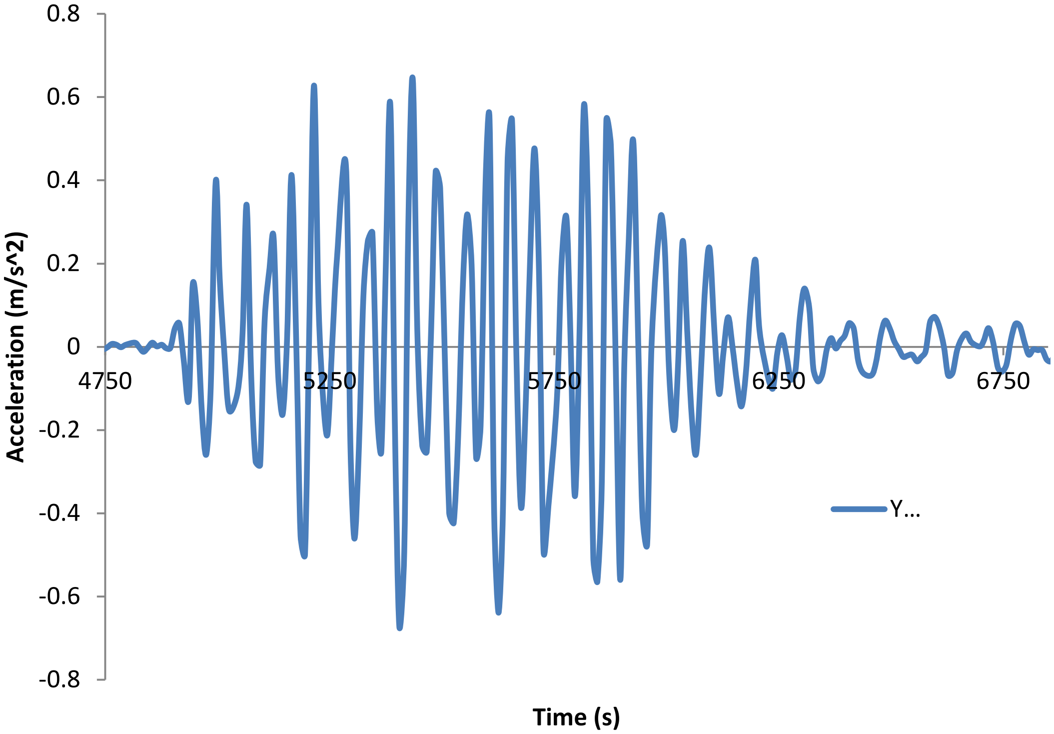


Figure 62: Vibration data for a series of moves taken on the Table Top

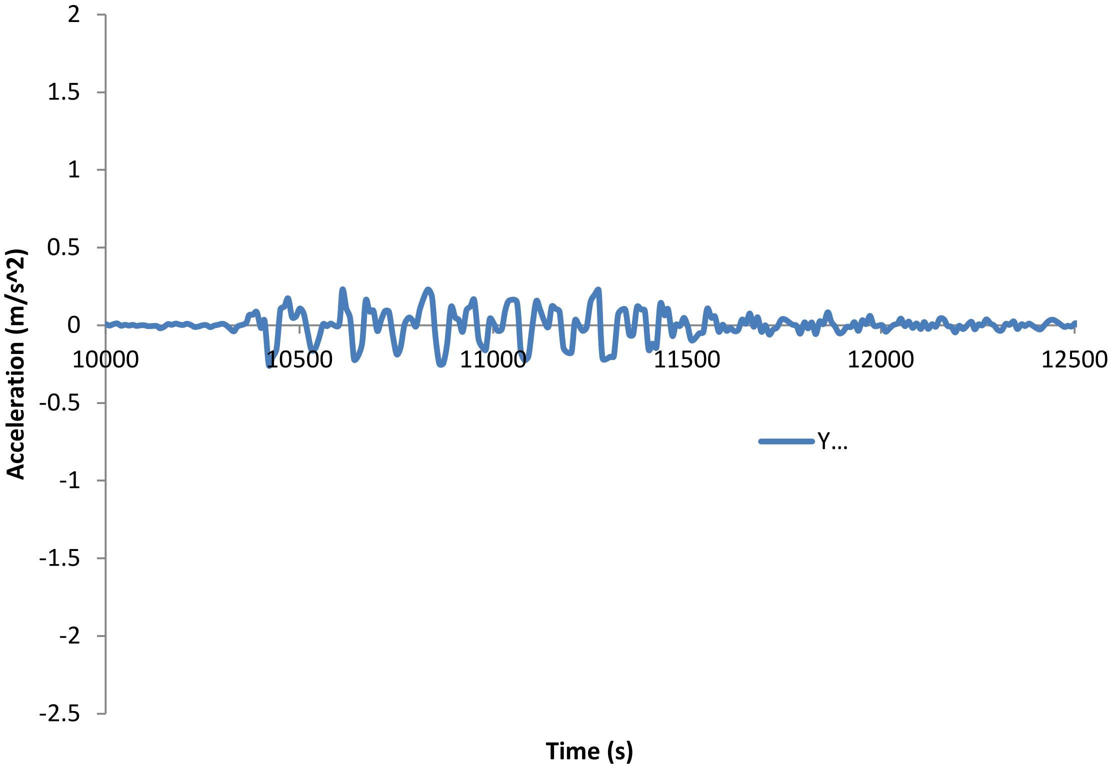


Figure 63: Vibration data for a series of moves taken at the top of the vision arch

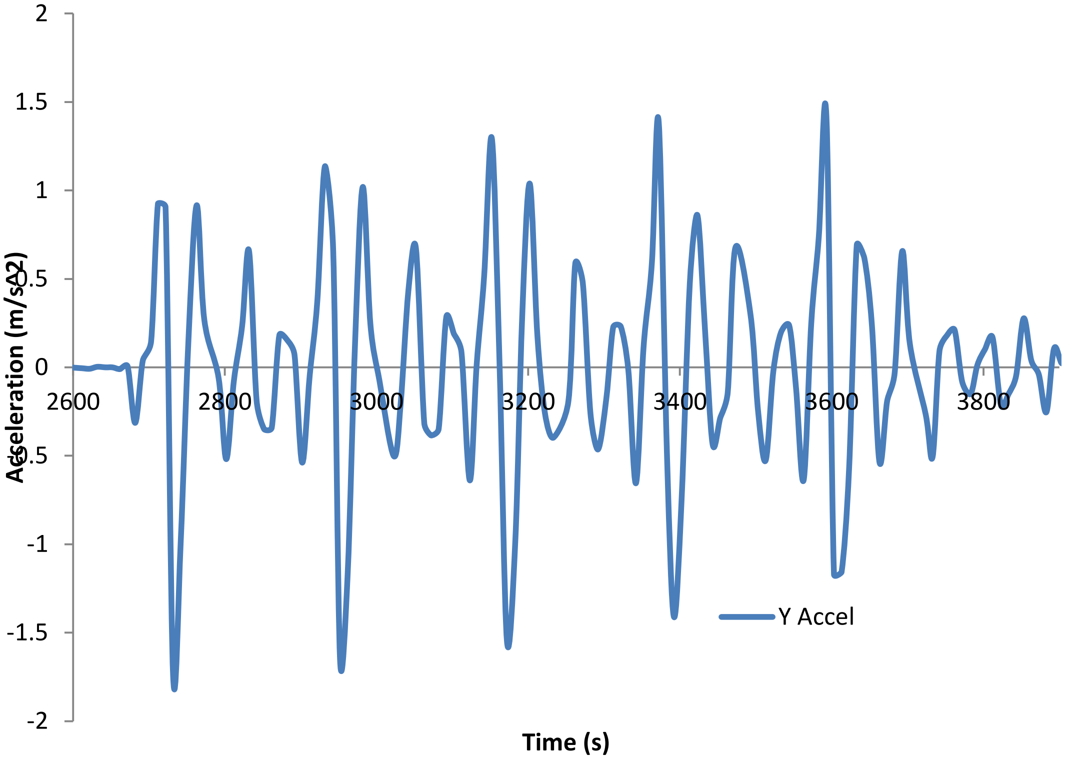


Figure 64: Vibration data for a series of moves, with the table and cabinet attached taken on the Plexiglas

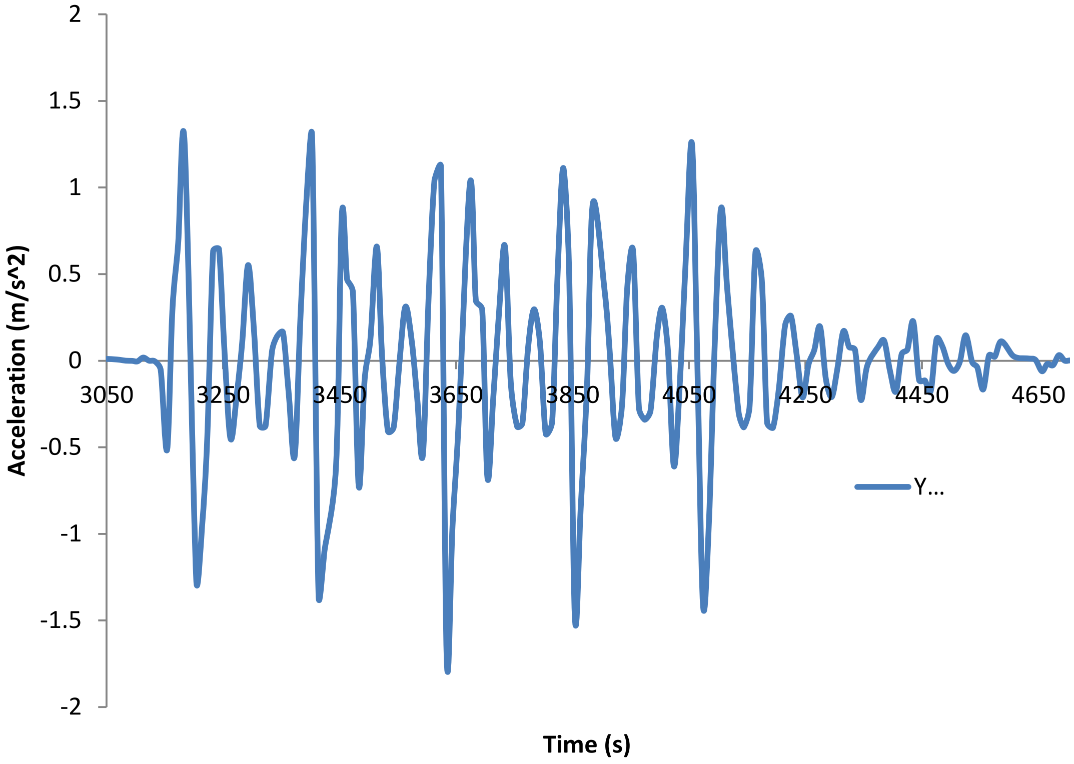


Figure 65: Vibration data for a series of moves, with the table and cabinet attached taken on the 8020 strut

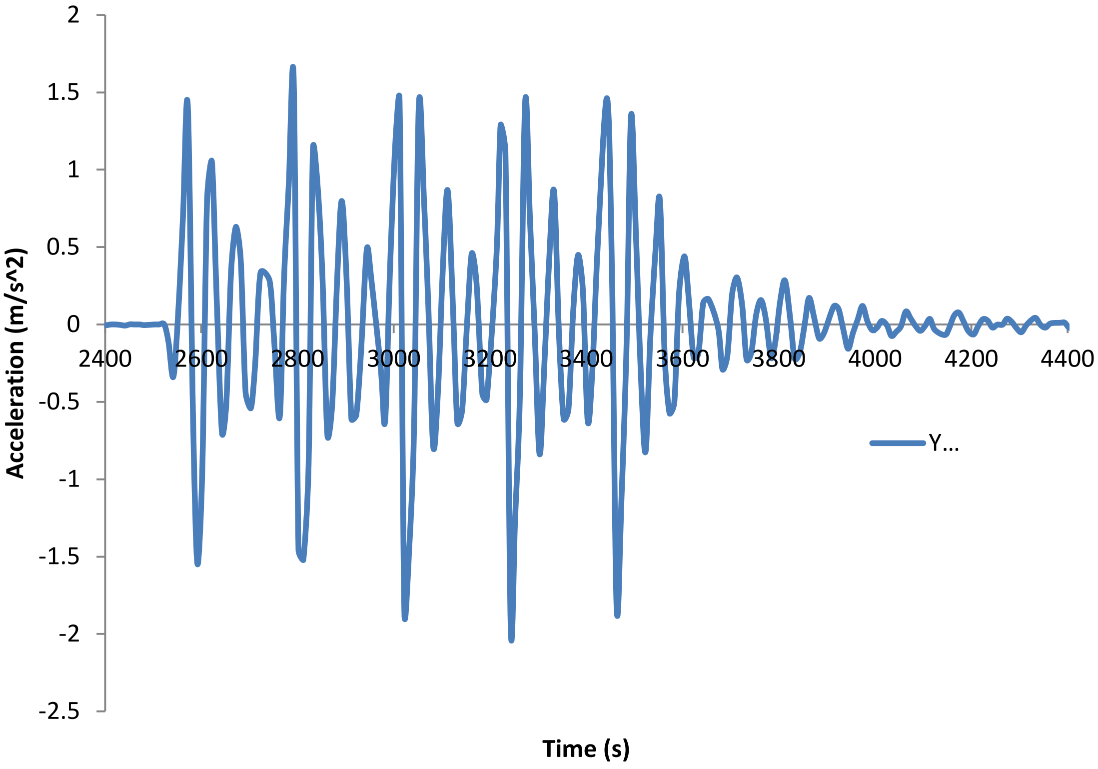


Figure 66: Vibration data for a series of moves, with the table and cabinet attached taken on the table top

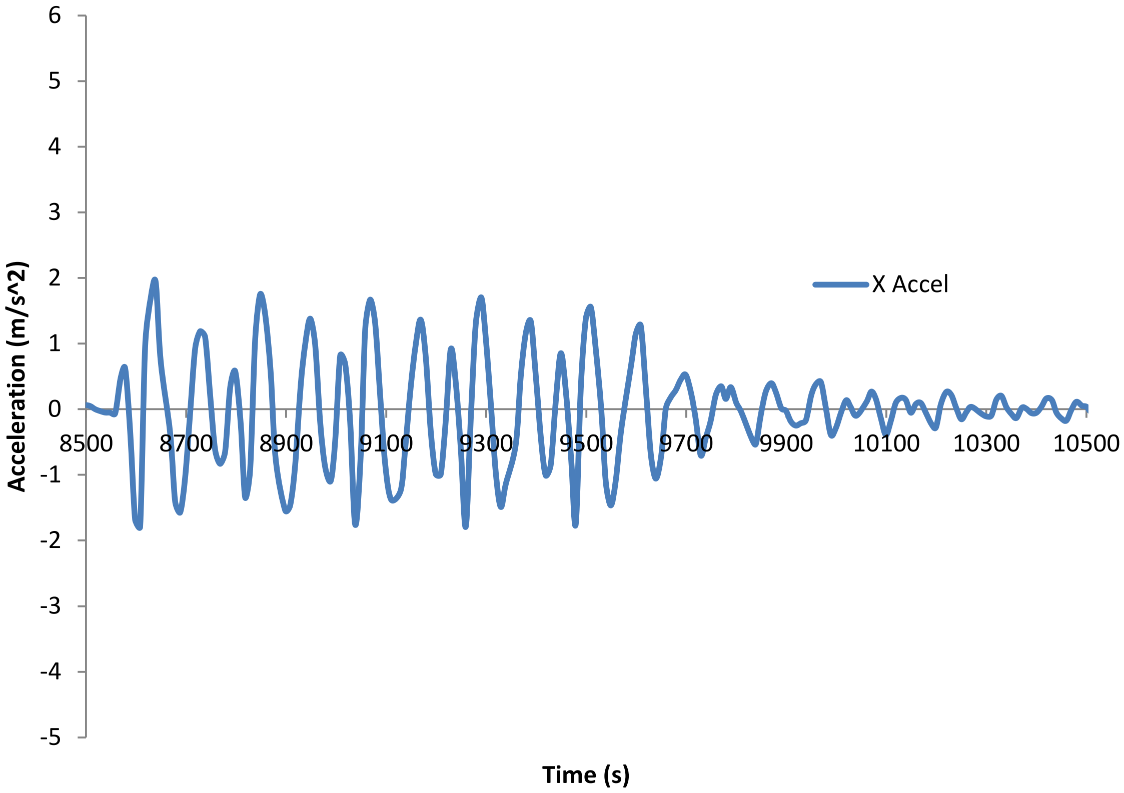


Figure 67: Vibration data for a series of moves, with the table and cabinet attached taken on the top of the vision Arch

Table 26: Results of the vibrational testing. The maximum amplitude and average frequency of the vibrations at each location are given.

	Long Move Unattached		Short Move Unattached		Short Move Attached	
Location	Max Amp. (m/s)	Frequency (Hz)	Max Amp. (m/s)	Frequency (Hz)	Max Amp. (m/s)	Frequency (Hz)
Plexiglas on Cabinet	-2.434	34.39	3.114	25.58	-1.746	19.04
8020 on Cabinet	2.470	33.64	3.775	25.62	-1.719	18.34
Top of Table	-0.539	29.26	0.643	18.31	-1.958	19.29
Vision Arch	0.640	-22.03	0.230	16.79	1.943	14.76

Inertia Ration Testing

This section describes the process used to determine the inertia ratios for the rotary and linear drive motors. The Moment of Inertia Identifier function in Sigmawin+ was used in both cases to determine the inertia ratio. Table 26 contains the parameters used to run the tests and Table 27 contains the results of each of the tests.

Table 27: The parameters used in the Inertia Identification Function.

Parameter	Value
Acceleration (min-1/s)	20000
Speed (Min-1)	1000
Moving Distance (Rotation)	2.5
Pn100:SPEED LOOP GAIN (0.1Hz)	400

Table 28: Results of the Inertia Identification test

Run	Linear Motor – Axis 3	Rotary Motor – Axis 4
1	645%	185%
2	625%	178%
3	632%	179%
4	631%	172%
5	633%	172%
6	635%	176%
Average	633%	177%

Appendix G: Motor Tuning

Linear Drive Motor

This section contains the details of the tuning of the linear drive motor of the goalie rod. The custom tuning function in Sigmawin+ was used to perform the test. Table 29 contains the JOG settings used in the test and Table 28 contains the procedure used to tune the motor. Figures 58-62 contain graphs of the data collected during the tuning of the motor and Table 30 contains the results of the Tuning.

Table 29: The tuning procedure used to tune the linear drive motor of the goalie rod

Run Number	Parameter State	Comments
1	Tuneless Mode	Long Settling Time
2	• Feed Forward Gain (FF) = 50 • Feed Back Gain (FB) =50	None
3	• FF = 37 • FB = 50 • Vibration Damping (VD) = 20 Hz	Vibration Damping Engaged at Recommendation of the System
4	• FF = 45 • FB = 50 • VD = 20 Hz	None
5	• FF = 65 • FB = 50 • VD = 20 Hz	None
6	• FF = 85 • FB = 50 • VD = 20 Hz	None
7	• FF = 85 • FB = 70 • VD = 20 Hz	None
8	• FF = 85 • FB = 100 • VD = 20 Hz	None
9	• FF = 110 • FB = 150 • VD = 0 Hz	Vibration Damping was Disengaged to Improve Performance. Vibration Sensor Did Not Trip After This
10	• FF =135 • FB = 175	Response Improving greatly
11	• FF =155 • FB = 200	Final Tuning. Further Modifications Did Not Improve Response

Table 30: The parameters used in the JOG move used to tune the linear drive motor

Parameter	Value
Move Distance (mm)	400
Motor Speed (min-1)	6000
Actuator Speed (m/s)	1.87
Acceleration Time (ms)	150
Number of Cycles	1

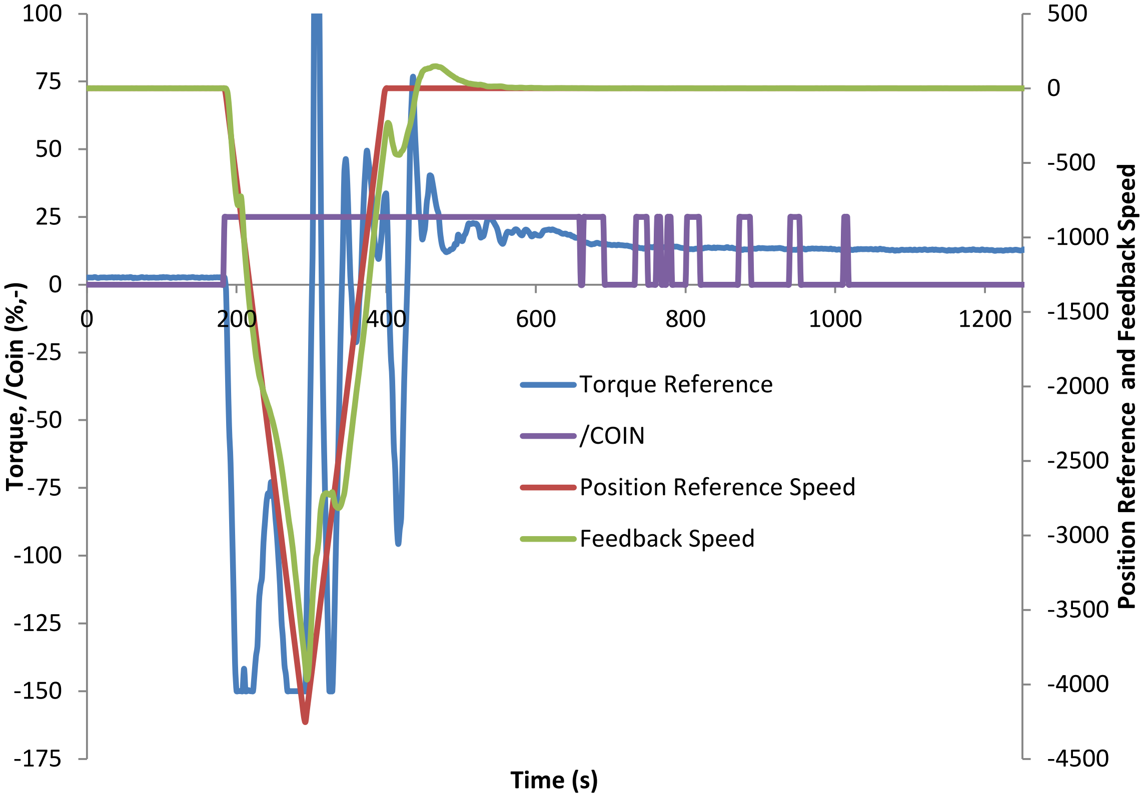


Figure 68: Plot of the motors performance in tuneless mode.

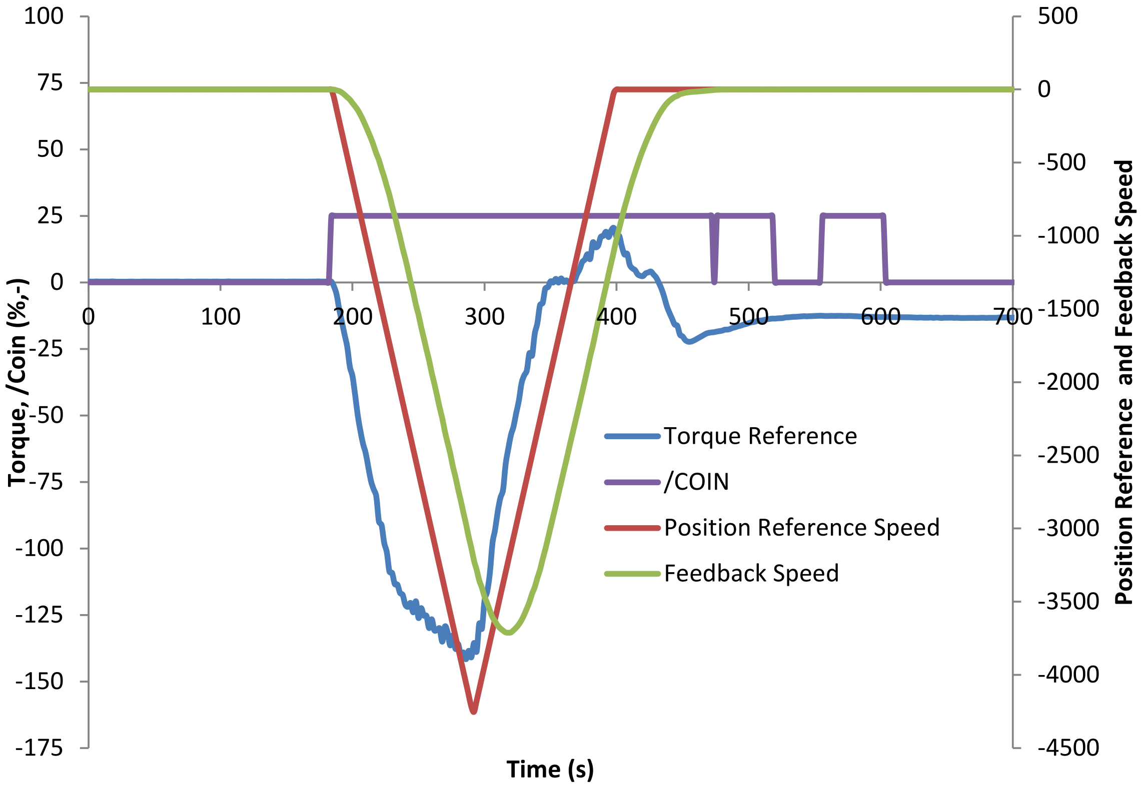


Figure 69: Plot of the motors performance with FF=35 FB=50 VD=20Hz.

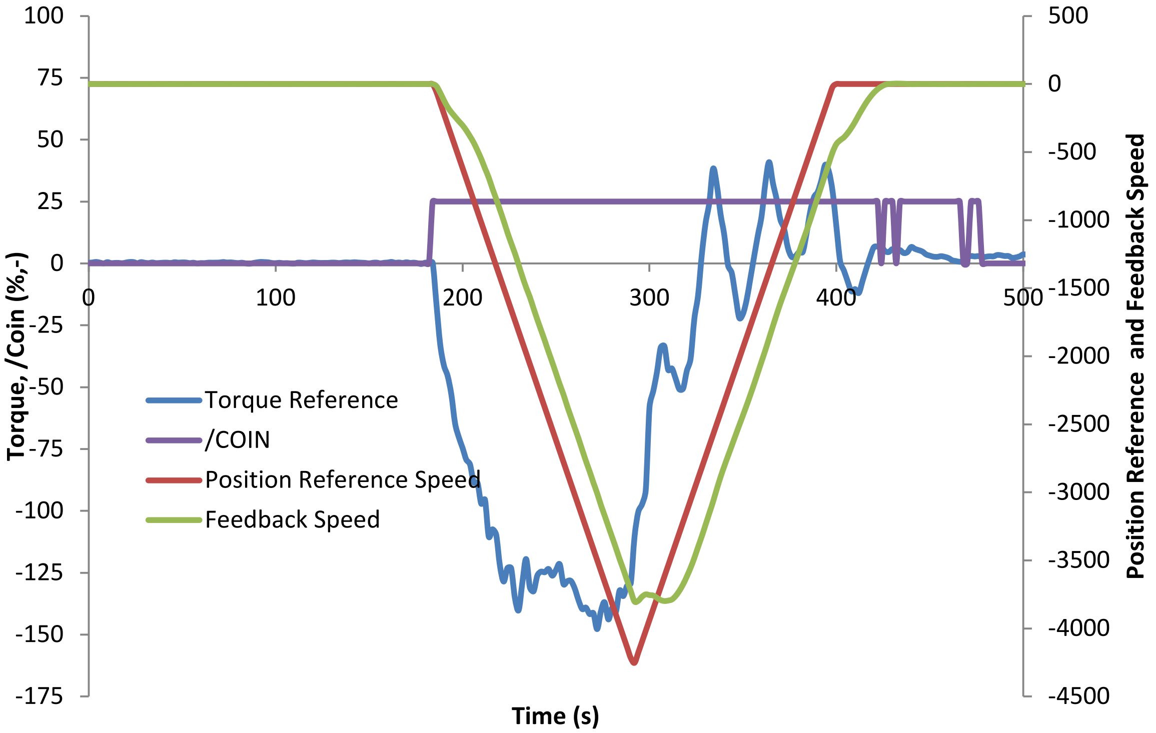


Figure 70: Plot of the motors performance with FF=85 FB=100 VD=20Hz.

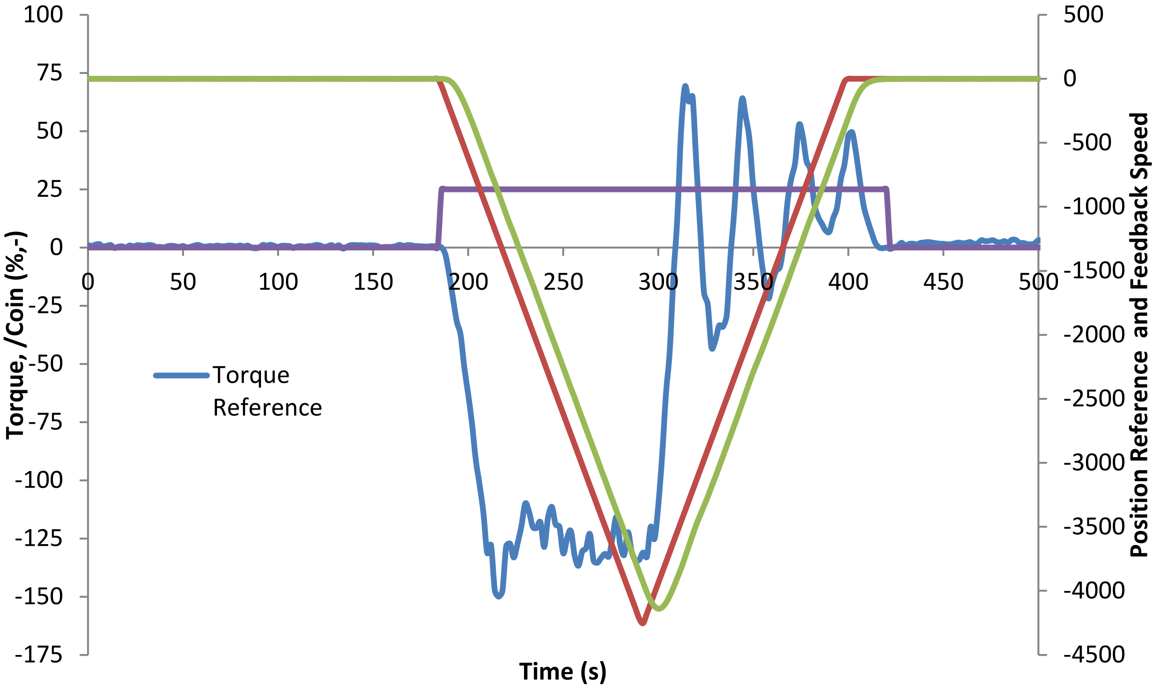


Figure 71: Plot of the motors performance with FF=110 FB=150 VD=0Hz.

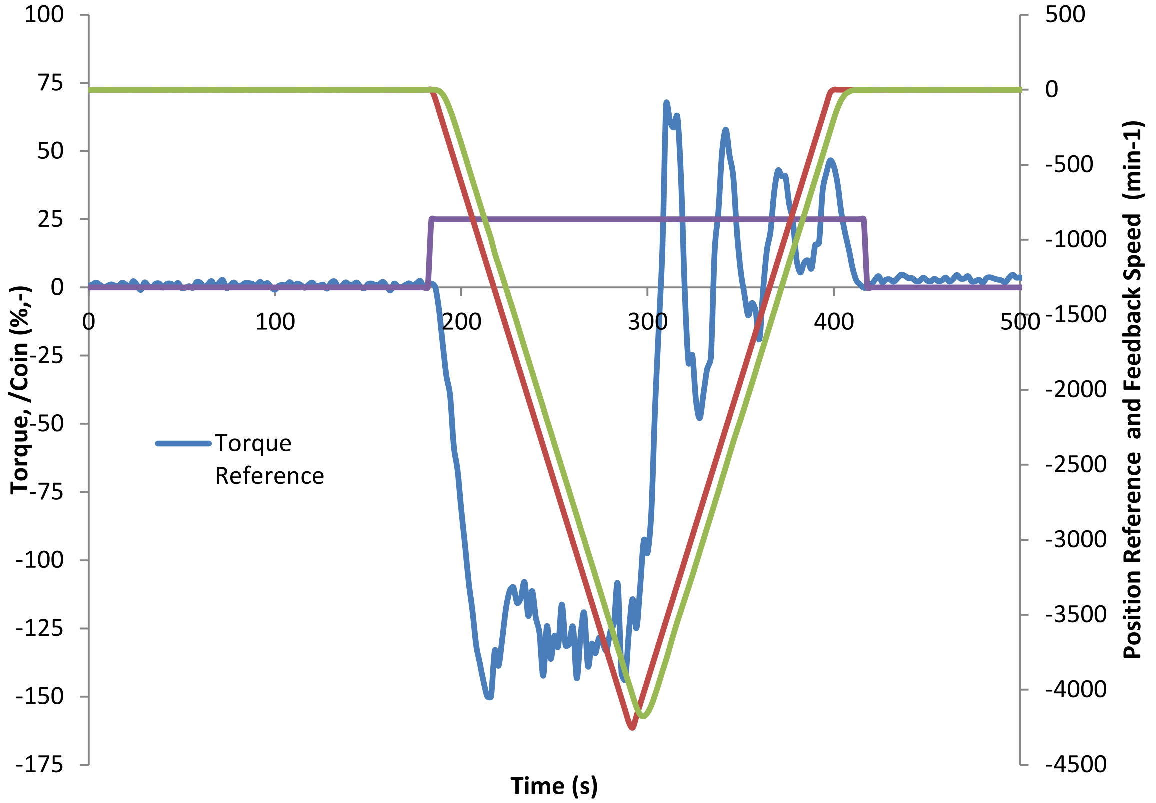


Figure 72: Plot of the motors performance with FF=155 FB=200 VD=0Hz.

Table 31: Results of the linear drive motor tuning

Tuneless Settling Time	Tuned Settling Time	Percent Improvement
612.72	17.34	97.1%

Rotary Motor

This section contains the details of the tuning of the rotary motor for the goalie rod. The tuning was completed using the autotune function in Sigmawin+. Table 31 contains the parameters used in the JOG function which the test moves were made using. Figures 63 and 64 contain the traces of the test moves performed to assess the performance of the motor after tuning. Table 32 contains the results of the tuning.

Table 32: Parameters used in the JOG test move

Parameter	Value
Move Distance (mm)	400
Motor Speed (min-1)	450
Actuator Speed (min-1)	112
Acceleration Time (ms)	150
Number of Cycles	3

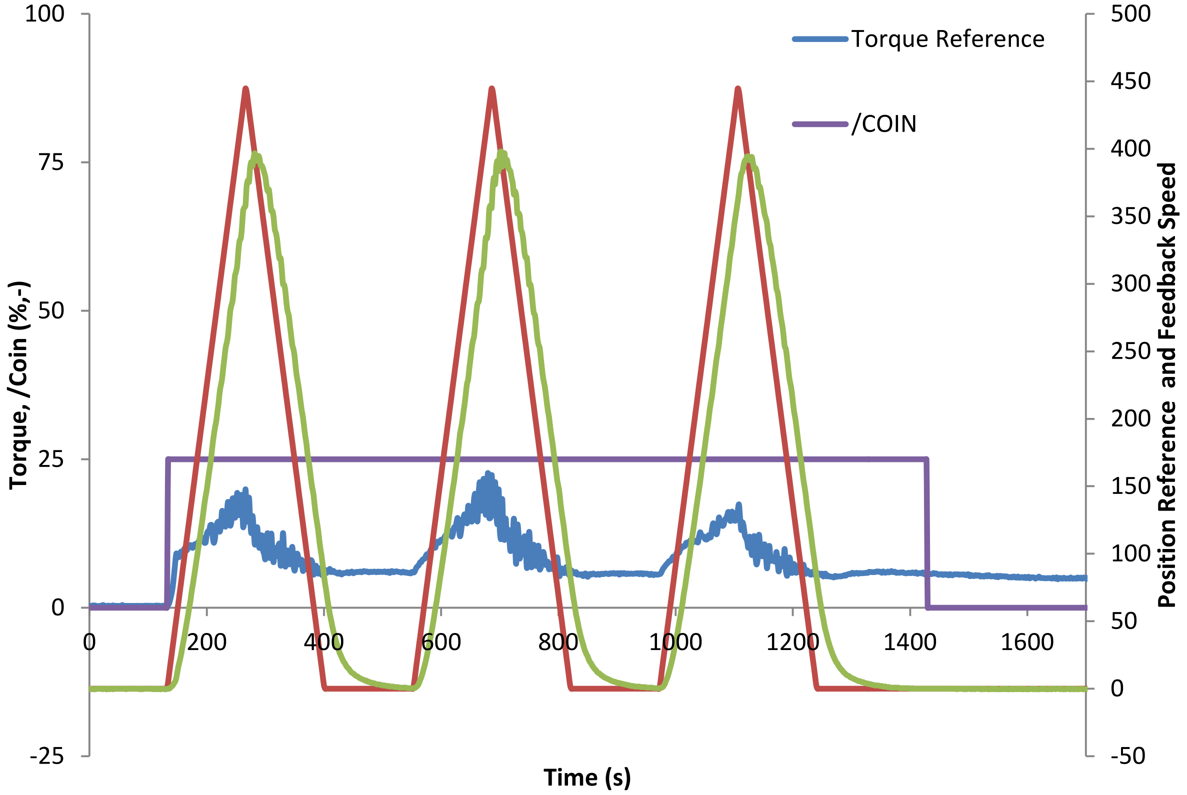


Figure 73: Plot of motor performance in tuneless mode

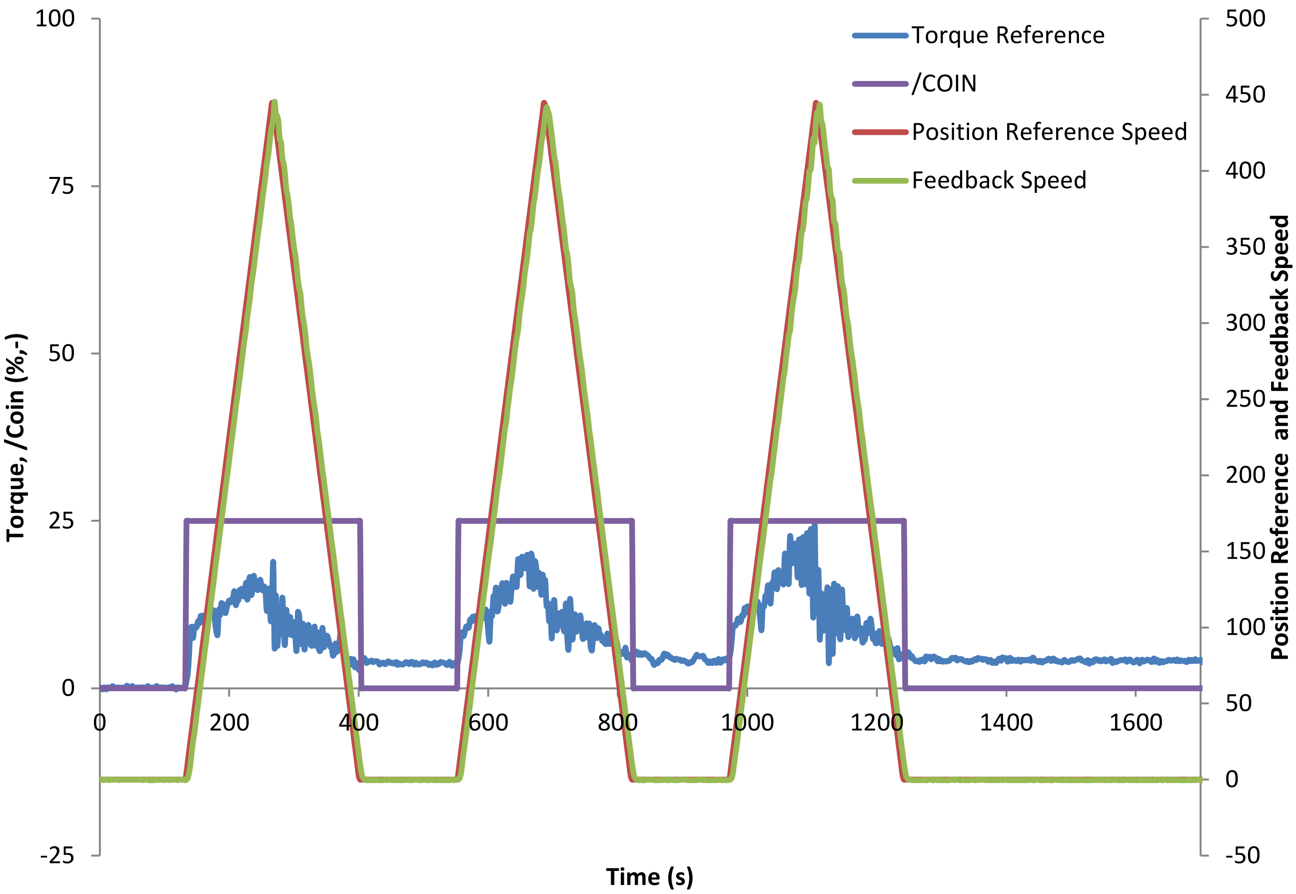


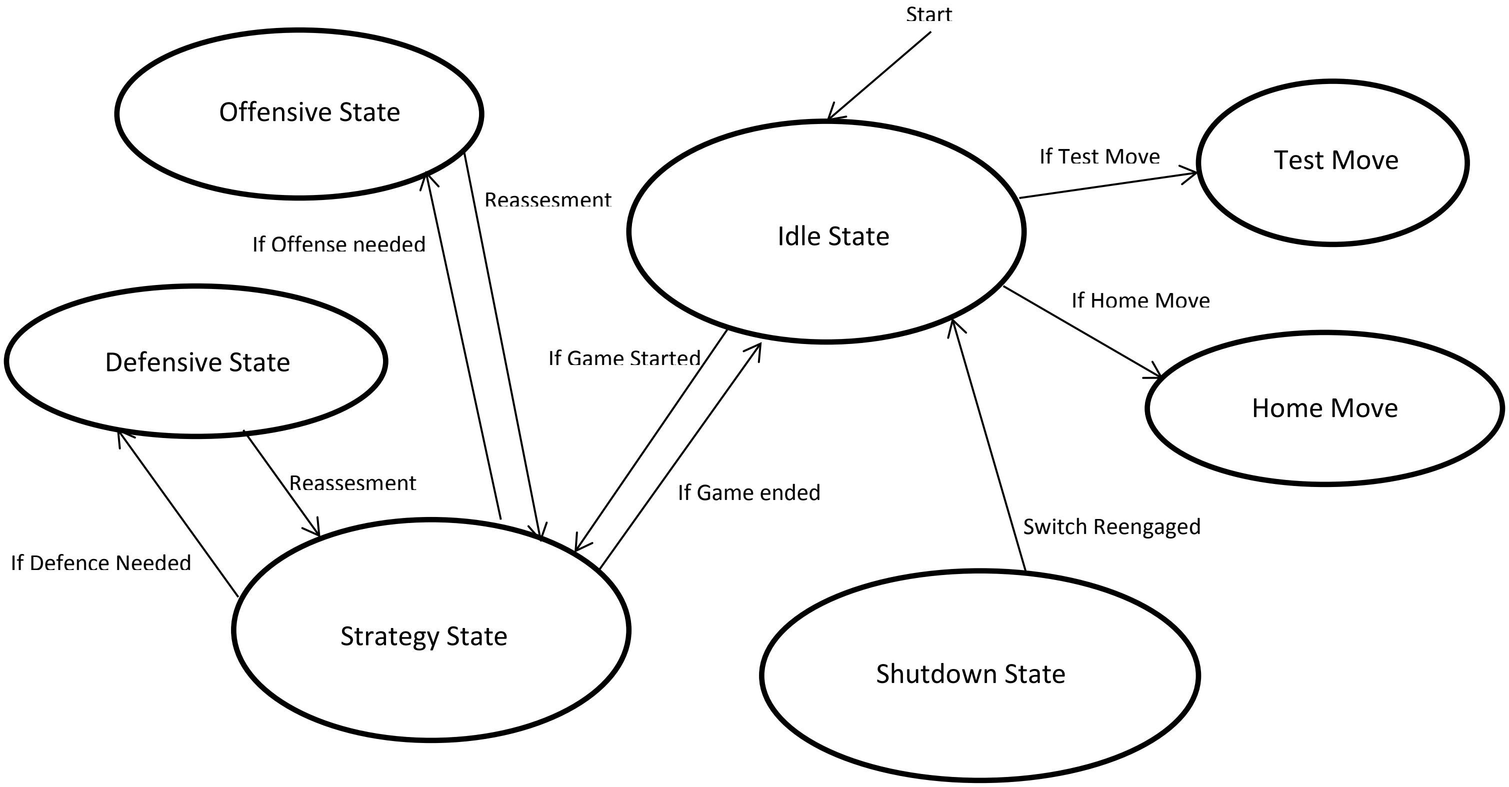
Figure 74: Plot of motor performance after autotuning

Table 33: Results of the rotary drive motor tuning

Tuneless Settling Time	Tuned Settling Time	Percent Improvement
189.6	4.62	97.6%

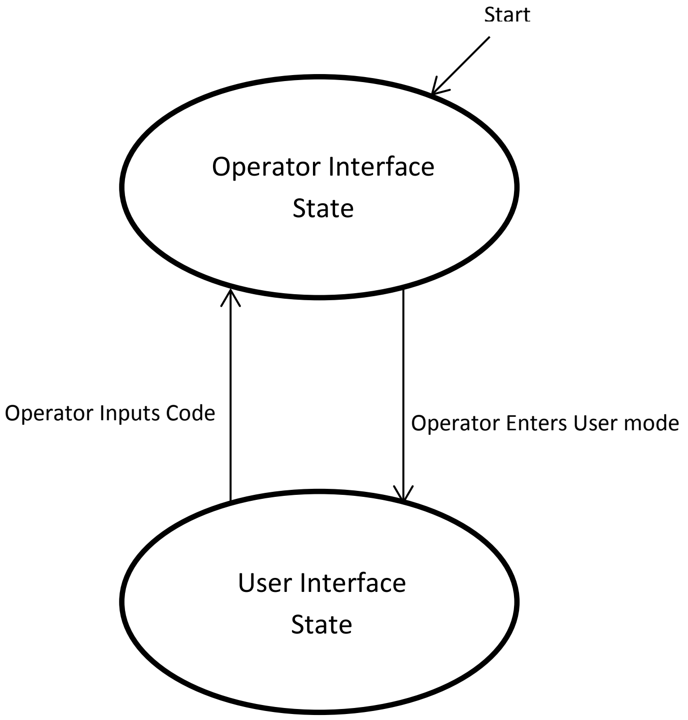
Appendix H: Programing State Diagrams

ROD CONTROL TASK

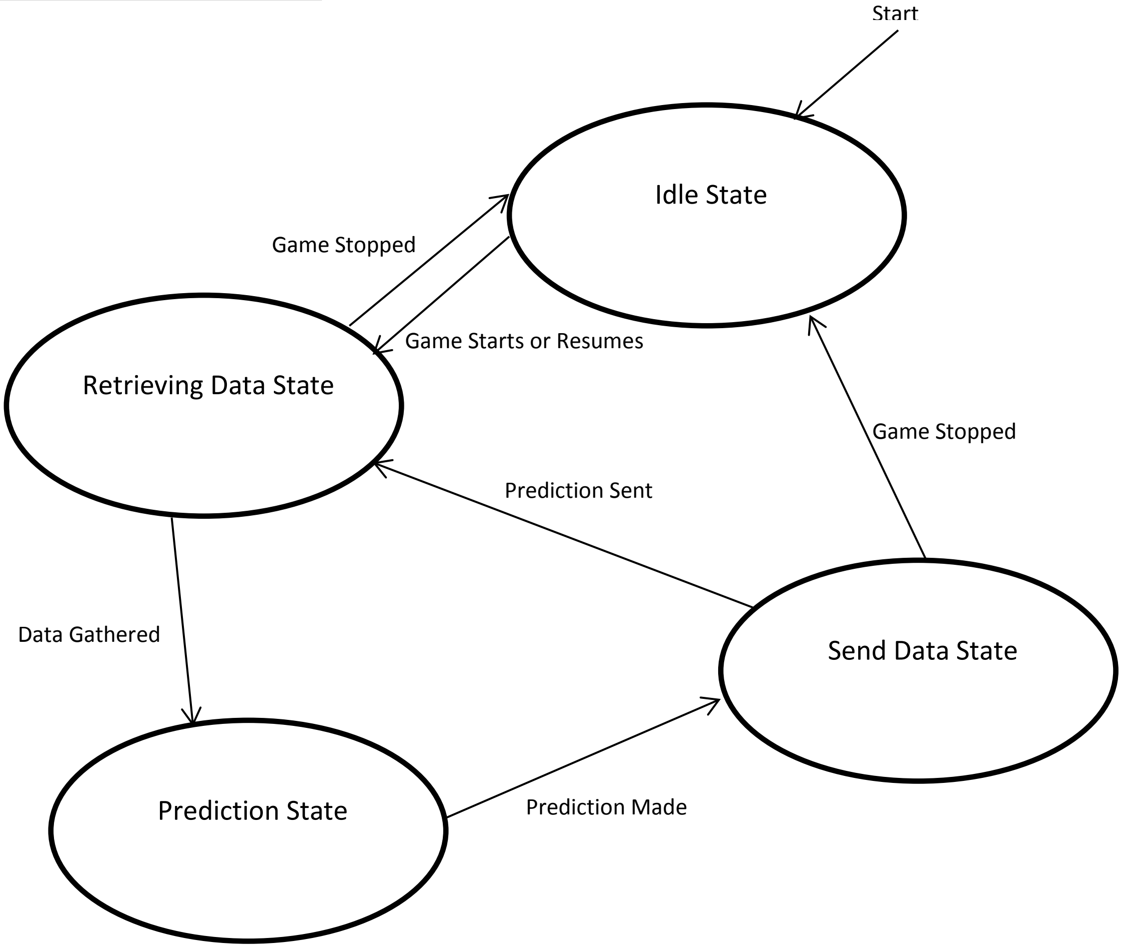


NOTE: All states can transition to Shutdown if a switch is tripped

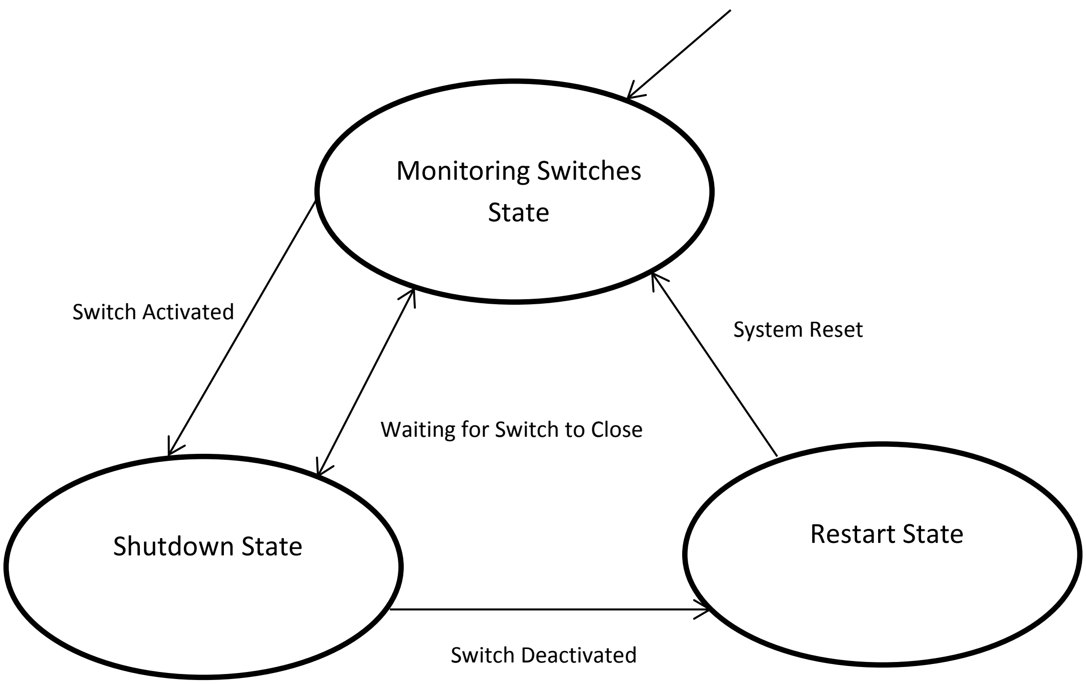
HMI TASK



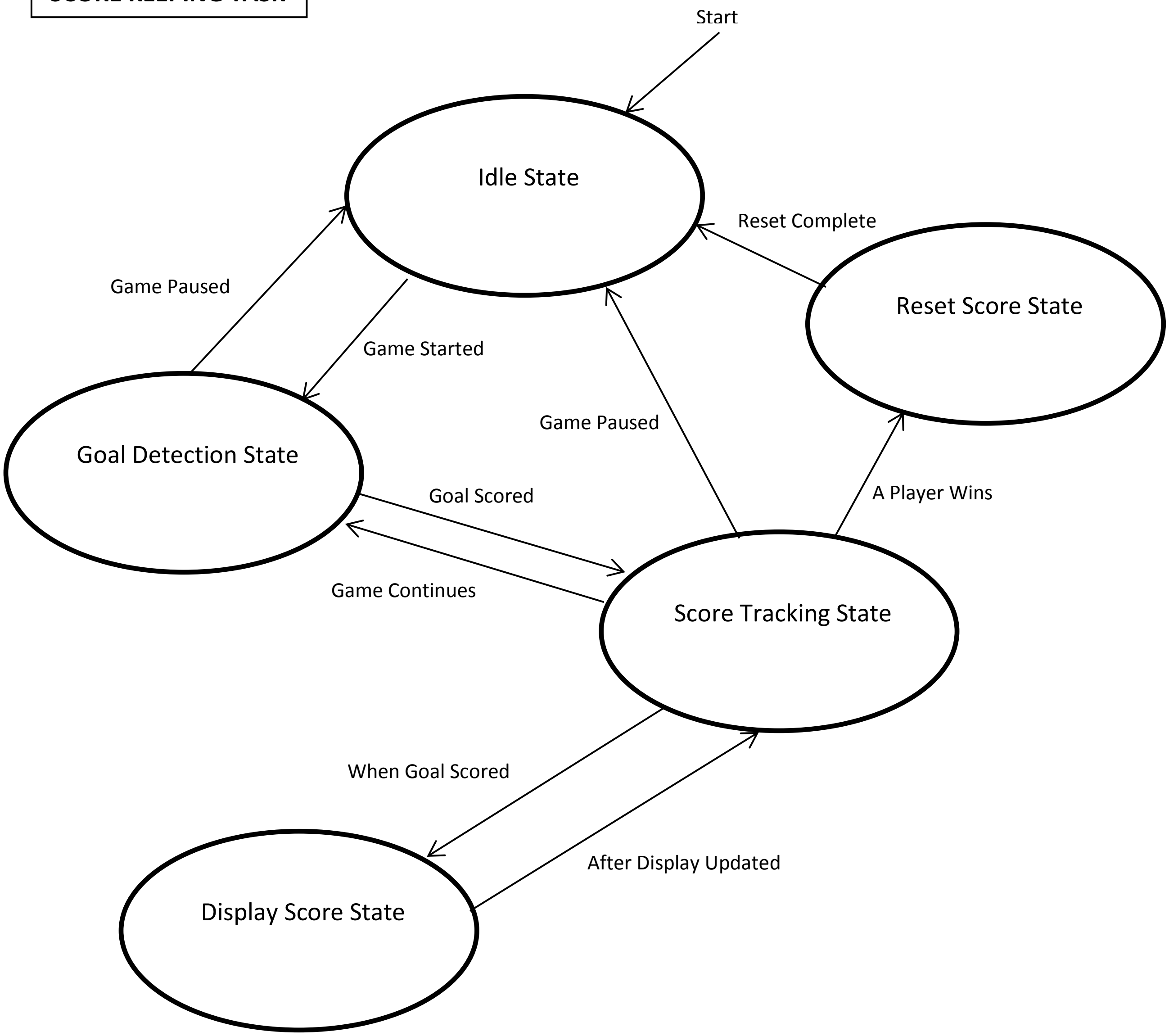
BALL TRACKING/PREDICTION TASK



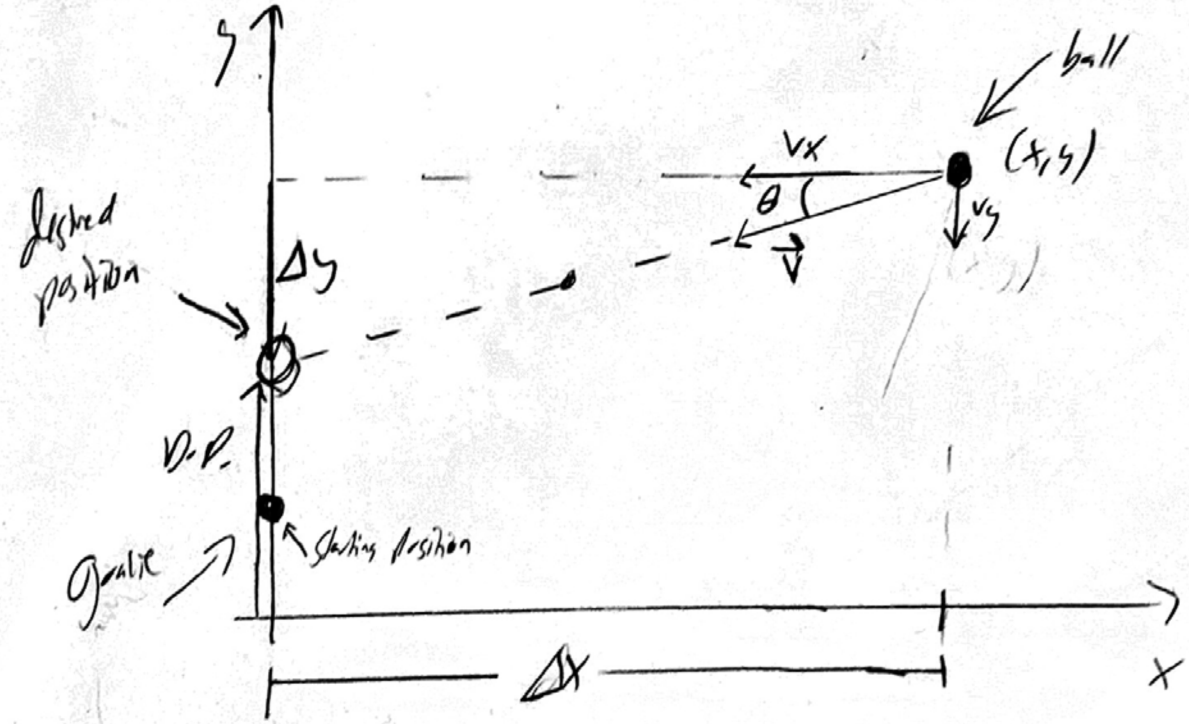
SAFETY SYSTEM TASK



SCORE KEEPING TASK



Ball Tracking Geometry 1-27-14



D.P. = $y + \Delta y$

Similar Triangles:

$$\frac{\Delta y}{\Delta x} = \frac{V_y}{V_x}$$

$$\Delta y = \frac{V_y}{V_x} \Delta x$$

$$\boxed{D.P. = y + \frac{V_y}{V_x} \Delta x}$$

* This will only work for non-banking shots

* Sideways passes must be detected as something else – maybe do this by setting a minimum V_x for a shot

Appendix I: Wiring Diagram

Wiring Schematic Overview

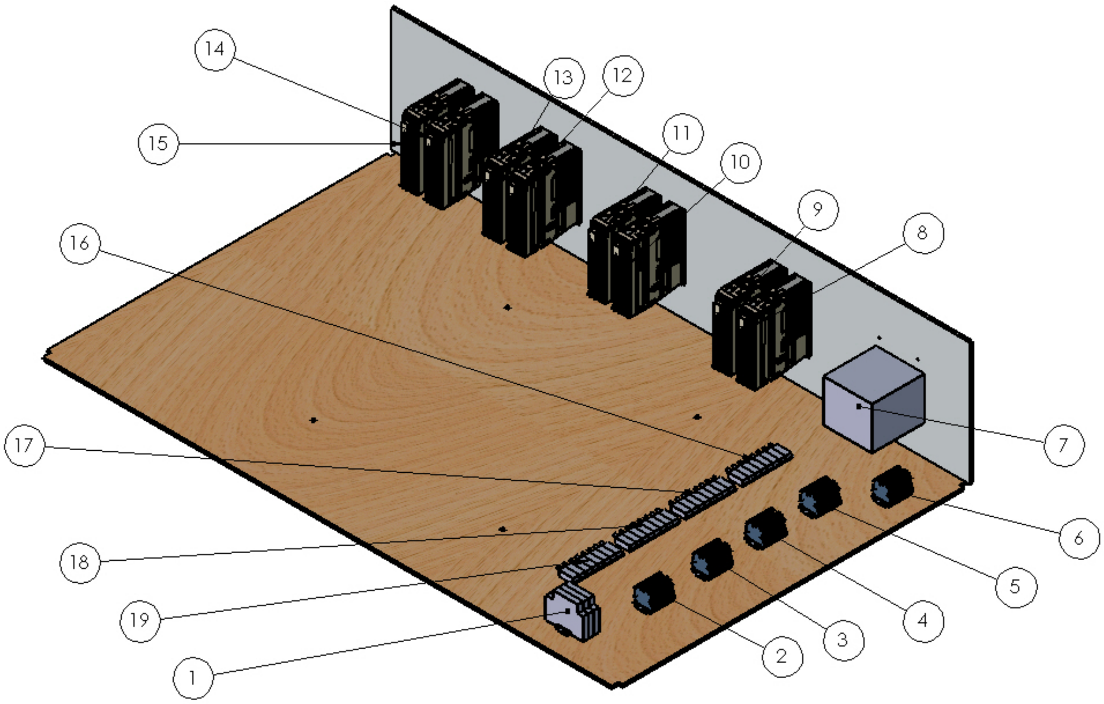


Figure 75: Module Framework

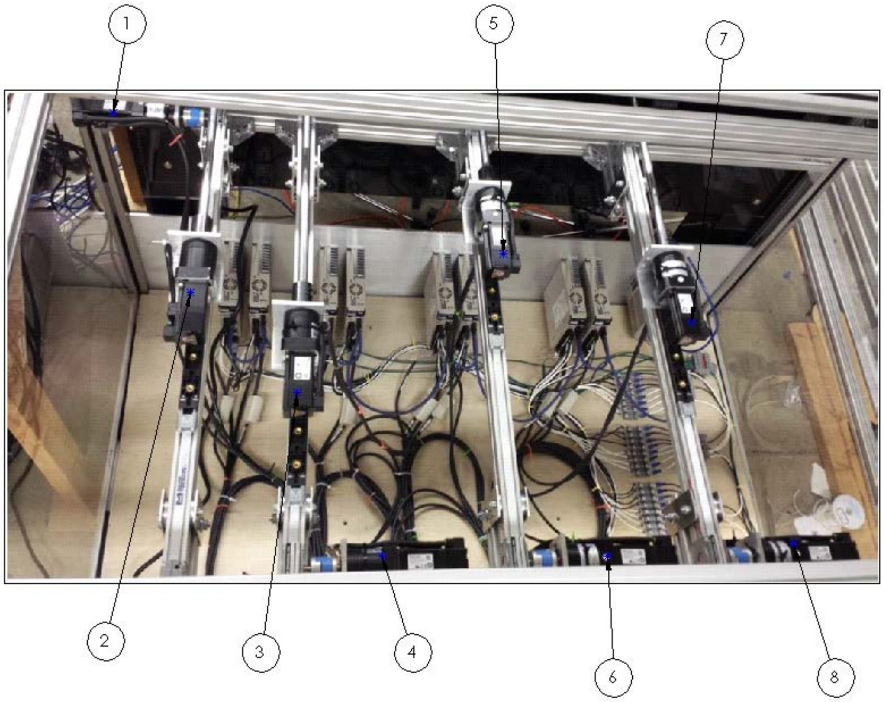


Figure 76: Motor Framework

Modules

Module 1: Circuit Breaker connected to power supply

Module 2-5: Circuit Breaker connected to fuses

Module 6: Circuit Breaker connected to amplifiers

Module 7: PLC

Module 8-15: Amplifier

Module 16-19 Fuses

Motor

Motor 1,4,6,8: Linear Drive Motors

Motor 2,3,5,7: Rotary Motors

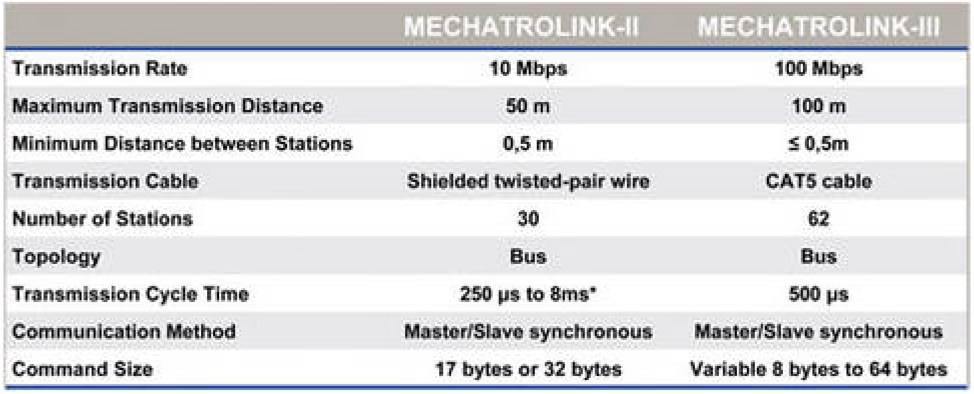
Wire Notation

W#/WS#: White Wire #

B#/BS#: Black Wire #

G#: Green Wire #

P#: Power Cable #



	MECHATROLINK-II	MECHATROLINK-III
Transmission Rate	10 Mbps	100 Mbps
Maximum Transmission Distance	50 m	100 m
Minimum Distance between Stations	0,5 m	≤ 0,5m
Transmission Cable	Shielded twisted-pair wire	CAT5 cable
Number of Stations	30	62
Topology	Bus	Bus
Transmission Cycle Time	250 µs to 8ms*	500 µs
Communication Method	Master/Slave synchronous	Master/Slave synchronous
Command Size	17 bytes or 32 bytes	Variable 8 bytes to 64 bytes

Figure 77: Metrolink Specifications

Wiring Schematic

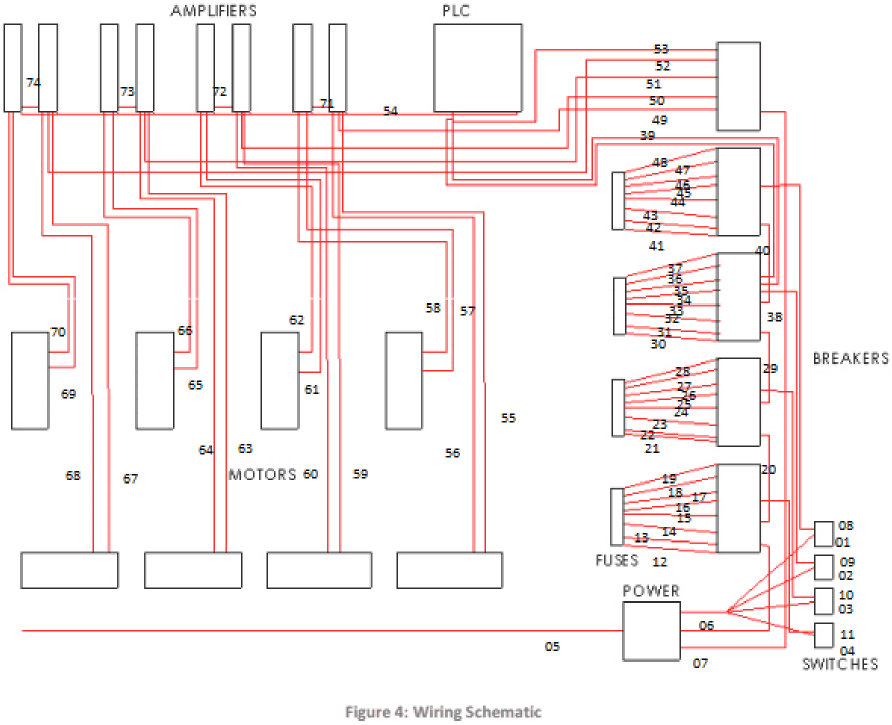


Figure 78: Wiring Schematic

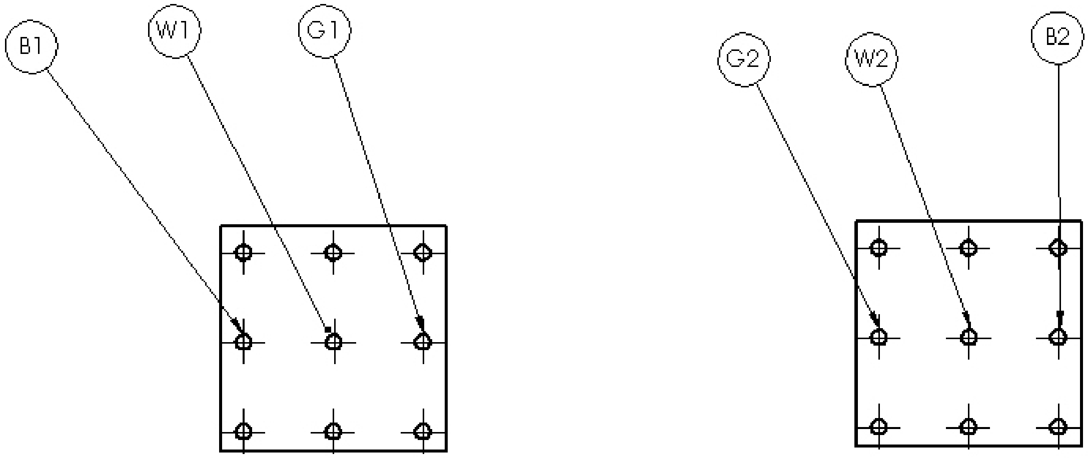


Figure 79: Module 1

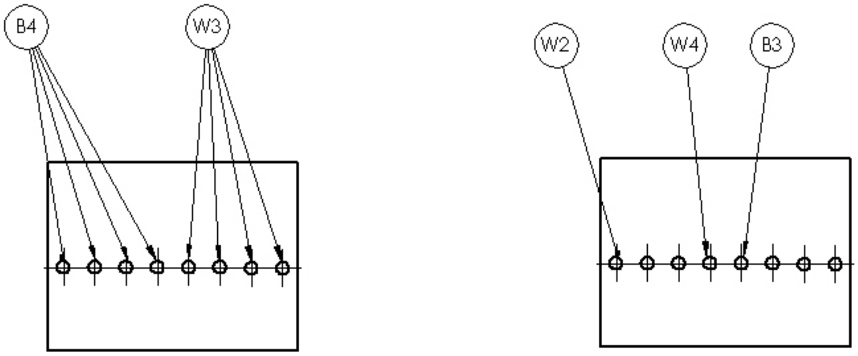


Figure 80: Module 2

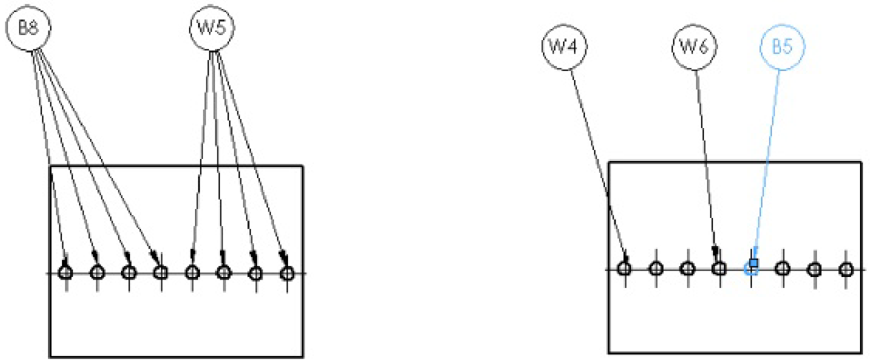


Figure 81: Module 3

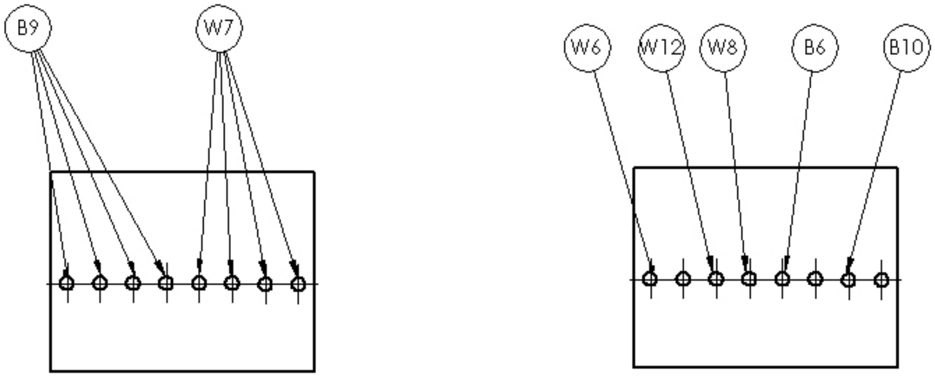


Figure 82: Module 4

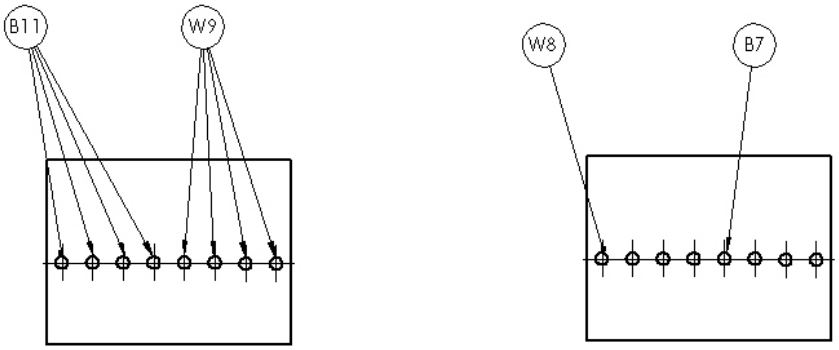


Figure 83: Module 5

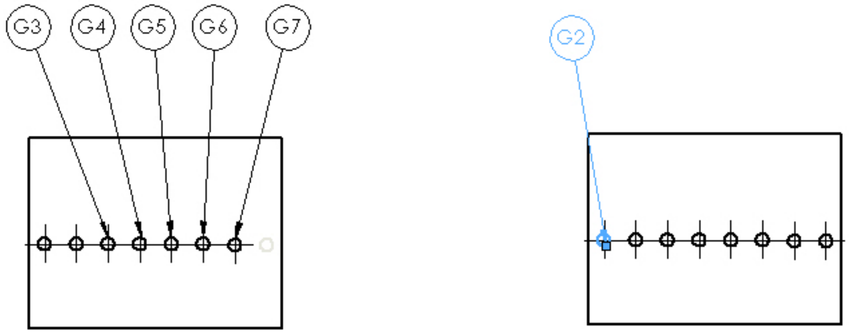


Figure 84: Module 6

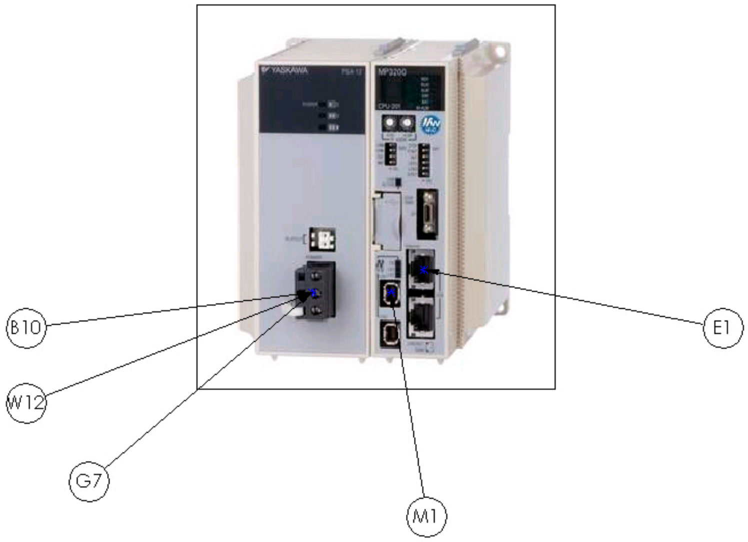


Figure 85: Module 7

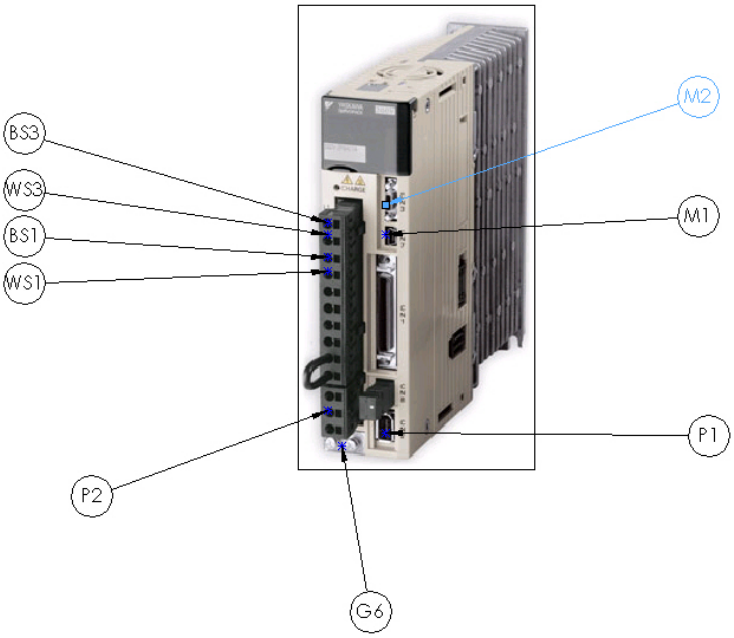


Figure 86: Module 8

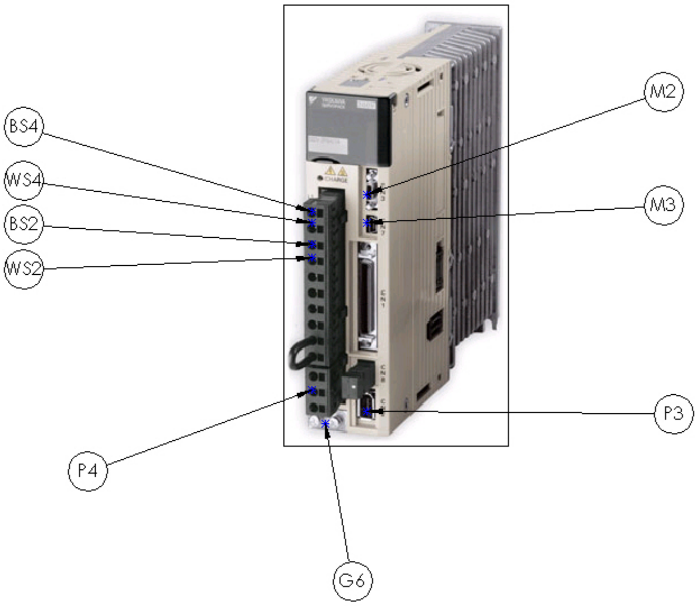


Figure 87: Module 9

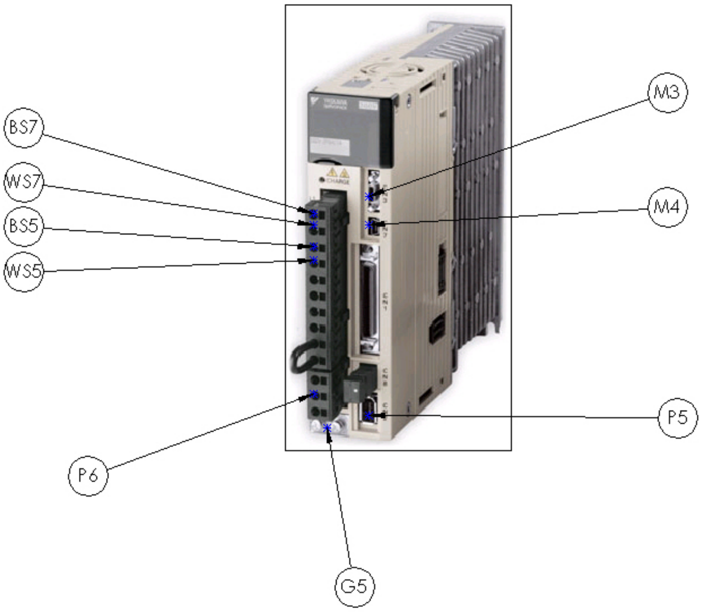


Figure 88: Module 10

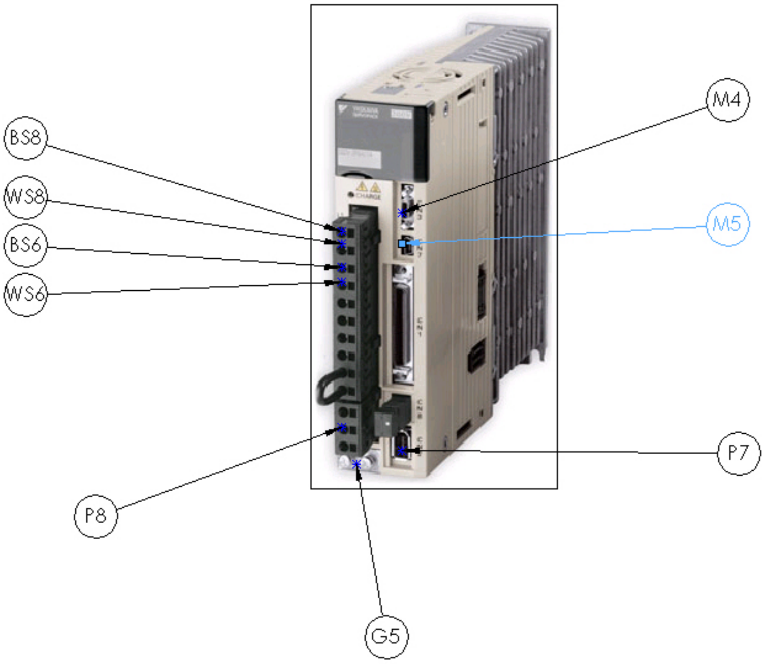


Figure 89: Module 11

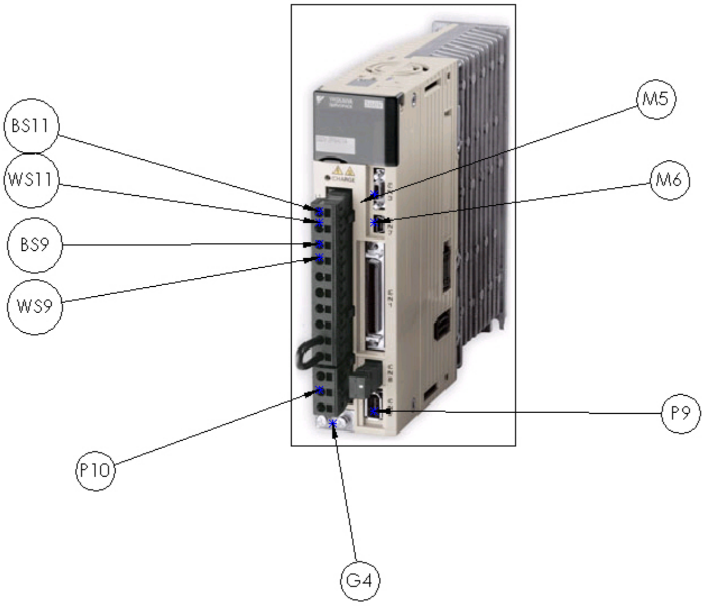


Figure 90: Module 12

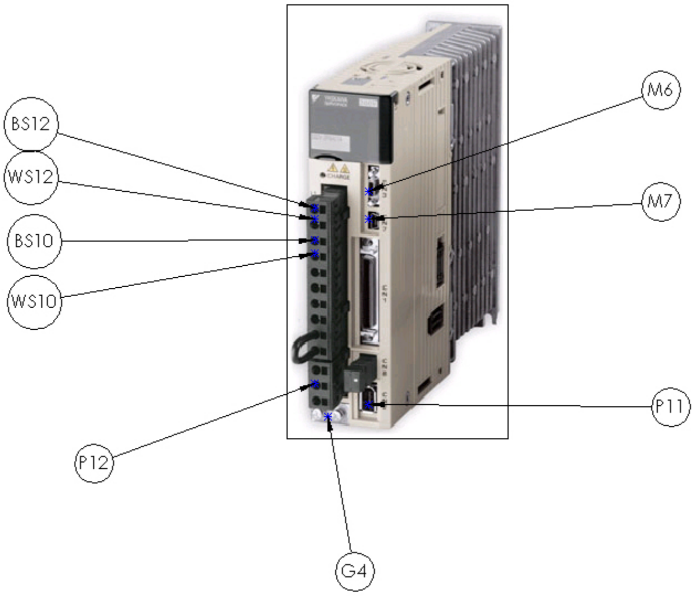


Figure 91: Module 13

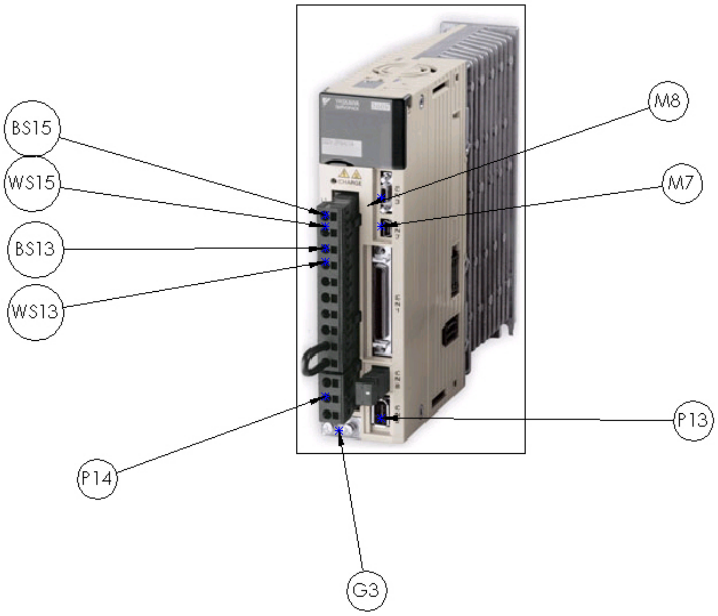


Figure 92: Module 14

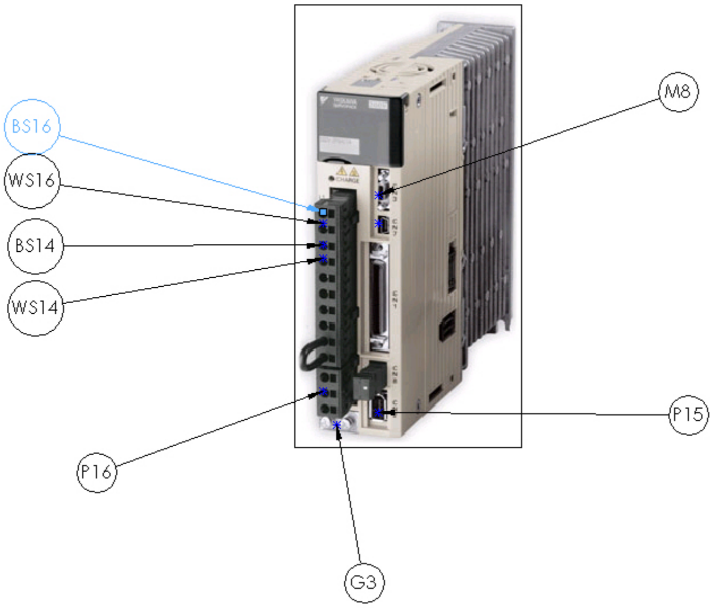


Figure 93: Module 15

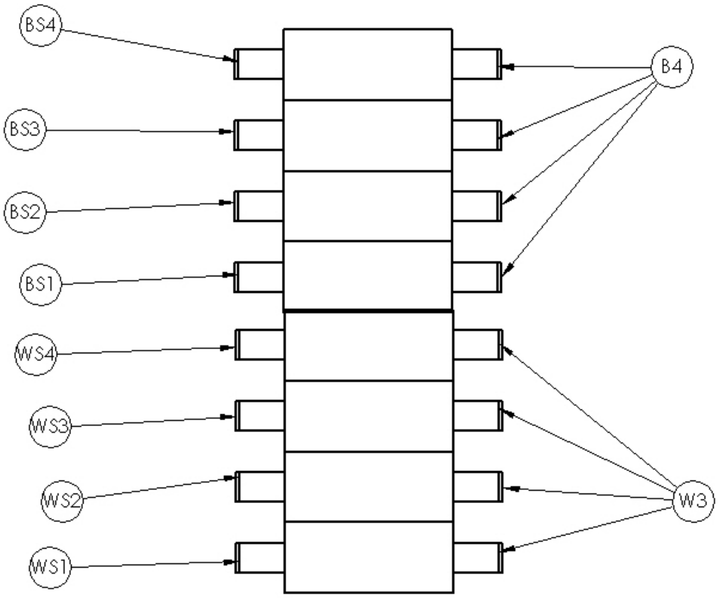


Figure 94: Module 16

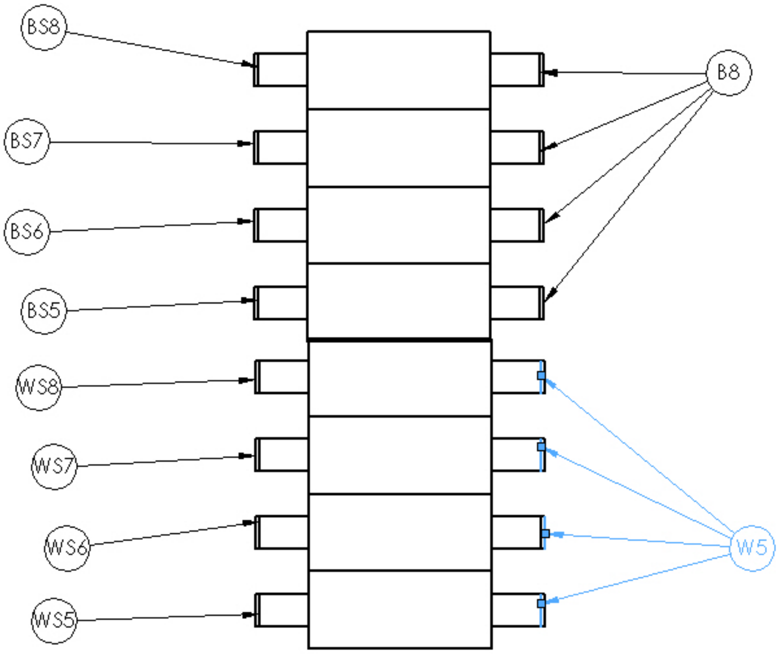


Figure 95: Module 17

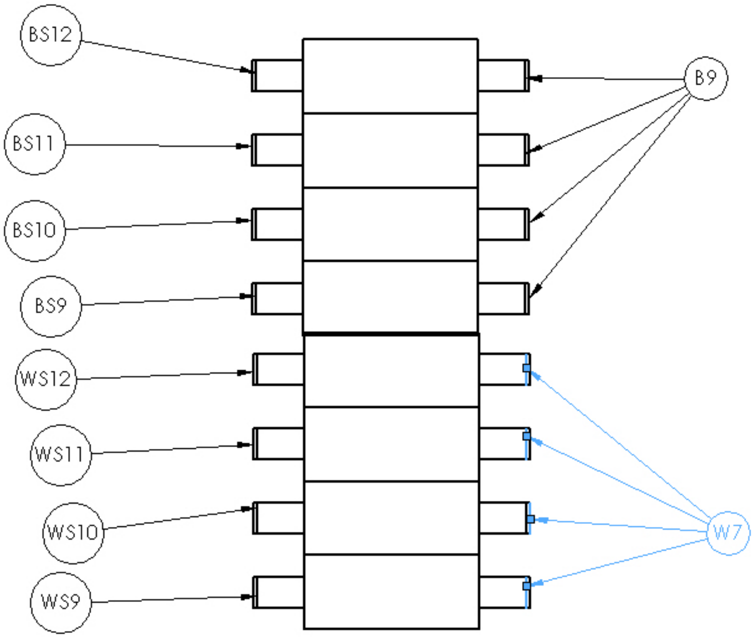


Figure 96: Module 18

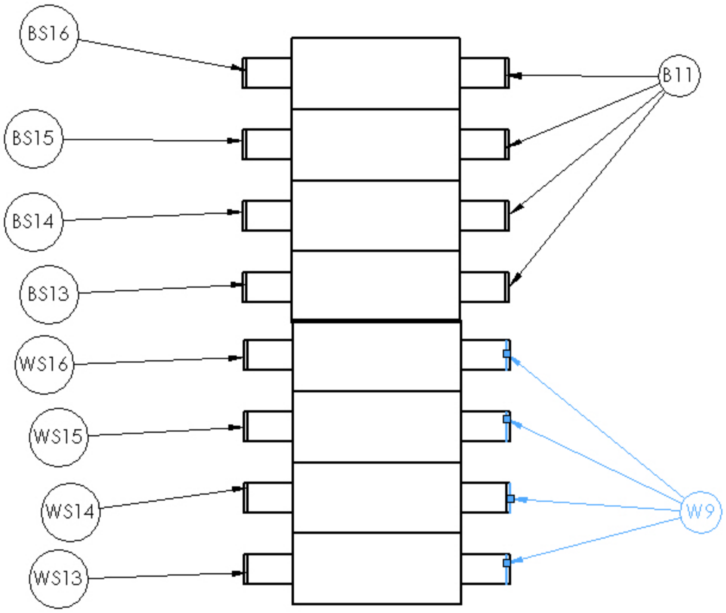


Figure 97: Module 19

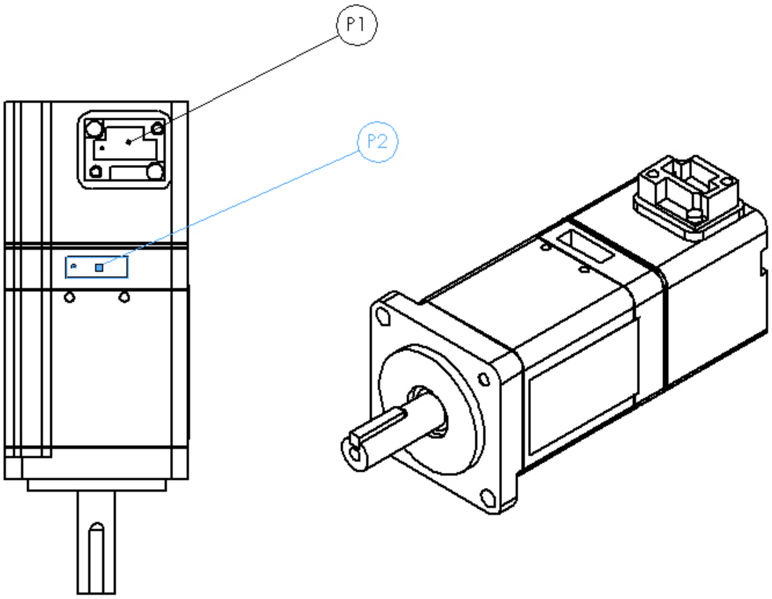


Figure 98: Motor 1

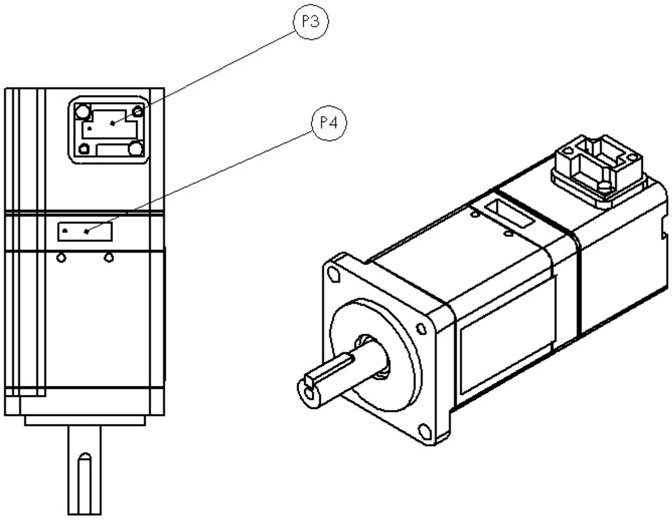


Figure 99: Motor 2

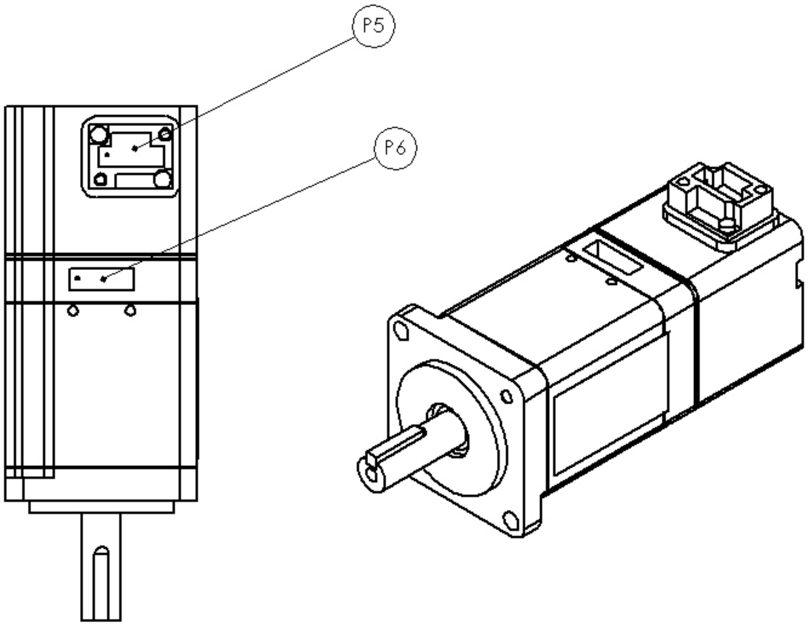


Figure 100: Motor 3

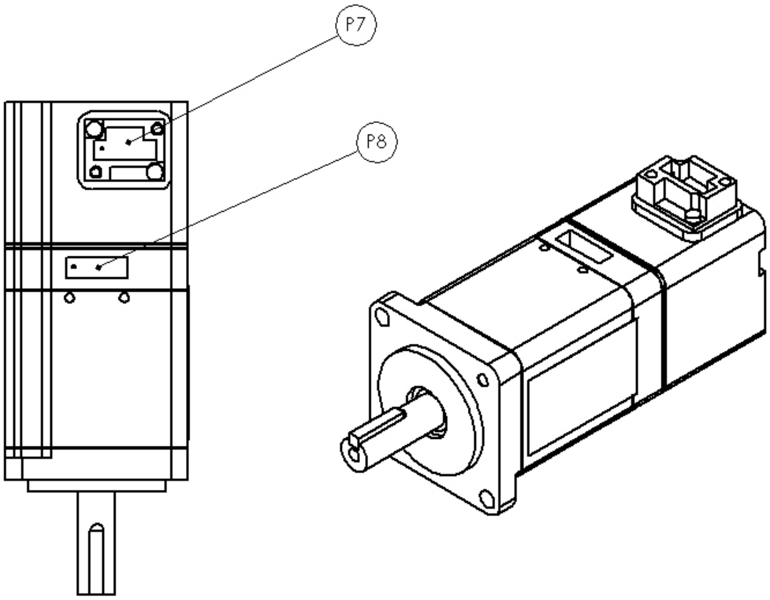


Figure 101: Motor 4

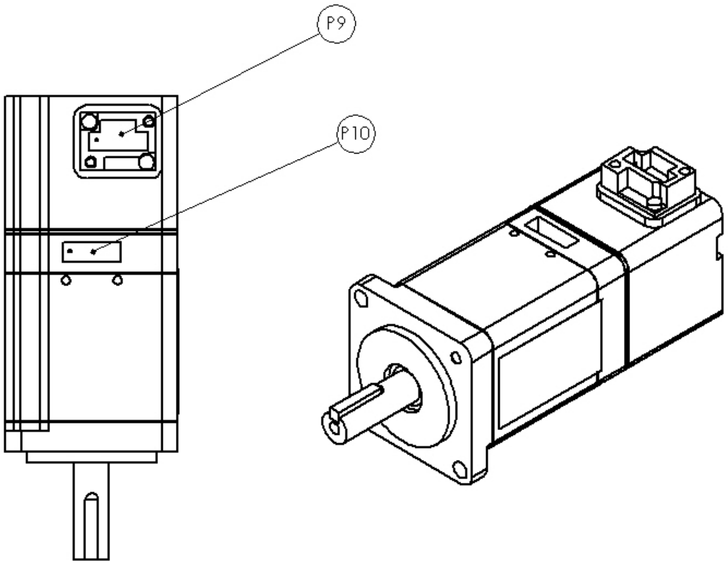


Figure 102: Motor 5

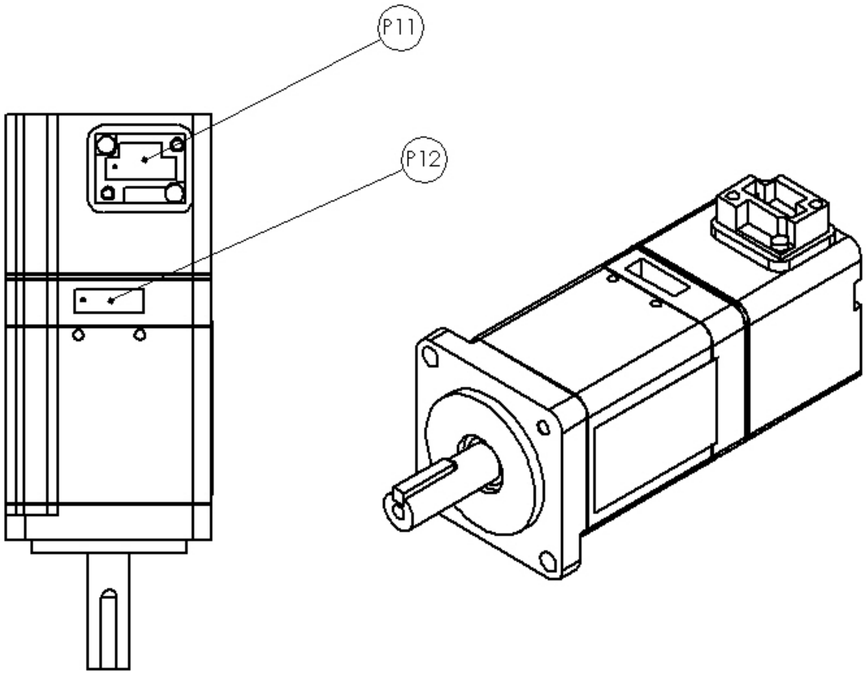


Figure 103: Motor 6

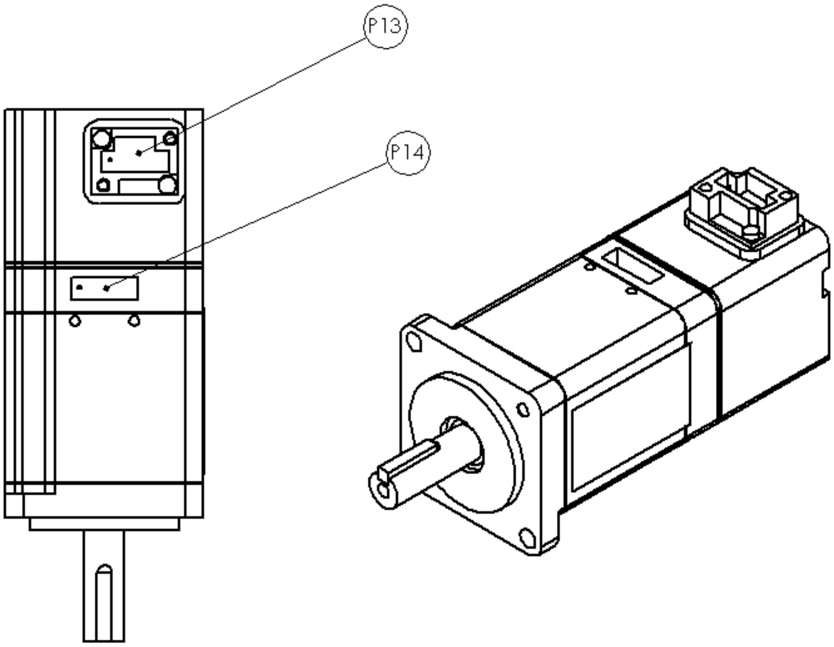


Figure 104: Motor 7

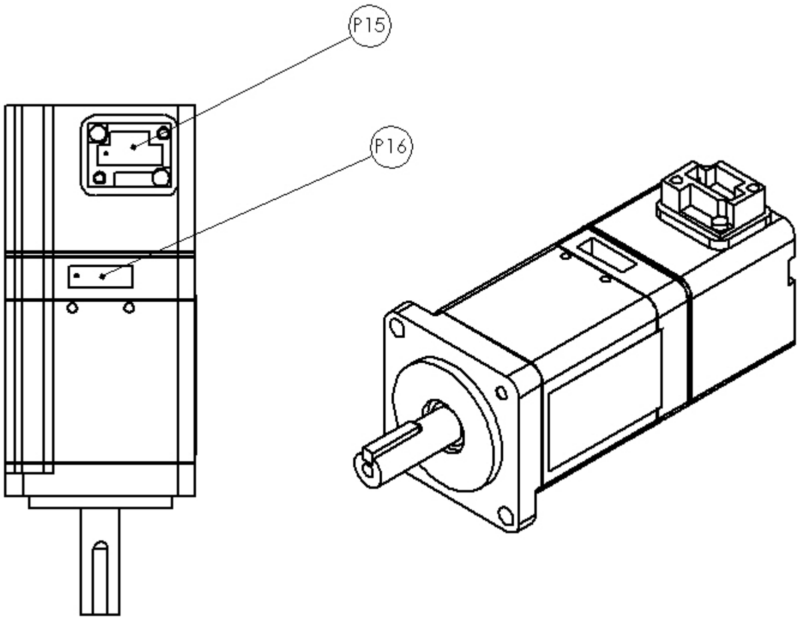
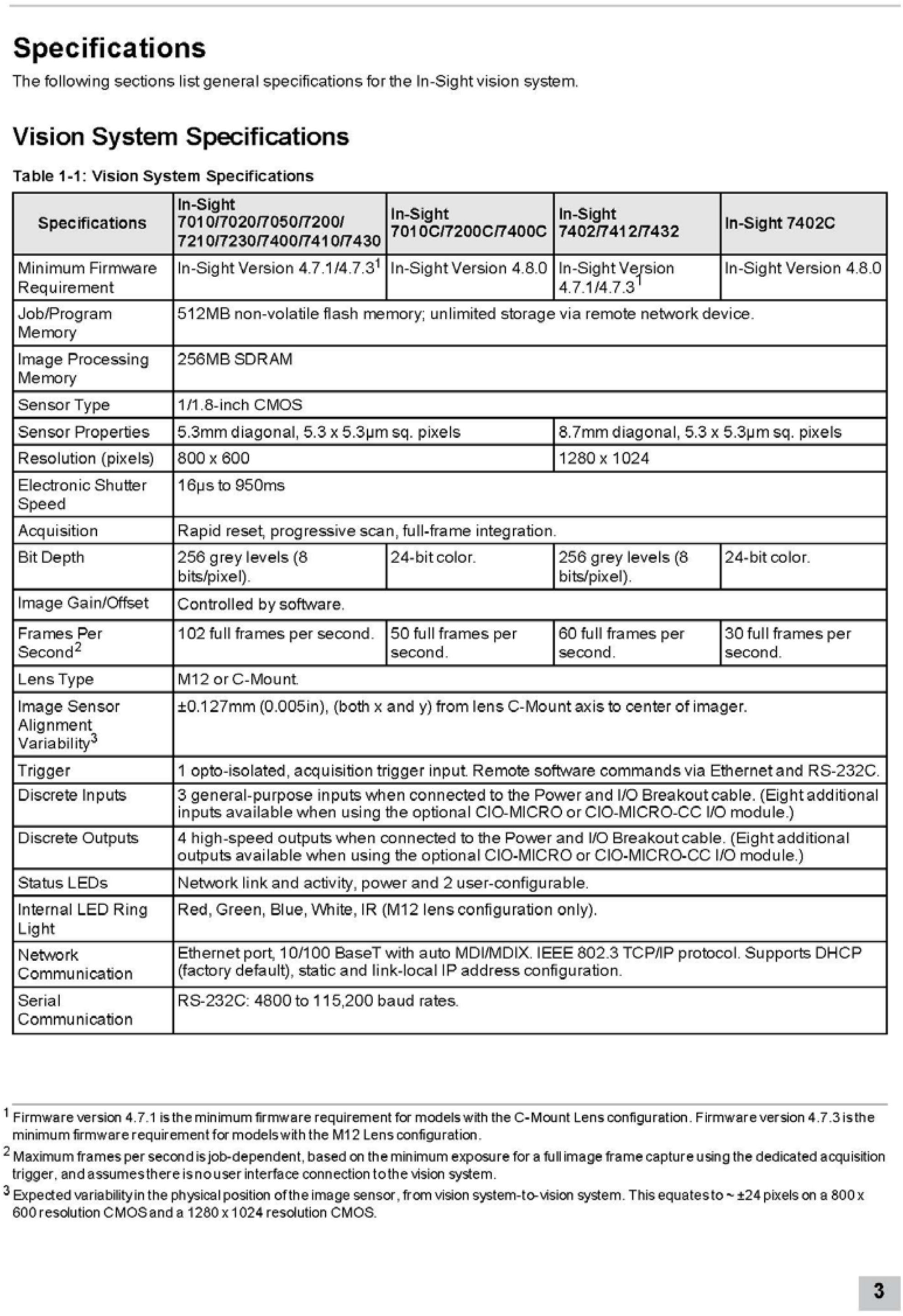


Figure 105: Motor 8

Appendix J: Vendor Data Sheets

Vision System



Specifications

The following sections list general specifications for the In-Sight vision system.

Vision System Specifications

Table 1-1: Vision System Specifications

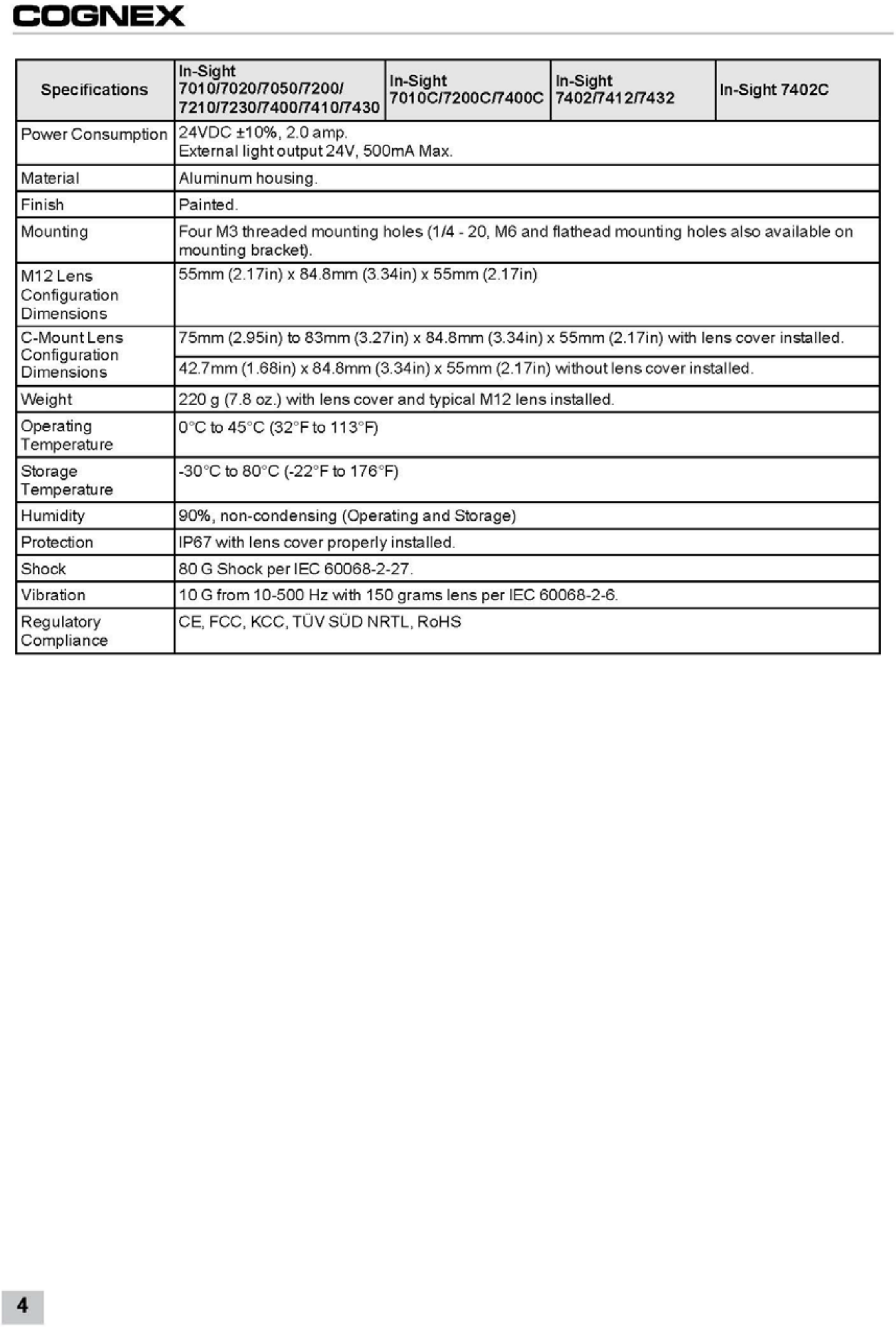
Specifications	In-Sight 7010/7020/7050/7200/ 7210/7230/7400/7410/7430	In-Sight 7010C/7200C/7400C	In-Sight 7402/7412/7432	In-Sight 7402C
Minimum Firmware Requirement	In-Sight Version 4.7.1/4.7.3[1]	In-Sight Version 4.8.0	In-Sight Version 4.7.1/4.7.3[1]	In-Sight Version 4.8.0
Job/Program Memory	512MB non-volatile flash memory; unlimited storage via remote network device.			
Image Processing Memory	256MB SDRAM			
Sensor Type	1/1.8-inch CMOS			
Sensor Properties	5.3mm diagonal, 5.3 x 5.3µm sq. pixels		8.7mm diagonal, 5.3 x 5.3µm sq. pixels	
Resolution (pixels)	800 x 600		1280 x 1024	
Electronic Shutter Speed	16µs to 950ms			
Acquisition	Rapid reset, progressive scan, full-frame integration.			
Bit Depth	256 grey levels (8 bits/pixel).	24-bit color.	256 grey levels (8 bits/pixel).	24-bit color.
Image Gain/Offset	Controlled by software.			
Frames Per Second[2]	102 full frames per second.	50 full frames per second.	60 full frames per second.	30 full frames per second.
Lens Type	M12 or C-Mount.			
Image Sensor Alignment Variability[3]	±0.127mm (0.005in), (both x and y) from lens C-Mount axis to center of imager.			
Trigger	1 opto-isolated, acquisition trigger input. Remote software commands via Ethernet and RS-232C.			
Discrete Inputs	3 general-purpose inputs when connected to the Power and I/O Breakout cable. (Eight additional inputs available when using the optional CIO-MICRO or CIO-MICRO-CC I/O module.)			
Discrete Outputs	4 high-speed outputs when connected to the Power and I/O Breakout cable. (Eight additional outputs available when using the optional CIO-MICRO or CIO-MICRO-CC I/O module.)			
Status LEDs	Network link and activity, power and 2 user-configurable.			
Internal LED Ring Light	Red, Green, Blue, White, IR (M12 lens configuration only).			
Network Communication	Ethernet port, 10/100 BaseT with auto MDI/MDIX. IEEE 802.3 TCP/IP protocol. Supports DHCP (factory default), static and link-local IP address configuration.			
Serial Communication	RS-232C: 4800 to 115,200 baud rates.			

---

[1] Firmware version 4.7.1 is the minimum firmware requirement for models with the C-Mount Lens configuration. Firmware version 4.7.3 is the minimum firmware requirement for models with the M12 Lens configuration.

[2] Maximum frames per second is job-dependent, based on the minimum exposure for a full image frame capture using the dedicated acquisition trigger, and assumes there is no user interface connection to the vision system.

[3] Expected variability in the physical position of the image sensor, from vision system-to-vision system. This equates to ~ ±24 pixels on a 800 x 600 resolution CMOS and a 1280 x 1024 resolution CMOS.

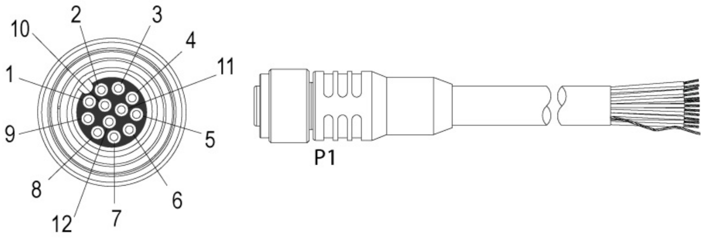


Specifications	In-Sight 7010/7020/7050/7200/ 7210/7230/7400/7410/7430	In-Sight 7010C/7200C/7400C	In-Sight 7402/7412/7432	In-Sight 7402C
Power Consumption	24VDC ±10%, 2.0 amp. External light output 24V, 500mA Max.			
Material	Aluminum housing.			
Finish	Painted.			
Mounting	Four M3 threaded mounting holes (1/4 - 20, M6 and flathead mounting holes also available on mounting bracket).			
M12 Lens Configuration Dimensions	55mm (2.17in) x 84.8mm (3.34in) x 55mm (2.17in)			
C-Mount Lens Configuration Dimensions	75mm (2.95in) to 83mm (3.27in) x 84.8mm (3.34in) x 55mm (2.17in) with lens cover installed.			
	42.7mm (1.68in) x 84.8mm (3.34in) x 55mm (2.17in) without lens cover installed.			
Weight	220 g (7.8 oz.) with lens cover and typical M12 lens installed.			
Operating Temperature	0°C to 45°C (32°F to 113°F)			
Storage Temperature	-30°C to 80°C (-22°F to 176°F)			
Humidity	90%, non-condensing (Operating and Storage)			
Protection	IP67 with lens cover properly installed.			
Shock	80 G Shock per IEC 60068-2-27.			
Vibration	10 G from 10-500 Hz with 150 grams lens per IEC 60068-2-6.			
Regulatory Compliance	CE, FCC, KCC, TÜV SÜD NRTL, RoHS			

Power and I/O Breakout Cable Specifications

The Power and I/O Breakout cable provides connections to an external power supply, the acquisition trigger input, general-purpose inputs, high-speed outputs, and RS-232 serial communications. The Power and I/O Breakout cable is not terminated.

Table 1-2: Power and I/O Breakout Cable Pin-Out



Pin#	Signal Name (I/O Mode)	Wire Color
1	IN 2	Yellow
2	IN 3	White/Yellow
3	HS OUT 2	Brown
4	HS OUT 3	White/Brown
5	IN 1/ RS-232 RECEIVE[1]	Violet
6	INPUT COMMON	White/Violet
7	+24VDC	Red
8	GROUND	Black
9	OUTPUT COMMON	Green
10	TRIGGER	Orange
11	HS OUT 0	Blue
12	HS OUT 1/ RS-232 TRANSMIT[2]	Grey
Shell	SHIELD	Bare Wire

Note:

- Cables are sold separately.
- Unused bare wires can be clipped short or tied back using a tie made of non-conductive material. Keep all bare wires separated from the +24VDC wire.

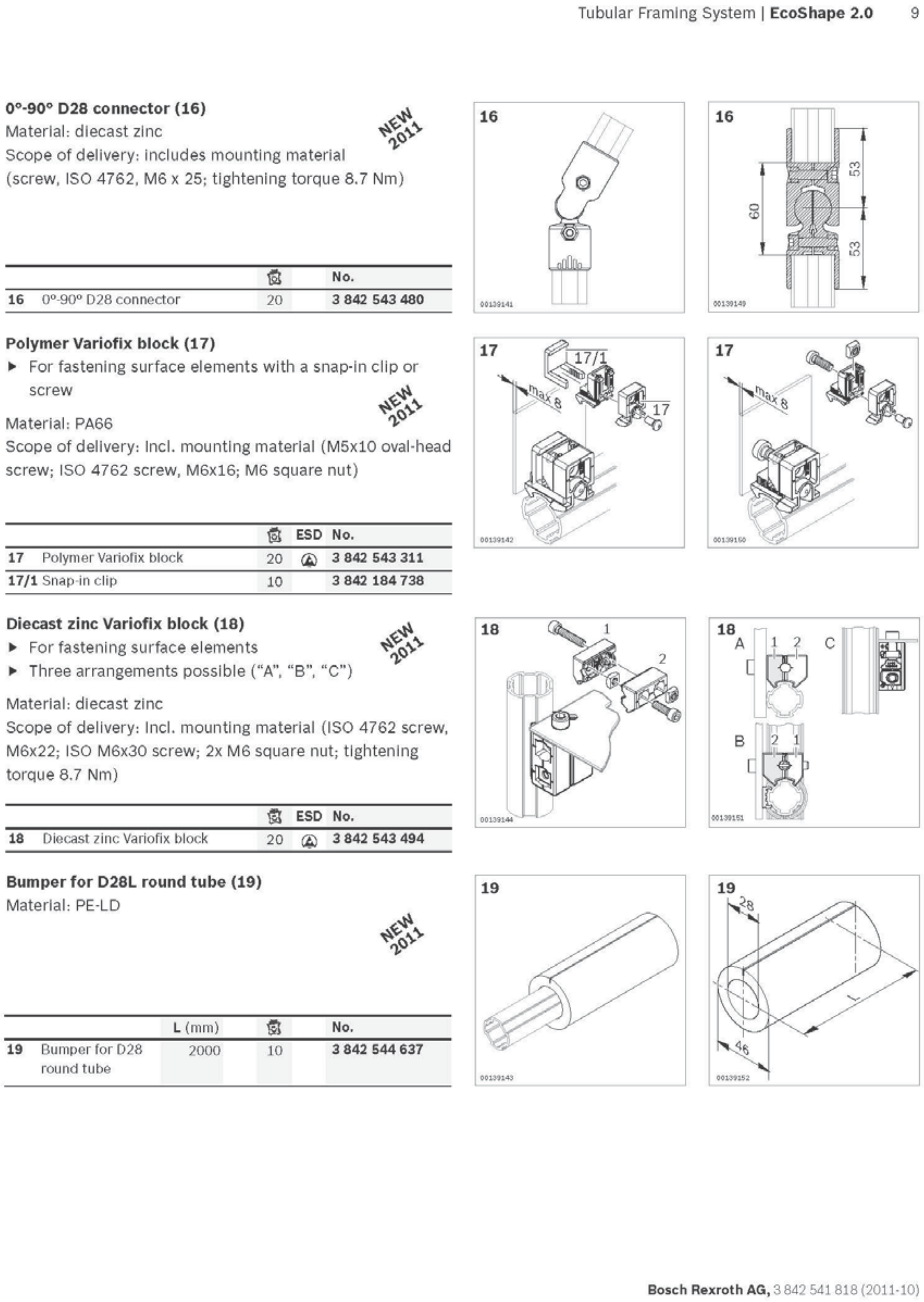
P/N 597-0145-01 Rev.B
Printed in the USA

---

[1] If hardware handshaking is required, an I/O module must be used.

[2] If hardware handshaking is required, an I/O module must be used.

Roof



0°-90° D28 connector (16)
Material: diecast zinc
Scope of delivery: includes mounting material (screw, ISO 4762, M6 x 25; tightening torque 8.7 Nm)

NEW 2011

		📦	No.
16	0°-90° D28 connector	20	**3 842 543 480**

Polymer Variofix block (17)

- For fastening surface elements with a snap-in clip or screw

NEW 2011

Material: PA66
Scope of delivery: Incl. mounting material (M5x10 oval-head screw; ISO 4762 screw, M6x16; M6 square nut)

		📦	ESD	No.
17	Polymer Variofix block	20	⚠	**3 842 543 311**
17/1	Snap-in clip	10		**3 842 184 738**

Diecast zinc Variofix block (18)

- For fastening surface elements
- Three arrangements possible ("A", "B", "C")

NEW 2011

Material: diecast zinc
Scope of delivery: Incl. mounting material (ISO 4762 screw, M6x22; ISO M6x30 screw; 2x M6 square nut; tightening torque 8.7 Nm)

		📦	ESD	No.
18	Diecast zinc Variofix block	20	⚠	**3 842 543 494**

Bumper for D28L round tube (19)
Material: PE-LD

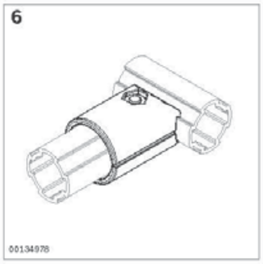
NEW 2011

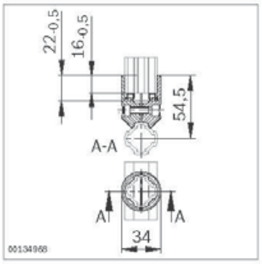
		L (mm)	📦	No.
19	Bumper for D28 round tube	2000	10	**3 842 544 637**

90° connector (6)

Material: diecast zinc
Scope of delivery: includes mounting material
(screw, ISO 4762, M6 x 25; tightening torque 8.7 Nm)

	🗑	ESD	No.
6 90° connector	20	⚠	**3 842 541 173**

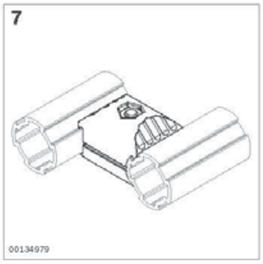


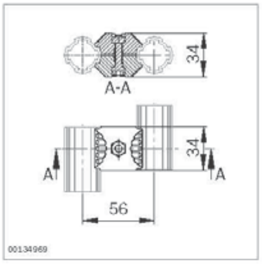


Parallel connector (7)

Material: diecast zinc
Scope of delivery: includes mounting material
(screw, ISO 4762, M6 x 25; tightening torque 8.7 Nm)

	🗑	ESD	No.
7 Parallel connector	20	⚠	**3 842 541 183**

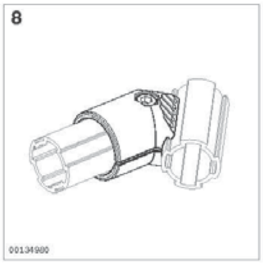


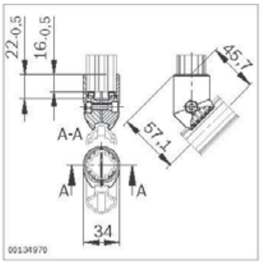


45° connector (8)

Material: diecast zinc
Scope of delivery: includes mounting material
(screw, ISO 4762, M6 x 25; tightening torque 8.7 Nm)

	🗑	ESD	No.
8 45° connector	20	⚠	**3 842 541 175**

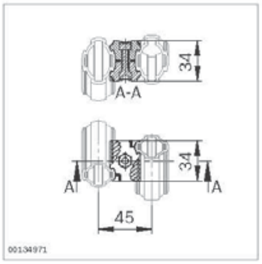




Cross connector (9)

Material: diecast zinc
Scope of delivery: includes mounting material
(screw, ISO 4762, M6 x 25; tightening torque 8.7 Nm)

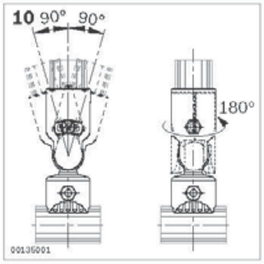
	🗑	ESD	No.
9 Cross connector	20	⚠	**3 842 541 181**

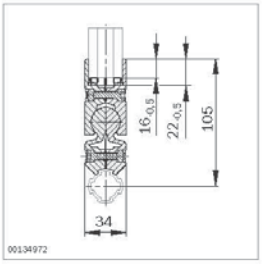


0°-90° connector (10)

Material: diecast zinc
Scope of delivery: includes mounting material
(screw, ISO 4762, M6 x 25; tightening torque 8.7 Nm)

	🗑	ESD	No.
10 0°-90° connector	20	⚠	**3 842 541 178**



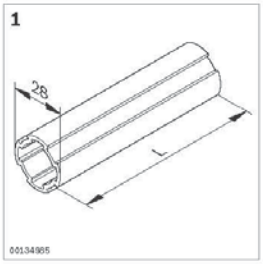


D28L round tube (1)

- 4 interfaces for attaching EcoShape connectors

Material: aluminum

		L (mm)	LE	No.
1	Round tube	50 to 5600	1	**3 842 996 191/L**
	D28L	5600	50	**3 842 541 211**



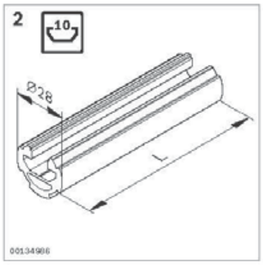
A = 1.8 cm^2
I_x = 1.33 cm^4
I_y = 1.33 cm^4
W_x = 0.95 cm^3
W_y = 0.95 cm^3
m = 0.48 kg/m

D28L, N10 round tube (2)

- A 10 mm slot for mounting accessories from the modular aluminum framing system

Material: aluminum

		L (mm)	LE	No.
2	Round tube	50 to 5600	1	**3 842 996 192/L**
	D28L, N10	5600	20	**3 842 541 213**



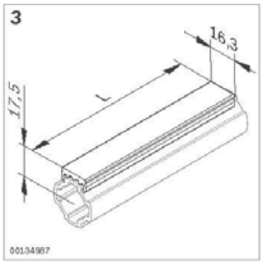
A = 2.63 cm^2
I_x = 1.17 cm^4
I_y = 1.97 cm^4
W_x = 0.84 cm^3
W_y = 1.41 cm^3
m = 0.71 kg/m

Slide rail (3)

- For creating a simple conveyor track or manual slide section by clipping slide rail onto a D28L round tube

Material: PVC; gray

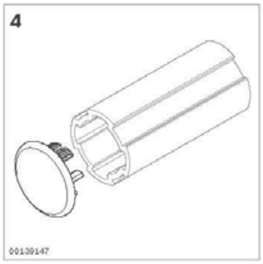
		L (mm)	📦	No.
3	Slide rail	2000	10	**3 842 541 196**



Cap (4)

Material: PA66; black

		📦	ESD	No.
4	Cap	20	⚠	**3 842 541 195**

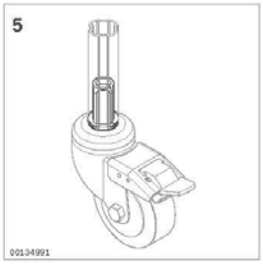


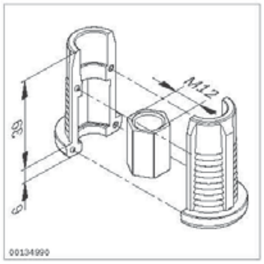
Threaded sleeve (5)

- For integrating a leveling foot or caster

Material: PA66; black

		📦	ESD	No.
5	D28L threaded sleeve	20	⚠	**3 842 541 185**





Lighting

SPECIFICATIONS Ribbon Star Supreme LED Light Strip - 118”

PRODUCT NAME Ribbon Star Supreme UL LED Strip Light - 118" (3m)

SKUs White: RL-SC-RSS-WW-10,
Warm White: RL-SC-RSS-WW-10

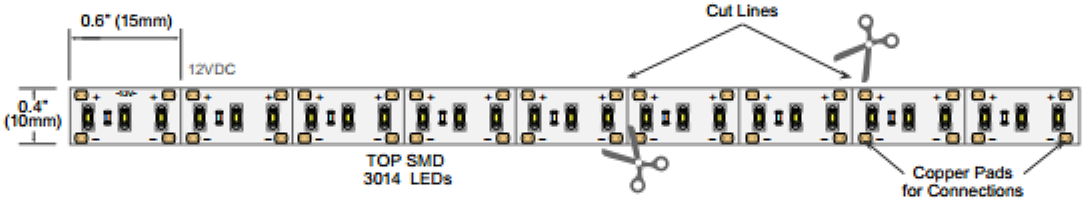
DESCRIPTION The Ribbon Star Supreme LED Ribbon is a super bright light source for many projects. This 118" (approximately 10 Feet) ribbon can be cut every 0.6” and has many quick connection options for easy connection and use. This ribbon comes with 3M™ mounting tape on the back, for the best long-term mounting solution. 3 SMD LEDs per 0.6” and 9.5W per foot of power required for this light. It can also be used for box signs, overhead lighting, cove, under cabinet, back-lighting and many more applications. It is recommended that you mount this strip to steel or aluminum, for a good heat sink. Requires a 12VDC Constant Voltage power supply.

DIMENSIONS 118” Long X 0.40” Wide X 0.07” Tall (3m L X 10mm W X 1.8mm T)

APPROVALS CE, RoHS, UL Recognized

LIFESPAN 50,000 Hours

WARRANTY 2 Year Manufacturers



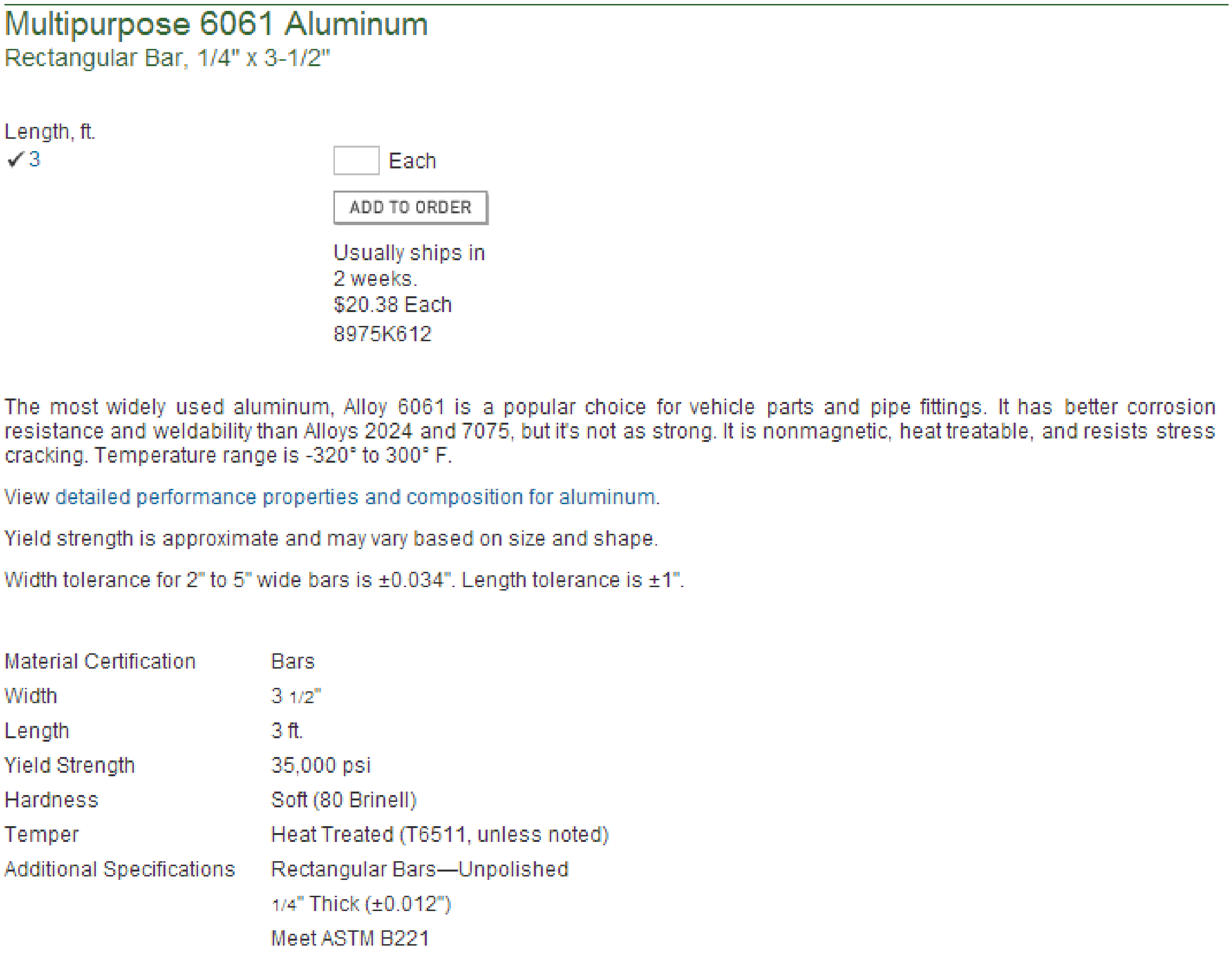
SKU	Color	Length	LED Qty	LED Type	Lumen	Beam Angle	Voltage (VDC)	Current (Amps)	Watts Required	Max Serial Connection
RL-SC-RSS-W-10	White 6500K	118" (3m)	612	3014 SMD Top LED	640 lm/ft	120°	12	7.9	9.5W per ft. 95W per 118"	12 Feet
RL-SC-RSS-WW-10	Warm Wht 3500K	118" (3m)	612	3014 SMD Top LED	542 lm/ft	120°	12	7.9	9.5W per ft. 95W per 118"	12 Feet

EcoLightLED.com
info@EcoLightLED.com

ECO LIGHT
LED Lighting Solutions

775-636-6060

Brackets



Multipurpose 6061 Aluminum

Rectangular Bar, 1/4" x 3-1/2"

Length, ft.

✓3

☐ Each

Usually ships in
2 weeks.
$20.38 Each
8975K612

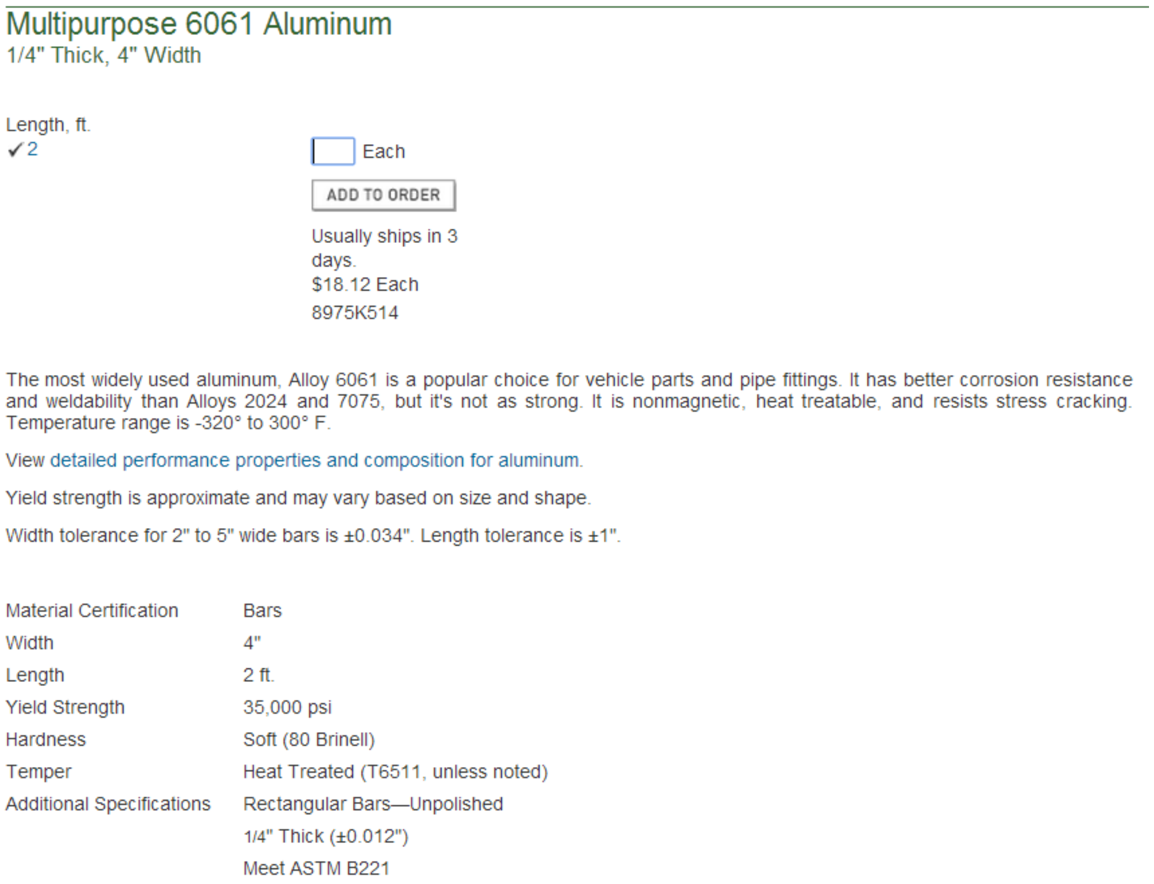
The most widely used aluminum, Alloy 6061 is a popular choice for vehicle parts and pipe fittings. It has better corrosion resistance and weldability than Alloys 2024 and 7075, but it's not as strong. It is nonmagnetic, heat treatable, and resists stress cracking. Temperature range is -320° to 300° F.

View detailed performance properties and composition for aluminum.

Yield strength is approximate and may vary based on size and shape.

Width tolerance for 2" to 5" wide bars is ±0.034". Length tolerance is ±1".

Material Certification	Bars
Width	3 1/2"
Length	3 ft.
Yield Strength	35,000 psi
Hardness	Soft (80 Brinell)
Temper	Heat Treated (T6511, unless noted)
Additional Specifications	Rectangular Bars—Unpolished 1/4" Thick (±0.012") Meet ASTM B221



Multipurpose 6061 Aluminum

1/4" Thick, 4" Width

Length, ft.	
✓2	☐ Each

Usually ships in 3 days.
$18.12 Each
8975K514

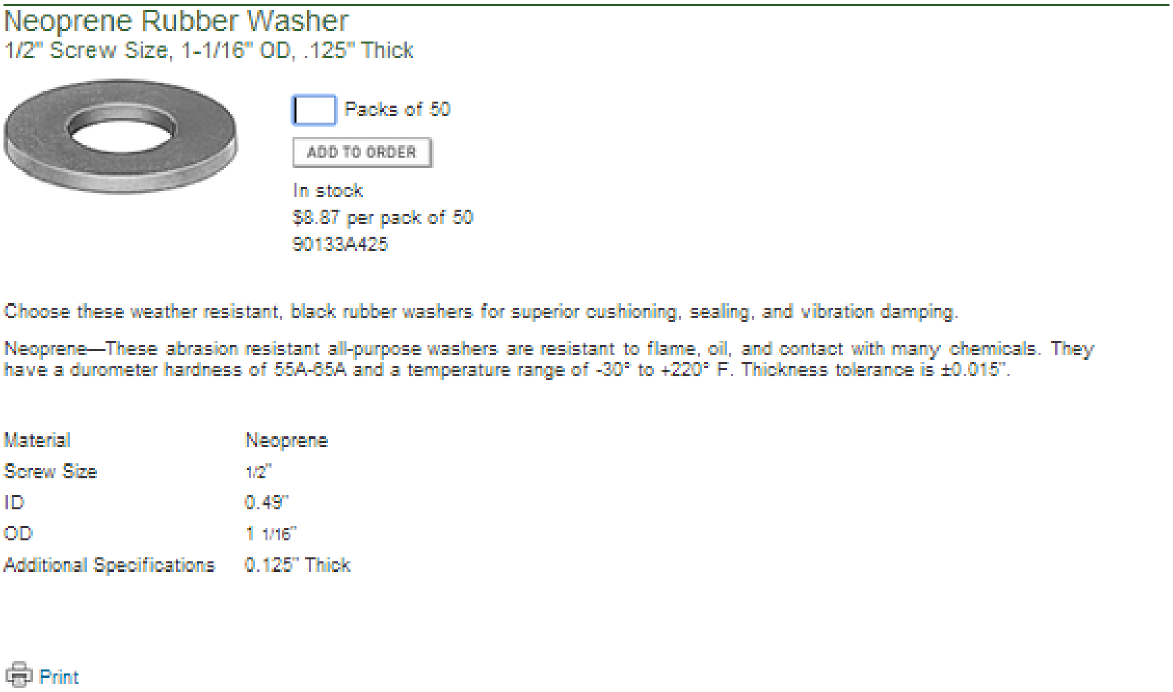
The most widely used aluminum, Alloy 6061 is a popular choice for vehicle parts and pipe fittings. It has better corrosion resistance and weldability than Alloys 2024 and 7075, but it's not as strong. It is nonmagnetic, heat treatable, and resists stress cracking. Temperature range is -320° to 300° F.

View detailed performance properties and composition for aluminum.

Yield strength is approximate and may vary based on size and shape.

Width tolerance for 2" to 5" wide bars is ±0.034". Length tolerance is ±1".

Material Certification	Bars
Width	4"
Length	2 ft.
Yield Strength	35,000 psi
Hardness	Soft (80 Brinell)
Temper	Heat Treated (T6511, unless noted)
Additional Specifications	Rectangular Bars—Unpolished 1/4" Thick (±0.012") Meet ASTM B221



Neoprene Rubber Washer

1/2" Screw Size, 1-1/16" OD, .125" Thick

☐ Packs of 50

ADD TO ORDER

In stock

$8.87 per pack of 50

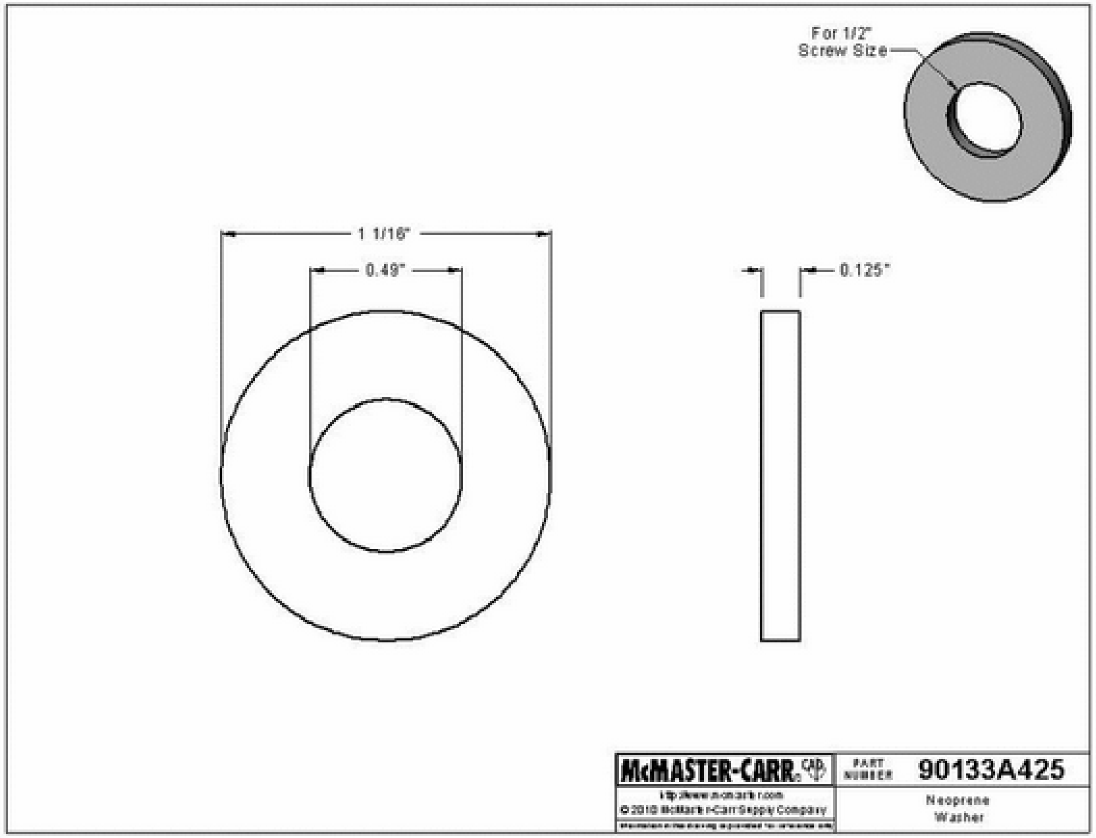
90133A425

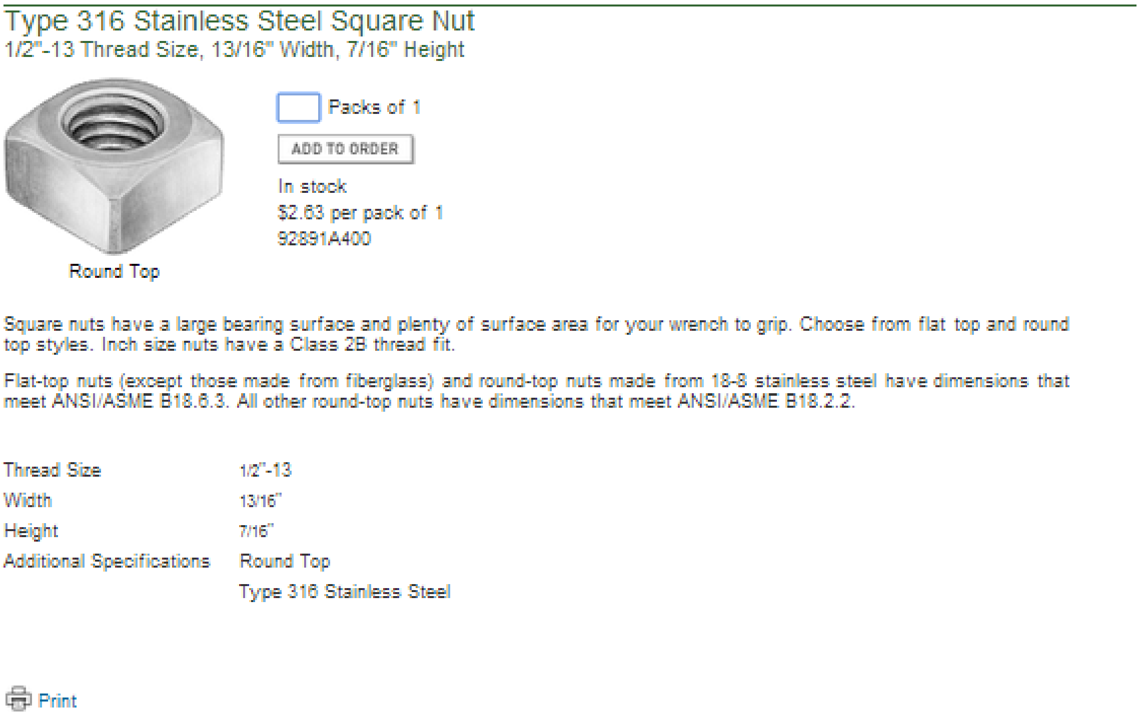
Choose these weather resistant, black rubber washers for superior cushioning, sealing, and vibration damping.

Neoprene—These abrasion resistant all-purpose washers are resistant to flame, oil, and contact with many chemicals. They have a durometer hardness of 55A-65A and a temperature range of -30° to +220° F. Thickness tolerance is ±0.015".

Material	Neoprene
Screw Size	1/2"
ID	0.49"
OD	1 1/16"
Additional Specifications	0.125" Thick

Print





Type 316 Stainless Steel Square Nut

1/2"-13 Thread Size, 13/16" Width, 7/16" Height

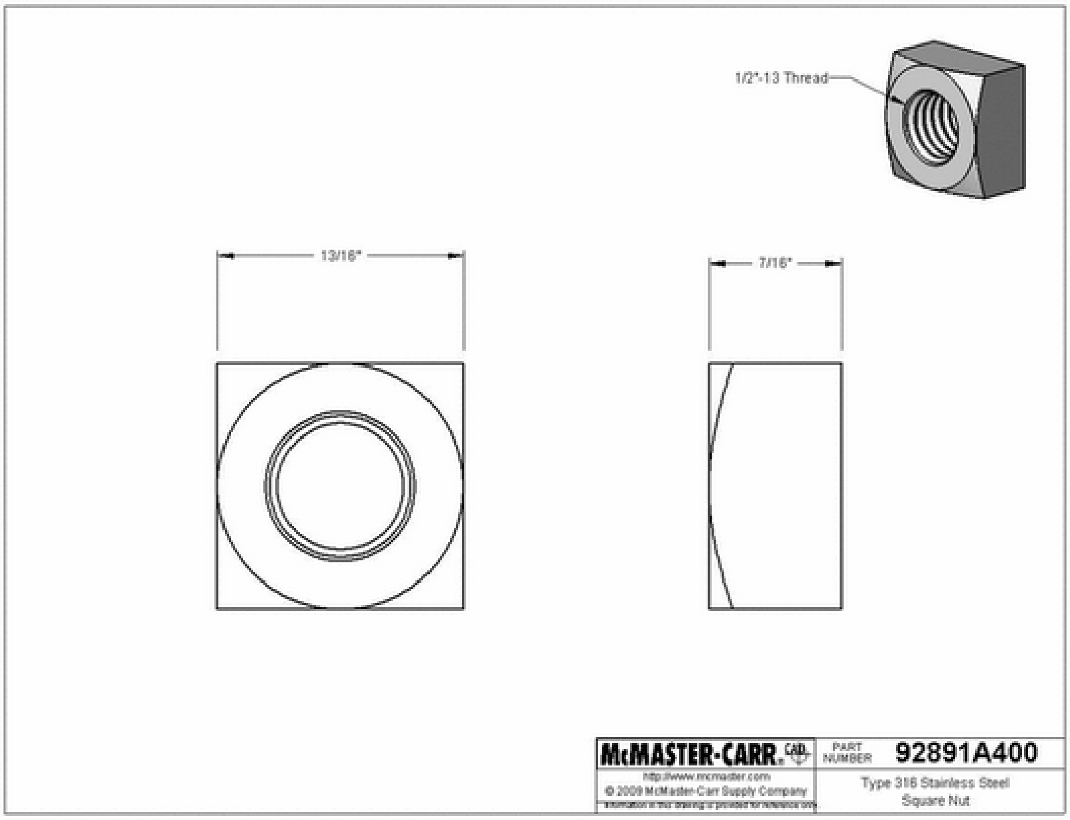
☐ Packs of 1

In stock
$2.63 per pack of 1
92891A400

Square nuts have a large bearing surface and plenty of surface area for your wrench to grip. Choose from flat top and round top styles. Inch size nuts have a Class 2B thread fit.

Flat-top nuts (except those made from fiberglass) and round-top nuts made from 18-8 stainless steel have dimensions that meet ANSI/ASME B18.6.3. All other round-top nuts have dimensions that meet ANSI/ASME B18.2.2.

Thread Size	1/2"-13
Width	13/16"
Height	7/16"
Additional Specifications	Round Top Type 316 Stainless Steel



Grade 5 Square Head Zinc-Plated Steel Bolt

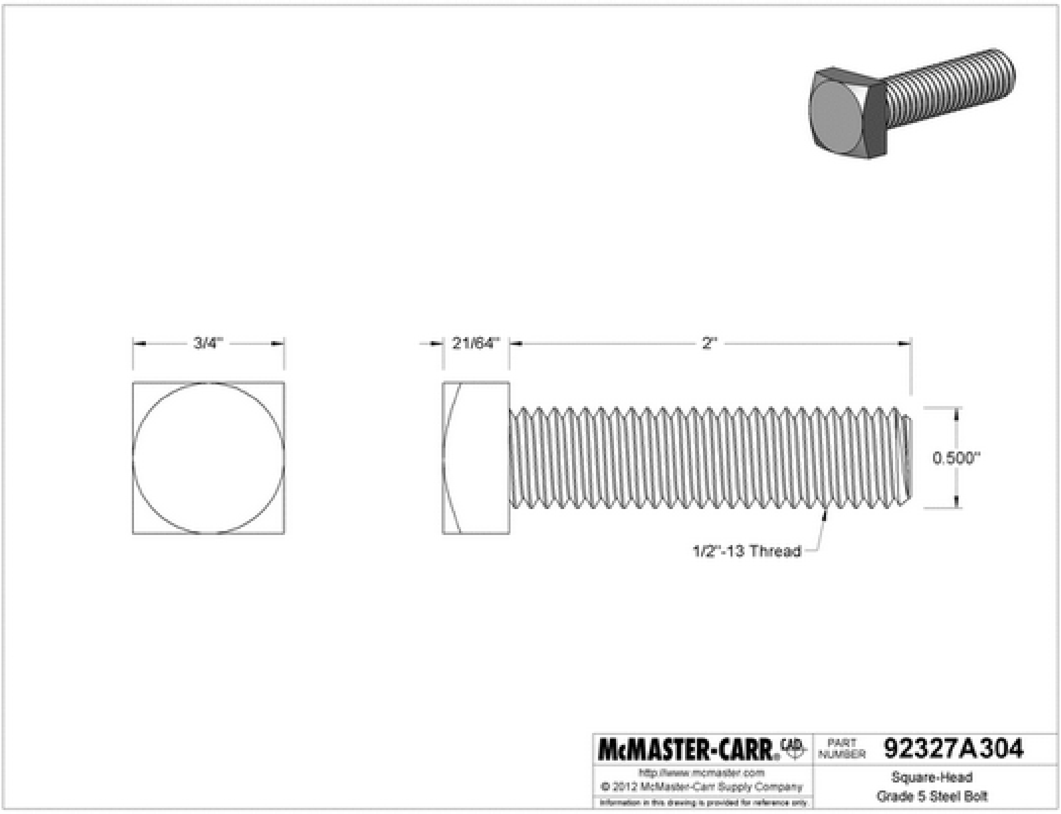
1/2"-13 Thread, 2" Length, Fully Threaded

In stock
$3.12 Each
92327A304

Square heads offer plenty of surface for your wrench to grip—and that means less slippage. All have a Class 2A thread fit. Bolt length is measured from under the head. For nuts see Square Nuts.

Medium-Strength Steel—Grade 5 bolts have a zinc yellow-chromate plating and are marked on the head with three radial lines to indicate Grade 5. Each bolt has a minimum Rockwell hardness of C25, minimum tensile strength of 120,000 psi, and meets ASME B18.2.1 and SAE J429.

Thread Size	1/2"-13
Length	2"
Head	
Height	21/64"
Width	3/4"
Additional Specifications	Medium-Strength Steel—Grade 5 Fully threaded.



Safety Switches

Magnetically Actuated Switch

Mini Rectangular, SPST-NO, 400V DC

In stock
$12.14 Each
65985K11

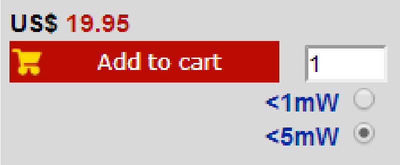
With no moving parts to wear out, these switches provide long-term, trouble-free service. When the magnet comes within the sensing distance of the switch, the switch actuates. When the magnet moves away, the switch resets. The switch is usually mounted in a stationary location such as a door frame while the magnet is mounted on a movable object such as a door. Switches with mounting slots do not include hardware; switches with mounting threads include two hex nuts.

Standard switches are designed for proximity-sensing applications such as detecting when a door or window is open. Magnet included. They are not for use on safety-guard doors found on machinery. DC-rated switches have two wire leads except SPDT switches have three. UL and C-UL recognized.

Amp Rating	2 amps @ 24V DC
Maximum Voltage	400 DC
Sensing Distance	0.25"
(A)	1.13"
(B)	0.25"
(C)	0.75"
Housing	Polyester
Mounting Information	0.13" Dia. Slots
Additional Specifications	Standard DC Rated—Switch One Circuit from Off to On (SPST-NO) Mini Rectangular

Laser Alignment Tool

YCHG-650C : Adjustable Cross-Hair Red Laser Module



US$ 19.95

Add to cart 1

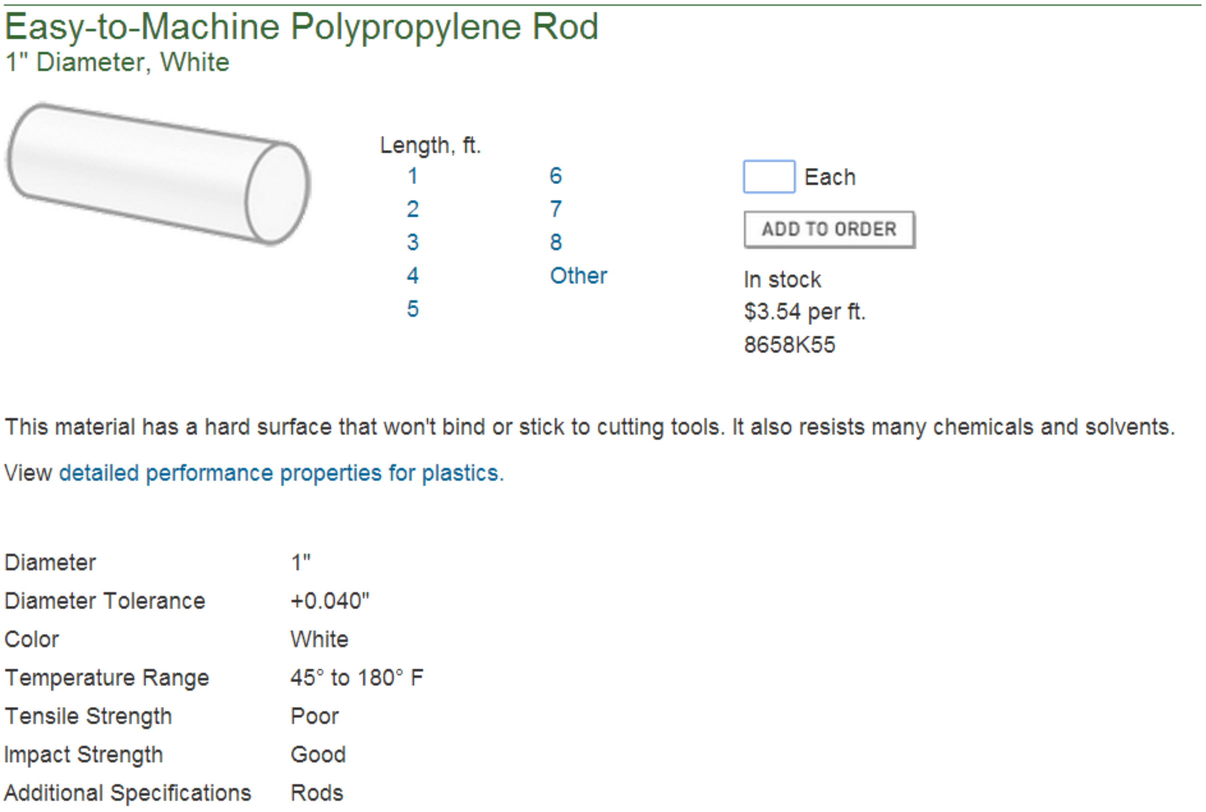
<1mW ○

<5mW ●

Self-contained adjustable cross-hair red laser module for alignment and positionning.

Specifications :

Laser Class :	II or IIIa
Wavelength :	650nm
Output Power :	<1mW, <5mW
Operation Voltage :	3V DC
Operation Current :	35mA
Operation Temp. :	-10C to +40C
Divergence :	<2.0 mrd
Beam :	Cross line (Adjustable focus)
Length :	35 mm
Diameter :	12 mm
Wires Length :	55 mm
Case Material :	Metal



Easy-to-Machine Polypropylene Rod

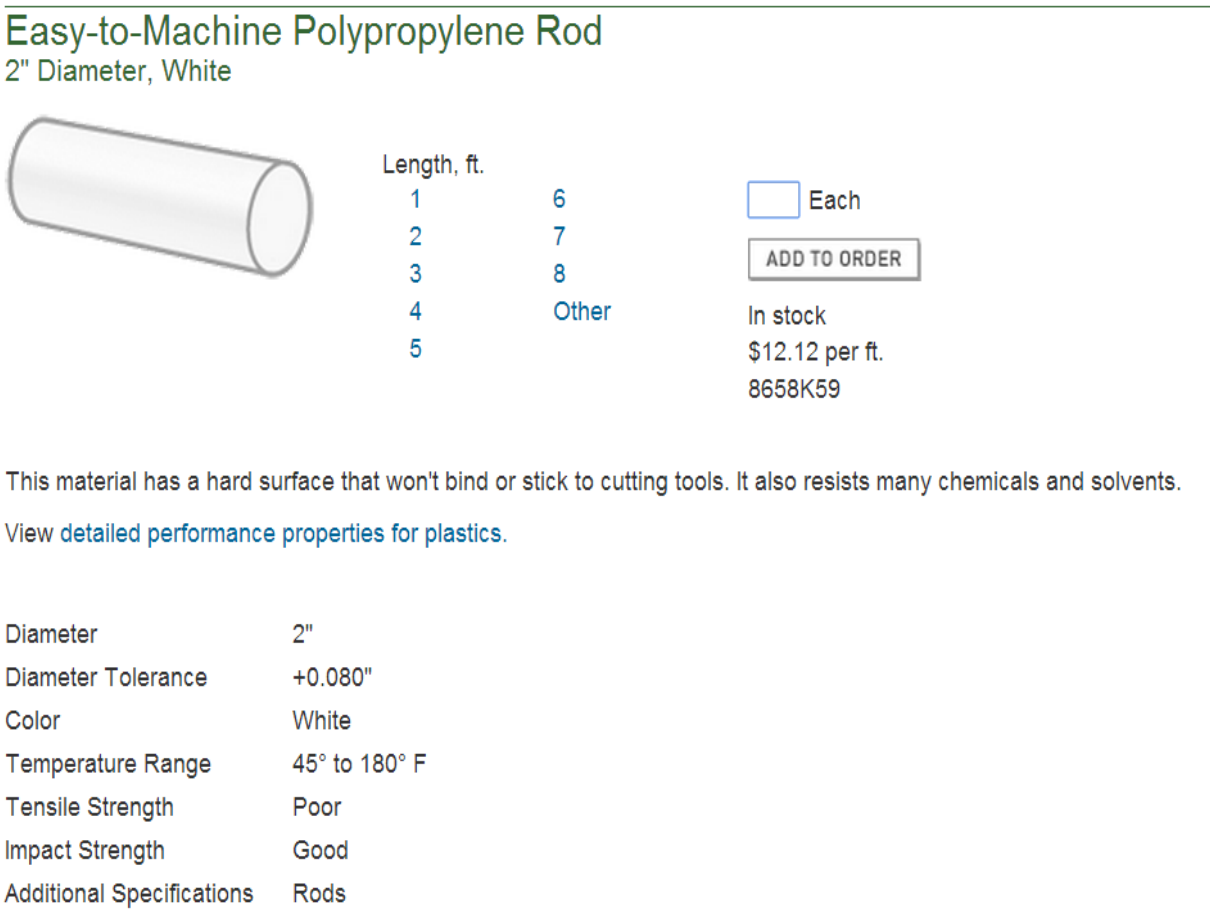
1" Diameter, White

In stock
$3.54 per ft.
8658K55

This material has a hard surface that won't bind or stick to cutting tools. It also resists many chemicals and solvents.

View detailed performance properties for plastics.

Diameter	1"
Diameter Tolerance	+0.040"
Color	White
Temperature Range	45° to 180° F
Tensile Strength	Poor
Impact Strength	Good
Additional Specifications	Rods



Easy-to-Machine Polypropylene Rod

2" Diameter, White

Length, ft.

1
2
3
4
5
6
7
8
Other

☐ Each

In stock
$12.12 per ft.
8658K59

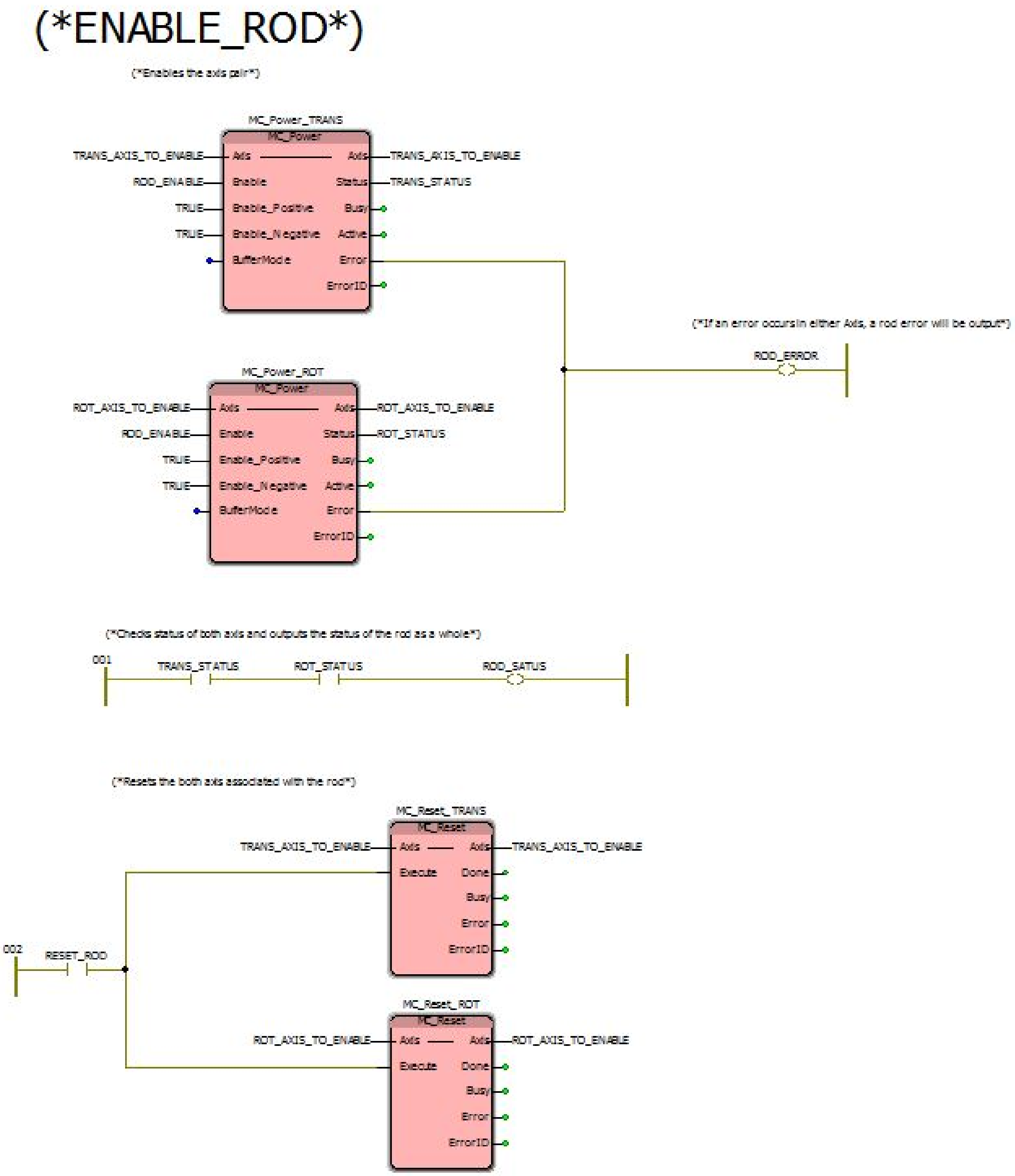
This material has a hard surface that won't bind or stick to cutting tools. It also resists many chemicals and solvents.

View detailed performance properties for plastics.

Diameter	2"
Diameter Tolerance	+0.080"
Color	White
Temperature Range	45° to 180° F
Tensile Strength	Poor
Impact Strength	Good
Additional Specifications	Rods

Appendix K: Pictures of Code

Rod Enable



(*ENABLE_ROD*)

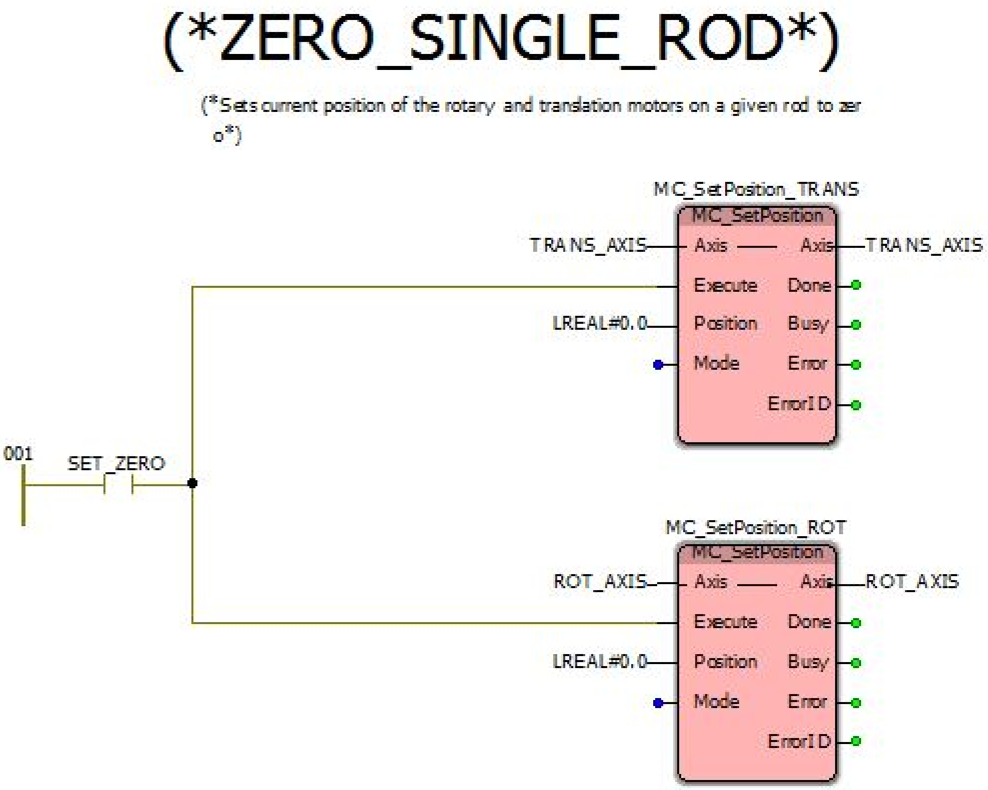
(*Enables the axis pair*)

(*If an error occurs in either Axis, a rod error will be output*)

(*Checks status of both axis and outputs the status of the rod as a whole*)

(*Resets the both axis associated with the rod*)

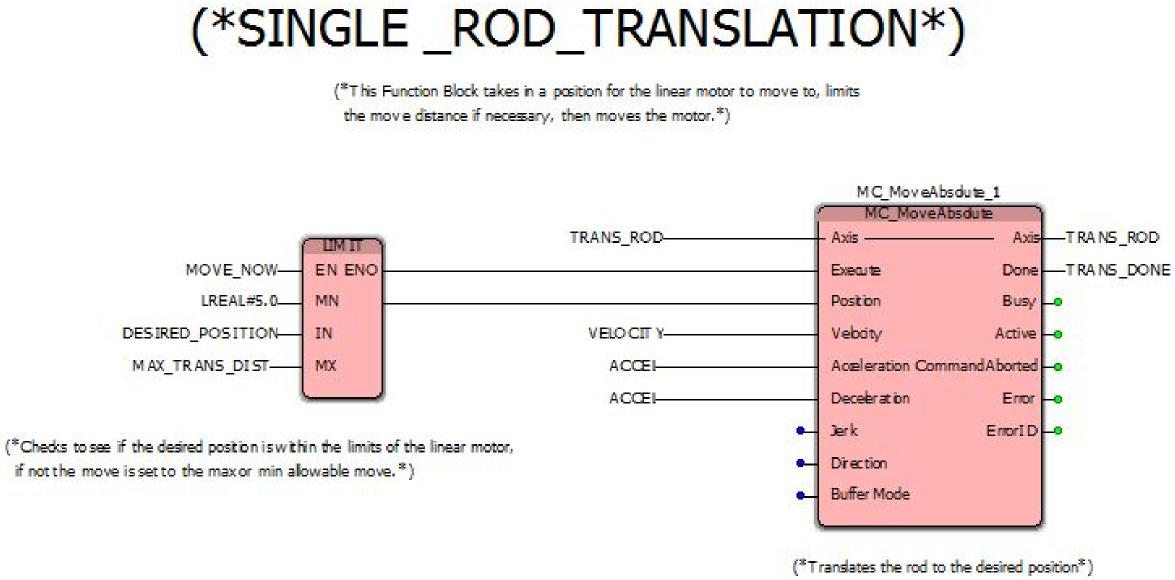
Zero Single Rod



(*ZERO_SINGLE_ROD*)

(*Sets current position of the rotary and translation motors on a given rod to zero*)

Single Rod Translation



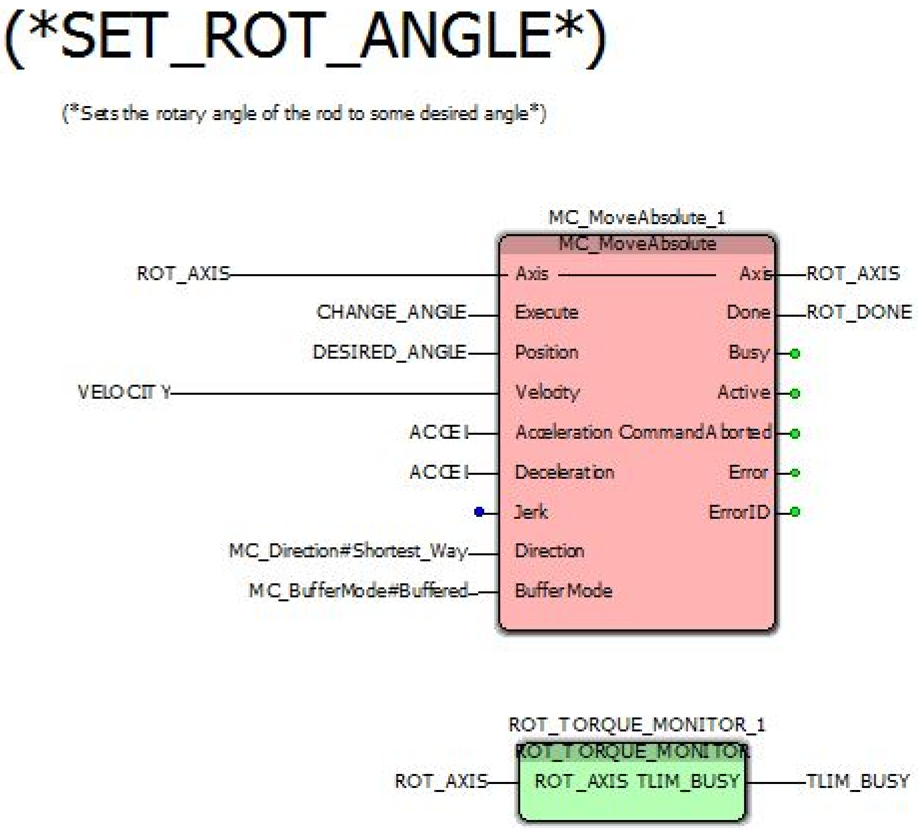
(*SINGLE _ROD_TRANSLATION*)

(*This Function Block takes in a position for the linear motor to move to, limits the move distance if necessary, then moves the motor.*)

(*Checks to see if the desired position is within the limits of the linear motor, if not the move is set to the max or min allowable move.*)

(*Translates the rod to the desired position*)

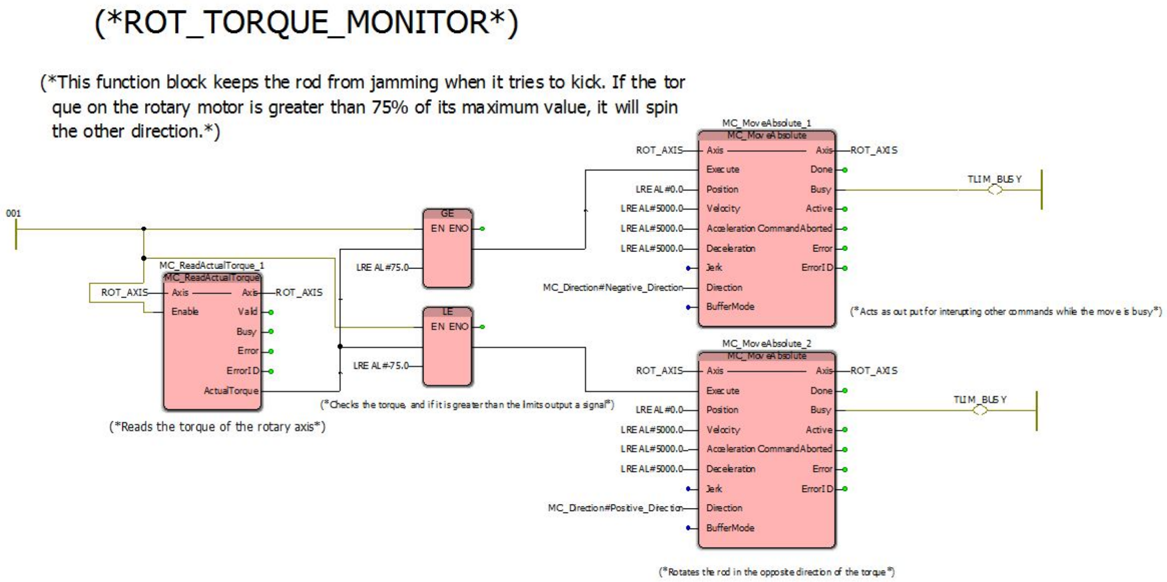
Set Rod Angle



(*SET_ROT_ANGLE*)

(*Sets the rotary angle of the rod to some desired angle*)

Rotational Roque Monitor



(*ROT_TORQUE_MONITOR*)

(*This function block keeps the rod from jamming when it tries to kick. If the tor
que on the rotary motor is greater than 75% of its maximum value, it will spin
the other direction.*)

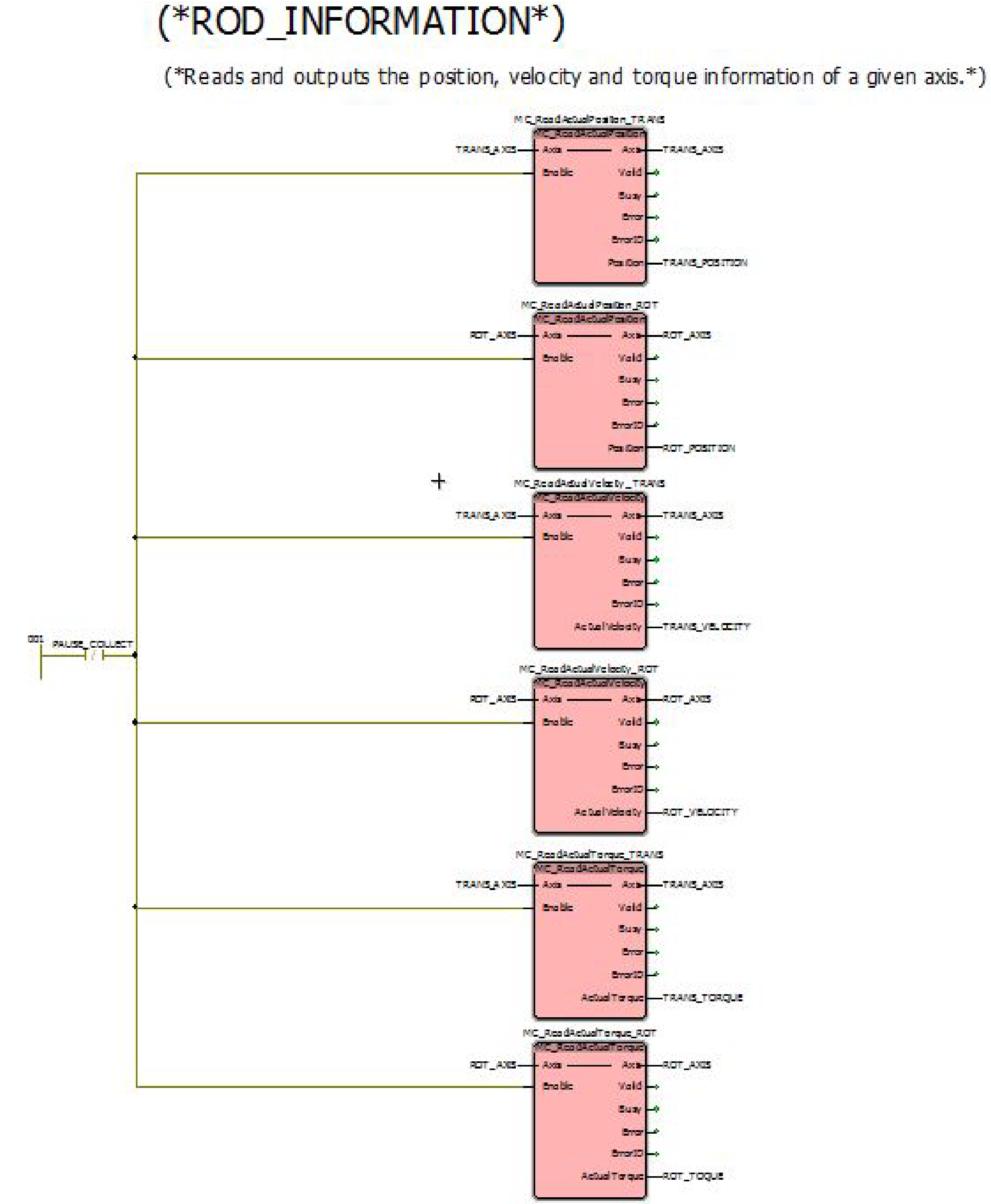
(*Reads the torque of the rotary axis*)

(*Checks the torque, and if it is greater than the limits output a signal*)

(*Acts as out put for interupting other commands while the move is busy*)

(*Rotates the rod in the opposite direction of the torque*)

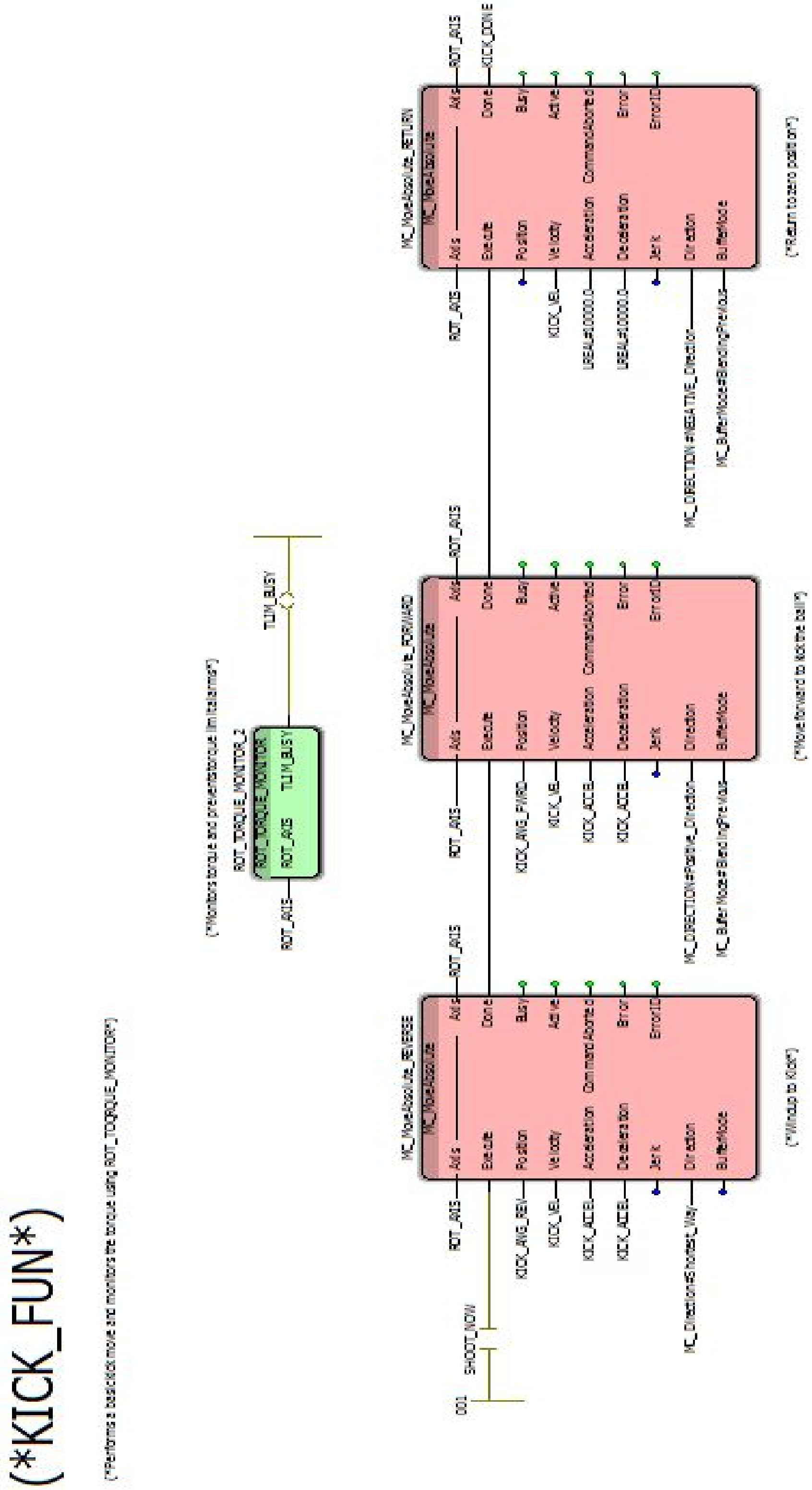
Rod Information



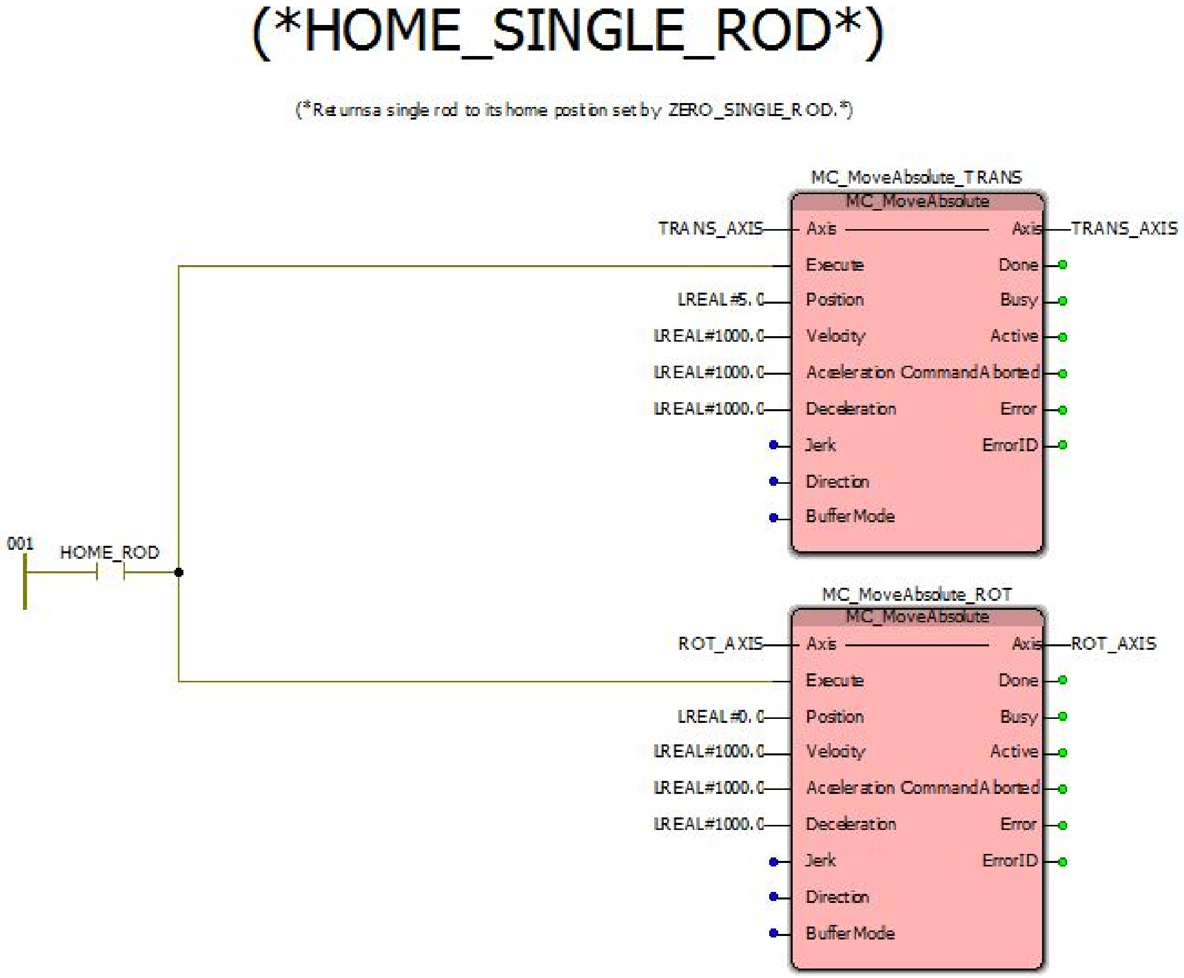
(*ROD_INFORMATION*)

(*Reads and outputs the position, velocity and torque information of a given axis.*)

Kick Function



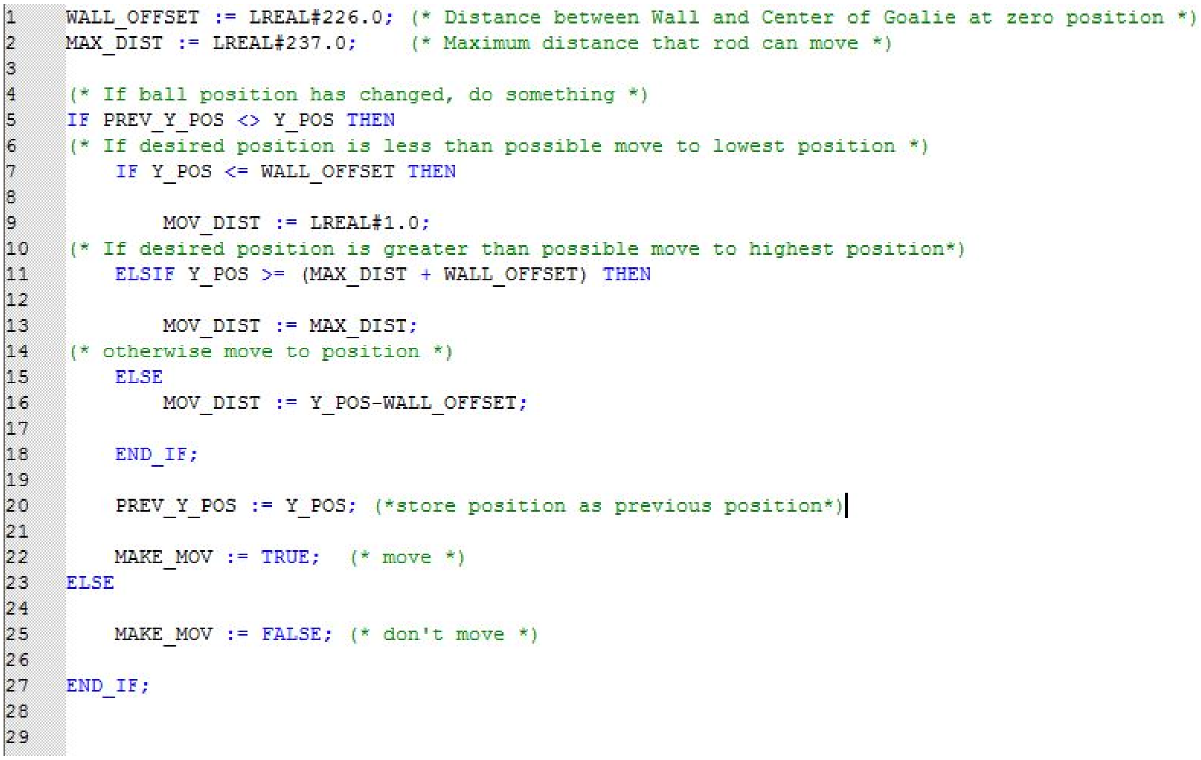
Home Single Rod



(*HOME_SINGLE_ROD*)

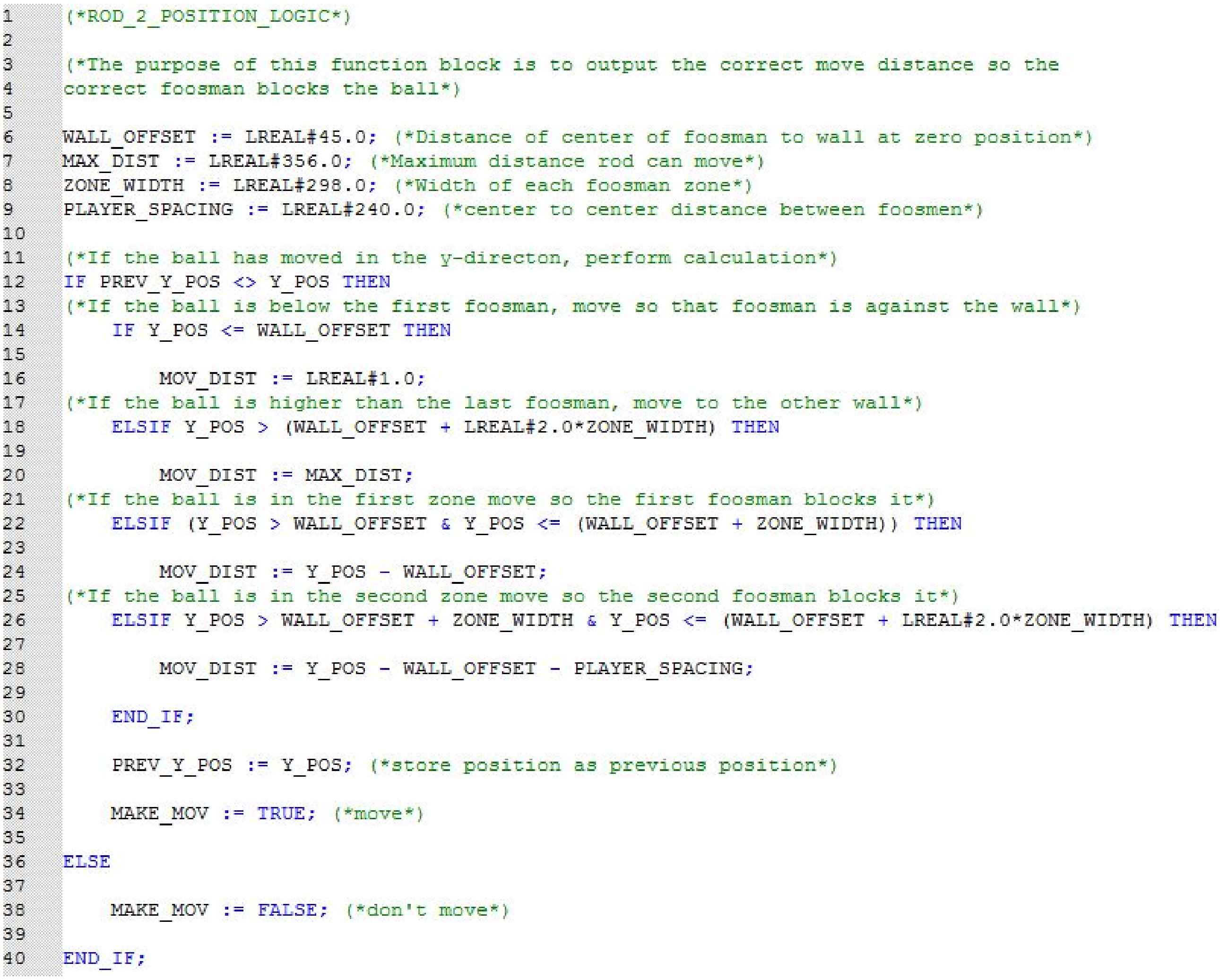
(*Returnsa single rod to its home position set by ZERO_SINGLE_ROD.*)

Rod Position Logic 1



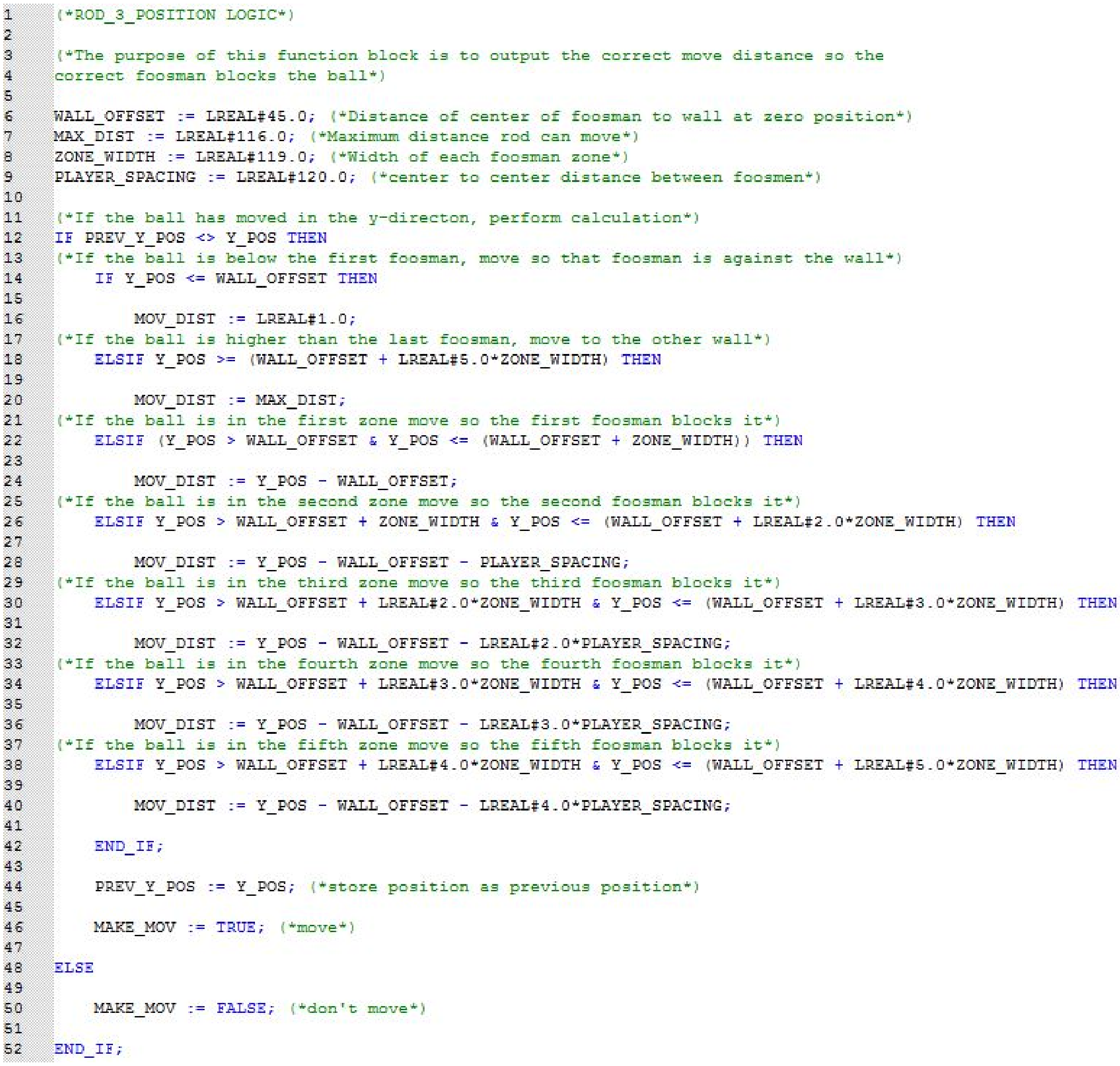
```
WALL_OFFSET := LREAL#226.0; (* Distance between Wall and Center of Goalie at zero position *)
MAX_DIST := LREAL#237.0;    (* Maximum distance that rod can move *)

(* If ball position has changed, do something *)
IF PREV_Y_POS <> Y_POS THEN
(* If desired position is less than possible move to lowest position *)
    IF Y_POS <= WALL_OFFSET THEN

        MOV_DIST := LREAL#1.0;
(* If desired position is greater than possible move to highest position*)
    ELSIF Y_POS >= (MAX_DIST + WALL_OFFSET) THEN

        MOV_DIST := MAX_DIST;
(* otherwise move to position *)
    ELSE
        MOV_DIST := Y_POS-WALL_OFFSET;

    END_IF;

    PREV_Y_POS := Y_POS; (*store position as previous position*)

    MAKE_MOV := TRUE;  (* move *)
ELSE

    MAKE_MOV := FALSE; (* don't move *)

END_IF;
```

Rod Position Logic 2



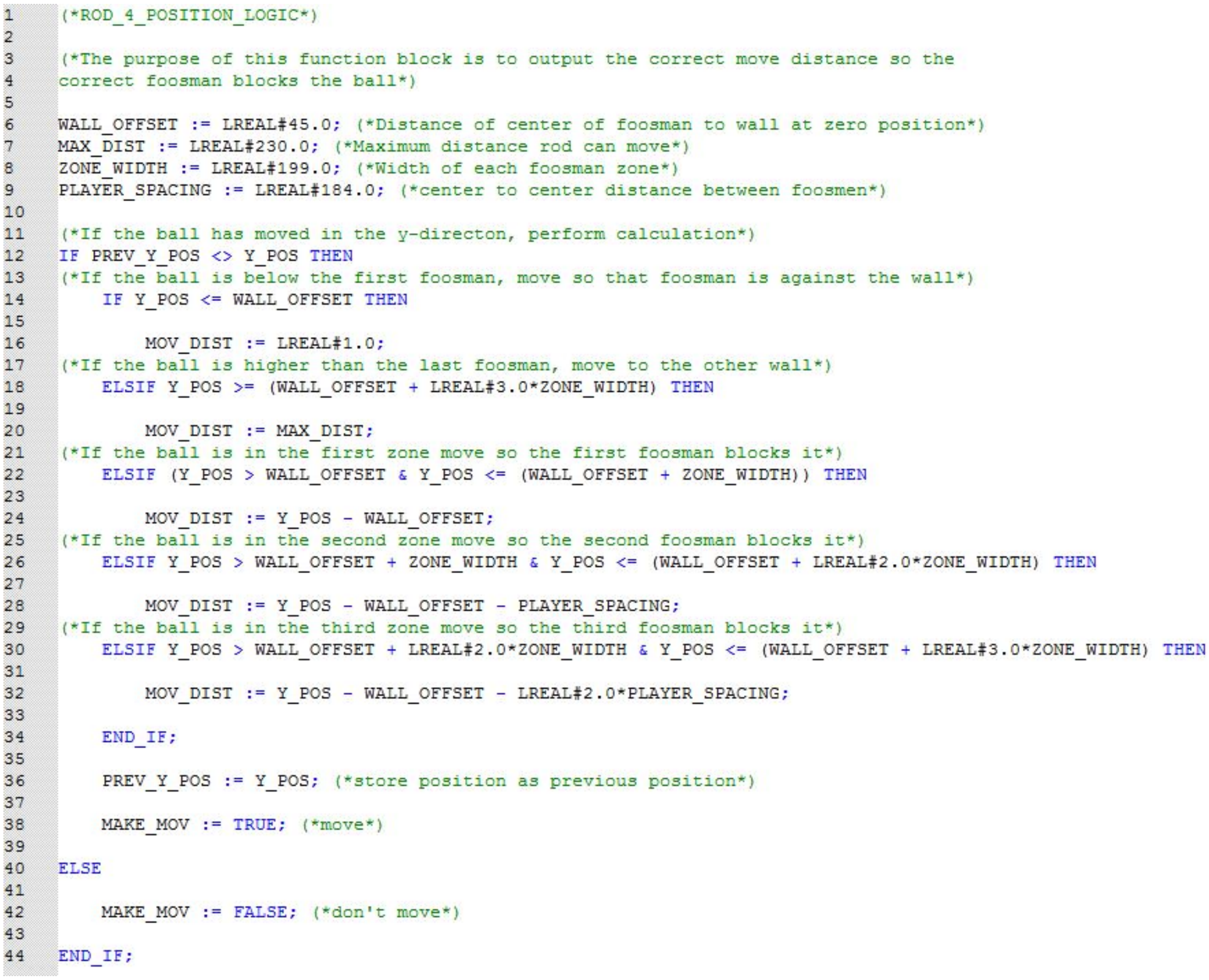
```
(*ROD_2_POSITION_LOGIC*)

(*The purpose of this function block is to output the correct move distance so the
correct foosman blocks the ball*)

WALL_OFFSET := LREAL#45.0; (*Distance of center of foosman to wall at zero position*)
MAX_DIST := LREAL#356.0; (*Maximum distance rod can move*)
ZONE_WIDTH := LREAL#298.0; (*Width of each foosman zone*)
PLAYER_SPACING := LREAL#240.0; (*center to center distance between foosmen*)

(*If the ball has moved in the y-directon, perform calculation*)
IF PREV_Y_POS <> Y_POS THEN
(*If the ball is below the first foosman, move so that foosman is against the wall*)
    IF Y_POS <= WALL_OFFSET THEN

        MOV_DIST := LREAL#1.0;
(*If the ball is higher than the last foosman, move to the other wall*)
    ELSIF Y_POS > (WALL_OFFSET + LREAL#2.0*ZONE_WIDTH) THEN

        MOV_DIST := MAX_DIST;
(*If the ball is in the first zone move so the first foosman blocks it*)
    ELSIF (Y_POS > WALL_OFFSET & Y_POS <= (WALL_OFFSET + ZONE_WIDTH)) THEN

        MOV_DIST := Y_POS - WALL_OFFSET;
(*If the ball is in the second zone move so the second foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#2.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - PLAYER_SPACING;

    END_IF;

    PREV_Y_POS := Y_POS; (*store position as previous position*)

    MAKE_MOV := TRUE; (*move*)

ELSE

    MAKE_MOV := FALSE; (*don't move*)

END_IF;
```

Rod Position Logic 3



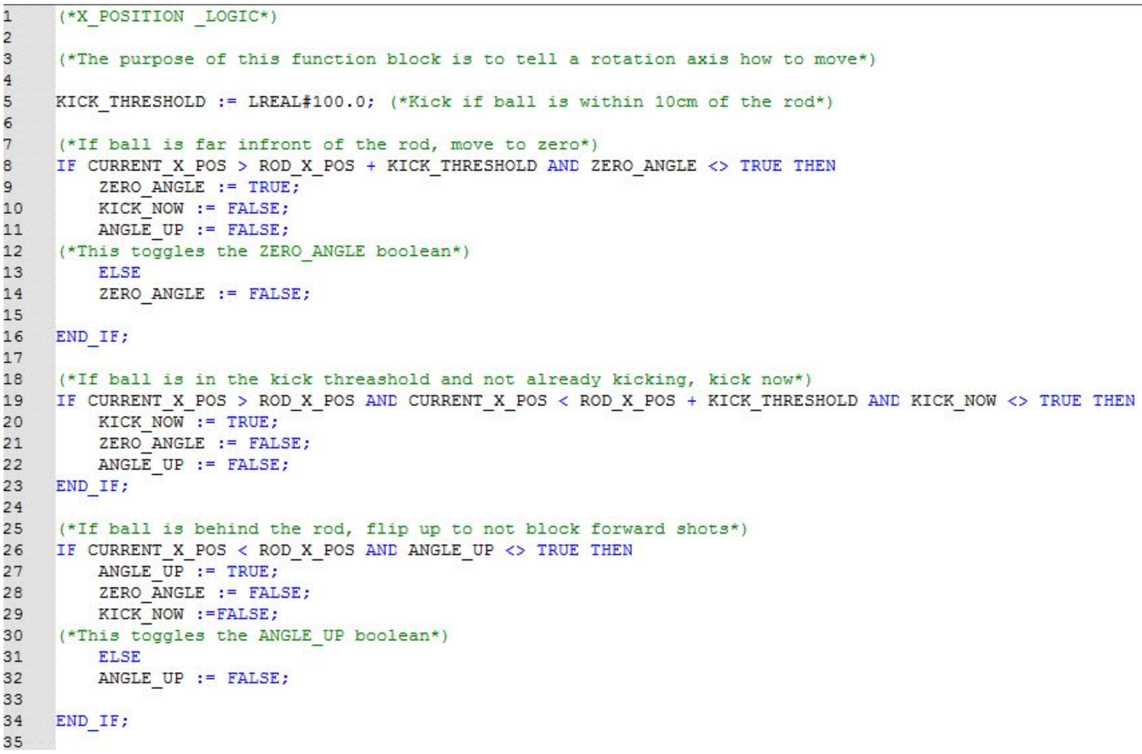
```
(*ROD_3_POSITION LOGIC*)

(*The purpose of this function block is to output the correct move distance so the
correct foosman blocks the ball*)

WALL_OFFSET := LREAL#45.0; (*Distance of center of foosman to wall at zero position*)
MAX_DIST := LREAL#116.0; (*Maximum distance rod can move*)
ZONE_WIDTH := LREAL#119.0; (*Width of each foosman zone*)
PLAYER_SPACING := LREAL#120.0; (*center to center distance between foosmen*)

(*If the ball has moved in the y-directon, perform calculation*)
IF PREV_Y_POS <> Y_POS THEN
(*If the ball is below the first foosman, move so that foosman is against the wall*)
    IF Y_POS <= WALL_OFFSET THEN

        MOV_DIST := LREAL#1.0;
(*If the ball is higher than the last foosman, move to the other wall*)
    ELSIF Y_POS >= (WALL_OFFSET + LREAL#5.0*ZONE_WIDTH) THEN

        MOV_DIST := MAX_DIST;
(*If the ball is in the first zone move so the first foosman blocks it*)
    ELSIF (Y_POS > WALL_OFFSET & Y_POS <= (WALL_OFFSET + ZONE_WIDTH)) THEN

        MOV_DIST := Y_POS - WALL_OFFSET;
(*If the ball is in the second zone move so the second foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#2.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - PLAYER_SPACING;
(*If the ball is in the third zone move so the third foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + LREAL#2.0*ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#3.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - LREAL#2.0*PLAYER_SPACING;
(*If the ball is in the fourth zone move so the fourth foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + LREAL#3.0*ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#4.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - LREAL#3.0*PLAYER_SPACING;
(*If the ball is in the fifth zone move so the fifth foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + LREAL#4.0*ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#5.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - LREAL#4.0*PLAYER_SPACING;

    END_IF;

    PREV_Y_POS := Y_POS; (*store position as previous position*)

    MAKE_MOV := TRUE; (*move*)

ELSE

    MAKE_MOV := FALSE; (*don't move*)

END_IF;
```

Rod Position Logic 4



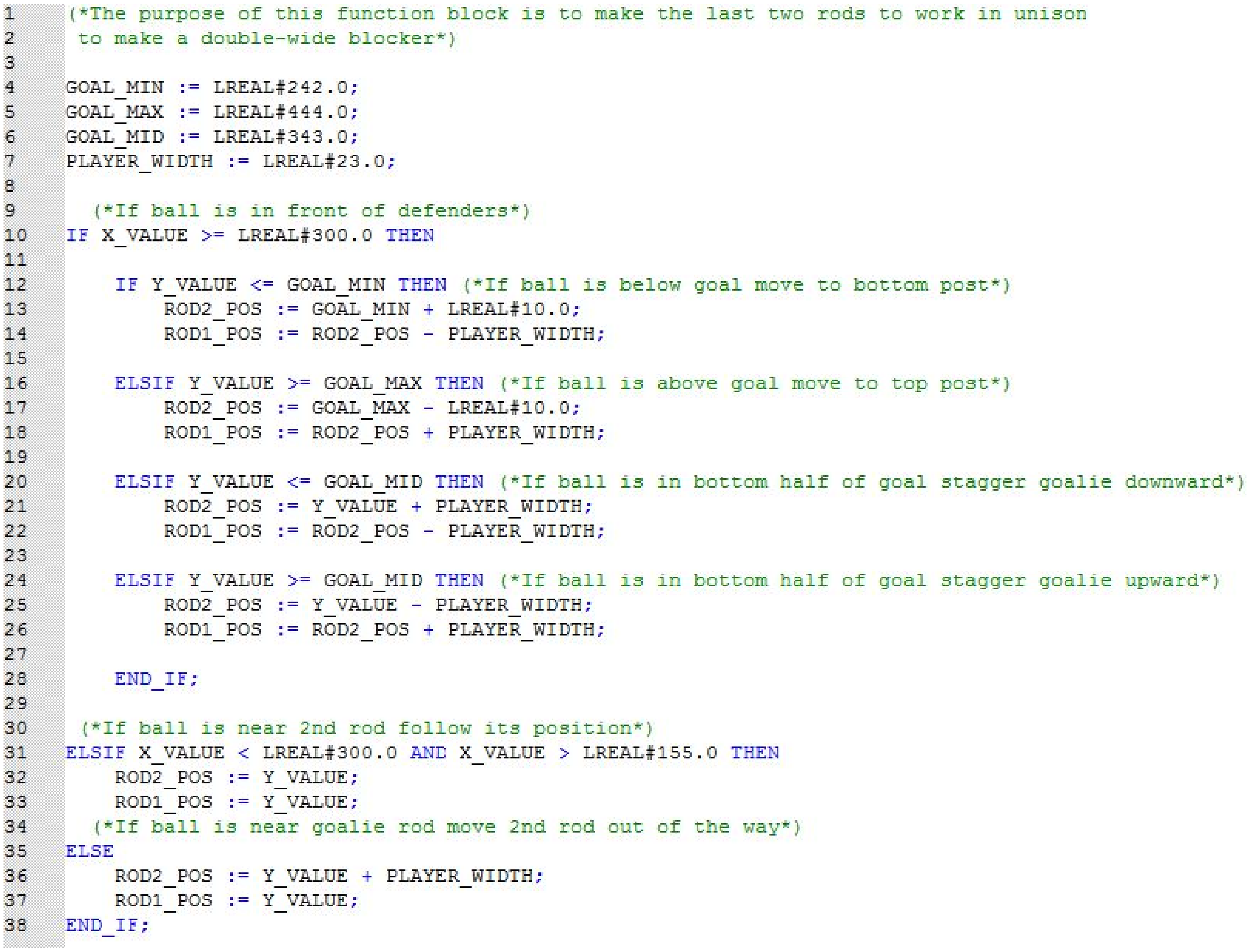
```
(*ROD_4_POSITION_LOGIC*)

(*The purpose of this function block is to output the correct move distance so the
correct foosman blocks the ball*)

WALL_OFFSET := LREAL#45.0; (*Distance of center of foosman to wall at zero position*)
MAX_DIST := LREAL#230.0; (*Maximum distance rod can move*)
ZONE_WIDTH := LREAL#199.0; (*Width of each foosman zone*)
PLAYER_SPACING := LREAL#184.0; (*center to center distance between foosmen*)

(*If the ball has moved in the y-directon, perform calculation*)
IF PREV_Y_POS <> Y_POS THEN
(*If the ball is below the first foosman, move so that foosman is against the wall*)
    IF Y_POS <= WALL_OFFSET THEN

        MOV_DIST := LREAL#1.0;
(*If the ball is higher than the last foosman, move to the other wall*)
    ELSIF Y_POS >= (WALL_OFFSET + LREAL#3.0*ZONE_WIDTH) THEN

        MOV_DIST := MAX_DIST;
(*If the ball is in the first zone move so the first foosman blocks it*)
    ELSIF (Y_POS > WALL_OFFSET & Y_POS <= (WALL_OFFSET + ZONE_WIDTH)) THEN

        MOV_DIST := Y_POS - WALL_OFFSET;
(*If the ball is in the second zone move so the second foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#2.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - PLAYER_SPACING;
(*If the ball is in the third zone move so the third foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + LREAL#2.0*ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#3.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - LREAL#2.0*PLAYER_SPACING;

    END_IF;

    PREV_Y_POS := Y_POS; (*store position as previous position*)

    MAKE_MOV := TRUE; (*move*)

ELSE

    MAKE_MOV := FALSE; (*don't move*)

END_IF;
```

Rotation Logic



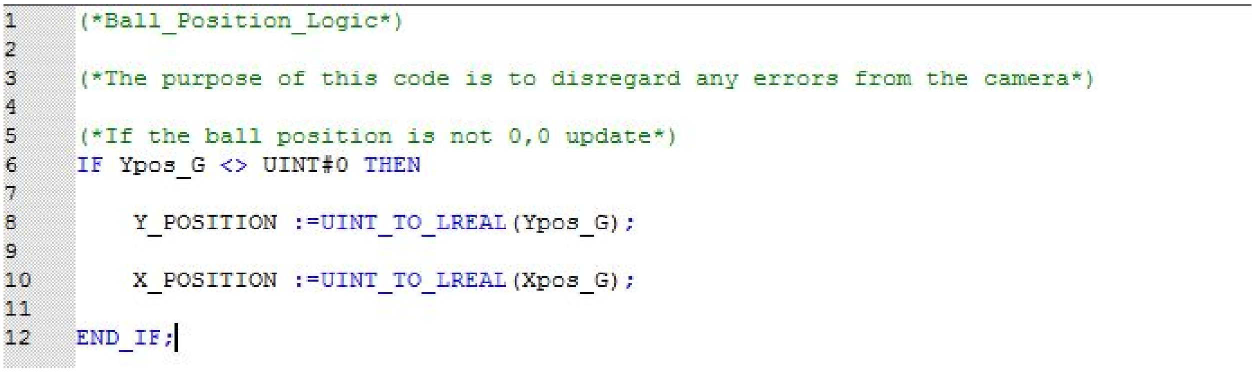
```
(*X_POSITION _LOGIC*)

(*The purpose of this function block is to tell a rotation axis how to move*)

KICK_THRESHOLD := LREAL#100.0; (*Kick if ball is within 10cm of the rod*)

(*If ball is far infront of the rod, move to zero*)
IF CURRENT_X_POS > ROD_X_POS + KICK_THRESHOLD AND ZERO_ANGLE <> TRUE THEN
    ZERO_ANGLE := TRUE;
    KICK_NOW := FALSE;
    ANGLE_UP := FALSE;
(*This toggles the ZERO_ANGLE boolean*)
    ELSE
    ZERO_ANGLE := FALSE;

END_IF;

(*If ball is in the kick threashold and not already kicking, kick now*)
IF CURRENT_X_POS > ROD_X_POS AND CURRENT_X_POS < ROD_X_POS + KICK_THRESHOLD AND KICK_NOW <> TRUE THEN
    KICK_NOW := TRUE;
    ZERO_ANGLE := FALSE;
    ANGLE_UP := FALSE;
END_IF;

(*If ball is behind the rod, flip up to not block forward shots*)
IF CURRENT_X_POS < ROD_X_POS AND ANGLE_UP <> TRUE THEN
    ANGLE_UP := TRUE;
    ZERO_ANGLE := FALSE;
    KICK_NOW :=FALSE;
(*This toggles the ANGLE_UP boolean*)
    ELSE
    ANGLE_UP := FALSE;

END_IF;
```

Defense Logic



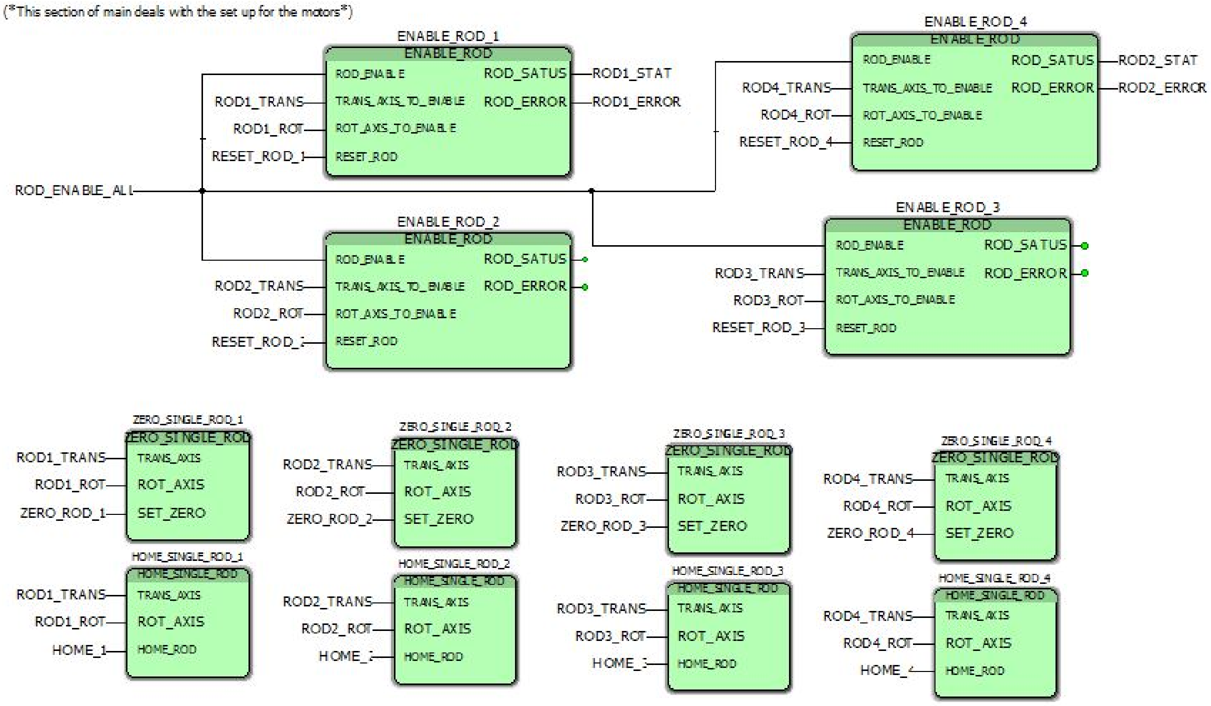
```
(*The purpose of this function block is to make the last two rods to work in unison
 to make a double-wide blocker*)

GOAL_MIN := LREAL#242.0;
GOAL_MAX := LREAL#444.0;
GOAL_MID := LREAL#343.0;
PLAYER_WIDTH := LREAL#23.0;

  (*If ball is in front of defenders*)
IF X_VALUE >= LREAL#300.0 THEN

    IF Y_VALUE <= GOAL_MIN THEN (*If ball is below goal move to bottom post*)
        ROD2_POS := GOAL_MIN + LREAL#10.0;
        ROD1_POS := ROD2_POS - PLAYER_WIDTH;

    ELSIF Y_VALUE >= GOAL_MAX THEN (*If ball is above goal move to top post*)
        ROD2_POS := GOAL_MAX - LREAL#10.0;
        ROD1_POS := ROD2_POS + PLAYER_WIDTH;

    ELSIF Y_VALUE <= GOAL_MID THEN (*If ball is in bottom half of goal stagger goalie downward*)
        ROD2_POS := Y_VALUE + PLAYER_WIDTH;
        ROD1_POS := ROD2_POS - PLAYER_WIDTH;

    ELSIF Y_VALUE >= GOAL_MID THEN (*If ball is in bottom half of goal stagger goalie upward*)
        ROD2_POS := Y_VALUE - PLAYER_WIDTH;
        ROD1_POS := ROD2_POS + PLAYER_WIDTH;

    END_IF;

 (*If ball is near 2nd rod follow its position*)
ELSIF X_VALUE < LREAL#300.0 AND X_VALUE > LREAL#155.0 THEN
    ROD2_POS := Y_VALUE;
    ROD1_POS := Y_VALUE;
  (*If ball is near goalie rod move 2nd rod out of the way*)
ELSE
    ROD2_POS := Y_VALUE + PLAYER_WIDTH;
    ROD1_POS := Y_VALUE;
END_IF;
```

Ball Position Logic

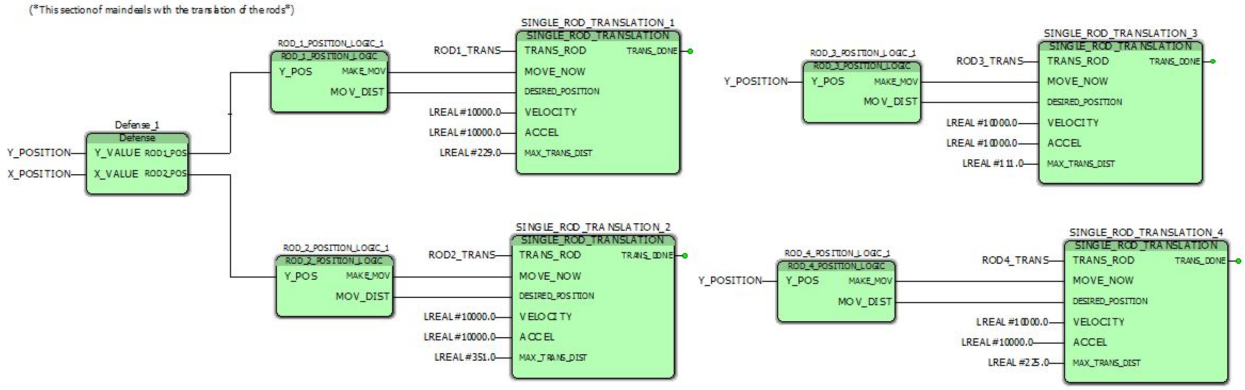


```
(*Ball_Position_Logic*)

(*The purpose of this code is to disregard any errors from the camera*)

(*If the ball position is not 0,0 update*)
IF Ypos_G <> UINT#0 THEN

    Y_POSITION :=UINT_TO_LREAL(Ypos_G);

    X_POSITION :=UINT_TO_LREAL(Xpos_G);

END_IF;
```

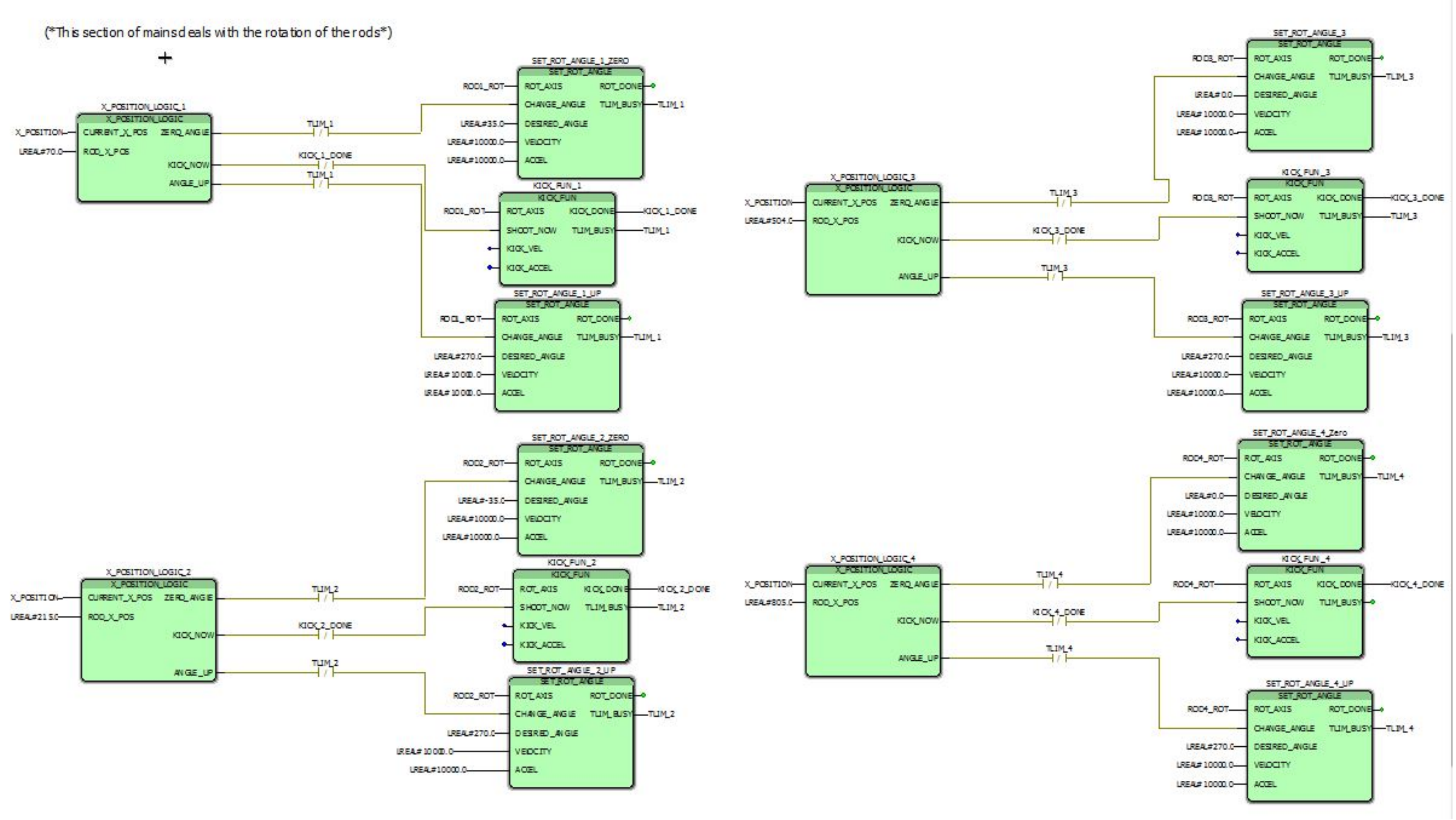
Main – Rod Setup



Main – Lateral Position Logic



Main – Rotational Logic



(*This section of mainsdeals with the rotation of the rods*)

Appendix L: User Manual

Foosball System Assembly

The following section contains the procedure for assembling the automated foosball system from a show-ready disassembly state. This section does not address the procedure used to prepare the assembled system for play.

STEP 1:

The motor cabinet is placed, using a pallet jack, into its desired position. The ground on which the cabinet is placed should be level.

STEP 2:

The foosball table is then placed between the brackets on the motor cabinet, oriented so the human player rods face outwards. The leg of the foosball table should be snuggly fit against the L brackets on the motor cabinet.

STEP 3:

The ½ in. holes in the legs of the foosball table should be aligned with the ½ in. holes in the brackets on the motor cabinet using the leveling feet on the foosball table. DO NOT ADJUST THE POSITION OF THE BRACKETS ON THE MOTOR CABINET.

STEP 4:

Place the shims between their respective plate brackets and place the short ½ in. bolts in the top plate bracket and the long ½ in. bolts in the lower plate bracket, ensuring a washer is inserted between the bolts and the brackets.

STEP 5:

Secure the bolts to the table using a washer and nut on the inside of the table lets. Use the pair of adjustable wrenches to tighten the bolt.

STEP 6:

Place the short ½ in. bolts in the top L bracket and the long ½ in. bolts in the lower L bracket, ensuring a washer is inserted between the bolts and the brackets.

STEP 7:

Secure the bolts to the table using a washer and nut on the inside of the table lets. Use the pair of adjustable wrenches to tighten the bolt.

STEP 8:

The human and computer controlled foosman rods are then inserted into the table and their bearings secured.

STEP 9:

The motors and rods are then connected by inserting the rods into the couplers attached to the motor. The coupler's bolts on the rod side are then tightened using a 3mm Hex key.

STEP 10:

Rotate the T-nut in the vision arch brackets so that the vision arch up rights can be inserted over the nuts.

STEP 11:

Place the vision arch up rights into the vision arch brackets from the front of the bracket.

STEP 12:

Tighten the vision arch bolts, using an adjustable wrench to secure the uprights into place.

STEP 13:

Place the vision arch crossbeam onto the 90 degree T-Slot braces on the vision arch uprights and secure the crossbeam in place using M8 T-bolts and an M13 socket wrench.

STEP 14:

Position the scoreboard on the vision arch crossbeam, using the markings on the beam as a guide for positioning. Secure the scoreboard in place using an M13 socket wrench.

STEP 15:

Insert the gap cover through the motor cabinet so that the gap is completely covered and the vertical portion of the cover rests in the T-Slot of the motor cabinet. Secure the cover in place using M8 T-bolts and a M13 socket wrench.

STEP 16:

Place the playfield cover on the top of the table such that the cover is square and secured to the table top using the Velcro strips on the table and cove.

STEP 17:

Assemble the roof (SEE ROOF ASSEMBLY SECTION).

STEP 18:

Place the roof on the brackets at the tops of the vision arch uprights and secure it in place using an M8 T-bolt and a M13 socket wrench. Use marks on the roof to center it over the table.

STEP 19:

Remove the sensor protector from the camera and screw in the lens.

STEP 20:

Connect the power and Ethernet cables to the camera and secure the cables to the vision arch upright on the side of the assembly with the L brackets.

STEP 21:

Place the power strip under the foosball table and connect it to a plug.

STEP 22:

Plug the PLC, Lighting and Camera into the power strip.

STEP 23:

Place the computer system and router on the side of the assembly with the L brackets.

STEP 24:

Plug the computer and router into the power strip.

STEP 25:

Connect the camera, PLC and computer Ethernet cables into the router.

STEP 26:

Connect the computer to the internet via Ethernet.

STEP 27:

Turn on the power strip.

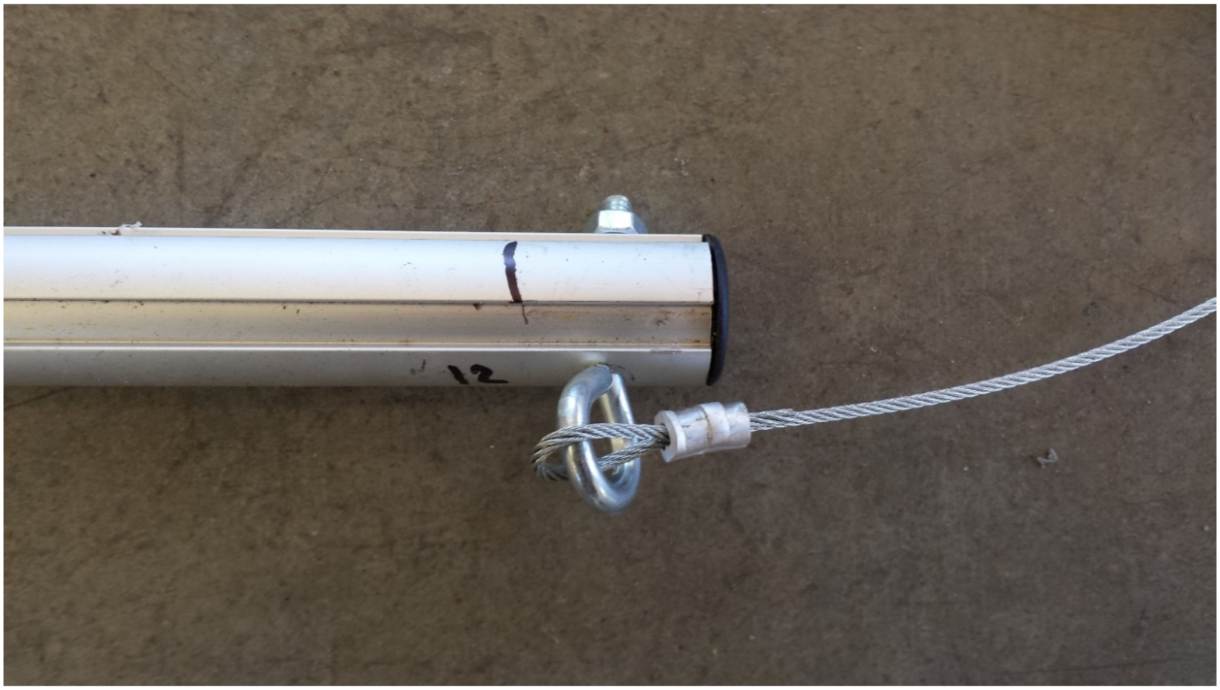
STEP 28:

The System is now ready to operate.

Roof Assembly

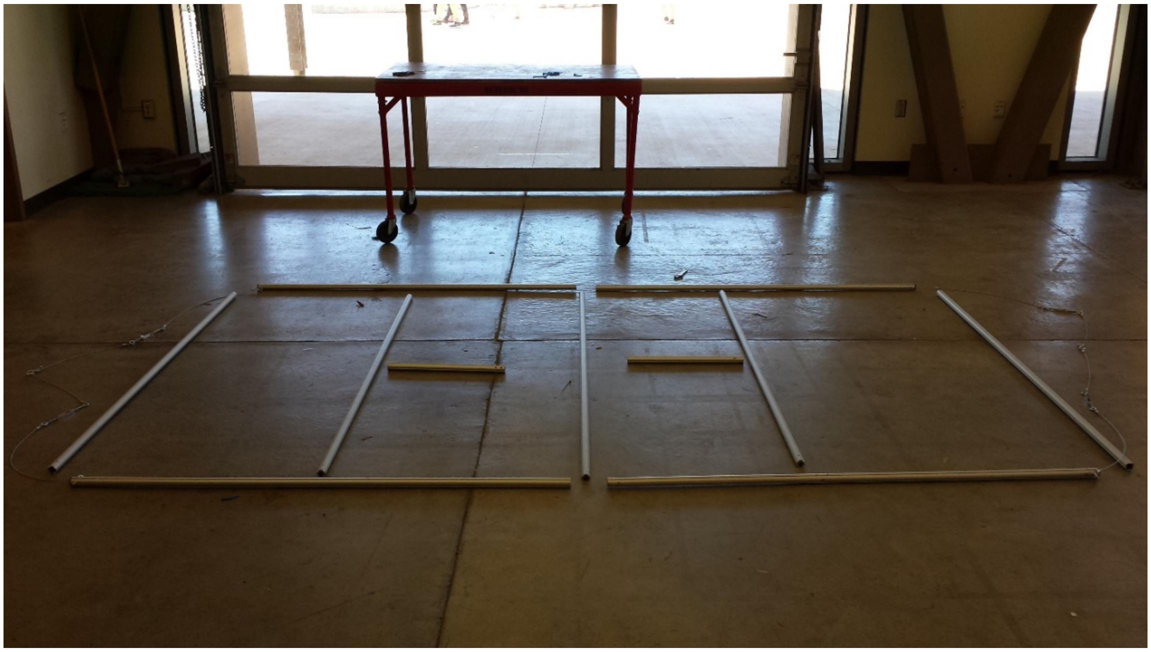
Step 1:

Attach the eyehooks of the wire rope sections with two turnbuckles to the ends of the rods with black end caps.



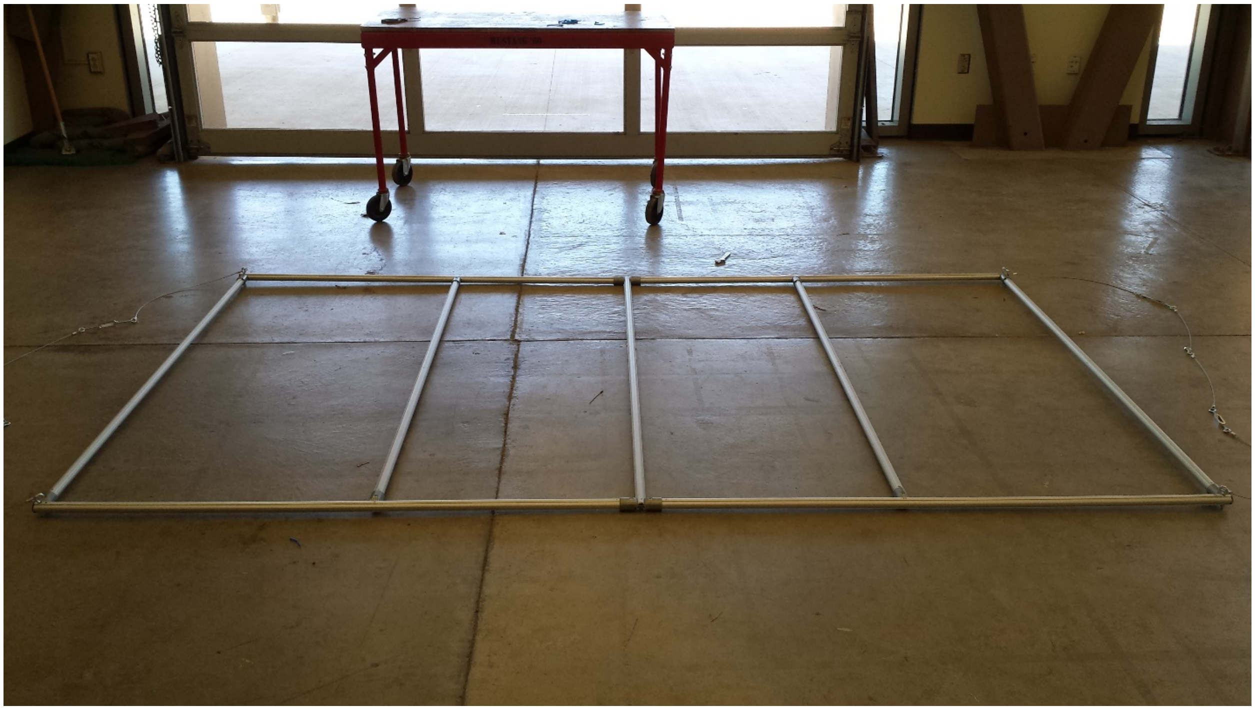
Step 2:

Lay out the rods in the correct shape by lining up the numbered joints.



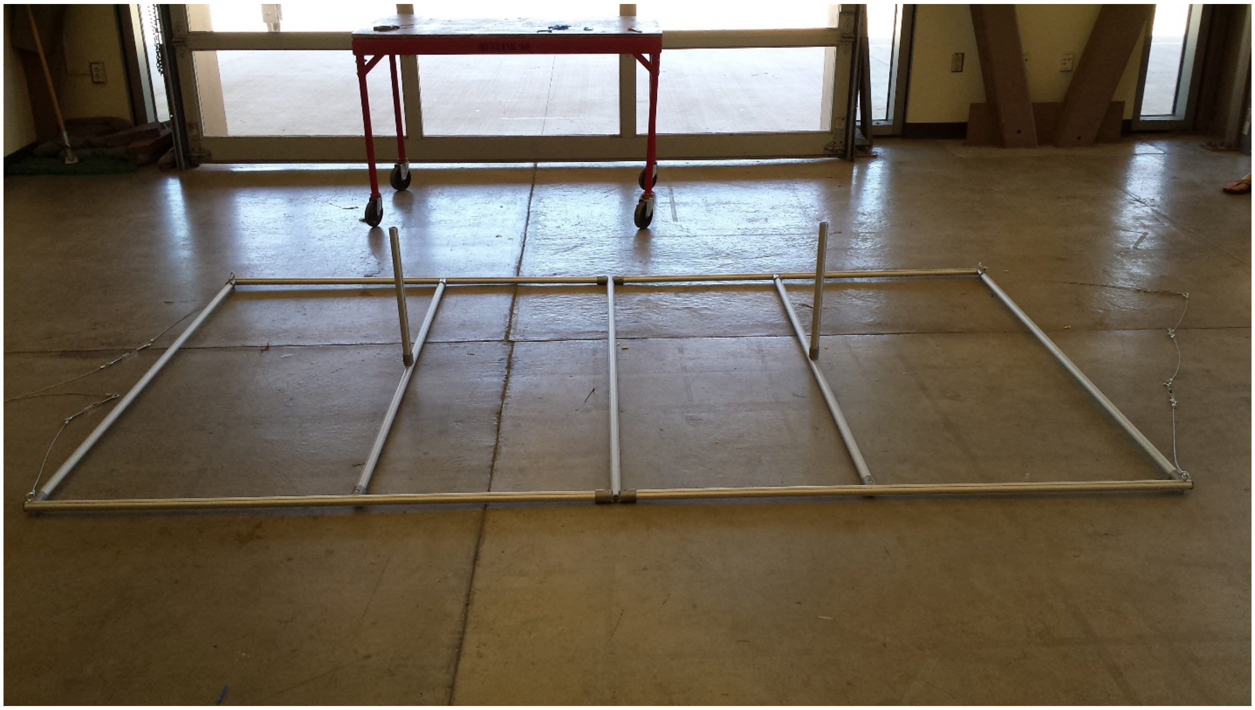
Step 3:

Attach the joints using 90° connectors and a 5mm allen wrench. Black lines show where the connectors should sit.



Step 4:

Attach the vertical uprights by lining up the connectors with the black lines.



Step 5:

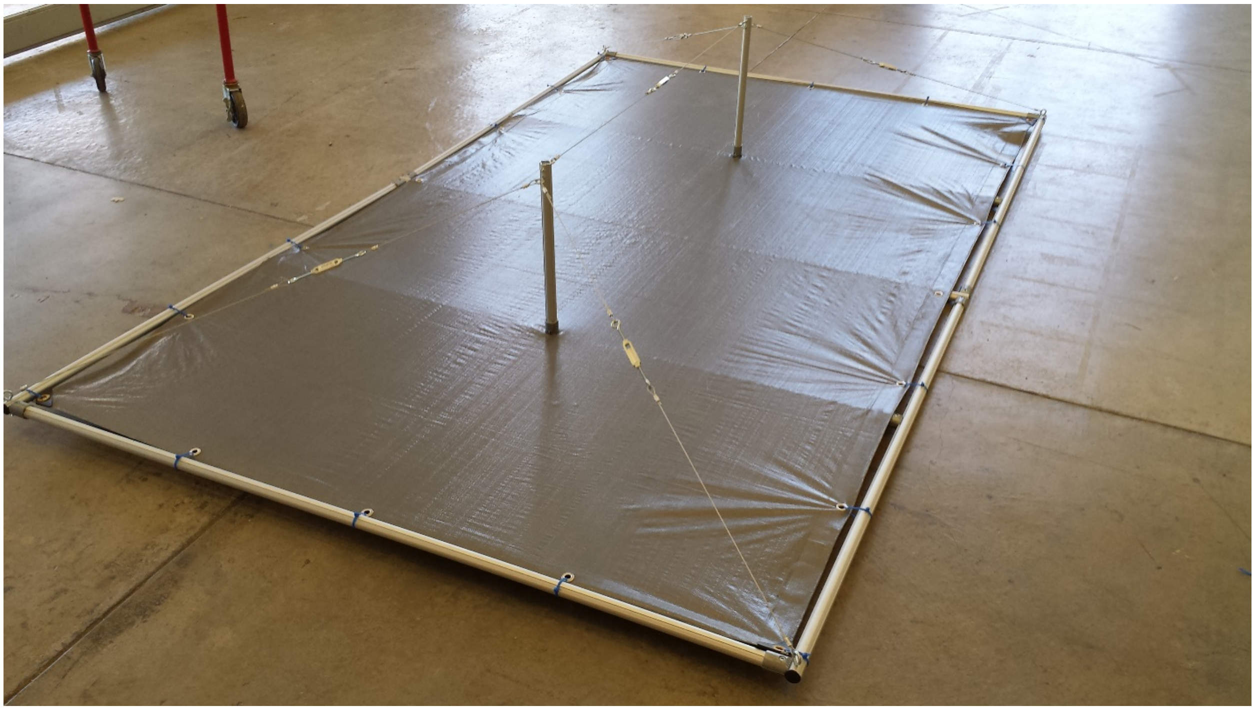
Align the holes in the tarp with the uprights and slide the tarp downward so it is flush with the frame.

Step 6:

Use zip ties to hold the tarp taught against the frame.

Step 7:

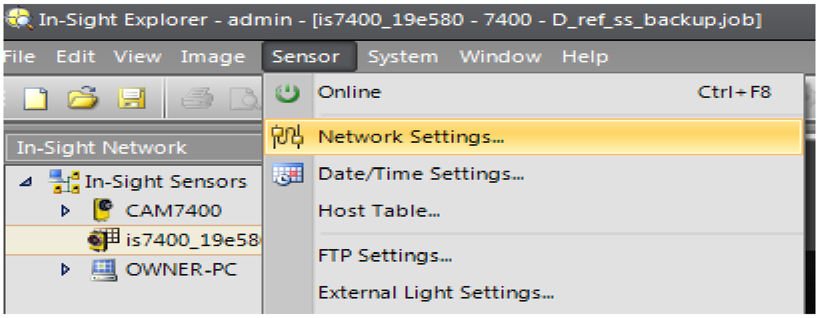
Bolt the middle eyehook to the lower hole of each upright and attach the third wire rope assembly to the top holes of the uprights. Finally tighten the turnbuckles evenly so the roof bends up slightly at the corners.



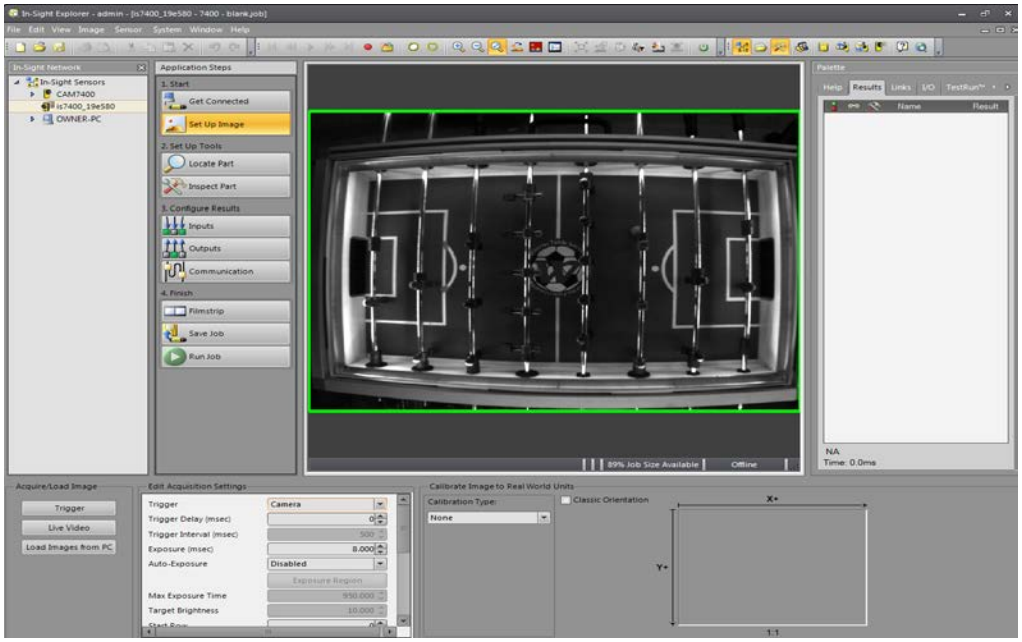
Basic Camera Setup

This procedure is for setting up an EIP networking configuration for the Cognex In-sight version 4.9 and Yaskawa's MotionWorks IEC 2 Pro version 2.4 software. The system at the time of this writing utilized an is7400 camera (Cognex) to communicate with an MP3200iec PLC (Yaskawa). This how to assumes that you will be using the In-sight Easybuild mode.

Step 1: Open and connect to the sensor in In-Sight Explorer. Network settings for the camera can be adjust under 'sensors' once In-sight detects auto-detects the device.



Step 2: Create a new Job, and in the 'Setup Up Image' define the desired trigger method. You may revisit the 'trigger interval' time once your job is running and you have run time values



Edit Acquisition Settings

Trigger: Camera

Trigger Delay (msec): 0

Trigger Interval (msec): 500

Exposure (msec): 8.000

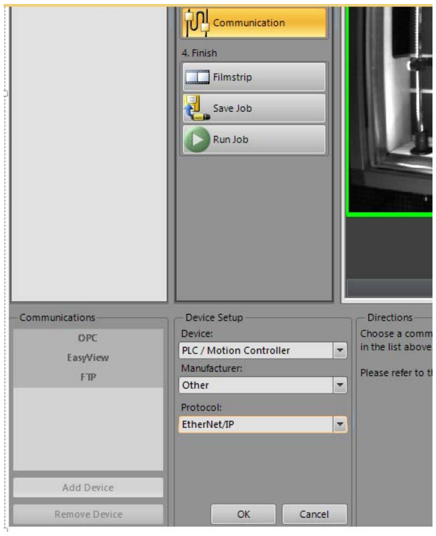
Auto-Exposure: Disabled

Max Exposure Time: 950.000

Target Brightness: 10.000

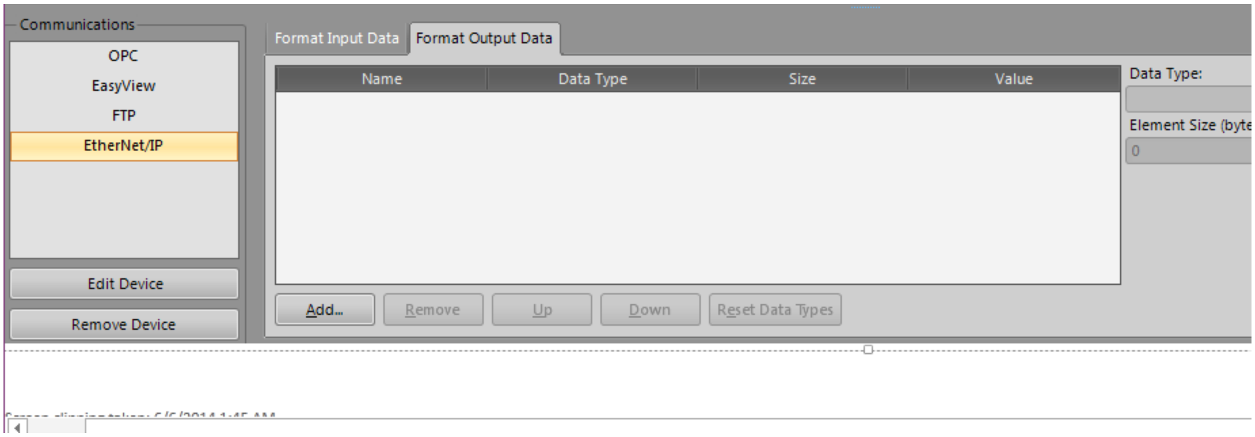
Calibration Type: None

Step 3: Under the Communication area, add a new device and select the appropriate settings

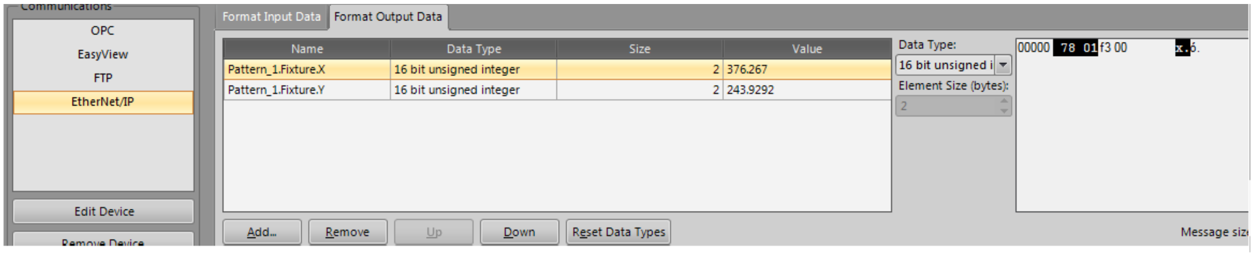


Device: PLC / Motion Controller
Manufacturer: Other
Protocol: EtherNet/IP

Step 4: Once you have added the Ethernet/IP device you will be able to access the inputs and outputs tabs to select Job variables you may desire to communicate to the PLC.



In this image the values displayed in the box on the far right are the hex values of our coordinates. These are the values that will show up in the Global memory allocated for the camera within MotionWorks, after we create the profile for the PLC.

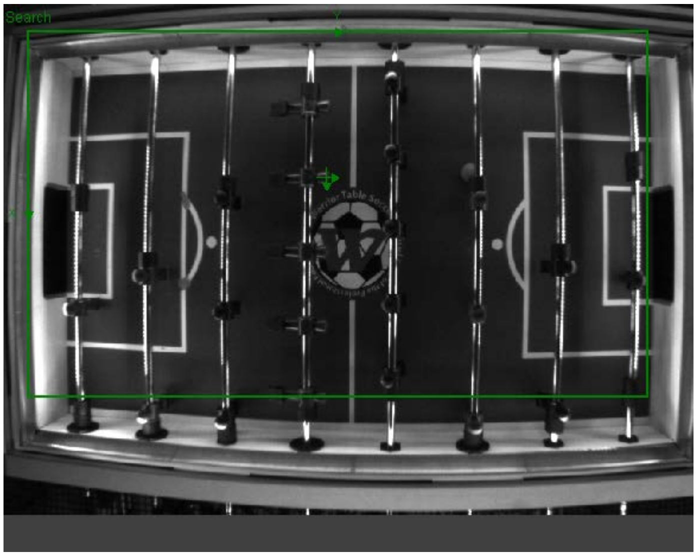


Name	Data Type	Size	Value
Pattern_1.Fixture.X	16 bit unsigned integer	2	376.267
Pattern_1.Fixture.Y	16 bit unsigned integer	2	243.9292

Data Type: 16 bit unsigned i
Element Size (bytes): 2

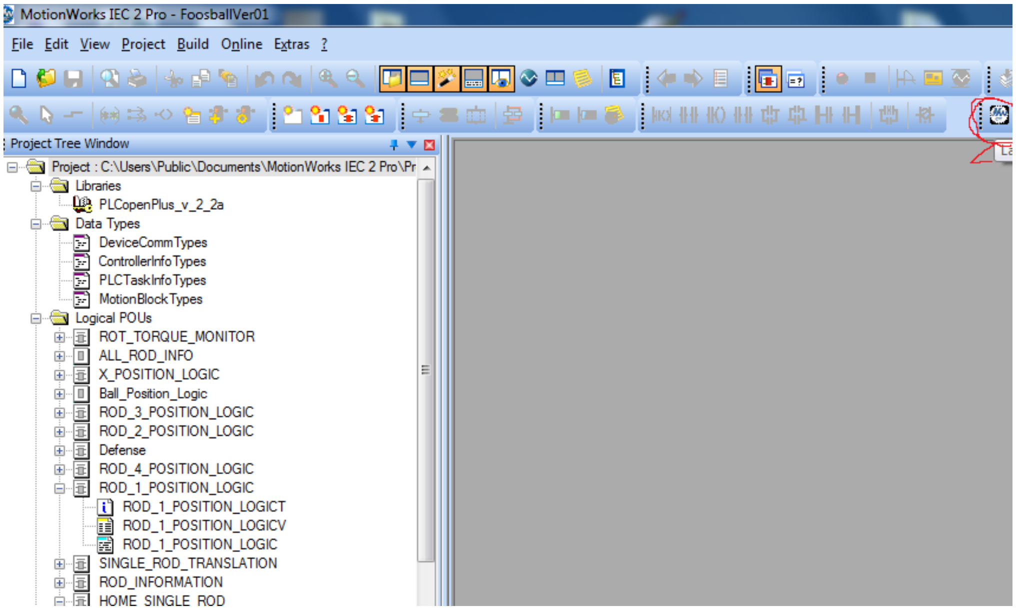
```
00000  78 01 f3 00          x.ó.
```

Step 5: Now create a job and have fun. This is an image of what you can expect to see after you have defined the search areas. The outer green box is the search region we designated for the pattern search, and the ball is indicated with the arrow since that portion of pixels more closely matches the defined model than does the partially obscured ball (on the right, mirroring).

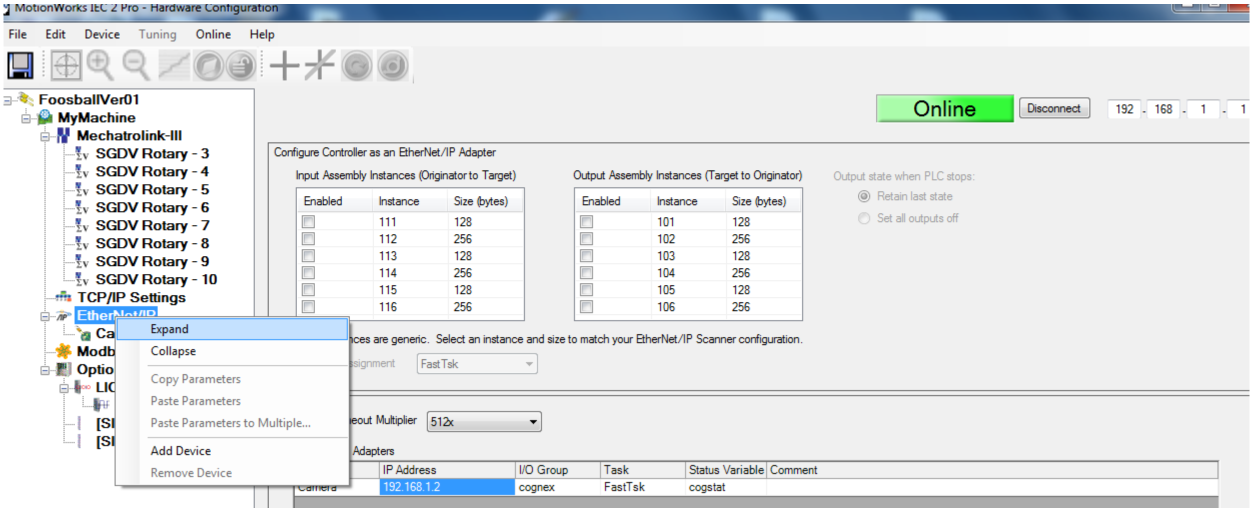


Step 6: At this point the camera is setup, once you save the job to the camera and select the option to make it the ‘run on start’ job, the next time you power the camera that will be the job that loads. Alternately, you can ‘go online’ with the camera in which case the job will start running then. Note that changes to parameters cannot be made while the camera is ‘online’.

Step 7: Open Yaskawa MotionWorks and open the Hardware Configuration editor.



Step 8: Once the hardware configuration editor is up, on the left, select the Ethernet/IP tab and add a new device for the camera.



Configure Controller as an EtherNet/IP Adapter

Input Assembly Instances (Originator to Target)

Enabled	Instance	Size (bytes)
☐	111	128
☐	112	256
☐	113	128
☐	114	256
☐	115	128
☐	116	256

Output Assembly Instances (Target to Originator)

Enabled	Instance	Size (bytes)
☐	101	128
☐	102	256
☐	103	128
☐	104	256
☐	105	128
☐	106	256

Output state when PLC stops:
● Retain last state
○ Set all outputs off

nces are generic. Select an instance and size to match your EtherNet/IP Scanner configuration.

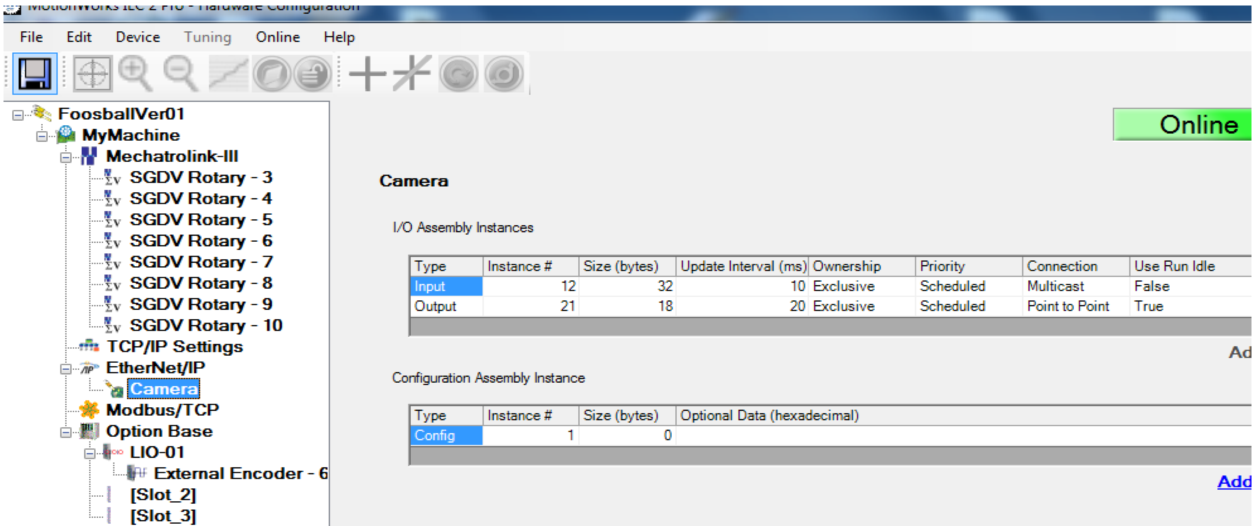
ssignment FastTsk

eout Multiplier 512x

Adapters

	IP Address	I/O Group	Task	Status Variable	Comment
Camera	192.168.1.2	cognex	FastTsk	cogstat	

Step 9: Once your device is created, (ours was called camera) select it, add the input, and output assembly instances. For the in7000 series we found that aside from keeping the custom instance sizes below 495 (+/- 5) bytes you should have no problems. Note that we were unable to use the default instance options seen in the image above because the ins7000 instances for EIP are not configurable. In addition, the 'Configuration Assembly Instance' must be configured as shown below. Refer to the In-sight help files for further guidance.



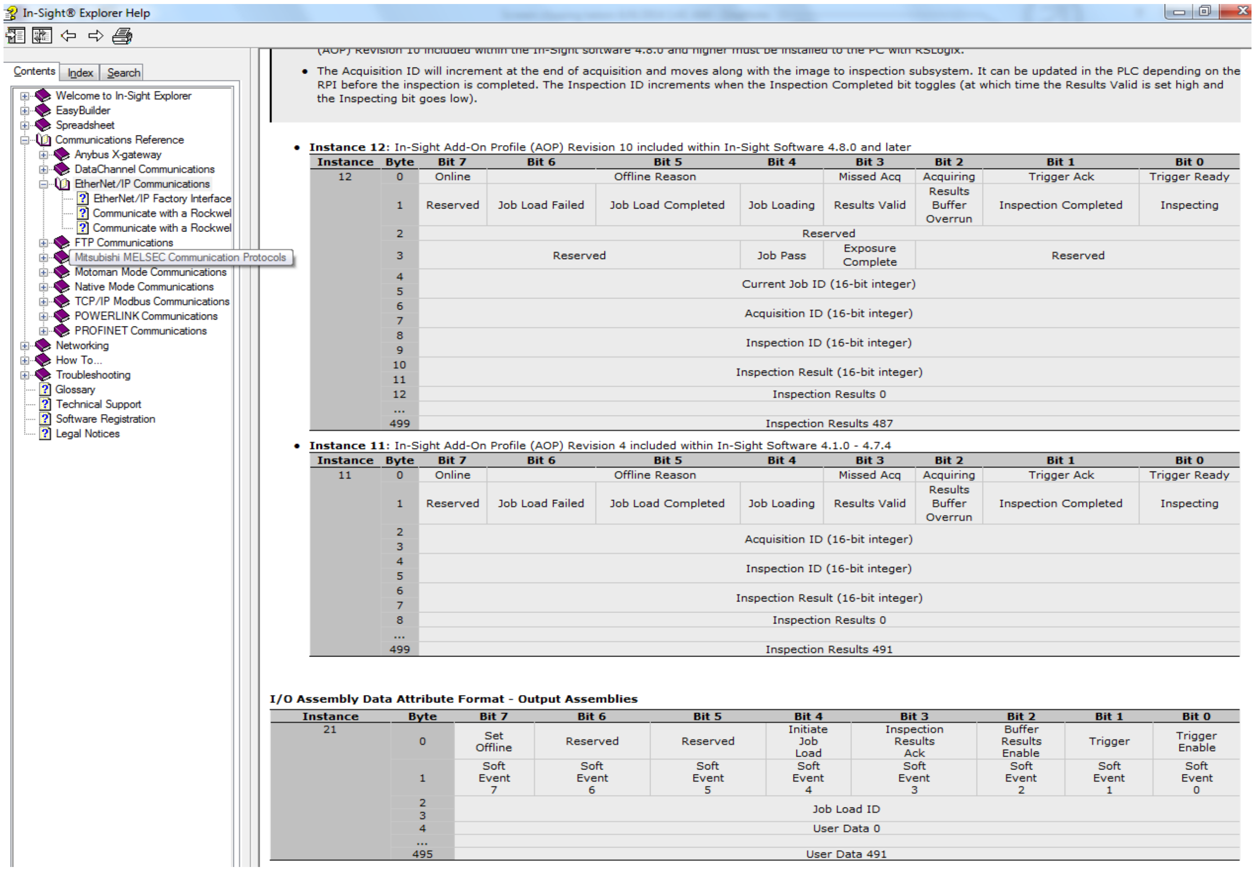
Camera

I/O Assembly Instances

Type	Instance #	Size (bytes)	Update Interval (ms)	Ownership	Priority	Connection	Use Run Idle
Input	12	32	10	Exclusive	Scheduled	Multicast	False
Output	21	18	20	Exclusive	Scheduled	Point to Point	True

Configuration Assembly Instance

Type	Instance #	Size (bytes)	Optional Data (hexadecimal)
Config	1	0	



(AOP) Revision 10 included within the In-Sight software 4.8.0 and higher must be installed to the PC with RSLogix.

- The Acquisition ID will increment at the end of acquisition and moves along with the image to inspection subsystem. It can be updated in the PLC depending on the RPI before the inspection is completed. The Inspection ID increments when the Inspection Completed bit toggles (at which time the Results Valid is set high and the Inspecting bit goes low).

- **Instance 12:** In-Sight Add-On Profile (AOP) Revision 10 included within In-Sight Software 4.8.0 and later

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
12	0	Online	Offline Reason			Missed Acq	Acquiring	Trigger Ack	Trigger Ready
	1	Reserved	Job Load Failed	Job Load Completed	Job Loading	Results Valid	Results Buffer Overrun	Inspection Completed	Inspecting
	2	Reserved							
	3	Reserved			Job Pass	Exposure Complete	Reserved		
	4 5	Current Job ID (16-bit integer)							
	6 7	Acquisition ID (16-bit integer)							
	8 9	Inspection ID (16-bit integer)							
	10 11	Inspection Result (16-bit integer)							
	12	Inspection Results 0							
	...								
	499	Inspection Results 487							

- **Instance 11:** In-Sight Add-On Profile (AOP) Revision 4 included within In-Sight Software 4.1.0 - 4.7.4

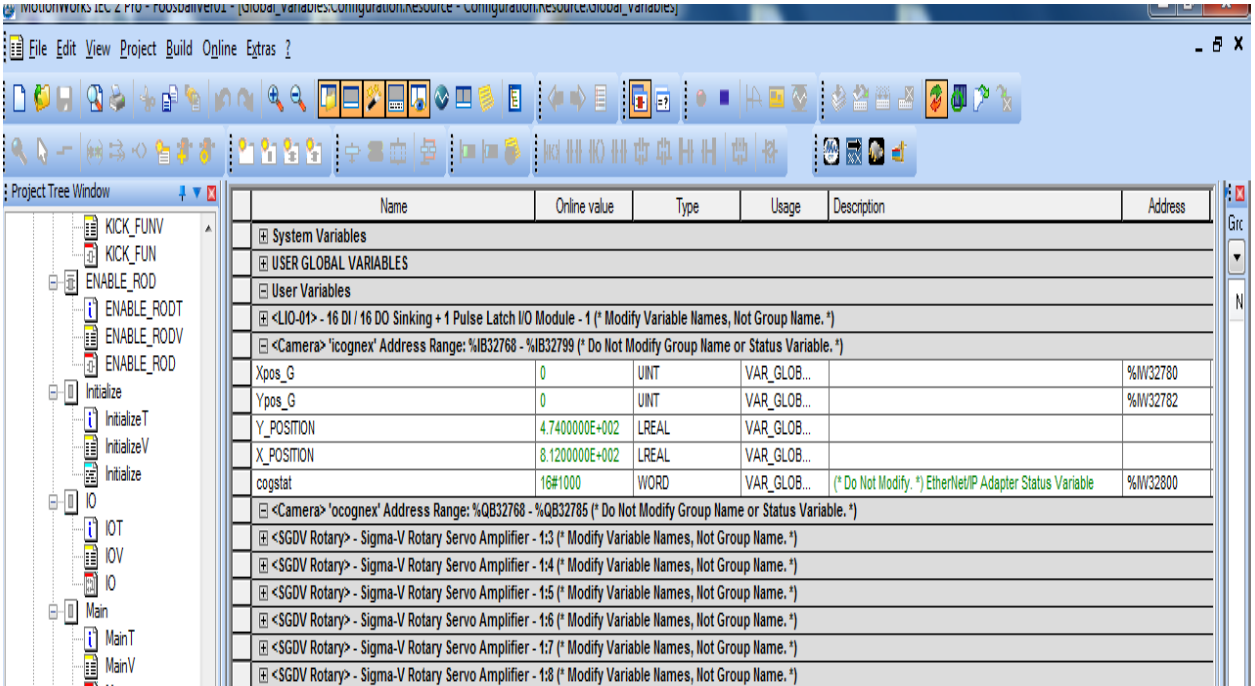
Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
11	0	Online	Offline Reason			Missed Acq	Acquiring	Trigger Ack	Trigger Ready
	1	Reserved	Job Load Failed	Job Load Completed	Job Loading	Results Valid	Results Buffer Overrun	Inspection Completed	Inspecting
	2 3	Acquisition ID (16-bit integer)							
	4 5	Inspection ID (16-bit integer)							
	6 7	Inspection Result (16-bit integer)							
	8	Inspection Results 0							
	...								
	499	Inspection Results 491							

I/O Assembly Data Attribute Format - Output Assemblies

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
21	0	Set Offline	Reserved	Reserved	Initiate Job Load	Inspection Results Ack	Buffer Results Enable	Trigger	Trigger Enable
	1	Soft Event 7	Soft Event 6	Soft Event 5	Soft Event 4	Soft Event 3	Soft Event 2	Soft Event 1	Soft Event 0
	2 3	Job Load ID							
	4	User Data 0							
	...								
	495	User Data 491							

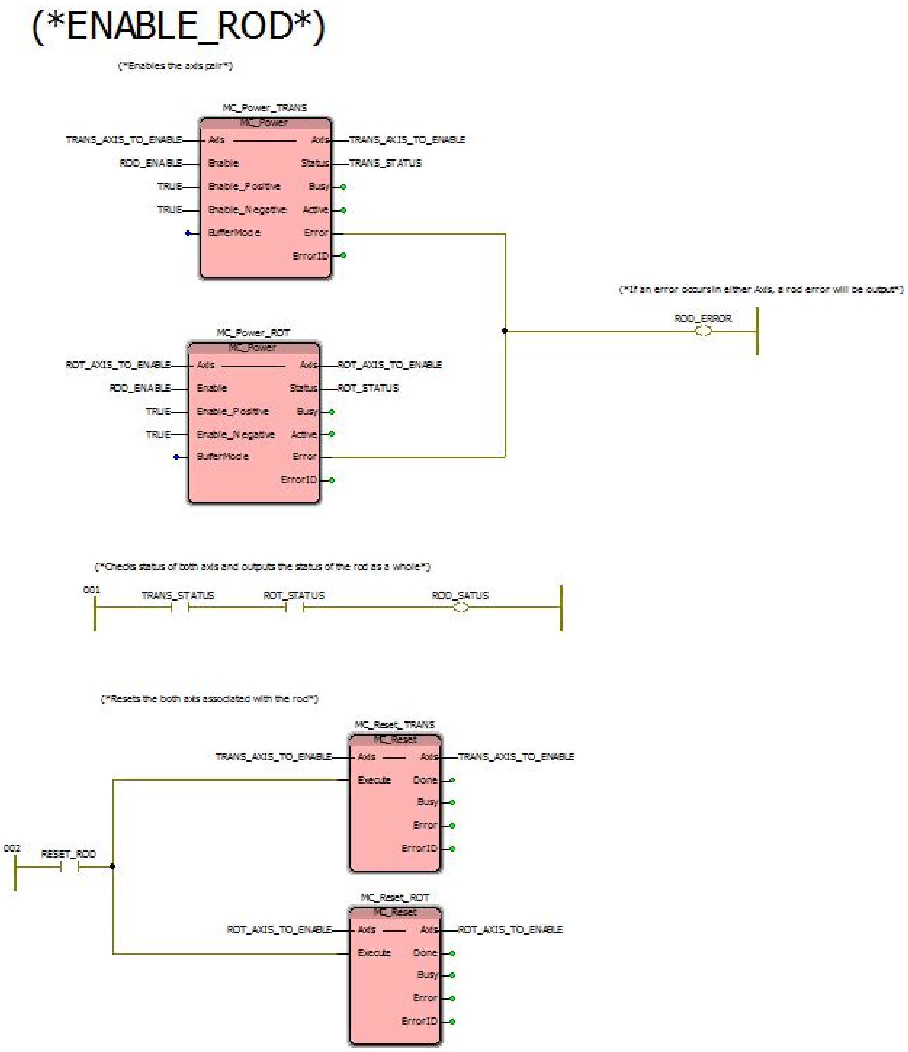
Step 10: Now that the Camera is setup in the hardware configuration, Save the profile and close the hardware editor. Return to MotionWorks and under the Global Variables tab you will have a block of memory allocated to the size of the instances you have selected. The only variable initially configured is the Status Variable.

At this point once the PLC and camera are running. You can go into the debug PLC program debug and see the status code in the Status Variable. x1000 indicates that the device is connected. Other status codes can be found in the Yaskawa documentation.



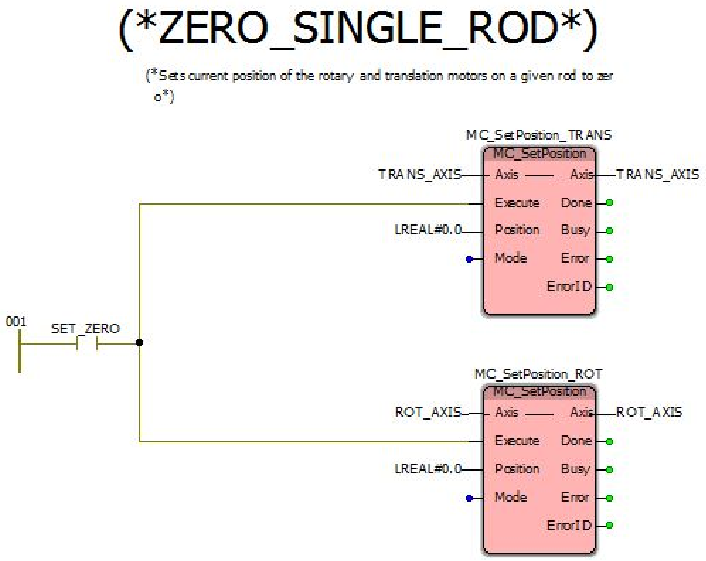
Name	Online value	Type	Usage	Description	Address
⊞ System Variables					
⊞ USER GLOBAL VARIABLES					
⊟ User Variables					
⊞ <LIO-01> - 16 DI / 16 DO Sinking + 1 Pulse Latch I/O Module - 1 (* Modify Variable Names, Not Group Name. *)					
⊟ <Camera> 'icognex' Address Range: %IB32768 - %IB32799 (* Do Not Modify Group Name or Status Variable. *)					
Xpos_G	0	UINT	VAR_GLOB...		%IW32780
Ypos_G	0	UINT	VAR_GLOB...		%IW32782
Y_POSITION	4.7400000E+002	LREAL	VAR_GLOB...		
X_POSITION	8.1200000E+002	LREAL	VAR_GLOB...		
cogstat	16#1000	WORD	VAR_GLOB...	(* Do Not Modify. *) EtherNet/IP Adapter Status Variable	%IW32800
⊟ <Camera> 'ocognex' Address Range: %QB32768 - %QB32785 (* Do Not Modify Group Name or Status Variable. *)					
⊞ <SGDV Rotary> - Sigma-V Rotary Servo Amplifier - 1:3 (* Modify Variable Names, Not Group Name. *)					
⊞ <SGDV Rotary> - Sigma-V Rotary Servo Amplifier - 1:4 (* Modify Variable Names, Not Group Name. *)					
⊞ <SGDV Rotary> - Sigma-V Rotary Servo Amplifier - 1:5 (* Modify Variable Names, Not Group Name. *)					
⊞ <SGDV Rotary> - Sigma-V Rotary Servo Amplifier - 1:6 (* Modify Variable Names, Not Group Name. *)					
⊞ <SGDV Rotary> - Sigma-V Rotary Servo Amplifier - 1:7 (* Modify Variable Names, Not Group Name. *)					
⊞ <SGDV Rotary> - Sigma-V Rotary Servo Amplifier - 1:8 (* Modify Variable Names, Not Group Name. *)					

This function block enables both motors on the given rod. It allows the rod to be enabled or disabled, and for the rod to be reset. It also informs the user if the rod was successfully enabled and if the motors have encountered an error during operation.



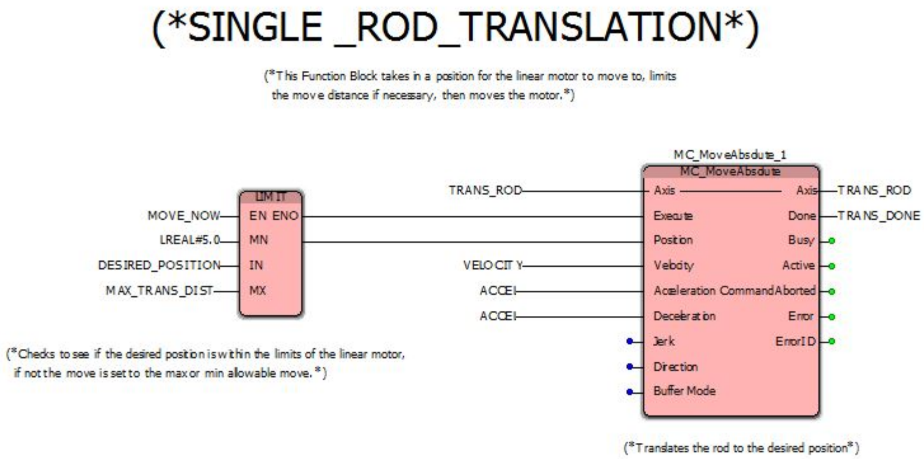
Zero Single Rod:

The purpose of this function block is to set the zero position for both the translational and rotational motors on the given rod. As there are no limit switches on motors, it is necessary to place the rods in the zero positions by hand. The rods should be zeroed after an alarm, an unexpected power down or the system, or if an error in position is noticed by the operator.



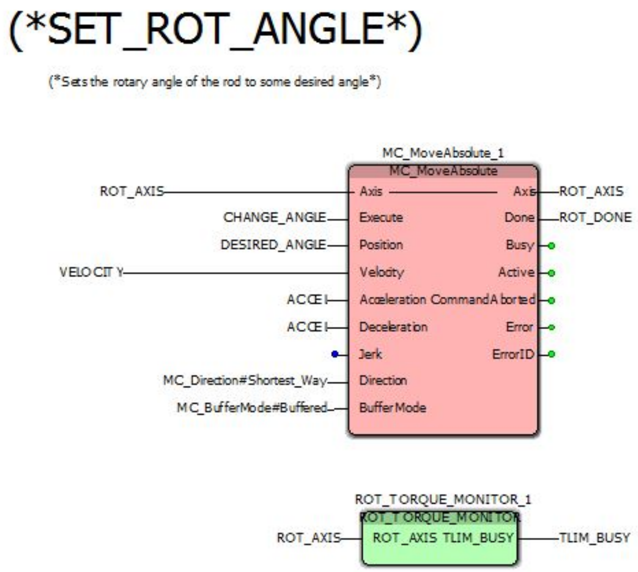
Single Rod Translation:

This function block is used to translate the given rod to a desired position within the physical limits of the system. The block accepts an input in the form of a real position. It then determines if the given position is within the minimum and maximum movement range and if it is not the result is saturated at the appropriate extreme. After the position is accepted or saturated, the move command is issued to the motor to translate to the desired location.



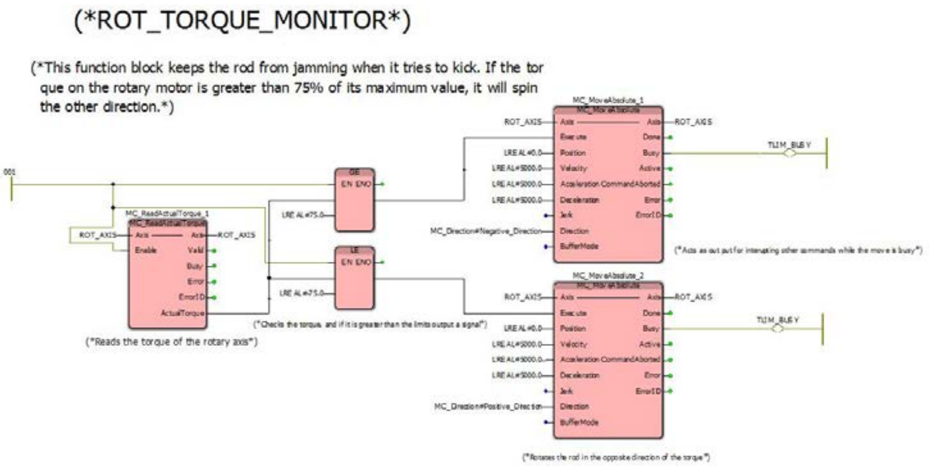
Set Rod Angle:

This function is used to set the angle of the rotational motor of a given rod. The desired angle is input into the function block and the move is executed. The direction taken by the motor is shortest route from its current position to the desired position. The block includes torque monitoring, which will prevent the rotary motor from overloading if the ball is caught beneath the foosman.



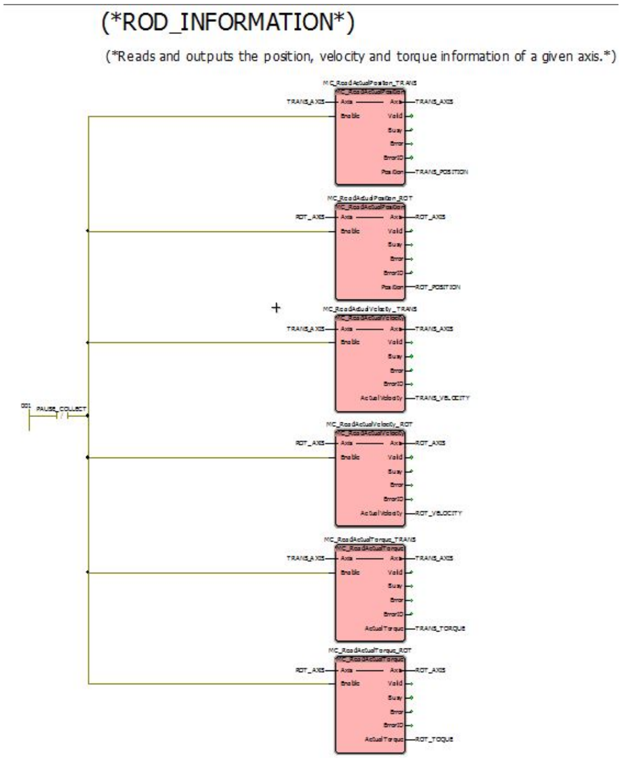
Rotational Torque Monitor:

This function block prevents the rotational motor from overloading during a move. The block is constantly monitoring the torque of the given motor, and in the event that the torque reaches the set limit, the block rotates the motor in the direction opposite to the increasing torque. It outputs a signal which can be used to prevent any other actions to be taken on the rod until the unjamming move is complete.



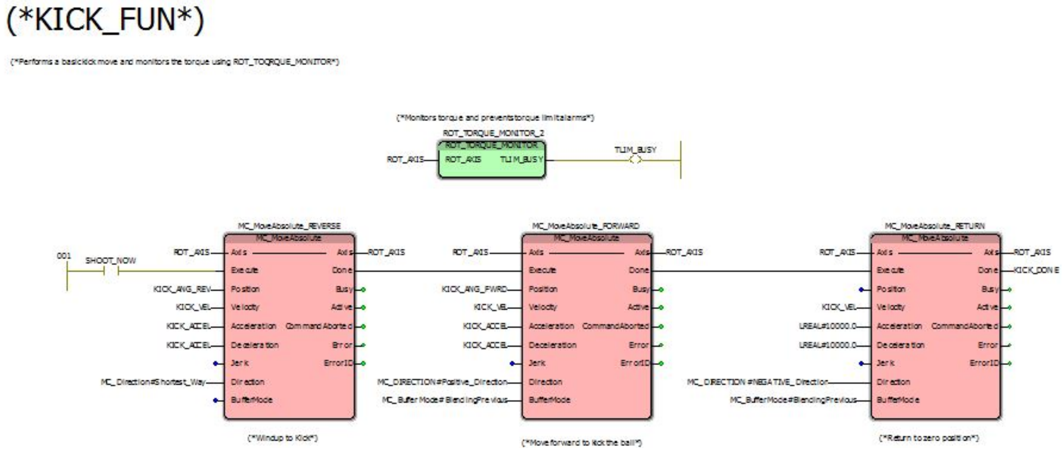
Rod Information:

This function block reads and outputs the current torques, positions and velocities of the translational and rotational motors of a given axis.



Kick Function:

This function block is used to execute a kick with the rotary motor of the given axis. The kick consists of three distinct moves. The first is the windup which moves the foosmen back in preparation for the kick. The next move is the kicking move, which sweeps the foosmen quickly forward through an arc which terminates near the foosmen's maximum reach. The final move is the return to zero move, which moves the rod back into the zero position. This block includes a torque monitoring block to prevent the rotary rod from overtorquing.

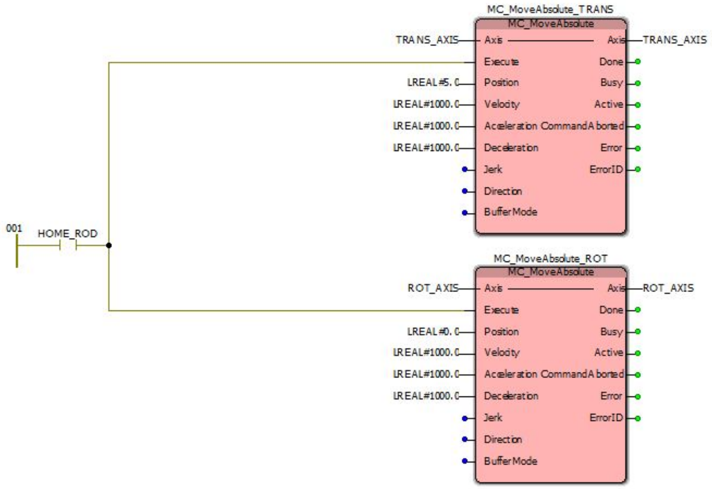


Home Single Rod:

This function moves the rotary and linear motors of a given axis back to their zero positions. The motors should be homed before operation of the system to ensure that zeroes are properly set.

(*HOME_SINGLE_ROD*)

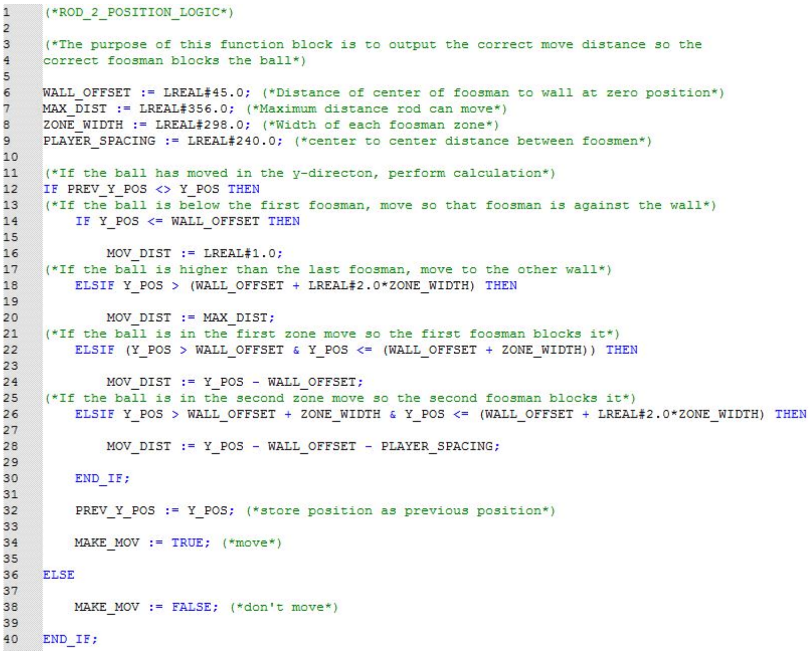
(*Returns a single rod to its home position set by ZERO_SINGLE_ROD.*)



Rod Position Logic:

The purpose of the rod position logic function blocks is to take the desired y position of the rode and tell the translation motor how much to move in order to place a foosman at that position.

These function blocks work using zones. For example, on the three man rod, there are three zones. Each zone is one third of the width of the table (excluding the width of the bumpers). If the desired y position is in the 1st zone, then the rod will move that distance minus the width of the bumper. If it is in the second zone, the rod will move that distance minus the width of the bumper and the distance between the first and second foosmen. This means that the second foosman will be at the desired y position. This pattern repeats for all of the different rods; the rods with more players just have more zones.

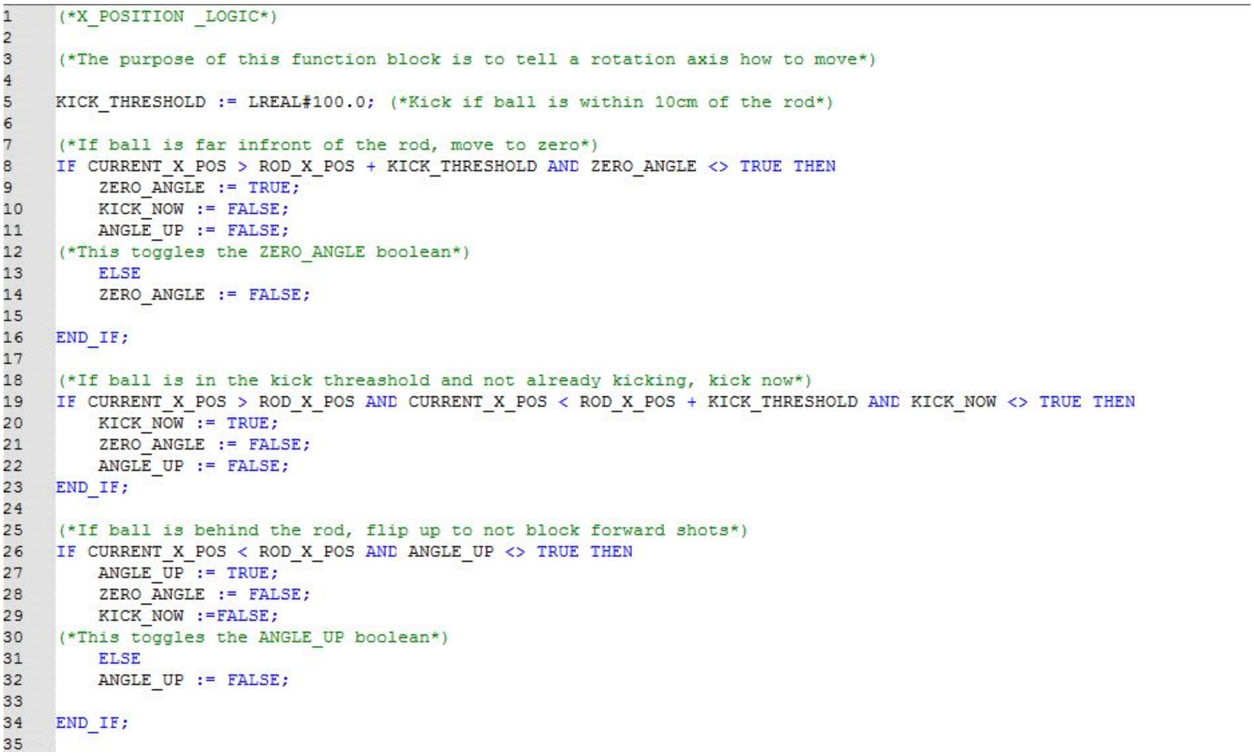


```
(*ROD_2_POSITION_LOGIC*)

(*The purpose of this function block is to output the correct move distance so the
correct foosman blocks the ball*)

WALL_OFFSET := LREAL#45.0; (*Distance of center of foosman to wall at zero position*)
MAX_DIST := LREAL#356.0; (*Maximum distance rod can move*)
ZONE_WIDTH := LREAL#298.0; (*Width of each foosman zone*)
PLAYER_SPACING := LREAL#240.0; (*center to center distance between foosmen*)

(*If the ball has moved in the y-directon, perform calculation*)
IF PREV_Y_POS <> Y_POS THEN
(*If the ball is below the first foosman, move so that foosman is against the wall*)
    IF Y_POS <= WALL_OFFSET THEN

        MOV_DIST := LREAL#1.0;
(*If the ball is higher than the last foosman, move to the other wall*)
    ELSIF Y_POS > (WALL_OFFSET + LREAL#2.0*ZONE_WIDTH) THEN

        MOV_DIST := MAX_DIST;
(*If the ball is in the first zone move so the first foosman blocks it*)
    ELSIF (Y_POS > WALL_OFFSET & Y_POS <= (WALL_OFFSET + ZONE_WIDTH)) THEN

        MOV_DIST := Y_POS - WALL_OFFSET;
(*If the ball is in the second zone move so the second foosman blocks it*)
    ELSIF Y_POS > WALL_OFFSET + ZONE_WIDTH & Y_POS <= (WALL_OFFSET + LREAL#2.0*ZONE_WIDTH) THEN

        MOV_DIST := Y_POS - WALL_OFFSET - PLAYER_SPACING;

    END_IF;

    PREV_Y_POS := Y_POS; (*store position as previous position*)

    MAKE_MOV := TRUE; (*move*)

ELSE

    MAKE_MOV := FALSE; (*don't move*)

END_IF;
```

Rotation Logic:

The purpose of the rotation function block is to allow the rod to rotate three different ways. First, if the ball is further up the field than the rod, the foosmen should be pointed down to block the ball. If the ball is near the rod, then it should kick. Finally, if the ball is behind the rod, it should flip up to avoid blocking kicks from the rods behind it.

This function block works by toggling variables which are attached to three different action blocks which are described above.

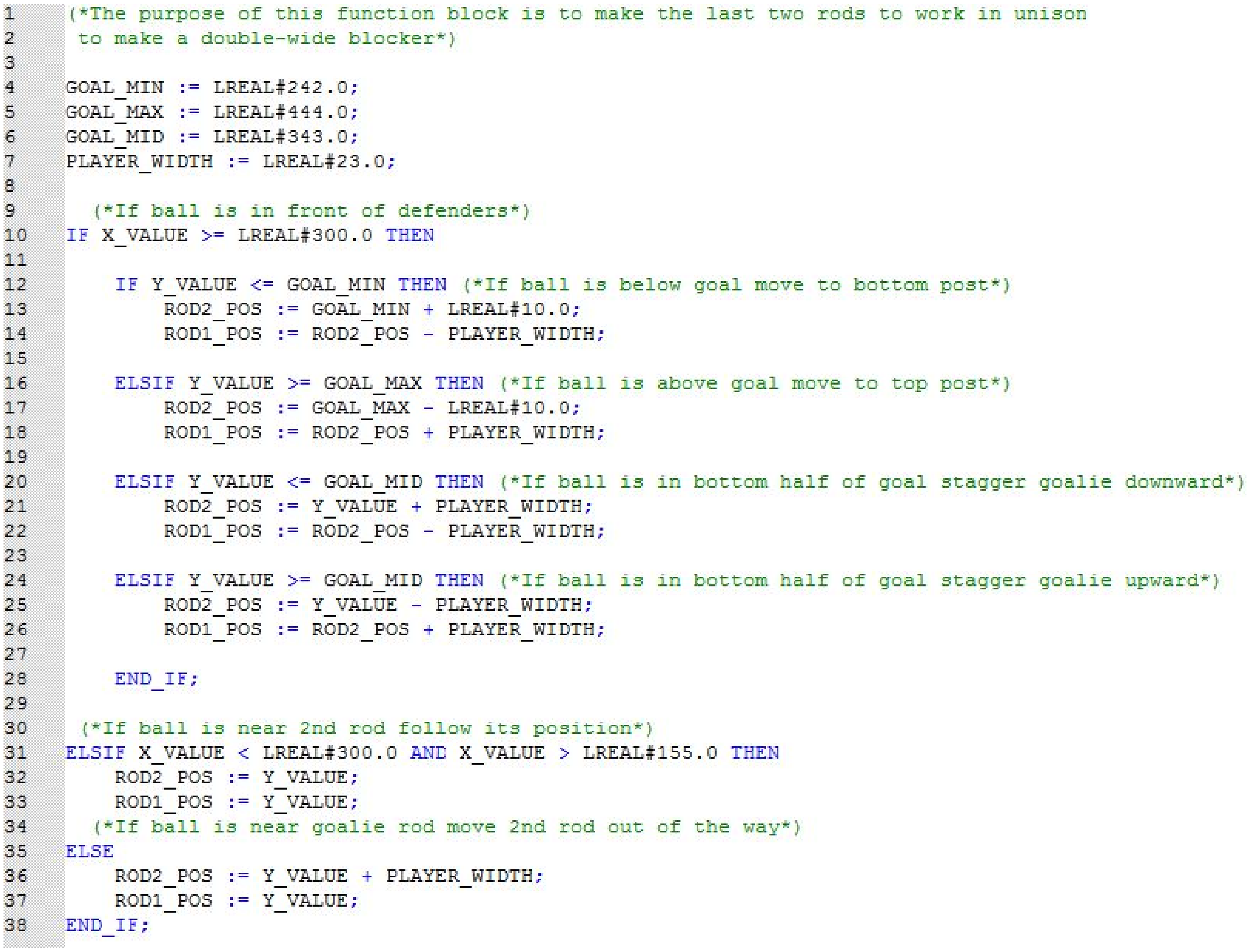


```
(*X_POSITION _LOGIC*)

(*The purpose of this function block is to tell a rotation axis how to move*)

KICK_THRESHOLD := LREAL#100.0; (*Kick if ball is within 10cm of the rod*)

(*If ball is far infront of the rod, move to zero*)
IF CURRENT_X_POS > ROD_X_POS + KICK_THRESHOLD AND ZERO_ANGLE <> TRUE THEN
    ZERO_ANGLE := TRUE;
    KICK_NOW := FALSE;
    ANGLE_UP := FALSE;
(*This toggles the ZERO_ANGLE boolean*)
    ELSE
    ZERO_ANGLE := FALSE;

END_IF;

(*If ball is in the kick threashold and not already kicking, kick now*)
IF CURRENT_X_POS > ROD_X_POS AND CURRENT_X_POS < ROD_X_POS + KICK_THRESHOLD AND KICK_NOW <> TRUE THEN
    KICK_NOW := TRUE;
    ZERO_ANGLE := FALSE;
    ANGLE_UP := FALSE;
END_IF;

(*If ball is behind the rod, flip up to not block forward shots*)
IF CURRENT_X_POS < ROD_X_POS AND ANGLE_UP <> TRUE THEN
    ANGLE_UP := TRUE;
    ZERO_ANGLE := FALSE;
    KICK_NOW :=FALSE;
(*This toggles the ANGLE_UP boolean*)
    ELSE
    ANGLE_UP := FALSE;

END_IF;

```

Defense Logic:

The purpose of the defense logic is to make the last two rods work in tandem in order to block more of the goal.

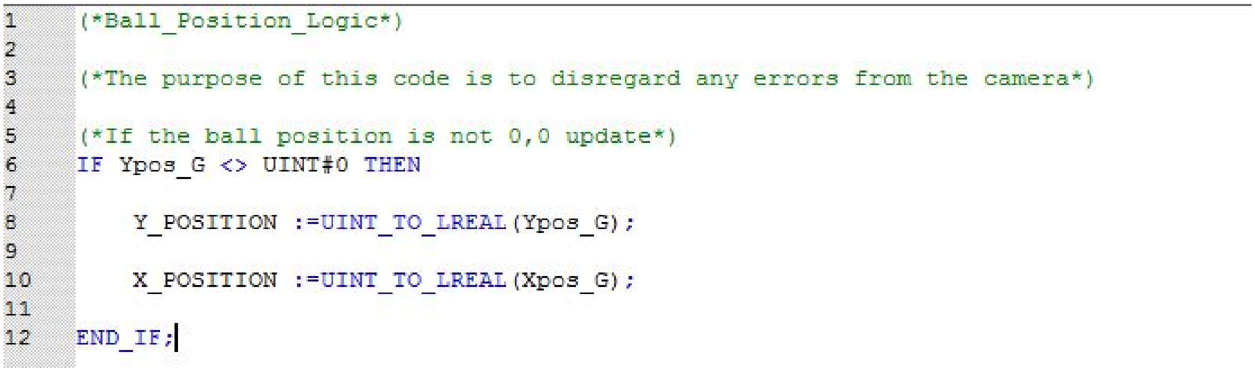
The first goal of this function block is to make sure that the last two foosmen stay within the goal as long as the ball is in front of the rods. If the ball is outside of the goal, the last two foosmen guard the post closest to the ball. Once the ball is in front of the goal, the foosman on the second rod stays slightly to the inside of the ball and the goalie staggers slightly to the outside of the ball. This essentially creates a double-wide defender.



```
(*The purpose of this function block is to make the last two rods to work in unison
 to make a double-wide blocker*)

GOAL_MIN := LREAL#242.0;
GOAL_MAX := LREAL#444.0;
GOAL_MID := LREAL#343.0;
PLAYER_WIDTH := LREAL#23.0;

  (*If ball is in front of defenders*)
IF X_VALUE >= LREAL#300.0 THEN

    IF Y_VALUE <= GOAL_MIN THEN (*If ball is below goal move to bottom post*)
        ROD2_POS := GOAL_MIN + LREAL#10.0;
        ROD1_POS := ROD2_POS - PLAYER_WIDTH;

    ELSIF Y_VALUE >= GOAL_MAX THEN (*If ball is above goal move to top post*)
        ROD2_POS := GOAL_MAX - LREAL#10.0;
        ROD1_POS := ROD2_POS + PLAYER_WIDTH;

    ELSIF Y_VALUE <= GOAL_MID THEN (*If ball is in bottom half of goal stagger goalie downward*)
        ROD2_POS := Y_VALUE + PLAYER_WIDTH;
        ROD1_POS := ROD2_POS - PLAYER_WIDTH;

    ELSIF Y_VALUE >= GOAL_MID THEN (*If ball is in bottom half of goal stagger goalie upward*)
        ROD2_POS := Y_VALUE - PLAYER_WIDTH;
        ROD1_POS := ROD2_POS + PLAYER_WIDTH;

    END_IF;

 (*If ball is near 2nd rod follow its position*)
ELSIF X_VALUE < LREAL#300.0 AND X_VALUE > LREAL#155.0 THEN
    ROD2_POS := Y_VALUE;
    ROD1_POS := Y_VALUE;
  (*If ball is near goalie rod move 2nd rod out of the way*)
ELSE
    ROD2_POS := Y_VALUE + PLAYER_WIDTH;
    ROD1_POS := Y_VALUE;
END_IF;
```

Ball Position Logic:

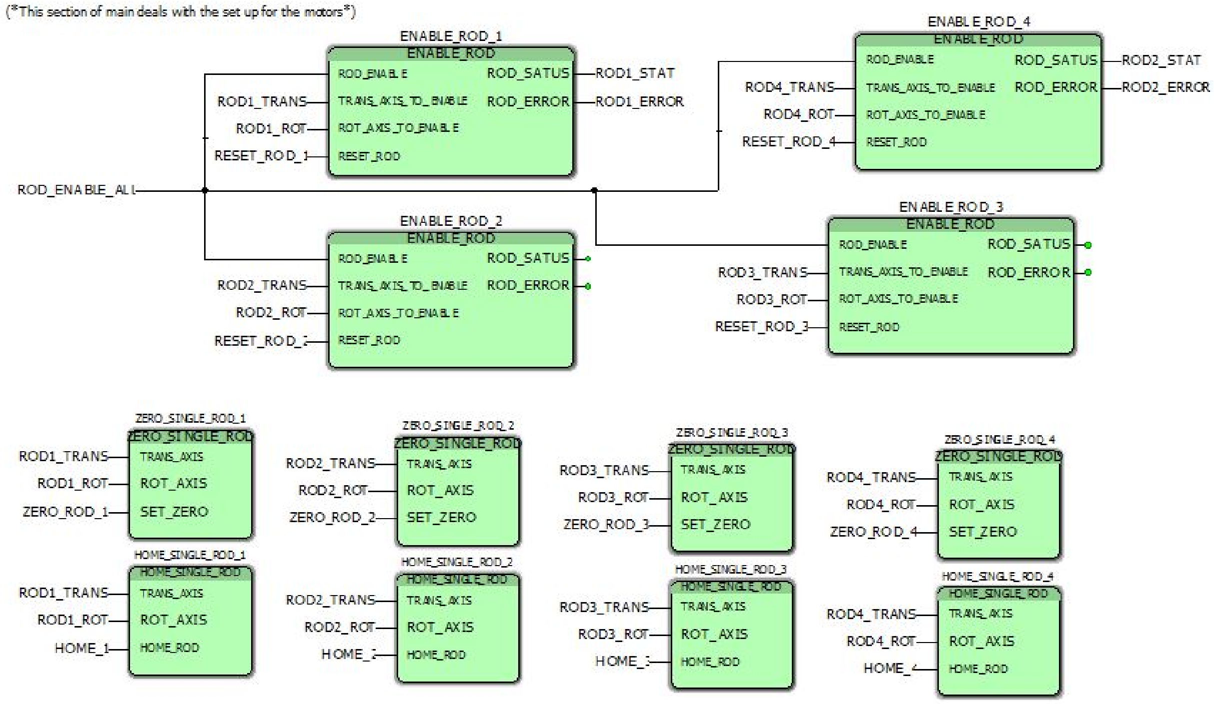
This is a standalone program, executed before main, which reads the incoming ball position data from the vision system and passes that information onto main via global variables. When the ball is lost by the camera, zeros are sent to the PLC, and so zeros are rejected by this function and the previous valid position is maintained by the system.



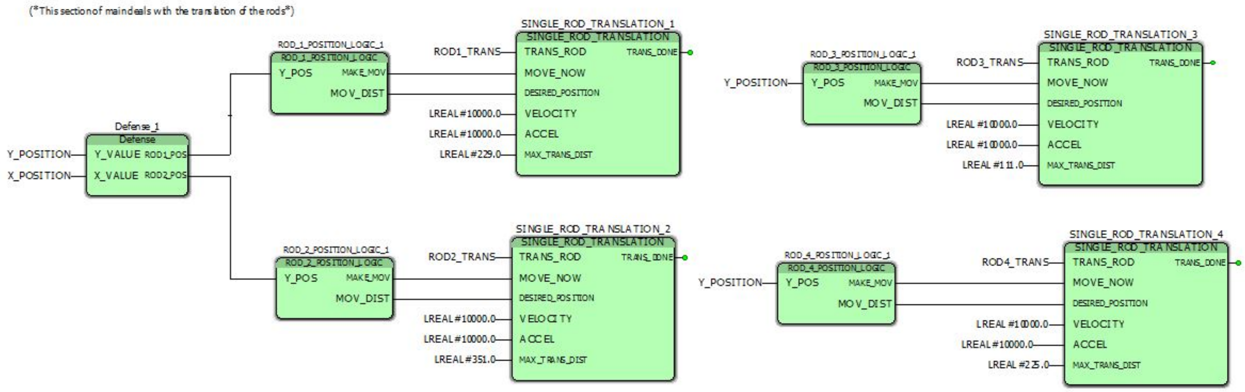
```
(*Ball_Position_Logic*)

(*The purpose of this code is to disregard any errors from the camera*)

(*If the ball position is not 0,0 update*)
IF Ypos_G <> UINT#0 THEN

    Y_POSITION :=UINT_TO_LREAL(Ypos_G);

    X_POSITION :=UINT_TO_LREAL(Xpos_G);

END_IF;
```

Main:

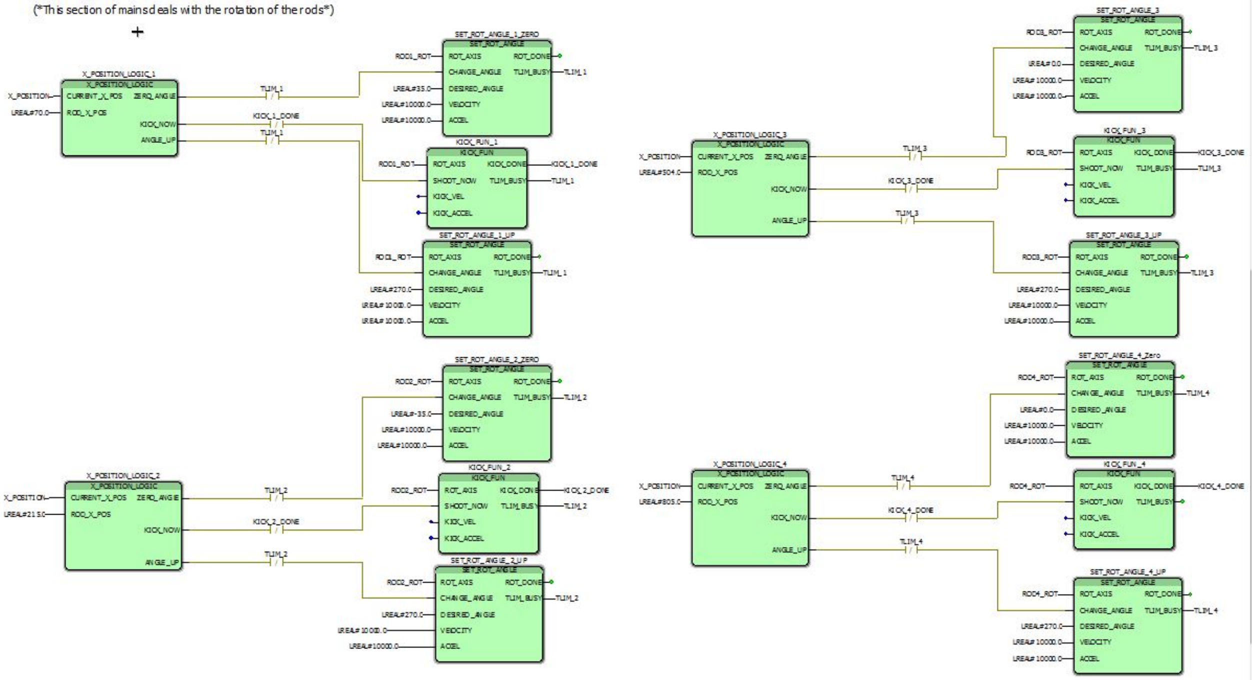
Main is an amalgamation of each of the function blocks described above into a working and playable, if simple, AI program. It also includes the necessary blocks to perform set up before play begins. It is broken into three main sections. The first contains the rod enables for the rods, the zero position functions for the rods and the homing functions for the rods. The next section contains the translational logic for the rods and the translation functions required to move each rod. It also contains the defensive logic used on the last two rods. The final section contains the rotational logic for the rods and the set angle and kick functions required to move the rods.



(*This section of main deals with the set up for the motors*)



(*This section of main deals with the translation of the rods*)



(*This section of mains deals with the rotation of the rods*)

Setup Explanation:

When the system has been full assembled and is fully powered, its relatively easy to operate the foosball program. This section contains the procedure for preparing the program for play.

Step 1:

Open Motionworks and load FOOSBALLV01 into Motionworks.

Step 2:

Perform a Download Changes to ensure that the program file is loaded into the PLC.

Step3:

Enter debug mode, this will cause real time values to appear in association with the variables in the program.

Step 4:

Turn on Toggle Boolean Mode using the following path ONLINE>TOGGLE BOOLEAN

Step 5:

Open the MAIN POU in the editing window.

Step 6:

Move the rods into their home positions, and press the set zero variables on each zeroing function block.

Step 7:

Enable the rods by setting the enable rod variable to true.

Step 8:

The system should begin operating at this point. Simply place a ball in the field.

Motor Tuning

This section contains information on the tuning of the motors using Sigmawin+, Yaskawas motor tuning software. It contains a link to an tutorial produced by Yaskawa on the basics of tuning using Sigmawin+. It also contains information on the tuning the motors which cannot be found in the video, but is useful.

Yaskawa Tutorial

The Yaskawa tutorial contains most of the information required to use the autotuning function, jog function and trace function in Sigmawin+. Each of these functions is important in tuning and assessing the performance of the motors. Figure ## contains an image of the tutorial video and the direct link to the video.

Sigmawin+ contains both an autotuning function and a manual tuning function. As the inertia ratios of the motors are relatively low, the autotuning function is more than sufficient to accurately tune the motor. This function detects the inertia ratio of the load, adjusts the gains to improve performance and applies any filters that are required to improve the performance in the motors.

The jog and trace functions are used in tandem to monitor the performance of the motors. The jog function is used to perform moves with the motor being assessed. It allows the user to set a pattern of moves a desired torque, speed and position. The trace function monitors the motor and outputs graphical information about the motors. Used together, the jog and trace function can be used to monitor the motors performance during a variety of different moves.

Sigma-5 Tuning
Quick Start Video

Class ID: TRM801-Sigma5-TuningQuickstart
Doc ID: eLV.Sigma5.01.TuningQuickStart
Rev 1.00 July 14, 2008

Yaskawa Electric America
Technical Training Services

Sigma-5 Advanced Auto-Tuning Quickstart

Figure 106: The link to training video https://www.youtube.com/watch?v=_9TW9wodQ8M

Connecting Motors to Sigmawin+

This section describes the steps required to connect a motor and motor amplifier to Sigmawin+ for tuning.

STEP 1:

Connect the computer to the motor amplifier using a USB to micro USB cable. The micro USB is plugged into the CN7 port of the motor amplifier.

STEP 2:

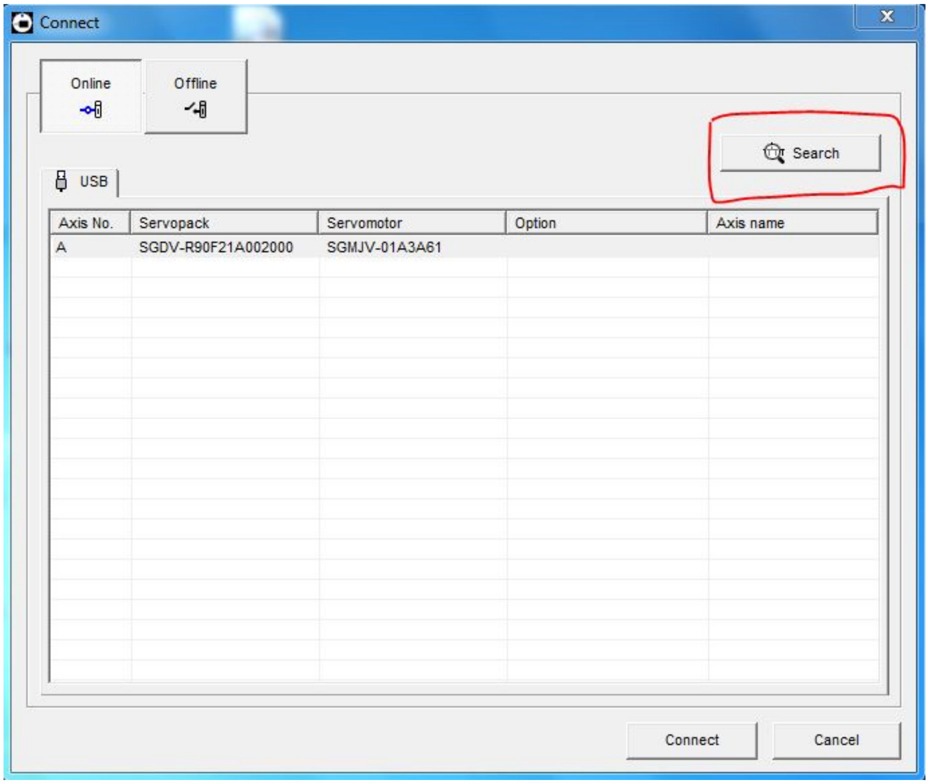
Power on the PLC and motors.

STEP 3:

Open The Sigmawin+ software.

STEP 4:

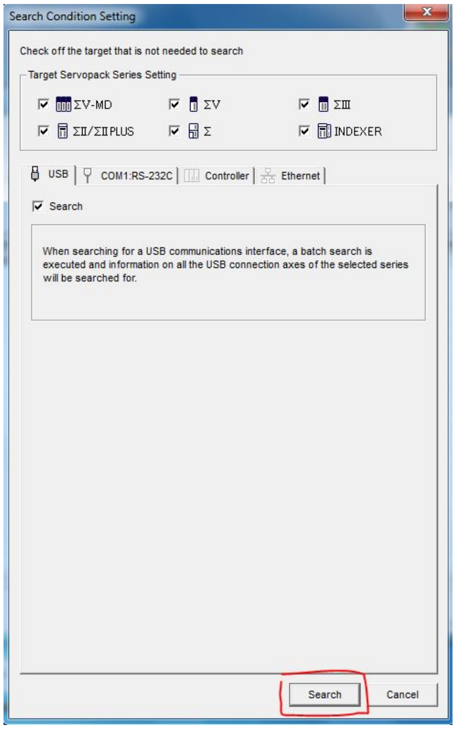
When the Sigmawin+ Connect window opens, click the search button to open the Search Condition Setting window.



Axis No.	Servopack	Servomotor	Option	Axis name
A	SGDV-R90F21A002000	SGMJV-01A3A61		

STEP 5:

Select the USB tab in the Search Condition Setting menu, then hit the search button.



Check off the target that is not needed to search

Target Servopack Series Setting

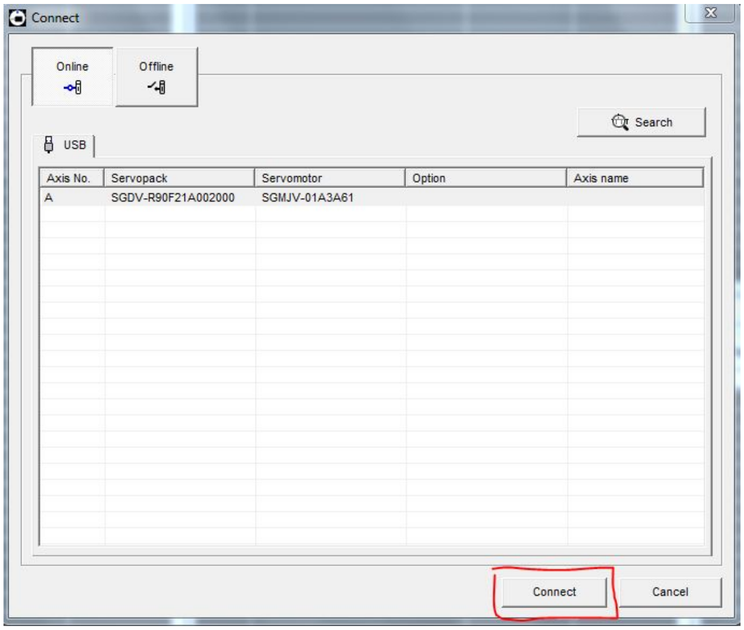
☑ ΣV-MD ☑ ΣV ☑ ΣIII
☑ ΣII/ΣII PLUS ☑ Σ ☑ INDEXER

☑ Search

When searching for a USB communications interface, a batch search is executed and information on all the USB connection axes of the selected series will be searched for.

STEP 6:

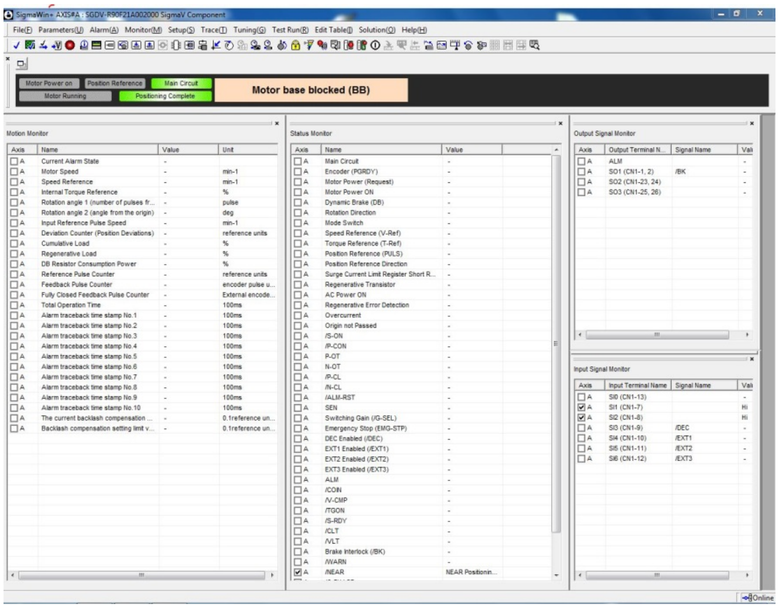
Once the motor has been found by the software, the Connect window will be brought up. The motor can be selected and the connect button clicked.



Axis No.	Servopack	Servomotor	Option	Axis name
A	SGDV-R90F21A002000	SGMJV-01A3A61		

STEP 7:

The motor is now connected to Sigmawin+ and the main Sigmawin+ window is open.



Motion Monitor

Axis	Name	Value	Unit
☐ A	Current Alarm State	-	
☐ A	Motor Speed	-	min-1
☐ A	Speed Reference	-	min-1
☐ A	Internal Torque Reference	-	%
☐ A	Rotation angle 1 (number of pulses fr...	-	pulse
☐ A	Rotation angle 2 (angle from the origin)	-	deg
☐ A	Input Reference Pulse Speed	-	min-1
☐ A	Deviation Counter (Position Deviations)	-	reference units
☐ A	Cumulative Load	-	%
☐ A	Regenerative Load	-	%
☐ A	DB Resistor Consumption Power	-	%
☐ A	Reference Pulse Counter	-	reference units
☐ A	Feedback Pulse Counter	-	encoder pulse u...
☐ A	Fully Closed Feedback Pulse Counter	-	External encode...
☐ A	Total Operation Time	-	100ms
☐ A	Alarm traceback time stamp No.1	-	100ms
☐ A	Alarm traceback time stamp No.2	-	100ms
☐ A	Alarm traceback time stamp No.3	-	100ms
☐ A	Alarm traceback time stamp No.4	-	100ms
☐ A	Alarm traceback time stamp No.5	-	100ms
☐ A	Alarm traceback time stamp No.6	-	100ms
☐ A	Alarm traceback time stamp No.7	-	100ms
☐ A	Alarm traceback time stamp No.8	-	100ms
☐ A	Alarm traceback time stamp No.9	-	100ms
☐ A	Alarm traceback time stamp No.10	-	100ms
☐ A	The current backlash compensation ...	-	0.1reference un...
☐ A	Backlash compensation setting limit v...	-	0.1reference un...

Status Monitor

Axis	Name	Value	
☐ A	Main Circuit	-	
☐ A	Encoder (PGRDY)	-	
☐ A	Motor Power (Request)	-	
☐ A	Motor Power ON	-	
☐ A	Dynamic Brake (DB)	-	
☐ A	Rotation Direction	-	
☐ A	Mode Switch	-	
☐ A	Speed Reference (V-Ref)	-	
☐ A	Torque Reference (T-Ref)	-	
☐ A	Position Reference (PULS)	-	
☐ A	Position Reference Direction	-	
☐ A	Surge Current Limit Register Short R...	-	
☐ A	Regenerative Transistor	-	
☐ A	AC Power ON	-	
☐ A	Regenerative Error Detection	-	
☐ A	Overcurrent	-	
☐ A	Origin not Passed	-	
☐ A	/S-ON	-	
☐ A	/P-CON	-	
☐ A	P-OT	-	
☐ A	N-OT	-	
☐ A	/P-CL	-	
☐ A	/N-CL	-	
☐ A	/ALM-RST	-	
☐ A	SEN	-	
☐ A	Switching Gain (/G-SEL)	-	
☐ A	Emergency Stop (EMG-STP)	-	
☐ A	DEC Enabled (/DEC)	-	
☐ A	EXT1 Enabled (/EXT1)	-	
☐ A	EXT2 Enabled (/EXT2)	-	
☐ A	EXT3 Enabled (/EXT3)	-	
☐ A	ALM	-	
☐ A	/COIN	-	
☐ A	/V-CMP	-	
☐ A	/TGON	-	
☐ A	/S-RDY	-	
☐ A	/CLT	-	
☐ A	/VLT	-	
☐ A	Brake Interlock (/BK)	-	
☐ A	/WARN	-	
☑ A	/NEAR	NEAR Positionin...	

Output Signal Monitor

Axis	Output Terminal N...	Signal Name	Val
☐ A	ALM		-
☐ A	SO1 (CN1-1, 2)	/BK	-
☐ A	SO2 (CN1-23, 24)		-
☐ A	SO3 (CN1-25, 26)		-

Input Signal Monitor

Axis	Input Terminal Name	Signal Name	Val
☐ A	SI0 (CN1-13)		-
☑ A	SI1 (CN1-7)		Hi
☑ A	SI2 (CN1-8)		Hi
☐ A	SI3 (CN1-9)	/DEC	-
☐ A	SI4 (CN1-10)	/EXT1	-
☐ A	SI5 (CN1-11)	/EXT2	-
☐ A	SI6 (CN1-12)	/EXT3	-

Autotuning Error

In the course of tuning the linear drive motors, an error in the atotuning feature was encountered. The motors were successfully autotuned once and all subsequent attempts at autotuning the motors encountered positioning errors of some kind. The solution to the problem is to reset the motors parameters back to factory default. The following procedure describes the steps required to set any of the motors back to their default settings.

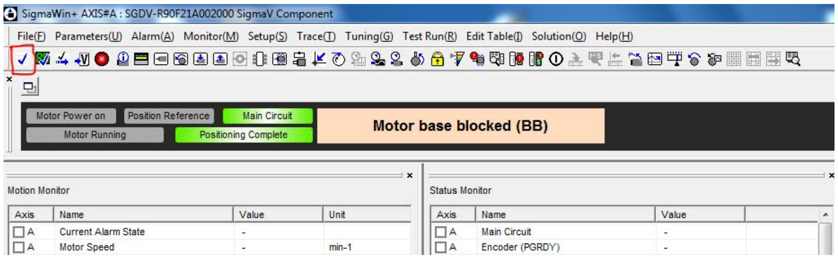
Disclaimer: Reinitializing the motors or changing individual motor parameters can cause the system to act in unexpected ways. Use caution when testing the motors and always save the last working parameter set as back up and guide.

STEP 1:

Connect the motor to the Sigmawin+ software using the guide above.

STEP 2:

When in main Sigmawin+ dialog is open, select the Edit Parameters button.



SigmaWin+ AXIS#A : SGDV-R90F21A002000 SigmaV Component

Motor base blocked (BB)

Motion Monitor

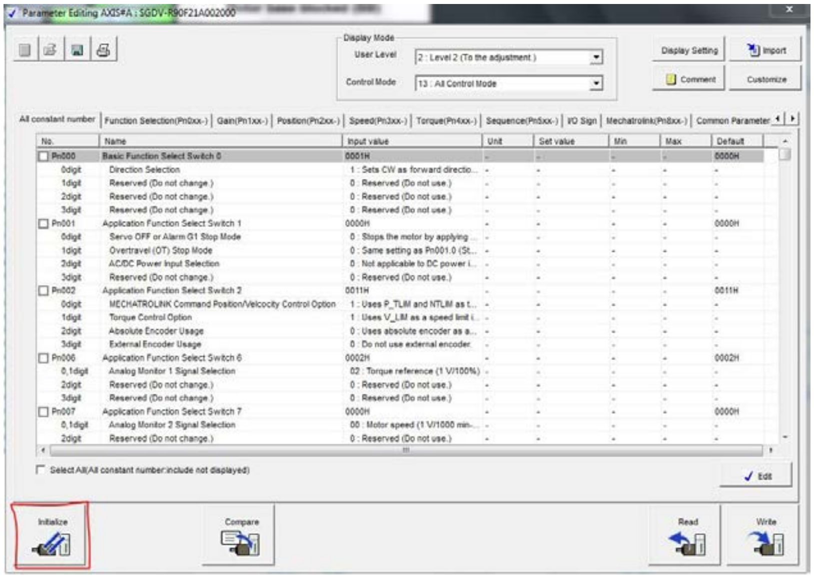
Axis	Name	Value	Unit
☐ A	Current Alarm State	-	
☐ A	Motor Speed	-	min-1

Status Monitor

Axis	Name	Value
☐ A	Main Circuit	-
☐ A	Encoder (PGRDY)	-

STEP 3:

When the Parameter Editing window opens click the Initialize button.



Parameter Editing AXIS#A : SGDV-R90F21A002000

Display Mode
User Level: 2 : Level 2 (To the adjustment)
Control Mode: 13 : All Control Mode

No.	Name	Input value	Unit	Set value	Min	Max	Default
☐ Pn000	Basic Function Select Switch 0	0001H	-	-	-	-	0000H
0digit	Direction Selection	1 : Sets CW as forward directio...	-	-	-	-	-
1digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-
2digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-
3digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-
☐ Pn001	Application Function Select Switch 1	0000H	-	-	-	-	0000H
0digit	Servo OFF or Alarm G1 Stop Mode	0 : Stops the motor by applying ...	-	-	-	-	-
1digit	Overtravel (OT) Stop Mode	0 : Same setting as Pn001.0 (St...	-	-	-	-	-
2digit	AC/DC Power Input Selection	0 : Not applicable to DC power i...	-	-	-	-	-
3digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-
☐ Pn002	Application Function Select Switch 2	0011H	-	-	-	-	0011H
0digit	MECHATROLINK Command Position/Velocity Control Option	1 : Uses P_TLIM and NTLIM as t...	-	-	-	-	-
1digit	Torque Control Option	1 : Uses V_LIM as a speed limit i...	-	-	-	-	-
2digit	Absolute Encoder Usage	0 : Uses absolute encoder as a...	-	-	-	-	-
3digit	External Encoder Usage	0 : Do not use external encoder.	-	-	-	-	-
☐ Pn006	Application Function Select Switch 6	0002H	-	-	-	-	0002H
0,1digit	Analog Monitor 1 Signal Selection	02 : Torque reference (1 V/100%)	-	-	-	-	-
2digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-
3digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-
☐ Pn007	Application Function Select Switch 7	0000H	-	-	-	-	0000H
0,1digit	Analog Monitor 2 Signal Selection	00 : Motor speed (1 V/1000 min-...	-	-	-	-	-
2digit	Reserved (Do not change.)	0 : Reserved (Do not use.)	-	-	-	-	-

☐ Select All(All constant number:include not displayed)

STEP 4:

A Verification dialog will open; click the okay button to reinitialize the motor.

STEP 5:

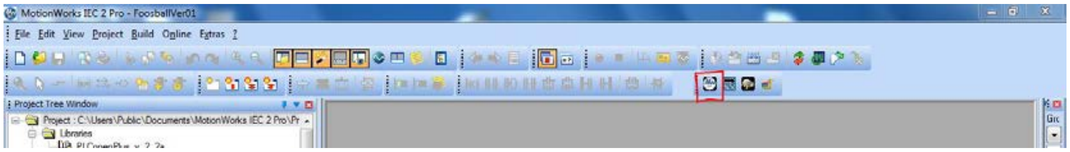
After the motor has been restored to its default settings, the motor can then be power cycled and then autotuned.

Saving Motor Parameters to the Program File

Saving changes made to the configurations of the motors or other hardware associated with the PLC is an important part of working with the PLC. The procedure used to save different configurations to the program file is the same regardless of the changes being made, but changes to motor parameters is used here as an example of the general procedure.

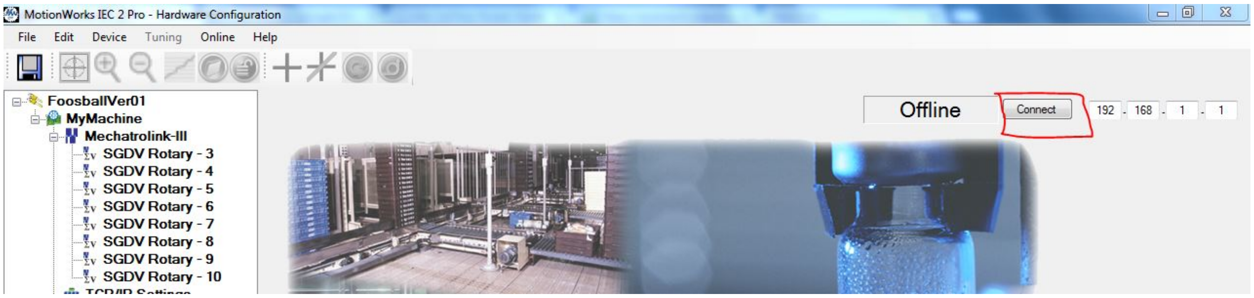
STEP 1:

Open Motionworks. When the main window opens, power on the motor system. After the PLC is fully powered, click the Hardware configuration button.



STEP 2:

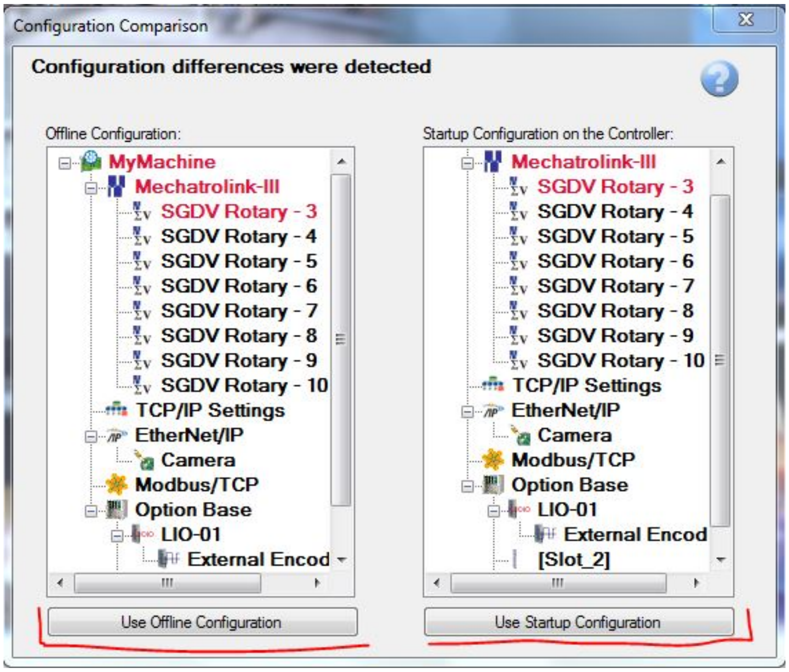
When the Hardware configuration window opens, click on the connect button.



STEP 3:

At this point two things can occur. If the configuration file stored on the PLC matches the current program file, the PLC will connect. If the two files conflict, a dialog will open showing two list of the hardware connected to the PLC, with the hardware where there is a configuration mismatch highlighted in red. These highlighted sections can be right clicked to show the differences between the two configurations. The left hand list, the Offline Configuration, are the configurations stored in the program file. The list on the right, the Start Configurations, are the configurations saved on the PLC and the Amplifiers.

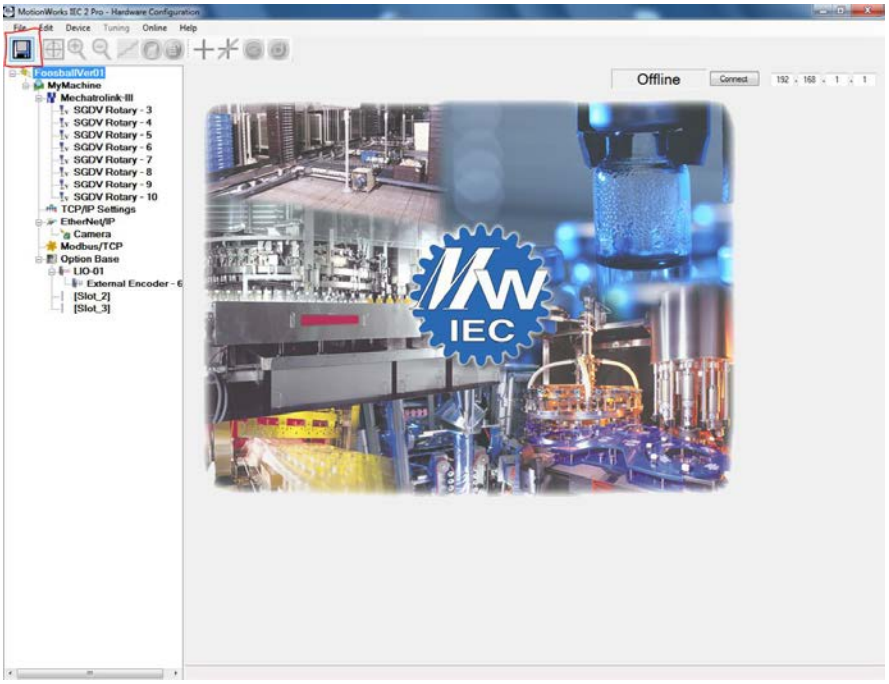
Because, in this example, the amplifier settings where changed outside of the Hardware configuration menu, the Startup Configuration button is clicked. If changes were made to the configuration while offline, the Offline changes button would be clicked.



Configuration differences were detected

STEP 4:

The configuration can now be saved to the PLC and Program File by pressing the save button.



Appendix L: Reference Material

[1] Alsalihi, A., Najjar, K., Van Scoy, B., & Zifer, J. (2011). Automated Foosball Table, University of Akron.

[2] Automated Foosball Table, A project at DTU (2007) Technical University of Denmark. Retrieved from http://foospmp.myl.dk/

[3] *Foosball Playing Robot* (2007). Georgia Institute of Technology. Retrieved from http://www.eskibars.com/projects/foosball_robot/

[4] Hijkoop, A., Steinbuch. (2007). Designing a foosball table actuator. Eindhoven University of Technology. Retrieved from http://www.mate.tue.nl/mate/

[5] (2008, January 9) Robotic Foosball [video file]. Retrieved from http://www.youtube.com/watch?v=KFAfwtsxarU

[6] (2011 April 17) Table soccer robot playing [video file]. Retrieved from http://www.youtube.com/watch?feature=player_embedded&v=m6C4SOxfNGQ#!

[7] tueindhoven (2011, July 8) Semi-automated Football Table – Eindhoven University of Technology [video file]. Retrieved from
http://www.youtube.com/watch?v=0qE_a0wFR
a0

[8] (2012, April 27) University of Akron Foosbot [video file]. Retrieved from http://www.youtube.com/watch?v=_UTl8UTE19Y

[9] Weigel, T. (2005). KiRo - *A Table Soccer Robot Ready for the Market*. University of Freiburg. Retrieved from ftp://ftp.informatik.uni-freiburg.de/papers/ki/weigel- icra2005.pdf.

[10] Graham, J et. All (2013, October 10, 2013) Automated Foosball Table Concept Design Report – Cal Poly. Courtesy of team Foos-Ro-Dah.