4. Evaluation

4.1. Overview of Testing Requirements

The tests described in this section ensure that the robot meets every design constraint. The tests performed on the hardware and software portions of the robot guarantee that each performs satisfactorily. Table 4.1 below shows the division of design constraints and provides an overview of how each was tested.

Table 4.1 Testing Set summary categorized by Design Constraints

Design Constraint	Test Set Description
Dimensions	The completely assembled robot was measured to ensure its dimensions do not exceed contest restrictions.
Navigation	The robot's ability to navigate the playing field correctly was tested. This required testing the following hardware components: stepper motors, line sensors, and sonar sensors. This testing also required the certification of the navigational software used for maneuvering, range finding, and motor control.
Speed	The robot's ability to meet speed requirements was tested by performing stepper motor testing and navigational software testing.
Package Identification and Manipulation	The robot's ability to accurately identify, collect, and store all 12 packages was tested. This required testing the following hardware components: servos, stepper motors, storage chutes, turntable, barcode scanner, and gripper. This testing also required the certification of the collection software used for servo and motor control, barcode reading, PIC communication, and package extraction.
Weight	The completely assembled and loaded robot was weighted to ensure it met the 20-pound weight limit.
Power	Each power supply was tested for correct voltages and currents. Each supply will also be tested for adequate battery life. The printed circuit board (PCB) was also test for correct power trace routes.

4.2 Detailed Description of Each Subsystem Test

4.2.1 Software

4.2.1.1 Barcode Scanner

The packages used during the competition are labeled with a barcode to indicate the ultimate destination, Plane A, B, or C, of each package. Software must be able to enable and disable power to the scanner and correctly read the data bytes of a barcode to determine the proper action of the loading and unloading mechanism. The objective of this test is to verify that software enables and disables power to the scanner, recognizes each barcode appropriately as either Package A, Package B, Package C, or Unrecognized, and in the event that no barcode is present, exits the function.

4.2.1.1.1 Barcode Scanner Test Procedure

The primary microprocessor was programmed to utilize the Scan_Barcode() function 12 times, because there are 12 packages in total, and to transmit the results of each scan serially as shown in Table 4.2.1.1.1.

Table 4.2.1.1.1 Serial Communication for Each Return Integer of Scan_Barcode():

Return Integer	Serial Communication
1	Package A
2	Package B
3	Package C
4	Unrecognized Scan
5	Invalid Barcode
6	No Barcode Present
< 1 or > 6	Error

In the event that an incoming data byte is corrupted a software reset will trigger and either “Error: Start bit is 1” or “Error: Stop bit is 0” will be transmitted serially. To monitor the results, an RS232 interface was used to connect serially from the robot’s primary microprocessor to a computer; the program HyperTerminal was used to monitor the serial transmissions. During the test, the results were monitored for when all three known barcodes were scanned, for when an unknown barcode is scanned, and for the event in which no barcode is present.

4.2.1.1.2 Barcode Scanner Test Certification

Initial software tests, as outlined in Section 4.1.1.1.1, were unsuccessful. The error message “Error: Start bit is 1” was continuously transmitted. Additional coding was added to instruct the program to serially transmit any and all data bytes received. The additional coding revealed that the first frame of data bytes had been received and that the error was occurring as the second frame of bytes was being received. This indicated that the problem could be a timing issue with the software. Perhaps the program was spending too much time reading the first frame of bytes and missing all or part of the second frame. To validate this theory, an oscilloscope was used to determine the time delay between frames. However, not only was it discovered that there is sufficient delay between each frame for the microprocessor to receive all the data, but also that the problem went away. Because the oscilloscope provides some capacitance to settle noise, it was determined that the error condition was occurring due to some noise on the clock line in between data frames. To solve this problem, a small capacitance, on the order of 2.2 nF, was added to the clock line of the scanner interface, as shown in Figure 4.2.1.1.2, to settle any noise present.

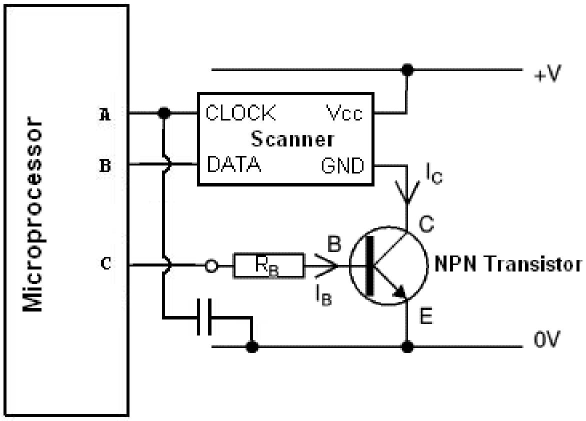


Figure 4.2.1.1.2: Wiring Diagram of Scanner Interface with Capacitor

4.2.1.2 Navigation

Under the software category of navigation, there are two major features with regard to step calculation that need to be addressed and tested: traveling distance and angle turning. To calculate the number of steps necessary for each motor to turn, it is imperative to know the radius of the wheels being used and the number of steps per revolution of the motor.

4.2.1.2.1 Traveling Distance

In order to test the accuracy of the step calculations, a program titled StepTest.c was created. This program accepts a distance in inches from the user and then calculates and outputs the number of steps needed for each motor to travel the given distance.

4.2.1.2.1.1 Distance Test Procedure

A program, StepCount.c, was written that calculated the number of steps that each stepper motor must travel to cover a given distance. The PIC4620 was then programmed with StepCount.c. Using the HyperTerminal application, a distance was entered in inches, and the number of steps returned by the PIC4620 was noted. This number was then compared to the steps calculated by hand. With this data, the accuracy of the program was determined.

4.2.1.2.1.2 Distance Test Certification

The accuracy of the program's step calculation for distance is near 100%. Since the motors cannot move a fraction of a step, the step count is calculated as an integer. As a result, any floating point ending is truncated from the final result. This is why the accuracy cannot reach absolute 100%. The data from Figure 4.2.1.2.1.2.b shows, however, that the step count is always at least 98% accurate. Figure 4.2.1.2.1.2.a displays the results from the program's calculation compared to that of the hand calculation. As one can see, the results are almost identical. A full chart of result values can be seen in Appendix A.

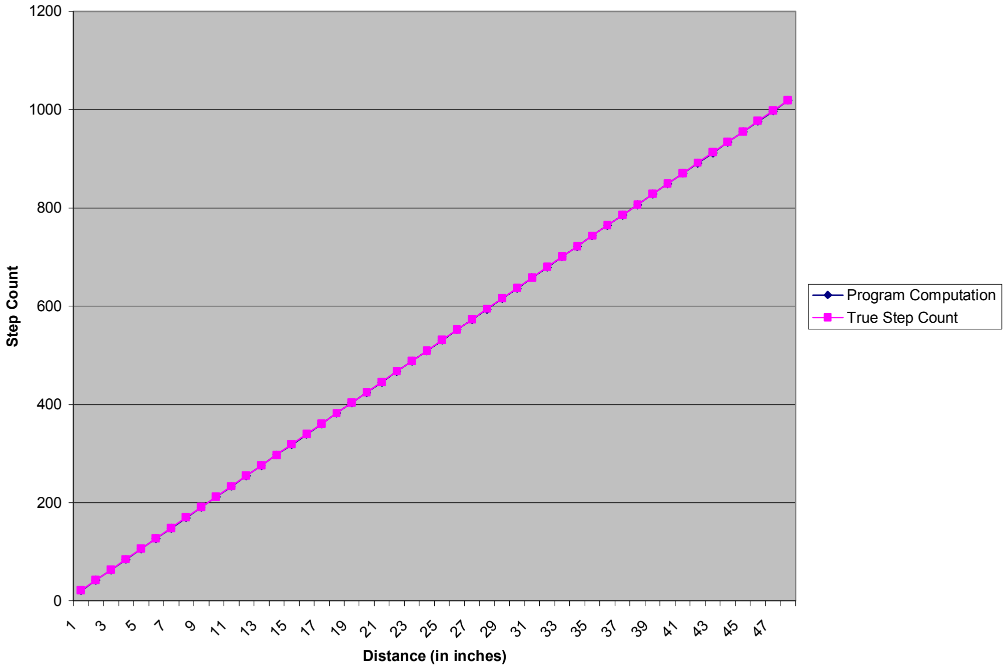


Figure 4.2.1.2.1.2.a: Program Step Count and True Step Count with Respect to Distance

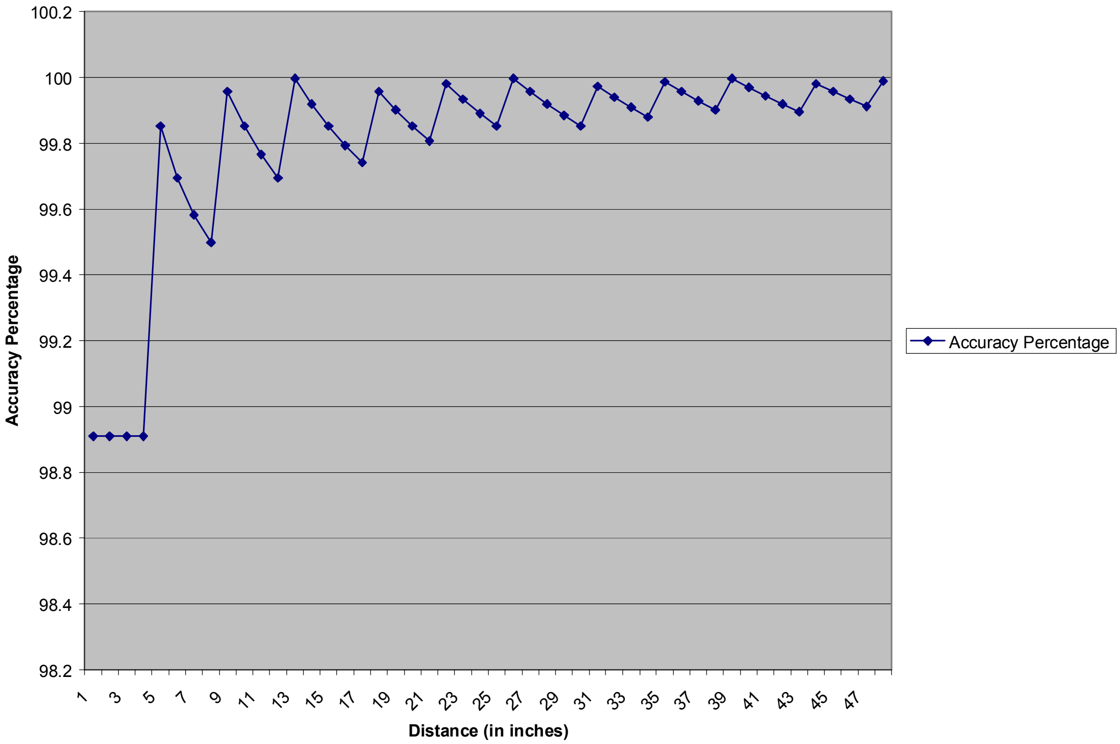


Figure 4.2.1.2.1.2.b: Accuracy of Program Step Count with Respect to True Step Count

4.2.1.2.2 Angle Turning

To turn the robot, it was decided to use a spinning approach in which both wheels turn in opposite directions. This approach was decided upon because the center of the robot would ideally stay in the same location. However, assuming that both wheels turn the same distance, it was necessary to calculate the distance that each wheel should move to turn the robot to a certain angle.

4.2.1.2.2.1 Angle Test Procedure

A program, AngleTest.c, was written that calculated the number of steps that each stepper motor must cover to move by a given angle. The PIC4620 was then programmed with StepCount.c. Using the HyperTerminal application, an angle, in degrees, was entered and the number of steps returned by the PIC4620 was noted. This number was then compared to the steps calculated by hand. With this data, the accuracy of the code was determined.

4.2.1.2.2.2 Angle Test Certification

Table 4.1.2.2.2.2: Steps per Angle

Angle	Computed Steps	Hand Calculation	Accuracy Percentage
30	44	44.4	99
45	66	66.7	99
60	88	88.9	99
90	133	133.3	99.8
120	177	177.8	99.6
135	200	200	100
150	222	222.2	99.9

180	266	266.7	99.8

Like the software for distance travel, the software for angle testing computes a step count with nearly 100% accuracy. Furthermore, like the software for distance travel, the truncation of floating point endings attributes to the lack of absolute 100% accuracy.

4.2.1.3 Servo and Stepper Motor Control

Software-generated square wave signals are used to control both the servos and the stepper motors for the top half of the robot. Servos are natively controlled by PWM signals, and using the BiMOS 5804 ICs allows the stepper motors to be controlled by square wave signals. Preliminary testing has shown that the generated PWM signals must be accurate and consistent within 0.05ms in order to keep the servos stationary at the intended angle. The BiMOS 5804 IC requires a duty cycle of at least 3us for each intended step [4.1]. Thus even short generated square wave signals must be the intended length. The test procedure is as follows.

4.2.1.3.1 Square Wave Generation Test Procedure

The PIC18LF242 was programmed with servo1.hex, the program used to control PWM output using serial input. Each PWM signal output from the PIC was tested using an oscilloscope to ensure the square wave was 5V, had the correct duty cycle, and a fluctuation in width less than 0.05ms. Table 4.2.1.3.1.a shows the HyperTerminal commands used during the PWM tests.

Table 4.2.1.3.1.a: Hyperterminal Input for Servo Signal Test

PWM Under Test	Signal Activation	Change Pulse Width
Chute A (RC1)	@ 0 1	! 0 XXX
Chute B (RC0)	@ 1 1	! 1 XXX
Chute C (RA5)	@ 2 1	! 2 XXX
Claw Gripper (RA2)	@ 6 1	! 6 XXX
Turntable (RA3)	@ 8 1	! 8 XXX

Each output pin used for stepper motor control was then tested. The oscilloscope was used in capture mode to ensure each pulse was 5V and the signal's duty cycle was consistently 3us. Table 4.2.1.3.1.b shows the HyperTerminal commands used for this portion of the test.

Table 4.2.1.3.1.b: HyperTerminal Input for Stepper Motor Signal Test

PWM Under Test	Signal Activation
Chute A (RB5)	$ 0 20 XXX 1
Chute B (RC3)	$ 1 20 XXX 1
Chute C (RA1)	$ 2 20 XXX 1
Claw Arm (RC5)	$ 3 20 XXX 1

4.2.1.3.2 Square Wave Generation Test Certification

The PWM test and stepper motor square wave test were completed successfully. The software generated PWMs for each pin were correctly formed and had a maximum of 29us duty cycle fluctuation. Fluctuation error occurred on each pin for duty cycles greater than 0.72ms. The errors were caused by the PIC's lack of precision in translating the step number into an exact duration for the duty cycle. Table 4.2.1.3.1 shows the results of each pins test.

Table 4.2.1.3.2: Servo PWM Test Results

PWM	Avg. Duty Cycle Step: 0	Avg. Duty Cycle Step: 128	Avg. Duty Cycle Step: 255	Max Fluctuation
RC1	0.72ms	1.58ms	2.44ms	0.029ms
RC0	0.72ms	1.58ms	2.44ms	0.029ms
RA5	0.72ms	1.58ms	2.44ms	0.028ms
RA2	0.72ms	1.58ms	2.44ms	0.029ms
RA3	0.72ms	1.58ms	2.44ms	0.028ms

The square waves created for stepper motor control are produced without use of the programmable PIC timers and without any duty cycle modifications. Therefore no fluctuations were found during testing. Each of the four pins displayed correctly formed square waves with 3us duty cycles.

4.2.1.4 PIC Communication

Communication between the two microcontrollers is essential to make the robot function. The PIC18LF4620 handles the main logic and navigation for the task while the PIC18LF242 controls loading and unloading movements. The PIC18LF4620 sends movement instructions to the PIC18LF242 over the serial communication protocol. Missing or dropping an instruction will possibly result in a task failure. The testing procedure for PIC communication is as follows.

4.2.1.4.1 PIC Communication Test Procedure

The PIC18LF242 was programmed with servo1.hex and the PIC18LF4620 was programmed with TestCom.hex. TestCom.hex uses serial communication to repeatedly send commands for PWM signal zero to move to step 0 and then step 255 once per second. The two PICs' TX and RX pins were connected and PWM zero's output was observed for ten seconds to ensure no command from the main PIC was dropped.

4.2.1.4.2 PIC Communication Test Certification

The PIC communication test described above was performed and the results are displayed in Figure 4.2.1.4.2. Due to the interrupt-driven implementation of the serial communication all aspects of the test were 100% successful.

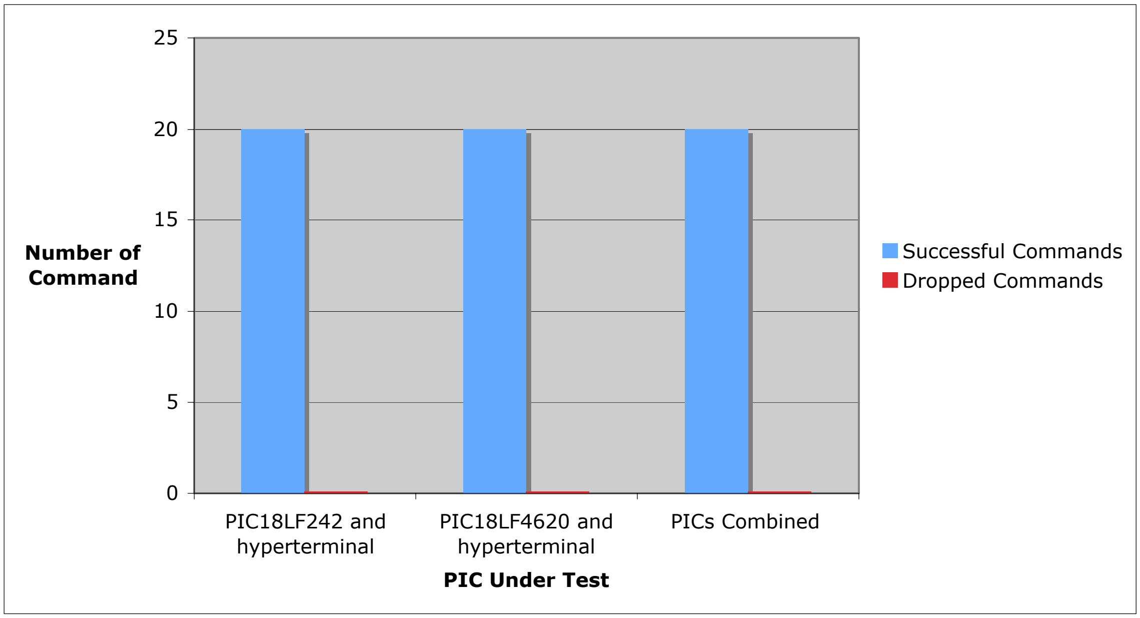


Figure 4.2.1.4.2: PIC Communication Test

4.2.2 Hardware

4.2.2.1 Linear Actuator and Extractor Claw Test

The linear actuator that drives the thrusting movement of the bottom extractor claw must reliably pull all 12 blocks out of the vertical package chute during each round. The first few blocks are the most difficult to extract because of the weight of the other packages on top of them. However, the last few blocks pulled are more likely to fall at an angle, which can make them challenging to extract as well. The linear actuator subsystem test verifies that the extractor claw and linear actuator can repeatedly pull blocks from the package chute for a wide range of weights on top of the block to be extracted.

4.2.2.1.1 Linear Actuator and Extractor Claw Test Procedure

To test the extractor system, the robot was programmed to repeatedly pull blocks from the package chute. The chute was loaded up with 12 blocks and the robot was run from its loading position in front of the chute. If the robot pulled the block from chute and dropped it in the appropriate spot for loading, the test was considered a success. If it failed, the block was removed by hand and the robot was allowed to try to pull the next block.

4.2.2.1.2 Linear Actuator and Extractor Claw Test Certification

The robot was unable to pull any blocks reliably when the stack was higher than 5 to 6 blocks. When the stack is this high, the weight of the other blocks puts a great deal of pressure on the package to be extracted. With its current power supply, the lower linear actuator is not able to overcome this pressure. Future tests will determine whether higher voltages will be enough to eliminate this problem or if the motor will have to be replaced.

4.2.2.2 Servo Torque and Range

Due to the robot design and the weight of the packages each servo requires a considerable amount of torque. The turntable servo requires the most because it must be able to rotate the robot accurately when the robot is fully loaded with all 12 packages. The turntable servo also needs the largest range of motion, at least 180 degrees. The other servos need only 90 degrees of rotation to operate the modified gripper claws on the robot. Three Hitec servo models were selected for use on the robot: the Ultra Torque HS-645MG, the Deluxe HS-475HB, and the Low Profile HS-77BB. Each servo was tested using different criteria.

4.2.2.2.1 Servo Torque and Range Test Procedure

The PIC18LF242 was programmed with servo1.hex. A HS-645MG servo was installed in the Lynxmotion turntable apparatus and rotated between 0 and 255 steps to test for a range of motion of at least 180 degrees. An eight-pound weight was set atop the turntable to simulate body weight and the rotation range was tested again. A storage chute apparatus was then affixed to the turntable with a HS-475HB servo in it. Four blocks were inserted into the chute, and the turntable was rotated from 0 to 255 multiple times to ensure no blocks were flung from the chute. Finally, a HS-77BB servo was used with the gripper claw kit to test its ability to extract packages from the FedEx chute.

4.2.2.2.2 Servo Torque and Range Test Certification

The HS-645MG servo test was completed successfully. A full range of rotation was determined to be approximately 187 degrees, and added weight did not hinder the servos range or rate of rotation. The HS-475HB servo test was also completed satisfactorily. However, single blocks were flung from the chute in 15% of test executions. This error can be contributed to the faulty construction of the chute and its inability to evenly distribute the servo’s pressure on all four blocks. The HS-77BB servo test showed that it was capable of retrieving a package from the chute. Even with all 12 packages in the chute the gripper showed no sign of slipping.

4.2.2.3 Hardware Tests for Navigation

Under the hardware aspect of navigation, three major features need to be addressed: distance travel, angle turning, and torque & speed. Software was written to compute the number of steps for the robot to travel, but only through actual hardware testing can the accuracy of the code be verified.

4.2.2.3.1 Distance

Though the code written for step calculation with respect to distance proved to be accurate within tolerance, it is necessary to observe if the calculated step count for a given distance can precisely move the robot by that distance.

4.2.2.3.1.1 Distance Test Procedure

The PIC4620 was programmed with a program that allowed the user to input a distance into the HyperTerminal application. The robot would then compute the required steps and move a distance. After the program was loaded, a distance was entered into the HyperTerminal application. The distance that was actually traveled was then measured. The distance that was actually traveled was compared to the distance entered. This was repeated for various distances.

4.2.2.3.1.2 Distance Test Certification

Table 4.2.2.3.1.2.a below shows the results of the aforementioned test.

Table 4.2.2.3.1.2.a: Distance Test Results

Distance Entered (inches)	Distance Traveled (inches)	Accuracy Percentage
3	2.875	95.83
6	5.875	97.92
9	9.0	100
12	11.75	97.92
18	17.875	99.31
24	23.25	96.88
30	28.75	95.73
36	34.50	95.83
42	40.25	95.83
48	46.00	95.83

For small distances, the step count proves to be very accurate. However, for large distances, the step count can prove to be off by as much as 2 inches. The cause for the lack of accuracy can be partially attributed to the fact that the locomotion motors cannot move a fraction of a step. However, this does not give sufficient reason as to why the robot can miss up to 2 inches in distance. Fortunately, the greatest distance that the robot must travel is from the starting square to the package chute – a distance of approximately 28 inches.

4.2.2.3.2 Angle Turning

Though the code written for step calculation with respect to angles proved to be accurate within tolerance, it is necessary to observe if the calculated step count for a given angle can precisely move the robot to that angle.

4.2.2.3.2.1 Angle Test Procedure

The PIC4620 was programmed with StepMotor_Test_Angles.c. This program allowed the user to enter an angle into the HyperTerminal application. The microcontroller then computed the required steps and turned the robot by that angle amount. An angle was entered into the HyperTerminal application. The angle that the robot actually turned was then measured by using a protractor. This process was repeated for various angles. An average for each angle was computed by repeating the procedure three times for each angle.

4.2.2.3.2.2 Angle Test Certification

Table 4.2.2.3.2.2.a below shows the results of the Hardware Angle Test.

Table 4.2.2.3.2.2.a: Angle Test Results

Angle Entered (in degrees)	Angle Turned (in degrees)	Accuracy Percentage
30	32	93.33
45	43	95.55
60	60	100
90	90.0	100
120	120	100
135	133	98.52
150	145	96.67
180	177	98.33

Though the written code cannot move the robot by a given angle at 100% accuracy, the results prove that the written code provides sufficient movement of the robot. The lack in accuracy can be attributed to the fact that the wheels are not equally spaced from the center of the robot. Furthermore, because it is not possible for the robot to move a fraction of a step, the loss of the fractional portion of the calculated step count also contributes to the loss in accuracy. Fortunately, the addition of sensors to the robot can help overcome this weakness.

4.2.2.3.3 Torque and Speed

The two stepper motors that control the wheels of the robot have an inverse torque-speed characteristic. Hence, as the rotational speed of the robot's wheels increase, the torque supplied by the motors decreases. It is desired to have the robot move as quickly as possible, while still being able to precisely move the weight of the robot.

The speed of the robot is determined by the period of the signal that the stepper motor IC's receive. As the period of this signal decreases, the rotational speed of the motors increases. Since the weight of the top half of the robot is significantly greater than the weight of the robot's base, it was necessary to perform tests to see at which speeds the robot's stepper motors could move the weight of the upper half.

4.2.2.3.3.1 Torque & Speed Test Procedures

The PIC4620 was programmed with the route traversal code. This program drove the robot along a path on the game board. A weight plate was placed on top of the robot base. The robot was then run along the path to observe if the motors ran into any problems. This process was repeated for different weights. If the robot experienced any difficulty with a given weight, then the speed was reduced for that weight until the robot could move properly.

4.2.2.3.3.2 Torque & Speed Test Certification

Table 4.2.2.3.3.2 below shows the results of the Torque & Speed test.

Table 4.2.2.3.3.2.a: Weight Test Results

Weight (pounds)	Speed (Signal Period in ms)	Result
3	8	Pass
5	8	Fail
6	8	Fail
8	8	Fail
11	8	Fail
5	10	Pass
6	10	Pass
8	10	Fail
11	10	Fail
8	12	Pass
11	12	Pass

Each of the weights listed above were run through the course multiple times. It was found that the robot's stepper motors could move up to 11 pounds with a signal of a period of 12 milliseconds. Given that the upper half of the robot weighs approximately 8 pounds, this speed provides ample torque to move the robot around the game board.

4.2.2.4 Chutes: Gripping Strength / Weight Test

Since each chute holds the blocks for a single airplane they must be strong enough to hold four packages. The chute gripper subsystem strength test verifies that a powered gripper can hold at least four blocks without allowing the blocks to become dislodged while driving or twisting.

4.2.2.4.1 Chute Test 1

4.2.2.4.1.1 Chute Test Procedure

Four blocks were added to the gripper and the servo controller PIC was told to hold the gripper closed. Then the gripper was moved around randomly to see if the blocks would fall out or if it would hold the blocks in place.

4.2.2.4.1.2 Chute Test Certification

The blocks held in the gripper without falling out even when the chute was rotated. After experimenting with the servo, it was found that a servo draws between 400mA and 600mA of current (depending on the torque of the servo). In order to reduce the drain on the batteries, rubber bands were added to hold the gripper closed. The test was repeated after rubber bands were placed on the back side of the gripper. The second test was also executed while the servo was not connected to the PIC so that the gripping force could be held constant at zero. The gripper was held horizontally to demonstrate the worse case condition and the test is shown in Figure 4.2.2.4.2.

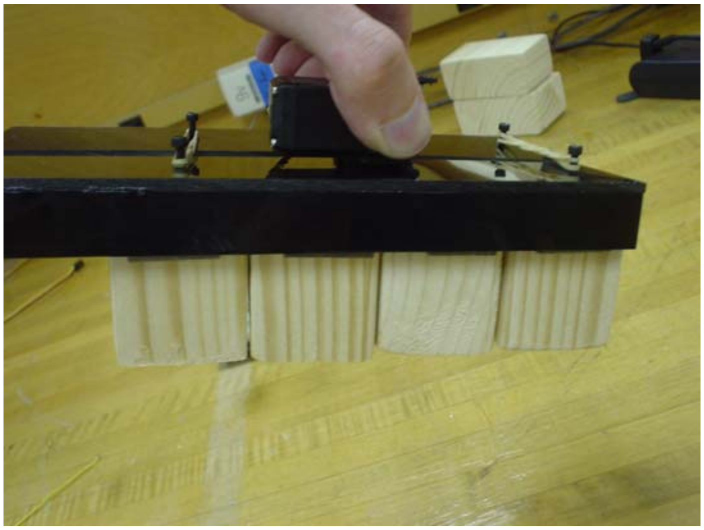


Figure 4.2.2.4.2: Gripper Holding Four Blocks with Rubber Bands

4.2.2.4.2 Chute Test 2

The stepper motors which move the chute grippers up and down need to be strong enough to raise and lower a chute while holding four blocks. The elevator weight subsystem test verifies that an elevator can run without binding with four blocks.

4.2.2.4.2.1 Chute Test Procedure

Four blocks were added to the chute gripper and the servo controller PIC told the elevator to go to the top. If the elevator bound and did not reach the top then the speed was reduced. However, if the elevator did not bind then the speed was increased until an optimal speed was determined.

4.2.2.4.2.2 Chute Test Certification

The elevator will move up and down while it is empty, but once the weight of four packages is added it starts binding and is not 100% accurate. A plate was added to hold the elevator's screw in a plumb position and this seems to help. The elevator only works correctly about 70% of the time. The speed is determined by time between pulses and each pulse is equivalent to a step. The elevator can run down at a maximum speed of 7 and up at 20.

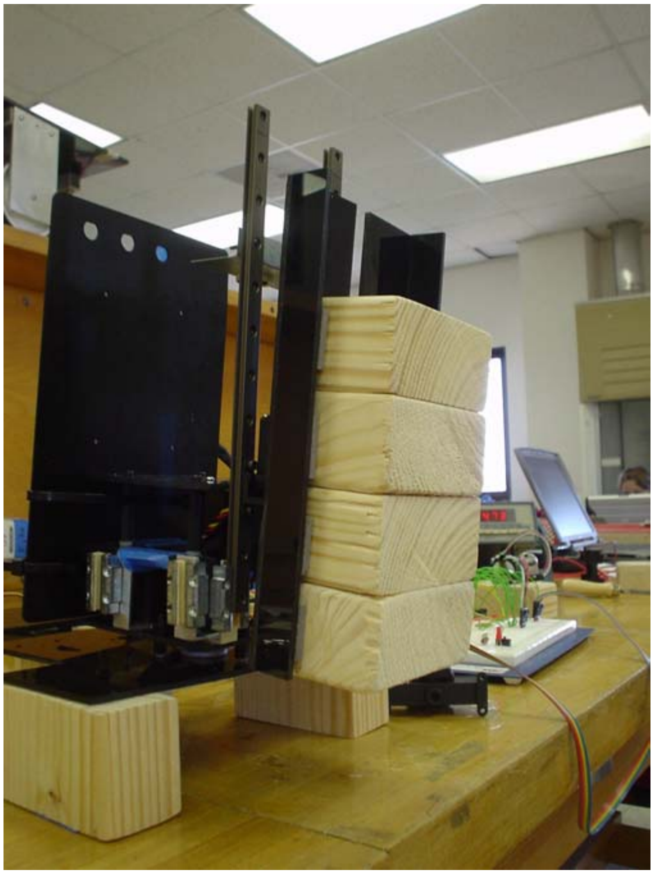


Figure 4.2.2.4.4: Elevator Weight Test

If the barcode reader makes an incorrect read then a block will be placed into an inappropriate chute. If a block is placed into an incorrect airplane then two points will be deducted from the team's score. If the robot only picks up four blocks into a single chute and one of those blocks is destined for an incorrect plane, then the robot is not able to gain all the possible points for the airplane. The overall score of the round will be higher if a package is placed into an incorrect airplane along with the four correct packages, so each chute has been designed to hold up to six wooden blocks. Testing with six packages in a chute has not been accomplished yet because four blocks are binding the elevator every so often and if four blocks does not work then six will not work.

4.2.2.5 Sonar Sensors

Sonar sensors are vital to approach the airplanes as close as possible without moving them out of position. Thus, it is necessary to confirm that they return distances that are accurate enough to accomplish this goal.

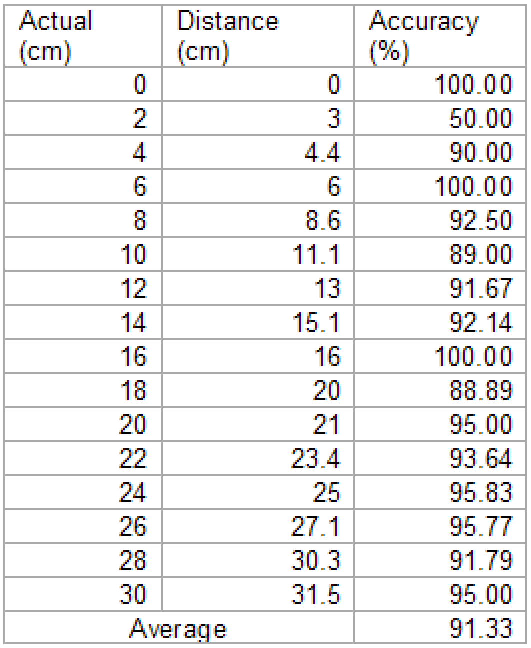
4.2.2.5.1 Testing Procedure

The accuracy of the sonar sensors was measured by first mounting a single sonar sensor to a vertical board. Next, a movable object was placed directly in front of the sensor. The sensor was then triggered to return distance from the object. The sensor's output was then compared to the actual distance to calculate the error in the measurement. This procedure was repeated for various distances.

4.2.2.5.2 Test Certification

The sensors were tested at distances ranging from 0cm to 30cm. As depicted in Table 4.2.2.5.2.a, the sensors had an average accuracy of 91.3%. As shown in Figure 4.2.2.5.2.a, the sonar modules had little variation from the actual distance being measured.

Table 4.2.2.5.2.a: Sonar sensor accuracy results



Actual (cm)	Distance (cm)	Accuracy (%)
0	0	100.00
2	3	50.00
4	4.4	90.00
6	6	100.00
8	8.6	92.50
10	11.1	89.00
12	13	91.67
14	15.1	92.14
16	16	100.00
18	20	88.89
20	21	95.00
22	23.4	93.64
24	25	95.83
26	27.1	95.77
28	30.3	91.79
30	31.5	95.00
Average		91.33

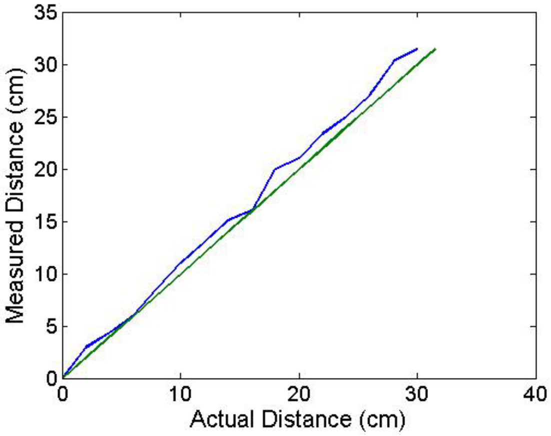


Figure 4.2.2.5.2.a: Sonar Sensor Accuracy Plot

4.2.2.6 Line Sensors

Line sensors are used to accurately follow the white paths leading to the preset destinations. The sensors must be able to distinguish black and white shades on all possible surface areas. Various tests are performed to test the reliability of the line sensors. The testing procedure is denoted below.

4.2.2.6.1 Test Procedure

The sensitivity of the line sensors was tested to determine the necessary pull-up resistance. This was accomplished by replacing the 330Ω resistor with a 1kΩ potentiometer as shown in Figure 4.2.2.6.1.a. The output voltage of the line sensor was monitored while the potentiometer was swept over an interval from 0Ω to 500Ω.

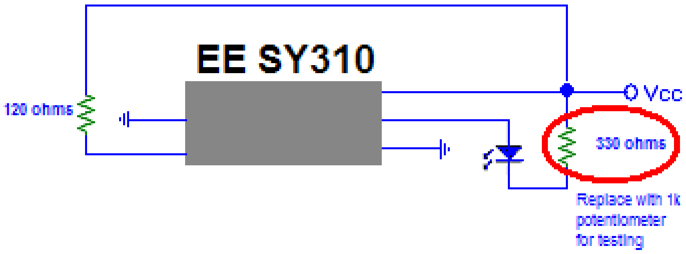


Figure 4.2.2.6.1.a: Line Sensor Sensitivity Test Schematic

The range of the line sensors was tested by monitoring the voltage of the output terminal while the sensor was raised from 0mm to 10mm above the ground. This test determined the necessary height required by the line sensors to obtain accurate readings.

4.2.2.6.2 Test Certification

The results of the sensitivity test are located in Table 4.2.2.6.2.a and Figure 4.2.2.6.2.a. As shown, the voltage output of the line sensor when over black falls below 4.7V at 350Ω. Therefore, all resistances greater than 350Ω are invalid for the application. Also, the output of the sensor when over white does not fall to 0.10V until 300Ω. This negates using any resistances less than 300Ω. Consequently, the optimal range of values for the pull-up resistance is 300Ω to 350Ω.

Table 4.2.2.6.2.a: Line Sensor Sensitivity Test Results

Resistance (Ω)	White (V)	Black (V)
0	4.57	4.74
25	3.21	4.75
50	1.69	4.73
75	0.29	4.73
100	0.22	4.75
125	0.27	4.73
150	0.22	4.72
175	0.19	4.74
200	0.17	4.74
225	0.15	4.74
250	0.14	4.71
275	0.13	4.71
300	0.12	4.72
325	0.11	4.7
350	0.11	4.72
375	0.1	4.67
400	0.09	4.66
425	0.09	4.63
450	0.09	4.65
475	0.08	4.63
500	0.08	4.63

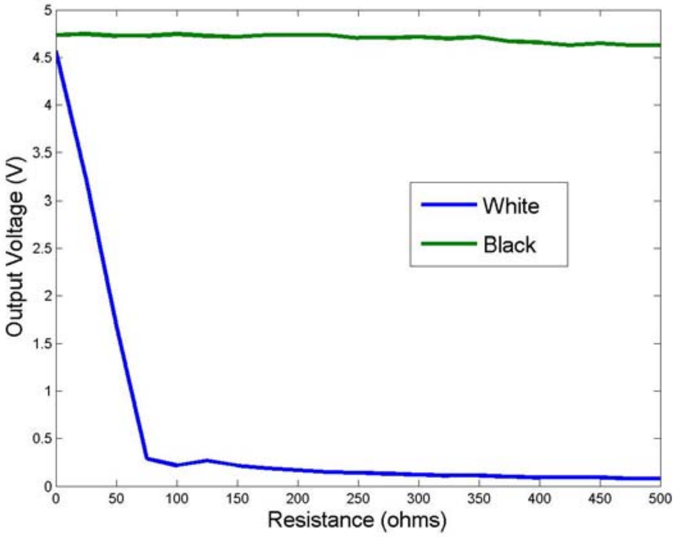
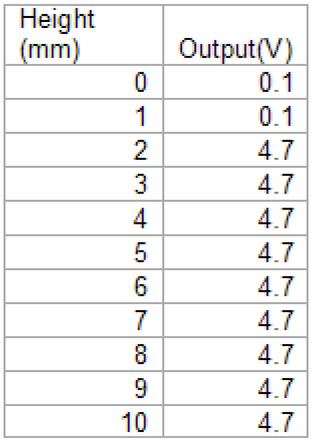


Figure 4.2.2.6.2.a: Line Sensor Sensitivity Test Plot

The results of the range test are shown in Table 4.2.2.6.2.b. The table reveals that the sensors need to be placed at any distance between 2mm and 10mm.

Table 4.2.2.6.2.b: Line Sensor Height Test Results



Height (mm)	Output(V)
0	0.1
1	0.1
2	4.7
3	4.7
4	4.7
5	4.7
6	4.7
7	4.7
8	4.7
9	4.7
10	4.7

4.2.2.7 Barcode Scanner

If there are no barcodes present, the scanner enters standby mode and only reenters operational mode when triggered by sufficient movement of either the scanner itself or some other object within its range. Because it cannot be guaranteed that movement of the extractor claw or the movement of the packages as they are being extracted from the package chute will trigger the scanner to enter operational mode, it is necessary to be able to control this feature of the scanner from software. For the sake of power consumption, it is also desirable to be able to be able to power the scanner on and off with software.

In theory, once powered on, the scanner begins in operational mode and will immediately read any barcode present. To allow the microprocessor to control power to the scanner, an NPN transistor circuit was added to the hardware that will allow a logic ‘high’ from the microprocessor to power the scanner on and a logic ‘low’ to power the scanner off. The figure below shows the added NPN transistor circuit to the scanner interface.

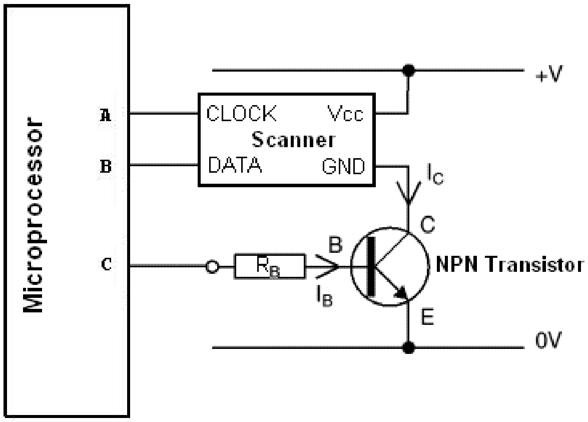


Figure 4.2.2.7.a: Scanner Interface with NPN Transistor Circuit

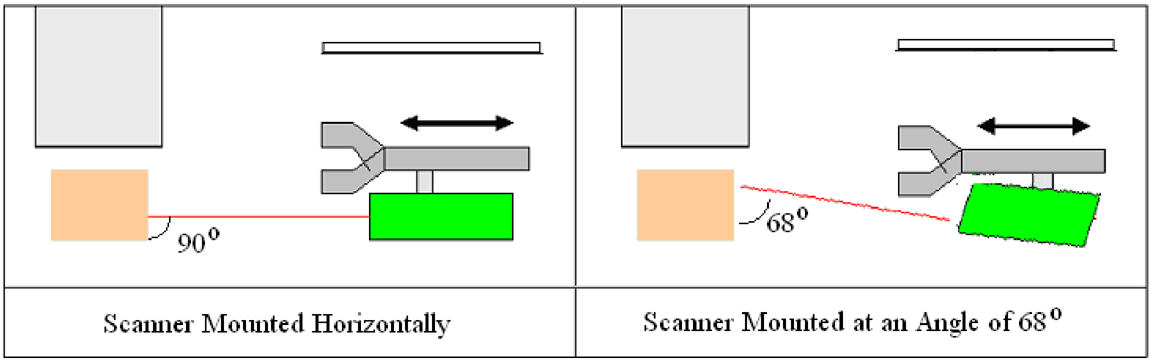
This test will initially be tested prior to mounting the scanner onto the extractor to prove that the scanner will power on when given a logic ‘high’ signal from port C, power off for a logic ‘low’ signal, and immediately read any barcodes within its range. After the initial test is complete, the scanner will then be mounted to the extractor claw and the procedure performed again to ensure the scanner can read barcodes from its mounted position.

4.2.2.7.1 Barcode Scanner Test Procedure

The purpose of this test is to confirm if the scanner can immediately read any barcode present once powered on from software. To begin, the scanner interface shown in Figure 4.2.2.7.a was implemented. A simple program that will toggle Port C from logic 'low' to 'high' and vice versa at the push of a button was programmed onto the microprocessor. Lastly, a barcode was placed within the scanner's range and the data and clock lines were configured to be monitored with an oscilloscope.

4.2.2.7.2 Barcode Scanner Test Certification

The power and scan test described in Section 4.2.2.7.1 was initially performed prior to mounting the scanner onto the robot. From the initial test, it was found that the addition of the NPN transistor circuit operated adequately as a switch and that the scanner immediately scanned any barcode within its range when powered. A second test was performed after the scanner was mounted to the robot's extractor claw, during this test the scanner would power on and off as required but was unable to read the majority of the barcodes within its range. It was diagnosed that because the scanner was mounted horizontally beneath the extract claw, the emitted laser was perpendicular relative to the face of the barcode and somehow unable to receive any reflection. The problem was remedied by mounting the scanner such that the emitted laser was approximately 68^{o} relative to the face of the barcode as shown in Figure 4.2.2.7.2.a.



Scanner Mounted Horizontally

Scanner Mounted at an Angle of 68^{o}

Figure 4.2.2.7.2: Mounting of the Barcode Scanner and Resulting Laser Angle

4.2.2.6 Dimensions Test

For the robot to be eligible to compete in the competition, it must fit within the starting size requirements: 8" x 8" x 12".

4.2.2.6.1 Dimensions Test Procedure

To ensure that the robot fit within the size constraints, a box was constructed from plywood. The volume of this box was equivalent to the volume of the size requirement of the competition robot. This box was then place over the robot to ensure that it fit.

4.2.2.6.2 Dimensions Test Certification

When the box was placed over the robot, the robot fit well within the dimensions. Hence, it was determined that the dimensions constraint was met.

4.2.2.7 Weight Test

The robot must weigh no more than 20 pounds. This constraint does not come from SECON rules but from practicality.

4.2.2.7.1 Weight Test Procedure

The robot was placed on a scale and its weight was measured.

4.2.2.7.2 Weight Test Certification

The robot was found to weigh 13.3 pounds. Hence, the robot met the weight constraint.

4.3 Detailed Description of Systems

4.3.1 Power System Test

Each of the many powered devices on the robot is grouped with similar devices into one of four categories. Each of these four categories: digital devices/sensors, servos, locomotion stepper motors, and linear stepper motors, has its own voltage rail and set of batteries. Each battery supply must run its group of devices for at least five minutes, the maximum length of a competition round.

To verify the reliability of the robot's power system, each category of devices will be tested separately under conditions designed to produce above-average current drain. The test conditions (program) for each category of devices are listed below in Table 4.3.1.a.

Table 4.3.1.a: Power Supply Test Programs

Device Category	**Supply Voltage**	**Power Supply Test Program**
Digital Devices/Sensors	7.2V	The robot scans a barcode every five seconds while also continuously taking readings from sonar and line sensors. At the same time, the slave PIC supplies PWM outputs that would rotate five servos between 0 and 90 degrees every five seconds. Correct operation is verified by correct readings from the scanner and all sensors, as well as the movement of servos connected to the slave PIC.
Servos	4.8V	Each storage chute gripper holds four blocks in the raised position while the turntable rotates to a random position (A, B, or C) every five seconds. At the same time, the extractor gripper holds a block being pulled back by a rubber band. Correct operation is verified by noting whether all blocks are held in place.
Locomotion Stepper Motors	7.2V	The robot continuously drives in a set loop around the playing field at its normal speed, stopping for ten seconds one time per loop. Correct operation is verified by completion of a loop in a time less than 120% that of the first loop.
Linear Stepper Motors	7.2V	While two elevators hold steady the weight of four blocks, the third continuously moves the weight of four blocks up and down the distance required by normal operation. At the same time, the bottom linear actuator extends and retracts against the force of a rubber band. Correct operation is verified by continuous smooth movement of all linear stepper motors.

With these test programs defined, testing of each device category can be carried out in a standard fashion.

4.3.1.1 Power System Test Procedure

To test each power supply, a freshly charged pack of batteries of the appropriate voltage will be connected to the power rail corresponding to the device category under test. Fresh batteries will also be connected to any other devices that will be required to run the test (PICs, servos, etc.), while other extraneous devices will be disconnected. The appropriate program from Table 4.3.1.a will then be run continuously until the device group fails. A multimeter will be connected to the battery supply to monitor the supply voltage during the test.

4.3.1.2 Power System Test Certification

Because of binding issues with both top and bottom linear actuators, the power system tests have been postponed. Once the binding issue is resolved and any necessary changes in power supply voltages have been made, all power supplies will be tested to ensure each will last for at least five minutes under the above-average drain conditions.

4.3.2 PCB Test

To prevent any catastrophic PCB failures and possible damage to circuit components, a few simple tests were run on each populated PCB before it was put into use.

4.3.2.1 PCB Test Procedure

Before any power was connected to the board, the resistance between all power rails and ground was measured to ensure that no shorts existed. Next, power was connected to the digital devices and sensors voltage rail and the rail probed to verify a steady 5V output from the voltage regulator. Finally, MPLAB was used to connect to both PICs via ICD2 to make sure they can be successfully programmed.

4.3.2.2 PCB Test Certification

One fully populated and two partially populated PCBs were verified using the above test. However, a partially populated board that was accidentally burnt during soldering was determined to be unusable after several test failures.

4.3.3 Loading System Test

The entire package loading system must operate reliably and quickly. In order to meet the goals set out in the robot’s design constraints, all 12 packages must be loaded onto the robot in under two minutes. The loading system test verifies that all loading components work together to accomplish this task in the given time.

4.3.3.1 Loading System Test Procedure

The robot will be programmed to complete a full competition run and supplied with fresh batteries. Once the robot comes to a complete stop at its loading position in front of the package chute, a timer will be started. The timer will be stopped once the robot begins moving away from the chute.

4.3.3.1 Loading System Test Certification

Binding problems with the linear stepper motors currently prevent the robot from loading all 12 blocks on its own. Once this problem is solved, the loading system test will be run to verify that the loading and sorting process takes meets the design constraint.

4.3.4 Turntable Movement

The turn table is a major component in the effectiveness of the robot. If the turntable does not align itself properly then the chutes will not be inline to grab a block properly.

4.3.4.1 Turntable Test Procedure

The slave PIC was used to move the turntable servo to each of its three positions. When the position was not correct, the angle was changed to align the chute with the front position until it was able to grab a block.

4.3.4.2 Turntable Test Certification

The turntable was able to turn 187 degrees and should be enough to accommodate for all three chutes. The turntable test was fairly successful. It was determined that the turntable could be accurately moved to the correct position, but cannot move to the location necessary for the chute to be aligned. Chute B is set to nine degrees and that makes Chute C have to be set to a minimum of 189 degrees if the chutes are all made perfectly. Since the turntable cannot be moved farther than 187 degrees, Chute C is not positioned correctly.

4.3.5 Gripper Movement/Strength

The bottom gripper has to be strong enough to pull a block out of the chute without letting it slip out of its grip. The worst case scenario is when there are twelve blocks in the chute because the pressure on the bottom block is the greatest.

4.3.5.1 Gripper Test Procedure

The gripper and linear actuator were set up on a temporary deck that allows them to pull a block out of the FedEx chute. The linear actuator was sent out and the gripper was told to grab the block. Then the linear actuator was told to retract. If the gripper was strong enough to hold the block then the block should be pulled out.

4.3.5.2 Gripper Test Certification

The block stayed in the chute because of the friction between the gripper and the block, so thumb tacks were added to the gripper. After the tacks were added then the block was successfully taken out of the chute. The strength of the gripper was determined to be great enough.

References

[4.1] "Bimos II Unipolar Stepper Motor Translator/Driver," 10 Oct 2005. [Online] Available: http://www.ece.msstate.edu/courses/design/ece4512/2005_fall/secon_II/docs/documentation/IC_5804.pdf

Appendix A: Distance Travel Software Test Results

Given Distance (in inches)	Computed Steps	Hand Calculation	Accuracy Percentage
1	21	21.2	98.9
2	42	42.5	98.9
3	63	63.7	98.9
4	84	84.9	98.9
5	106	106.2	99.9
6	127	127.4	99.7
7	148	148.6	99.6
8	169	169.9	99.5
9	191	191.1	100.0
10	212	212.3	99.9
11	233	233.5	99.8
12	254	254.8	99.7
13	276	276.0	100.0
14	297	297.2	99.9
15	318	318.5	99.9
16	339	339.7	99.8
17	360	360.9	99.7
18	382	382.2	100.0
19	403	403.4	99.9
20	424	424.6	99.9
21	445	445.9	99.8
22	467	467.1	100.0
23	488	488.3	99.9
24	509	509.6	99.9
25	530	530.8	99.9
26	552	552.0	100.0
27	573	573.2	100.0
28	594	594.5	99.9
29	615	615.7	99.9
30	636	636.9	99.9
31	658	658.2	100.0
32	679	679.4	99.9
33	700	700.6	99.9
34	721	721.9	99.9
35	743	743.1	100.0
36	764	764.3	100.0
37	785	785.6	99.9
38	806	806.8	99.9
39	828	828.0	100.0
40	849	849.3	100.0
41	870	870.5	99.9
42	891	891.7	99.9
43	912	913.0	99.9
44	934	934.2	100.0
45	955	955.4	100.0
46	976	976.6	99.9
47	997	997.9	99.9

48	1019	1019.1	100.0