Functional Specifications Report (Team 17 – Moo)

Laura Derr, David Pekker, Albert Sim

27 October 2000

Abstract

The objective of our robot, Moo, is to score more points than the opponent by firing Ballzooka balls through the hoop on the opposite end of the board or, if necessary, by dropping the balls into bins. To achieve this goal, Moo has a variety of systems to sense and manipulate its environment.

Moo is composed of a differentially steered mobile base and a baseball style shooter. In addition Moo has a variety of sensors to measure its position and state as well as that of an opponent robot.

1 Hardware

Moo is composed of two major components – the mobile base and the ball shooter. The base contains the standard ELEC 201 (Lego Lab) battery and robo-board as well as motors and sensors to drive the robot. The ball shooter has an integrated magazine and feeder and is mounted on top of the base.

1.1 Mobile Base

Moo’s base is a very solid structure, slightly under the allowed size of 30 by 30 cm. It houses the battery, robo-board, drive system, and sensors. The general layout of the base is pictured in figure 1.

The mobility is provided by a four wheel differential driving system with two driving wheels. The front wheels are the driving wheels. Each (5 HLU diameter) front wheel is powered by two large motors through a 15:1 gear box. To turn the robot, the front wheels are spun at different speeds, and/or in different directions and the rear wheels skid across the competition board.

A set of wall guidance wheels is also provided. When the robot strikes a wall at a glancing angle, the wheels help to straighten it out onto a path along the wall.

1.2 Shooter

On top of the mobile base sits the ball shooter. The shooter must be able to make basket from any where behind the centerline with a 3 degree aiming error. The ball shooter must also be able to get the balls from the ball dispensers, and feed them to the firing mechanism. The shooter must also be able to non-violently deposit the balls into the bins.

The ball shooter is composed of two counter rotating wheels, one on top of the other. A motor drives each wheel with no reduction. The shooter is mounted at an approximately 15 degree angle at approximately 5.5 inches above the ground so that balls fired from it go into the hoop.

To feed the balls into the shooter, we built a cardboard magazine (ramp) that extends to the left side of the robot, which can hold up to four balls at a time. Under the magazine, a set of LEDs is mounted to trigger the ball dispensers to load the magazine. The
balls are fed into the shooter via three gears placed behind the top shoot wheel. These gears rotate slowly so that only one ball can enter the shooting mechanism at a time. As the robot prepares to shoot, both the gears and the wheels of the shooter spin simultaneously, so the balls are fed individually into the shooter and subsequently spit at a high velocity into the hoop.

To drop the balls at low speed, one ball at a time is forced by the feeder gears through the shooting mechanism, which remains unpowered.

### 1.3 Sensors

The sensor package includes the following sensors:

- Two optical shaft encoders (revolution counters). The shaft encoders count the revolutions of two unpowered wheels mounted next to the driving wheels, (see Figure 1) and are used to measure change in position and direction.

- 2-5 optic reflectance sensors. These sensors may be used for navigation and line following.

- A photoresistor to determine position of lights embedded in the contest board, in particular the ignition of the starting light.

- One sensor to locate local maximum of light intensity. To accomplish this a pair of photoresistors enclosed in a spatial filter will be used. (See figure 2)

\[
\begin{array}{c}
\text{wells} \\
\text{open}
\end{array}
\]

\text{photo resistors}

Figure 2: Top view of spatial filter arrangement

- One IR beacon sensor to locate the direction and distance to the maximum IR emissions.

### 3 Conclusion

The above specifications should lead to Moo scoring 108 points (12 points * 9 balls = 108) on a perfect run. As always, the software will determine the outcome. So it is now time to build the software. Go MOO!

### 2 Software

Software consists of low, medium, and high levels. The low level is a library to provide the basic interface to the hardware. Medium level functions are used to abstract the high level decision making AI from the low level functions. High level decision making AI is going to be implemented in the form of a decision tree. The branches will lead different strategies based on state of the environment (which includes the dip switches that will be used to tell the AI of suspected strategy of opponent robot). For example, if the robot fails to get to a ball dispenser, it should try to acquire balls from the other ball dispenser. The tree must not have any "dangling" limbs. Also the tree must look at both success and all possible modes of failure.
Functional Specifications

Structure

The robot will be relatively large (9-11 inches in length and width) and roughly square-shaped. The battery will rest in the middle or rear of the robot and in front of or behind the roboboard, which will be mounted vertically. There will be a shooting mechanism that will be close to the front of the robot, ahead of a ball storage area and ramp that will funnel balls into the storage area. There will be two axles for wheels, located approximately the same distance from their respective ends of the robot.

Locomotion

The robot will be powered by front-wheel drive. Both front wheels will be driven by a separate Johnson motor, each with a gear reduction of 81:1. In addition, we will have two rear wheels for traction and stability. Directional changes will be accomplished by spinning the wheels in opposite directions. For instance, to turn to the right, the left motor will drive forward while the right motor drives in reverse, and vice versa if we need to go to the left. The rear two wheels will drag during these turns, but we don’t anticipate that this will create too much friction to stop the turn. The front wheels will be the largest size wheels and the rear ones will be medium-sized.

Sensors

Two touch sensors on the front corners and two touch sensors on the back corners of the robot will be mounted to detect impacts with objects such as the wall. Three infrared reflective sensors will be mounted on the underbelly of the robot, one on the center, one to its right, and one to its left avoiding alignment with the center of the gravity. The IR sensors will be used to track lines, detect which side of the board the robot is on, and, of course, to detect the starting light. In addition, two photocells (placed in long cardboard tubes to increase sensitivity) mounted on the front of the robot will be angled approximately 30° from the vertical. We will use these sensors to detect the light from the net to align our robot’s catapult. We will also mount two magnetic sensors on the underside of the robot to align it with the ball dispensers and properly catch the balls. We will also be mounting a single LED on the side of our robot to trigger the ball dispenser. Of course, we will also be mounting the required IR beacon.

Ball handling

The robot will include an internal space that can store up to six balls, and will funnel directly into the shooting mechanism. The shooter will consist of a pitching machine apparatus with two motors driving two separate wheels in the same direction. They will be spaced such that they will grab one ball at a time and propel them forward, into the opposite direction net. The shooter will be placed more towards the forward end of the robot to better distribute the weight of the battery. There will also be an inclined ramp that starts at the height of the ball dispenser that will direct the balls into the ball-storage space. Our magnetic sensors will make sure that our robot aligns properly with the ball
dispenser so that the ramp is directly underneath it and catches all the balls. We will place the ball that we start with in the storage space, until it is ready to be propelled.

**Program**

Our program will contain functions that keep our robot aligned along the board lines (using a moving median of readings from all three IR reflective sensors), find the light on from the net on the opposing side of the board (using the difference in readings from the two photocells), detect walls (by checking when our touch sensors are triggered). We also have routines that will fire the balls at the net, make sure that we in the proper position near the ball dispenser, and shine the light to dispense the balls. However, depending on what DIP switches are set, we will try for the right bin first, get those balls, fire, and try for the other bin, or vice versa.

**Functionality**

Our robot’s basic plan is to go directly to one bin, collect those balls, back up and fire at the net, then drive to the remaining bin and collect those balls if they are still there. After that, our robot will be programmed with some defensive actions such as blocking the net, ramming the opposing robot, or tricking the opposing robot’s sensor. This would require additional structure, such as a blocking piece of cardboard, a IR detecting sensor, or a light-bulb, but we have not decided which of those strategies to pursue, and will probably wait until the other parts of the robot are finished before turning to them.
Speed and size are the vital components of our favored strategy, which is to use a broom to essentially "sweep" the balls to our opponent's side of the board, and then use our own robot to guard the balls, preventing the opponent's robot from sending the balls back to our side of the board. Also important to the strategy are maneuverability, location of the sensors, and control.

To accomplish desired speed, we plan to use four Johnson 30029 motors. Each motor will be used to power one of the four main wheels of the robot. Naturally, to make the most of each motor's power, a low-gear ratio will be used. Our current gearboxes have a ratio of 25:1, so this figure is not likely to change. Speed and acceleration will also depend on the weight of the robot; therefore, we intend to keep the weight of the machine extremely light by avoiding excessive use of parts.

Another component of mobility is turning and agility. We plan to have a relatively narrow wheel base in order to have a tight turning radius. Secondly, if we can involve all four wheels in the turning process, we will be able to turn much more quickly. Two wheels just being dragged while turning sap power, and slow the turn down. Another benefit is that we will have much greater control over the motion of our robot.

Our navigation will mostly come from programmed specifications about duration and speed of movement. However, we will use IR reflectance sensors to check our position on the board and follow lines. We might use a counter to check how much we have turned. Also we will use touch sensors to know to stop when we get to the corner.

The structure of the machine is another vital aspect of the strategy. The structure of the robot must be able to provide a good enough shield of the balls to keep them out of the opponent's possession. Therefore, our machine will likely possess a greater width than height. Most of this width will come from the broom or shield used to sweep and contain the balls, the main body of the robot will be as small as possible. The motors for each of the wheels will be placed very close to the ground, with the roboboard and battery directly above them. This type of structure will place most of the weight at the bottom of the robot and will create an excellent center of gravity.

Our strategy is rather simple at this point, so we believe that input to our board will be relatively simple. Depending upon what we believe our opponent's strategies and strengths to be, we will have our robot go to different balls first to sweep them to the corner. We will probably input this information with the dip-switches in the pre-start moments. Our robot will need to know when to start based on the start lights, so we will use a light sensor positioned under our robot, which will be plugged into one of the analog ports. Also, for basic position knowledge we will use to IR reflectance sensors, one on each side of our centerline, so that we can follow a line, and know when we change sides of the board. These will also go to analog ports. We are considering using a counter to determine the number of rotations of at least one of our wheels to determine our degree of turn. This would go to another analog port. At this point, we do not plan to attempt to track the other robot. Touch sensors will tell us when we hit the wall; we plan to have at least two, possibly more, located on the ends of our broom so we can be sure that we have pinned the balls in the corner. These will go to digital ports.

Our main outputs are the four motors to the wheels. Each wheel will be represented by one of the motor ports. Also, we might use one or more motors to manipulate the broom. Of course we will power the IR beacon with its port. We have no plans for other outputs at this time.
Group 27
Roy Ha
Michael Haag
Jessica Watkins

Lego Lab Functional Specification

The function of our robot is based innately with our strategy. Our strategy is one of speed. We wish to be able to collect balls quickly and efficiently and then impede the movements of the other robot. So, ball collection and mobility become the key aspects of our robot’s functions.

To aid in mobility, we will design a light-weight, agile robot. Since the ability of the motors to move our robot is directly related to the weight of the robot, having a light-weight robot will enable us to gain a speed advantage. Of course, this will entail a possible handicap in structural integrity. However, we are confident that a small, well-built, and fast robot will be able to play the game against any robot for a sufficient period of time before possibly being damaged. Another aid in mobility is the use of large wheels. Although this affects the amount of power we can provide and affects the gear ratio, the large wheels will make movement simple and makes turning somewhat easier, since most of the friction in turning is now directed at the points of contact of the turning wheels. Also, with such large wheels, a single powerful motor is sufficient for each wheel. We do not have to worry about using two (or more) motors for each side of the robot.

Ball collection is to be handled by a large scoop. During a good portion of the game, the scoop should be retracted. This acts something like a shield, since the scoop should be relatively flat against the front of the robot. The scoop should be powered by a single motor which pulls on a rope, much like a drawbridge. The scoop should be strong enough to lift 3 or 4 racquetballs, which means one of the higher torque Johnson motors. Also, to simplify control, a sensor should be used to tell the robot that the scoop is fully lifted. Sensors on the end of the scoop will also tell the robot when a wall is encountered.

Our strategy depends on knowing when walls are reached or when the black/white boundary is crossed. We use two IR reflective sensors to both follow a line and to know which side of the board we are on. The touch sensors on the end of the scoop indicate when the wall is reached. Two other touch sensors will be used on the front of the robot for when the scoop is fully lifted. We have not decided whether or not to put other touch sensors on the other 3 sides of the robot.

Navigation consists of moving in a square around the board. The lines will be followed, via the 2 IR reflective sensors until either the black/white boundary is reached or a wall is reached. So, a combination of the touch sensors and the IR reflective sensors is used.
Lego Lab Functional Report

Ideal Structure:
The layout of our robot will be pretty close to a one-by-one square with expandable arms that are anchored to the frame of the robot. When the arms are expanded, the robot will span a good portion of the board. The frame will be divided into two parts. The front part is designed to be modular with the board on top of the battery and drive motors. This portion is designed so that the battery and motors can be easily accessible. The back half of the robot serves as a platform for the folded arms and the other motors for expanding the arms. We want our robot to have a lot of torque so that it will not be able to be pushed by other robots; therefore, we need to have a high gear ratio. We also want our robot to be moderately fast both in movement and in expanding its arms so that it gets to the other side of the board before the other robot can come over. We are relying on getting the middle balls over to the other side of the board and not allowing the other bot to bring any balls to our side in order to win.

Ideal Sequence of Events:
Option 1: Shaft Encoding
1. Have the bot drive in a quarter-circle to the center of the board. This will be done with shaft-encoders.
2. The bot then expands the arms.
3. Line-follow until we reach the other end of the board. More accurately, just keep on going until time runs out, bulldozing anything in our way in the process.
Option 2: Wheel Exchange
1. Drive straight to the center of the board.
2. Raise initial set of wheels so that drive wheels touch the ground. The drive wheels are positioned at right angles to the initial wheels. This way, we avoid shaft encoding and turning, which will save us time.
3. Same as above.

Ports needed:
Option 1: Two motor ports are required for the main drive motors. Two motor ports are also required for expanding the arms. Three analog ports are required for IR sensors for line following. Two digital ports are required for shaft encoding. One analog port is required for the IR sensor that detects the initial start light.
Option 2: Same ports are required except that two additional motor ports are required for the “initial” drive phase.

Current Structure:
Our current robot has a robust frame that is modular for easy repair. Several arm designs are being tested, including an “accordion” type arm, an arm that flings out, and an “arm of death” that invariably injures one of our lab members when tested. We are saving that arm for last-ditch efforts. Line following and shaft encoding are currently being worked on.