Lab 6 Competition Robot

This laboratory assignment accompanies the book, Embedded Microcomputer Systems: Real Time Interfacing, by Jonathan W. Valvano, published by Thomsen Publishing, copyright © 2006. The robot materials were funded in part by a grant from Tivioli.

Goals
- Design a robot that can move forward/backward, turn left/right, and collect/shoot balls,
- Interface motors, and sensors to the 9S12,
- Implement pulse-width modulation, input capture and output compare,
- Write low-level device drivers for the motors and sensors,
- Develop a high-level control system,
- Use communication skills to work effectively as team.

Review
- Chapter 8 on interfacing DC and stepper motors,
- Chapter 6 on pulse-width measurement and input capture,
- Chapter 2 on finite state machines,
- Chapter 13 on control systems,
- Construction guide for the erector set,
- Data sheets for GP2Y0A21YK, IRF540, L293, Ping))) and TIP120.

Starter files
- PWM, OC, IC, ADC, FuzzyLogic, LCD, Moore

To win this competition, your team must combine mechanical, electrical and computer skills. Each robot has one switch (in addition to the reset switch) that the students will push to start the competition when instructed by the TA-referee. The **reset** switch (the normal hardware reset) should cause all motors to stop and will be used for safety purposes. You will push your **start** switch once at the start of the competition signifying the start of the 180-second competition. You are not allowed to touch or move your robot any time during the 180-second competition. The robot must run autonomously; there will be no wireless input commands or buttons are pushed during the competition. There will be 30 bonus points (10%) awarded for running the robot software on top of the Lab 5 OS.

The Arena

Figure 6.1 shows the arena, which is about 4 feet by 8 feet, flat, and divided in half by a 1.5-inch high wall, called the *net*. The *orientation lines* are made with black tape of different widths. The tape has a different optical reflectance than the white floor. The *side walls* and the *end wall* are about 3.5 inches tall. The *walls* are gloss white like the floor. The ball will bounce off the *side walls*. All dimensions are approximate; please refer to the actual arena for more accurate measurements. There are two zones: your side and the opponent's side.

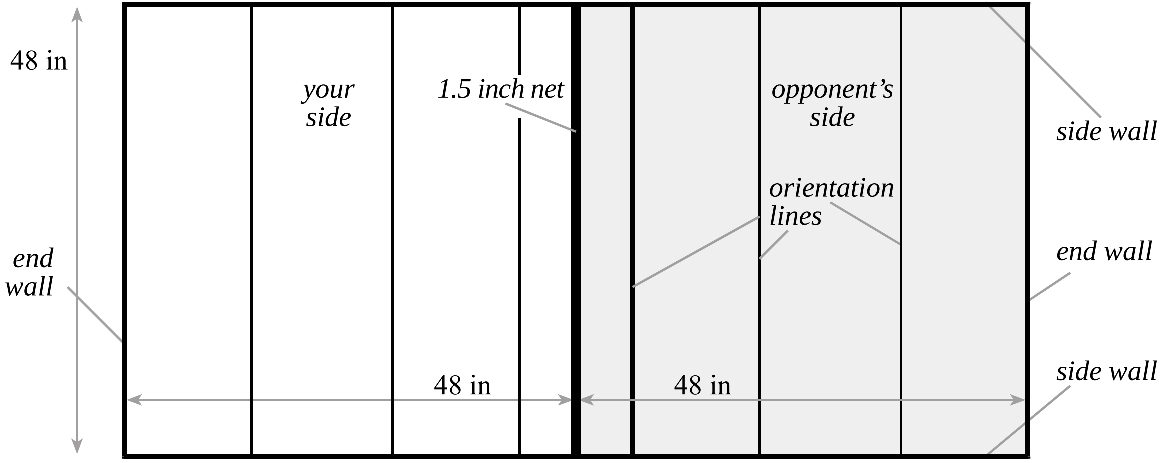


Figure 6.1. Rough sketch of the arena (the grey lines do not exist).

The Beginning and End

Two robots compete against each other, and the starting sides will be chosen by the TA-referee. This information will be announced during the previous game so teams may prepare their robots. The starting position of a robot defines its side of the arena for scoring purposes. The students will have 60 seconds to place their machines into the arena from the time the TA-referee calls them to set up. Your robot must be touching your *end wall* at the start, but its lateral position and orientation are up to you. When the two teams are ready, the TA-referee will randomly place seven balls in the arena, three on each side and one on the *net*. Next, the TA-referee says, "Go", and each team will push its **start** button to activate its software. If a robot fails to start as expected, the TA-referee will award one *false start*, and the round will be repeated. Two *false starts* constitute a loss, and the faulty robot will be removed from the arena and placed in a position of shame. The remaining robot will be allowed to play without opposition. The powered portion of a competition will last 180 seconds. Your robot must stop all moving functions at the end of 180 seconds, and any robot that continues to move after 180 seconds will lose two points. The TA-referee will terminate a game early if neither robot appears to be making any progress. The TA-referee will terminate a game immediately if a robot presents physical danger to spectators, itself or the opposing robot.

The Scoring

Your robot scores two points if your robot causes a ball to go over the *net* and touch the *end wall* on your opponent's side of the arena. If your robot shoots the ball over the net, and it hits the *end wall* multiple times, only one two-point score will be awarded. No points are awarded for balls that hit the *end wall* that were not launched over the net. In other words, if you cause a ball to hit your own *end wall*, no points are awarded. If one moving ball hits a second ball, and the second ball hits the *end wall*, no points are awarded. If your robot shoots a ball over the *net*, and the ball comes to a stop, the possibility for a 2-point award is ended. No points are awarded if your robot causes the ball to leave the arena, even if it hits the *end wall*. If the score is tied at the end of the 180-second competition, the win goes to the robot with the fewest balls on their side. The balls are not to be touched by students. After your robot makes a +2 point score, the TA-referee will place the ball in the middle your arena. If your robot causes a ball to leave the arena, even if it bounces over an opponent's wall or off the opponent itself, the TA-referee will place the ball in your back corner. If you design your robot in such manner that it deflects a rolling ball over the wall, your robot will be considered to have "caused" the ball to leave the arena. At the end of the 180-second competition, your robot must stop all moving parts. After 180 seconds, the game ends, and no more points are awarded. Balls that hit the wall or fly over a wall after the 180-second competition do not result in points. The robot with the most points wins.

+2 points if your robot shoots the ball over the *net* and the ball hits your opponent's *end wall*
-1 point if you cause the ball to leave the arena (ball placed in your back corner)
-2 points if your robot does not stop moving all functions after the 180-second competition.

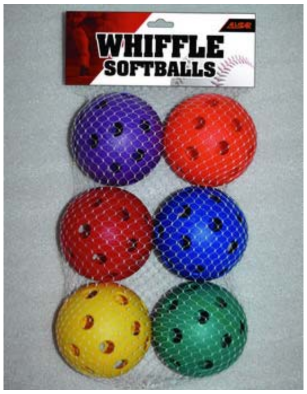
The Placement of the Balls

At the beginning of the game, you first place your robot into the arena, touching your *end wall*. Next, the TA-referee places the balls in such a manner that no robot has a clear advantage over the other. Sometimes the weather is windy and sometimes the tables not perfectly balanced. In these cases, the TA-referee will attempt to make the competition as fair as possible. The TA-referees will decide their own method of initial ball placement. He could toss the balls one at a time into the arena in a manner that causes it to hit at least two walls before stopping. The TA-referee may simply place the balls on the table using a random, but fair pattern. You will not know the initial pattern of the balls, except there will be one on the *net* itself. If your robot causes a ball to leave the arena, then the TA-referee will place the ball on your side at a place difficult for you to find (e.g., back corner).

The Strategy and Sportsmanship

The basic goal is to locate the balls and send them over the *net* to the other side of the arena. The robot is supposed to move in an intelligent manner. Your TA will not allow unintelligent robots to compete. If you are not sure what intelligent means, run your strategy by your TA. Robots designed to cause the ball to leave the arena will be disqualified. Bouncing that occurs after the ball hits a wall or an opponent is not penalized. Robots must move and react to the ball. No pieces may be intentionally dropped by your robot. The balls, the opponent's robot, and the arena may not be altered in any way. Pieces unintentionally dropped by your robot will be removed by the TA-referee. Good strategy, skillful ball handling, knowledge of your position/orientation, and reliability will be important. Because sensors can be unreliable, an effective solution will be robust, so that the robot acts in an appropriate manner even when presented with inaccurate sensor data. Your robot is allowed to squat, scoop, shoot, push, and kick the ball during the game. During the qualifying and preliminary rounds, the judges may declare a double loss, a double win, or no result, as appropriate. Good sportsmanship must be followed.

The Ball



The hard plastic "Whiffle Softballs" have a diameter of about 9 cm. This ball was designed for soft-toss softball batting practice. The balls can be reliably seen at distances from 10 to 50 cm (in a 9-cm wide rectangle) using the IR range sensor (GP2Y0A21YK). Because of the hard plastic and curved shape, it is difficult for the ultrasonic sensor to see the ball. The color of the ball may vary from match to match, but the IR range sensor seems to work similarly for the different color balls. The ball will roll and will bounce off the walls reasonably well. The ball may not be punctured, altered, or modified in any way.

The Robot

Each team will choose a name for its robot. Please limit your name to 20 characters including spaces. Robots will be constructed using materials from the robot kit. Input devices include one ultrasonic ranging sensor, three IR range sensors, four tactile touch sensors, and four reflective IR optical sensors. Motors include one servo, four geared DC motors, two high-RPM low-torque DC motors, two stepper motors and one solenoid. The *bounding box* is defined as the smallest 3-D box with 90 degree angles that could contain all the hard components of your robot. Any object with a well-defined volume is defined as hard, including grabbers, sensors, kickers, flippers and wheels. The *footprint* of your robot is defined as the floor of its bounding box, as shown in Figure 6.2. The footprint must be less than or equal to 9.5 inch by 9.5 inch at all times during the competition. Soft components of your robot such as rubber bands, string, and piano wire (whiskers) may extend 2 inches beyond the footprint. The maximum wheel diameter is 3.25 inches (to limit torque to the motors). The maximum *height* of the bounding box is 12 inches at all times during the competition. In other words, your robot can not unfold so that is becomes larger than the maximums of 9.5 inch wide, 9.5 inch long, and 12 inch high. It is acceptable if two inches of your robot extend into your opponent's half of the arena. Your wheels must be on the ground at all times. You can not go over the *net* during the competition.

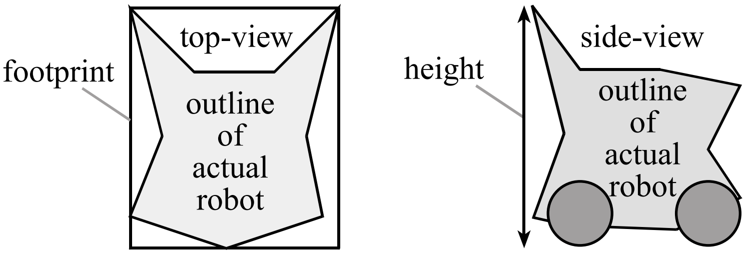


Figure 6.2. Definitions of footprint and height.

Three Laws of EE345M Robotics:

- A robot may not harm a human being, or, through inaction, allow a human being to come to harm.
- A robot must obey the orders given to it by TA-referees: John, and Lev.
- A robot must protect its own existence and the existence of the other robots, as long as such protection does not conflict with the First or Second Law.

The Judges Rule

- The judge is Valvano.
- Contestants should ask the judge about possible designs or strategies that may be questionable under any of the rules.
- Contest rules and procedures, or even the game, may have to be altered during the semester. As much notice as possible will be given.
- The judge may alter or eliminate any rule, or add rules, at any time.
- The TA-referees will decide any discrepancies in the contest play.
- All decisions of the TA-referees are final.

The Competition Format

The contest is a three phase competition, which are qualifying, preliminary, and finals. During the qualifying round, which will demonstrated to any TA or instructor during any regularly scheduled lab on or before

Friday May 1, your robot must "beat the brick" —win a game played against an inert opponent— in order to qualify for the next round. Qualified machines will then play three or four preliminary competitions, arranged at random. Preliminary competitions occur in the second floor lab on Monday 5/4 6:30-8pm. The results (+1 for a win, 0 for a tie, and -1 for a loss) of these preliminary games will be recorded and used only for seeding into the final competition. The total number of points scored during the preliminary rounds will be used as a tie-breaker in the seedings. The final competition will take place just outside the front door of ENS on Wednesday 5/6 and continue on Friday 5/8 10-11am. The final competition will be double-elimination, and the exact pairings will depend on the number of qualifying robots. During the finals, the TA-referee will award exactly one winner and one loser for each round. Depending on the number of qualifying robots, some robots with a high seeding may be awarded a bye for the first set of games, as shown in Figure 6.3.

Schedule (three arenas)

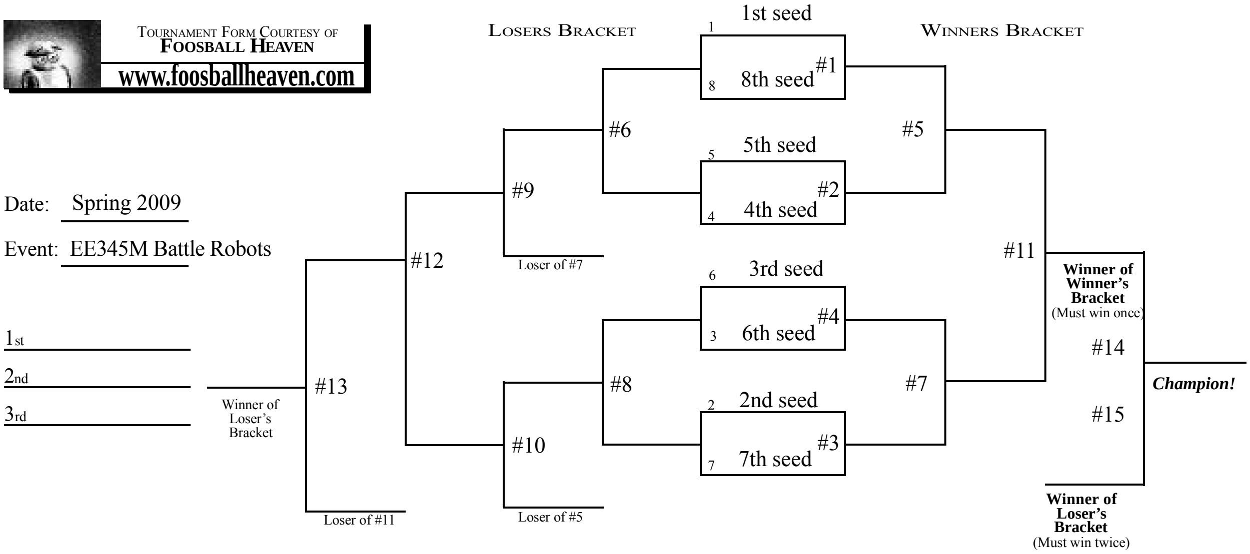


Figure 6.3. A double elimination finals with 8 robots will require 14 or 15 games.

Wednesday May 6, 10am
10:08, Game #1, Arena A
10:08, Game #2, Arena B

10:16, Game #3, Arena A
10:16, Game #4, Arena B

10:24, Game #5, Arena A
10:24, Game #6, Arena B

10:32, Game #7, Arena A
10:32, Game #8, Arena B

10:40, Game #9, Arena A
10:40, Game #10, Arena B

Friday May 8, 10am
10:08, Game #11, Arena A
10:08, Game #12, Arena B

10:16, Game #13, Arena B

10:24, Game #14, Arena A

10:32, Game #15, Arena A (if necessary)

Restrictions

Teams are encouraged to seek advice and help from any source, but the design, robot construction, and software must be exclusively their own work. All entries must be solely controlled by their onboard 9S12 computer(s). You may use any 9S12 board (e.g., 9S12C32, 9S12DP512, 9S12E128). It is also ok to employ multiple 9S12 controllers if you need more pins or more processing power. There can be no human intervention once the game begins. A robot that is moved by a team member during a game will be disqualified for that round. Also, team members touching the balls or the arena during play risk disqualification of their robot. Teams may qualify only one robot, and only that robot may play in the preliminary and final rounds. Modifications to your mechanical, electrical or software components following qualification are encouraged. Repair and reprogramming are allowed during the competitions only if time permits. A robot cannot be designed primarily to destroy the physical structure of its opponent. In particular, robots are not allowed to destroy their opponent's microprocessor board, wheels, power, or sensors. Jamming the opponent's sensors with light or sound is not allowed. However, stopping the ball before it hits your *end line* is an obvious strategy. No parts or substances may be deliberately dumped, deposited, or otherwise left to remain on the arena surface. A machine that appears to have been designed to perform such a function will be disqualified. Pieces that accidentally fall off robots will be removed from the arena during a game by the TA-referee at his discretion. No devices that move air (e.g., fans) are allowed. No adhesives or sticky substances (such as glue or tape) may be applied to any part of your robot, the opponent's robot, the arena, or the balls. Any machine that appears to be a safety hazard will be disqualified from the competition.

Robot Structure Restrictions

The kit has a lot of motors, but you will be allowed to purchase and use additional electro-mechanical components (motors, steppers, servos, solenoids) as long as they are powered from the one 8.4V battery. Additional electro-mechanical devices can be borrowed from the professor as long as there is sufficient quantity for all teams to have similar quantity. There is a box of random structural components (erector pieces), which may also be borrowed starting 1 week after you receive your kit. Teams may possess only one battery at a time. All kits contain approximately the same components, except there is variability in the optical and touch sensors. Electrical components from the second floor may be obtained (such as resistors, capacitors, transistors and IC chips). No batteries, servos, solenoids, or motors from the second-floor checkout may be used (because it is unlikely there will be sufficient quantity for everyone). You may not employ devices to increase the voltage above the 8.4 volts provided by the battery (e.g., no DC-DC converters). You can have no electrical, mechanical, chemical, or nuclear power source other than the provided 8.4 volt battery (e.g., no CO_2, butane, or gasoline may be used as a power source). You may use lubricants on the metal parts (not in the motors or sensors), as long as all components are cleaned prior to returning the kits. Rubber bands may be used as structural components (holding things together), but not used as an initial power source. For example, the motor can be used to transfer energy from the battery to a spring or rubber band. The spring or rubber band may then cause a flipper, a leg, or an arm to move or return back. In other words, humans can not preload potential energy into a spring or rubber band prior to the competition.

The use of temporary binding components such as nuts, bolts, rubber bands, cable ties, string, and wire is encouraged, but please don't use methods that permanently alter the pieces such as tape, hot glue, epoxy, or welding. You may solder wires to the motors, and sensors, as long as permanent damage is avoided. Drilling additional holes in the erector pieces is allowed, as long as the original usage of the piece is not altered. You may alter your own structural pieces (wood, plastic, metal) as you wish. The 9-inch aluminum base may be altered in any way you wish. Purely decorative items may be added, within the size limits and good taste.

The $50 Rule

Each team is allowed to purchase additional parts up to a $50 limit. These additional parts may include electrical components (such as sensors, transistors, amplifiers, motor drivers etc.), structural components (such as wood, metal, plastic, wheels, shafts, gears, chains, pulleys, and slip couplings), and motors (servos, solenoids, and DC motors). Repeating from above, you can not purchase or use additional batteries. Components from the second floor checkout or the instructor's office do not need to be logged and do not count towards this $50 limit. You do not need to measure the cost of readily available supplies such as nuts, bolts, rubber bands, cable ties, string, paper, cardboard, wire, your 9S12 boards, or your protoboards. To purchase an additional item, it must be readily available to all teams at that price. *Readily available* means, if it is purchased, it can be received 7 days later. Free components may be obtained as long as they are logged and equally available to all teams. The robot name, description, cost, and source must be first approved by a TA or instructor, then logged onto the Blackboard discussion forum, named *The $50 Rule*. All extra items must be logged within 5 days of its purchase. If you purchase a part, log it, then decide not to use it, the part does not count towards your $50 limit. After the component is purchased, the receipt must be shown to your TA (with date and cost). Any machines found with added parts that have not been documented in this fashion will be disqualified.

Each robot kit contains the following materials, partially donated by Tivioli
Amazon.com, Erector Special Edition Anniversary Set, 643 pieces, $109.99 each
  Case, DC motor, wheels, axles, drive belts, pulleys, large and small metal pieces, many nuts and bolts
  DC toy motor
BG Micro, www.bgmicro.com,

1, BAT1060, 8.4V/3.8AH, NiMH battery, $7 each (with charger $3) (no longer in stock)
2, MOT1050, Geared DC motor, Merkle-Korff S1627B ($7.95 each)

Parallax, www.parallax.com
1, Ultrasonic Ping))), $29.95

Sparkfun, www.sparkfun.com
3, GP2Y0A21YK IR range sensor, SEN-00242 ($9.95 each)
3, Jumper wires, SEN-08733 ($1.50 each)

All Electronics Corporation, http://www.allelectronics.com, Phone 1-800-826-5432
4, either SMS-189, MINI-SNAP-ACTION SWITCH, $0.50 each
or SMS-174, MINI-ACTION ROLLER SWITCH, $1.25 each
4, QRB1134 or 21L900, optical reflective object sensor, $1.44 each (Mouser)
1, SOL-89, Deltrol Controls #56597-60 INT. Spring-return, push-pull solenoid, $1.35

Jameco, 1-800-831-4242, www.jameco.com
2, 162191, HNGH12-1324Y Geared DC motor, 225RPM 12VDC, 65 mA, $16.95 each
2, 163395, 5017-935, 30 ohm, 280 mA, 400 steps/rotation stepper motor, $4.95 each (no longer in stock)
1, 157067, 14500 RBT geared servo motor (bag of servo parts), $17.95 each
2, 400961, 3 in Tires, Neoprene Foam Tread, 7.25 per pair
1, 358336, Ball and caster, $6.00 each

Mounting Hubs two 4mm shaft, two 5mm shaft (4mm shaft drilled to a 3/16 inch)
4, Jameco 386118, Mounting Hub, 7.99 per pair
or Robot Market Place, Part# 0-MHUB06, http://www.robotmarketplace.com, $8.49 pair

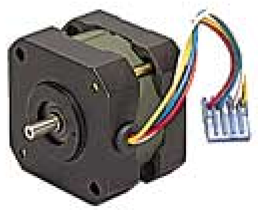
Electronics available for checkout on the second floor
IRF522, N CHANNEL MOSFET, drives up to 8A, $0.40 each
TIP120, NPN TO-220 DARLINGTON, $0.65 each
1N914, Switching diode
L293B, H-bridge driver, $1.15 each
7805 voltage regulator use with Servo

DC Motor specifications (HNGH12-1324Y)

Rated Voltage 12 VDC, 4 mm diameter shaft
No load 150 RPM at 8.4V, 50 mA (up to 500mA under load)
Operating Range 4.5 to 12 VDC
Current @ Max. Efficiency 145 mA
Speed @ Max. Efficiency 176 RPM
Torque @ Max. Efficiency 300 g-cm
Start torque: 700 g-cm
Gear Ratio: 30/1

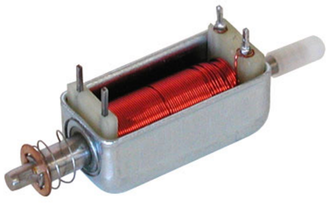
DC Motor specifications (Merkle-Korff S1627B)
240 RPM at 12V
No load 180 RPM at 8.4V, 250mA
4.8 mm (3/16 inch) diameter shaft

Stepper specifications



Step angle: 0.9 degrees
No. of phases: 2, Drive System: Bipolar
Voltage: 8.4 VDC
Phase resistance: 30 Ω
Current : 280 mA
Phase Inductance: 25 mH
Detent torque: 36 g-cm Holding Torque: 791 g-cm

Solenoid specifications
Spring-return, push-pull solenoid.
2 Ohm coil (it gets very hot with a continuous input).
Intermittent duty cycle (send very short pulses).
Metal-shielded body is 0.91" x 0.42" x 0.45" high.
0.1" diameter plunger extends 0.4" on each side of solenoid.
One end of plunger has a 0.11" diameter nylon cap.
Four pc pins on 0.75" x 0.2" centers.



HS-303 - Economy Standard Servo

For more information on servos, search the web site:
http://www.brookshiresoftware.com/

Servos are a popular mechanism to implement steering in robotics. Ranging from micro servos with 15oz-in torque to powerful heavy-duty sailboat servos, they all share several common characteristics. A servo is essentially a positionable motor. The servo "knows" two things: where it is (the actual position) and where it wants to be (the desired position). When the servo receives a position, it attempts to move the servo horn to the desired position. The task of the servo, then, is to make the actual position the desired position. The first step to understanding how servos work is to understand how to control them. All timing and electrical characteristics described here have been experimentally determined from a "HS-303 HiTec" servo. The servo is controlled by three wires: ground (black), power (red), and command (yellow). Power is usually between 4V and 6V and should be separate from system power (as servos are electrically noisy). Even small servos can draw over an amp under heavy load so the power supply should be appropriately rated. Though not recommended, servos may be driven to higher voltages to improve torque and speed characteristics. Servos are commanded through "Pulse Width Modulation," or PWM, signals sent through the command wire. Essentially, the width of a pulse defines the position. For example, sending a 1.5ms pulse to the servo, tells the servo that the desired position is 90 degrees. In order for the servo to hold this position, the command must be sent at about 50 Hz, or every 20 ms.

Position	**Pulse Width**
Minimum	0.5ms
Center	1.5ms
Maximum	2.5ms

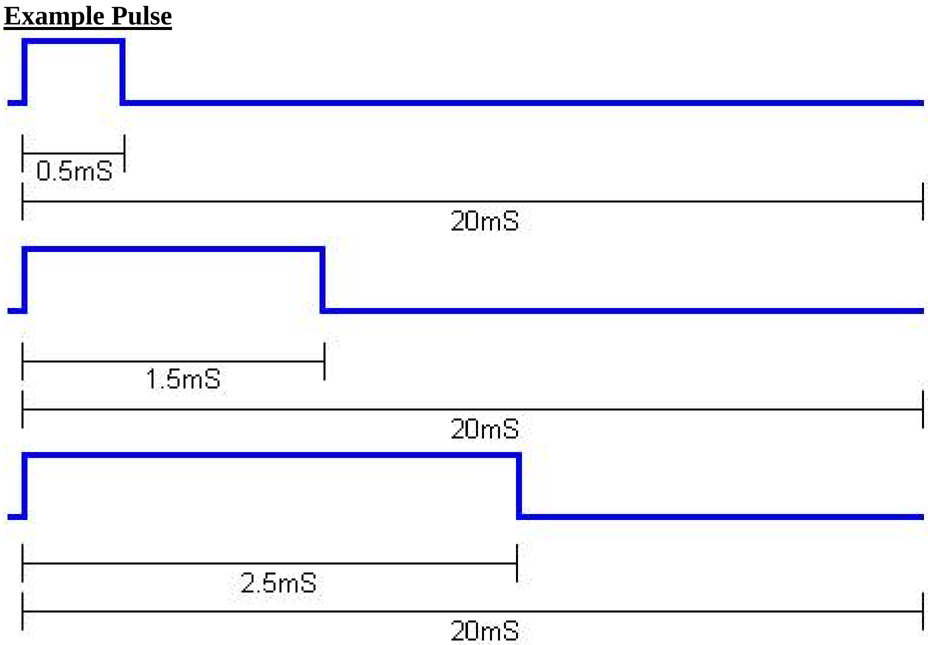


Figure 6.5. The timing constraints of the ***HS-303*** *servo (http://www.brookshiresoftware.com/).*

Figure 6.6. HS-303 servos (http://www.brookshiresoftware.com/).

If you were to send a pulse longer than 2.5 ms or shorter than 0.5 ms, the servo would attempt to overdrive and damage itself. Once the servo has received the desired position (via the PWM signal) the servo must attempt to match the desired and actual positions. It does this by turning a small, geared motor left or right. If, for example, the desired position is less than the actual position, the servo will turn to the left. On the other hand, if the desired position is greater than the actual position, the servo will turn to the right. In this manner, the servo "zeros-in" on the correct position. Should a load force the servo horn to the right or left, the servo will attempt to compensate. Note that there is no control mechanism for the speed of movement and, for most servos, the speed is specified in degrees/second. Indeed, one of the primary tasks of VSA is to synthesize servo speed control by stepping through a series of positions. *If you plan to use a servo, please add a 7805 regulator, creating two +5V sources, original one for the 9S12 and the second one for the servo.* If you connect the servo to the +5V used by the 9S12, then the servo will cause transient power losses to the 9S12, causing software resets.

Operating Voltage	4.8V		6.0V	
Torque	42 oz-in	3.3 kg-cm	49 oz-in	3.7 kg-cm
Speed @ 60 degrees	19 sec		15 sec	
	Standard		Metric	
Size L x W x H	1.6" x 0.8" x 1.4"		41 x 20 x 37mm	
Weight	1.9 oz		48.5g	

Table 6.1. HS-303 HiTec" servo specifications (http://www.brookshiresoftware.com/).

The 8.4V/3.6AH NiMH battery pack has two connectors. The most important design factor is to prevent the motor current (can be over an amp) from passing through the Adapt board (either Vin or GND). Figure 6.7 can be used to connect an Adapt C32 board to a protoboard. Figure 6.8 can be used to connect an Adapt DP512. Figure 6.8 could also be used to interface either Adapt board to a soldered or wire-wrapped board. The polarized RC-type connector (like the picture on the right) is used for charging the battery. It takes about 8 hours to fully charge the NiMH battery, using the 12V wall wart (with the matching RC-type connector), but please disconnect the other connector while charging. ***Please do not charge for more than 8 hours.*** The red (8.4V) and black (ground) wires must be connected to your protoboard for powering the DC and stepper motors (not the servo). If you plug the little connector directly into the 9S12C32 board, then interface the motors using Vin (pin 40), then motor current will destroy your Adapt board. Figure 6.7 shows the proper way to get both +8.4V and +5V power with a common ground. On the DP512, you need to plug 8.4V into the Adapt board (using the small connector) and plug 8.4V also into your motor circuit using the big connector. Be very careful that the motor current does not travel through the 9S12 board. Figure 6.8 should also be used when interfacing a servo. The 7805 and servo interface are part of the "motors" box shown in Figure 6.8.

One way to test the battery is to first drain the energy from the battery with a 15 ohm load. NiMH batteries have a uniform voltage until they are almost empty. Then the voltage drops quickly. So, once the voltage drops to 5V, we can consider the battery empty. Make sure the 15 ohm load can handle the 500 mA of current that will be flowing. It may take hours to drain the battery. Second, charge the battery for 8 hours (no more than 8). Third, discharge the battery again measuring the voltage across the load. If you use a variable resistance load, such as the motors, you should measure both voltage and current. You may not get the full 3.6 amp-hour specification, but you should get at least half this amount.

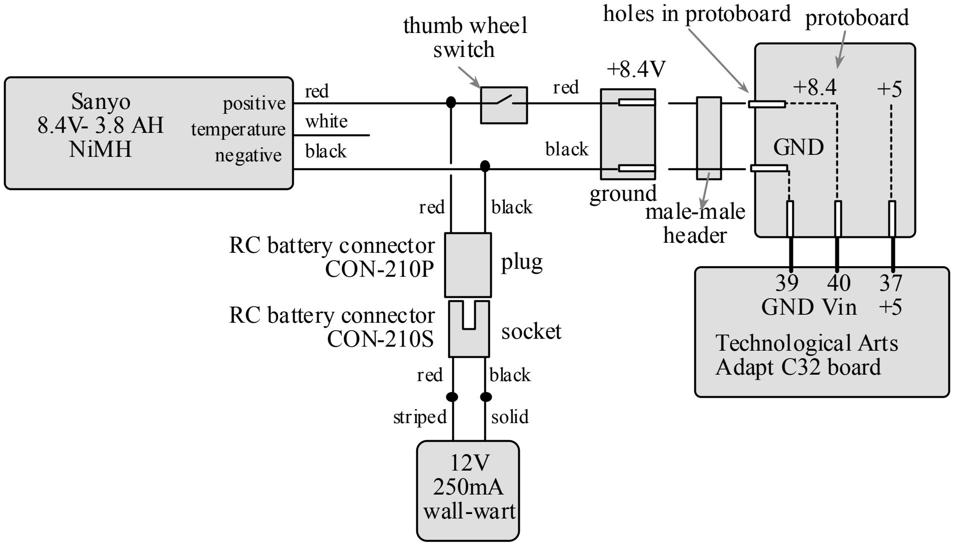


Figure 6.7. Battery connections for an AdaptC32 plugged into a protoboard.

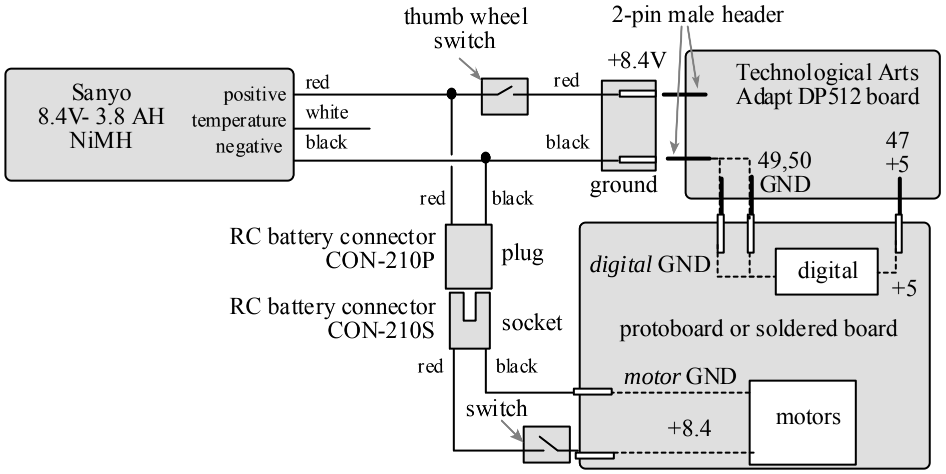


Figure 6.8. Battery connections for an AdaptDP512 board with protoboard or soldered board.

To interface the 21L900 optical sensor, connect black and green wires to ground. Connect the red wire through a 200 ohm resistor to +5V (red+black are the IR LED output). Connect the write wire through a 10k resistor to +5V (the white+green are the light sensitive transitor). The voltage on the white wire will be sensitive to optical reflectivity of the object 3-5mm from the sensor.

6A. Preparation (due Wednesday 4/8)

1: Form a robot design team with 4 members. The members do not have to be in the same lab section or have the same TA as the previous labs. The ideal team has at least one member with strengths in the areas of mechanical design (e.g., pulleys, belts, motors, and rack-and-pinion steering), electrical interfacing (e.g., transistors, currents, back EMF, servo interfacing, and sensor interfacing), power management (e.g., maintaining constant +5V power to the 9S12 while operating the motors), software design (e.g., device drivers, juggling multiple time-critical tasks, making it fit into the ROM space), high-level control, and project management (e.g., conflict resolution, report generation, and keeping on schedule). Part of your final project grade will be confidential peer evaluations, so choose your team wisely, then make a commitment to get this project finished. Please turn in the following:

robot name
first and last names of all team members
home phone numbers of all team members
email addresses of all team members

select at least two times each week for an official team project meeting (without the TA)
list all the regularly scheduled lab hours that it is possible for all your members to meet weekly with a TA
We will select one of these hours to be when your preparation and demonstrations will be due.

6A. Procedure (do this during your lab period)

1: Design and build the mechanical aspects of the robot. It must move, turn, and carry the electronics. Make a rough mechanical sketch of the robot showing how it moves, turns, carries, and senses. A detailed drawing will be required for the final report, but at this time only a rough sketch is required. I.e., just enough detail for the TA to understand your basic approach, but not enough detail for someone to build a duplicate.

2: Design the electronic interface between the steering and power motor(s) and the 9S12. We expect you to use the 8.4V NiMH battery, but you may use an external regulated supply during testing. Please test the circuitry before connecting to the 9S12. Snubber diodes must be used for all devices having an inductive load. Please consider the required current when designing the interfaces. Use a multimeter to measure the actual voltages and currents across the motors. Watch the +5V signal on the scope during times when motors are turned on and off to verify a constant +5V power supply line to the 9S12. Please thoroughly test all interfaces before connecting to the 9S12. Include low-level sensors like the wheel tachometers, used to measure wheel rotation speed.

3: Show how the system will be powered.

4: Design the low-level software drivers for the movement and steering motors. *It is a requirement that the drive motors use PI, PID, or fuzzy logic control (with sensor feedback).*

5: Write a simple high-level main program to test the movement and steering. Measure the maximum speed of the robot. Experimentally determine the best way to make turns. Calculate the accuracy of the turning algorithm. I.e., if you say turn +90, how many degrees does it actually turn? 6: Debugging a complex system like this is very hard. Please design features into the robot to assist in debugging.

6A. Deliverables (exact components of the lab report, to be included in the final 6C report)

A) Objectives (not required for this 6A)
B) Hardware Design
    1) Rough mechanical sketch of the robot (Procedure 1)
    2) Electrical circuit diagram for the motor interfaces (Procedure 2)
    3) Power supply circuitry (Procedure 3)
C) Software Design (printout of these software components)
    1) Low-level device drivers for the motor interfaces (header and code files) (Procedure 4)
    2) High-level test program to evaluate movement and steering (Procedure 5)
D) Measurement Data
    1) Give the voltage and currents of each of the motors used (Procedure 2)
    2) Give the robot speed and turning accuracy (Procedure 5)
E) Analysis and Discussion
    1) Minutes (date, time, duration, attendance, topics) for each team meeting

6A. Checkout (show this to the TA by Monday 4/20)

You should be able to demonstrate to the TA the driver motors spin at a constant speed both with the robot on the ground moving and lifted off the ground. You should demonstrate the robot moving and turning. Explain the debugging features of your device. From a software design perspective, the OS is at a low-level. Therefore, the OS needs to be designed in from the beginning, and not added on at the end. To get the bonus points for using the Lab5 OS in your robot, all three demonstrations (6A checkout, 6B checkout and the competition) must employ the OS.

6B. Preparation (do this before your lab period, but don't turn it in)

1: Develop a high-level plan of how your robot will compete. Your algorithm must involve abstractive methods. I.e., it must be layered, with a high-level algorithm separated from the low-level details of how the machine operates. A finite state machine like moore2.c is one example of how this abstraction might be implemented. You are of course allowed to develop your own approach, as long as there is a clear mapping from the high-level algorithm to the eventual C code. For example, if you decide to implement a finite state machine, draw the state graph for competition algorithm.

2: Develop an initial data flow graph of the hardware-software system. Include the motors, sensors, interface drivers, interrupt service routines, and strategic global variables.

3: Develop an initial call graph of the software system. Include foreground and background modules and their linkage (which ones call which).

4: Write the header file for the low-level sensor software driver.

6B. Procedure (do this during your lab period)

1: Design the electronic interfaces connecting the high-level sensors to the 9S12 (bumper switches, optical sensors to see lines on the arena floor, ultrasonic and compass). Again, please test the circuitry before connecting to the 9S12.

2: Design the software drivers for the detection of the walls, to avoid damaging the motors (robot safety is a high-priority process).

3: Design the software drivers for the sensors, creating software modules with prototypes for public functions in the header file and implementations of both public and private functions in the code file.

4: Write a second high-level main program to test the movement, steering and sensors. This program should demonstrate most of the middle-level building blocks that will be required for the competition. Monitor the power supply current during various operations (stopped, moving and turning).

5. Experimentally verify the robot can tell where it is on the arena.

6B. Deliverables (exact components of the lab report, to be included in the final 6C report)

A) Objectives (not required for this 6B)

B) Hardware Design

    1) Most recent electrical circuit diagram for the motor interfaces (6A Procedure 2)
    2) Most recent power supply circuitry (6A Procedure 3)
    3) Electrical circuit diagram for the sensor interfaces (6B Procedure 1)

C) Software Design (printout of these software components)

    1) Low-level device drivers for the motor interfaces (header and code files) (6A Procedure 4)
    2) Low-level device drivers for collision detection (header and code files) (Procedure 5)
    3) Low-level device drivers for the sensor interfaces (header and code files) (6B Procedure 2,3)
    4) High-level test program to evaluate movement, steering and sensing (6B Procedure 4)

D) Measurement Data

    1) Power supply current for various operations (6B Procedure 4).
    2) Accuracy of the positioning system, knowing where it is (6B Procedure 5)

E) Analysis and Discussion

    1) Minutes (date, time, duration, attendance, topics) for each team meeting
    2) List of the remaining major problems to solve

6B. Checkout (show this to the TA by Monday 4/27)

Show Lab 6B Procedure 5 to your TA. The robot should be able to go forward, turn left and turn right, moving around the arena for 180 seconds, during which time the system should be able to keep track of the position and orientation of the robot. Explain how you tested the system.

6C. Preparation (do this before your lab period, do this but don't turn it in)

1: Write the C code for the competition algorithm using abstractive methods. It should have a clear mapping from the high-level algorithm to the C code.

2: Make detailed mechanical drawings and/or photos of the robot showing how it moves, turns, carries, and senses. There should be enough detail so the robot could be duplicated.

6C. Procedure (do this during your lab period)

1: Debug the high-level competition algorithm.

2: Draw the final data flow graph of the hardware-software system. Include the motors, sensors, interface drivers, interrupt service routines, and strategic global variables.

3: Draw the final call graph of the software system.

4. Run the robot against the brick at least five times and measure your scores.

6C. Deliverables (exact components of the lab report, due Friday May 8)

A) Objectives (1/2 page maximum)

B) Hardware Design

    1) Final mechanical drawing of the robot (6C preparation)
    2) Final electrical circuit diagram for the motor interfaces (6A Section 2)
    3) Final power supply circuitry (6A Section 3)
    4) Final electrical circuit diagram for the sensor interfaces (6B Section 1)

C) Software Design (printout of these software components)

    1) Low-level device drivers for the motor interfaces (header and code files) (6A Section 4)
    2) Low-level device drivers for the sensor interfaces (header and code files) (6B Section 2)
    3) High-level competition algorithm (6C Section 2)
    4) Final data flow graph

- 5) Final call graph

D) Measurement Data
- 1) Scores during "beat the brick" competitions

E) Analysis and Discussion (1 page maximum)

F) Post-mortem concerning team member interactions (attached to the report)
- 1) Each team member evaluates each other team member including oneself
  - Simply list one or two weaknesses.
  - Simply list two or three strength characteristics.
- 2) Major failures in the way the team interacted (if any)
- 3) Major successes in the way the team interacted

G) Peer Review (each student submits independently and confidentially directly to the TA)

Classify each team member including oneself as:

- worked harder than average (explain), worked an average amount, worked less than average (explain)

6C. Checkout is the qualifying, preliminary and final rounds (show your robot to the class on demo day)

You should qualify, compete in the preliminary, then run the finals.

Lab 6 has a total of 300 points (counts for 3 labs.) Your Lab 6 grade will have five components. The first component is the demonstration of your 9S12-controlled robot that moves (6A checkout). The drive motors must be controlled using PI PID or fuzzy logic running in real time with rotational speed feedback. There will be no Lab6A report or Lab6B preparation. You will be awarded 50 points if your robot moves under 9S12 feedback control and no points if you fail to demonstrate motion by April 20. There will be no late checkouts of this part. You may demonstrate the motion to any of the TAs, or even to Valvano himself. The second component is the demonstration that your robot knows where it is (6B checkout). You will be awarded 50 points if your robot knows where it is during 180 seconds of moving and turning, and no points if you fail to demonstrate this by April 27. There will be no late checkouts of this part. You may demonstrate the motion to any of the TAs, or even to Valvano himself. The third component is qualifying round, where your robot must score more points than a brick. Qualifying is due by Friday, May 1. There will be no Lab6B report or Lab6C preparation. You will be awarded 50 points if your robot scores more points than a brick and no points if you fail to demonstrate these abilities by 5/1. There will be no late checkouts of this part. There will be 20-point bonus if you can qualify on or before 4/29. You can qualify with any TA. The fourth component is the scoring during the preliminary phase of the competition. The full 50 points will be awarded to teams that score high enough to be in the upper 50% seeding. Fewer points will be awarded for those teams in the bottom 50% seeding, depending on the specific scores and performances of the robot. The final competition will be 10-11am Wednesday 5/6 and Friday 5/8 10-11am. The last component is the Lab 6C report (worth 100 points), which is due by 5pm, Friday 5/8. This report will be graded in the usual manner.

I have additional parts that can be assigned to your kits, such as servos, optical sensors, switches, DC motors, springs, belts, cable ties, rubber bands, heat-shrink, wire, erector pieces, PN2222, IRF540, and 7805 regulators. I will dispense components as long as I have enough stock to allow all teams to have equal access. Unfortunately, I do not have enough batteries or Ping))) sensors for groups to have a second one.

My inventory procedures are limited, so please take good care of your kit, and return the kit in a manner that will make next year’s competition enjoyable for the next team using your kit.

Robot parts

http://www.robotstore.com/
http://www.robotmarketplace.com
http://www.sparkfun.com/
http://www.parallax.com/
http://www.towerhobbies.com/

Surplus parts

http://www.bgmicro.com/
http://www.allelectronics.com/

Full-service parts

http://www.jameco.com
http://www.digikey.com
http://www.mouser.com

Part search engine

http://octopart.com/