The Design and Implementation of Lego Volleyball Robots.

Derek Cassidy and Gerard Lacey

Department of Computer Science
Trinity College
University of Dublin

Abstract

The aim of the LEGO robot project was to develop a team of robots to compete in a scaled down version of volleyball. The competition was held in Sheffield, as part of the British Association for Science Annual exhibition, on the 13th of September 1999. The robots described in this technical report won the competition.

1. Introduction

The team consists of a two-robot team, a Sweeper and a Striker each with its own part to play in the overall strategy. This report describes the strategy used between the two robots and the design in both hardware and software of the two robots.
The robot volleyball was played on a 2.4m x 2.4m pitch which was divided into two courts. Each team consisting of one or two robots had the job of keeping the ball, which was an American softball, out of their court. Failing to do so within one minute lost valuable points to the other team.
The competition was held in Sheffield England, as part of the British Association for Science Annual exhibition, on the 13th of September 1999. The other universities taking part were the University of Manchester, the University of Edinburgh, the University of Wales, Cardiff and the University of Reading. The robots from this report won the competition, with Cardiff coming second.
The robotic team, which is described in this report, consists of a two robot team, one to act as a sweeper to push the ball to the walls, and the other to collect the ball from the wall and kick is across the line. The sweeping robot also had the job of defending the centreline from any attacking robots. Figure 1 is an outline of the strategy of the team.

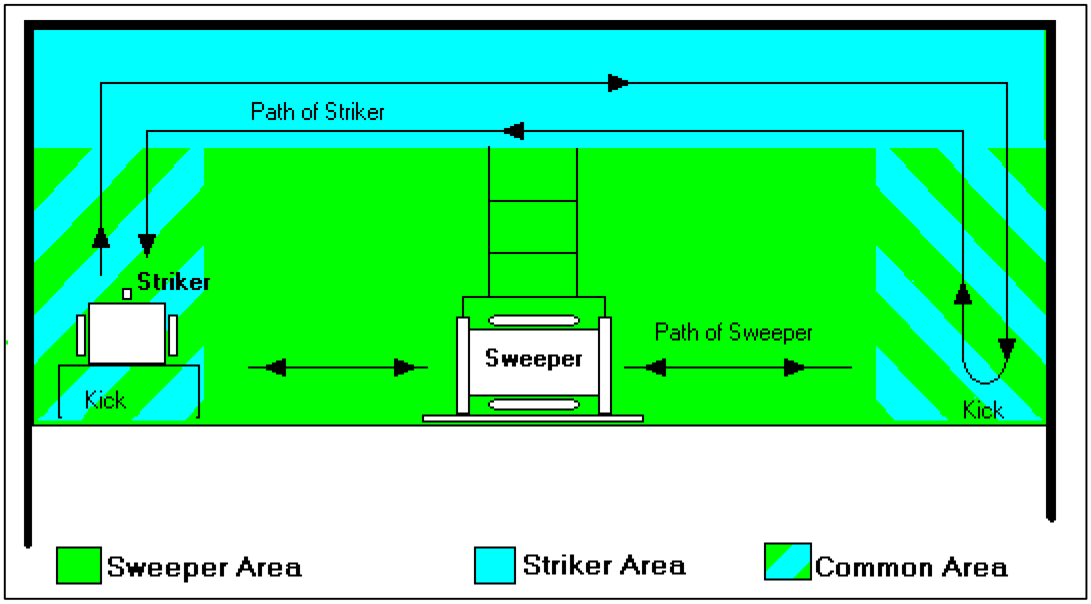
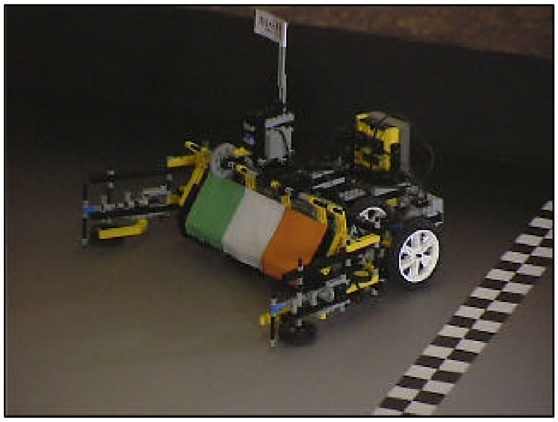
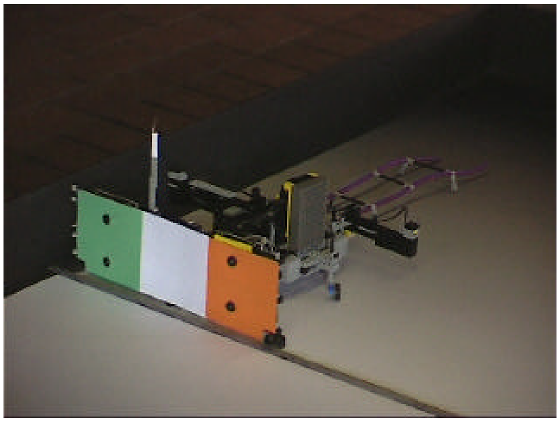


Figure 1 - Strategy of robots working as a team

Section 2 states the original goals of the competition, the actual rules of the competition itself and the constraints which had to be considered during the robots design.
Section 3 discusses the equipment used in the project. The competition was limited to LEGO parts apart from sonar sensors, which were built for the competition. The programming language used was called Not Quite C, which proved to be a very simple language to use but also powerful enough to use the RCX bricks to their full potential.
Section 4 is an outlined description of the building of the robots. It covers both the Sweeper and Striker design in hardware and software and also discusses the robots overall strategy on the playing field and how each robot takes on its own role in the strategy.
Section 5 gives an evaluation of the robots design and how they performed against the opposing robots in the competition environment itself.
Section 6 is the concluding chapter, which also discusses the areas in the design, which could have been improved if time permitted.
The final robot appearance can be seen in the photographs below:

2. Problem Statement

Goals

The main goal was for each team to build two working robots from LEGO, to play a type of volleyball. Although it is called volleyball the robots don't have to throw the ball across a net, the 'net' is only 6mm high so the robots must only push the ball across it. A team consisted of two robots and each robot was only allowed two RCX bricks maximum. The game was started by a service, where an American softball was placed anywhere inside one side of the pitch at the teams' discretion. The robots' goal was to get the ball out of their court within one minute. The robots also needed to last the full competition time, which was five minutes.

Rules

The rules of the competition were as follows:

1. There is a maximum starting 'foot print' size of 40 cm x 40cm or the equivalent area.
2. You are not allowed to drop or have parts 'accidentally' fall off your vehicle.
3. You are allowed to use any LEGO parts from the MINDSTORM, TECHNIC or other LEGO range. Non LEGO parts may only be used if agreed by the judges.
4. The robots may be programmed using any programming language or development tools.
5. The robots can be positioned anywhere in your half of the pitch before the game commences.
6. Once the game is started you are not allowed to interfere with your robot in anyway until the referee decides play should cease.
7. The robots must stay in you own half of the pitch.
8. The judges withhold the right to disqualify or penalise any robot that is considered un-sporting.
9. The pitch size is 2.4 x 2.4m square which is divided into two courts each being 1.2m x 2.4m in size. The edge of the pitch consists of a 100mm high black wall.

Constraints

One possible problem was when sending messages between robots, the messages from the other team could interfere with the robots. Therefore some rules about interference were necessary. The messages consist of single byte codes 0-255, so that each team was allocated its own private codes from 0-63 in the least significant last six bits, and two high-order bits be used as robot identifying codes 0-3, covering the 4 robot's in a competition. This means that each team should write their IR send and receive code in such a way that these first two bits could be easily changed (e.g. by adding 64, 128 or 192 to them) at competition time. Thus at the start of a match, the judges could allocate codes to the robots, and the robots could easily be adapted to communicate within their code regions. The robot cannot cross the line into the other side of the pitch or a penalty was incurred, therefore the robots need to be able to sense the centreline, which was 6mm square. The robots have one minute to get the ball out of their side of the pitch before they loose points so the robots need to be able to cover the entire pitch within this minute to have a reliable strategy.

3. Equipment

The robots are made mainly from standard LEGO parts and the only exception to this rule were the sonar sensors and "kitchen technology" i.e. paper, sellotape, glue and similar materials.

Sensors

There are three types of sensors used in the robots' design, infrared, touch and sonar.

The infrared sensors are part of the LEGO Mindstorms kits and they are used to detect a obstacles' colour at close range. The IR sensors can detect an object at roughly 15cm depending on background lighting. The output values ranges from 700 when the sensor detects a white object, to 800 where the sensor detects a black object.
The sensor value does flicker around a certain value, i.e. ±5 around a certain value, but this is reduced as the sensor is moved as close as possible to the object. This is due to the background light, and the closer the sensor is to the object the less background light can affect the reading.

The touch sensors used are basic micro-switches which have a zero or one output.

The third type of sensor was the sonar sensor, which is not part of the standard LEGO kits. One complete sensor consists of a pair of sonar sensors and each with a transmitter and receiver. The sensors cannot be placed closer than 3cm apart or the output will be incorrect. Each sensor (transmitter - receiver) pair is positioned at a roughly 45 degree angle.

The width of detection can be adjusted by changing the angle of each sensor pair:

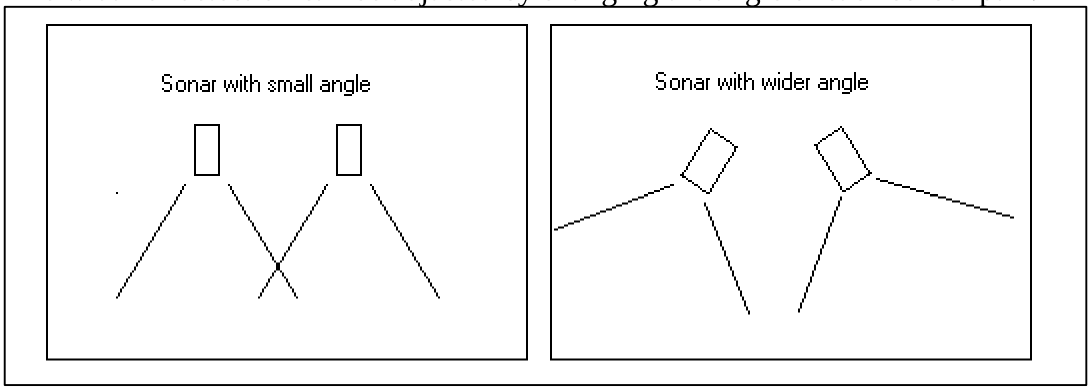


Figure 2 – Sonar Range Adjustments

The output values from the sensors are as follows:

Object	*Output Value*
Nothing	64
Near right	20
Near left	30
Far right	40
Far left	50
Near left & right	11

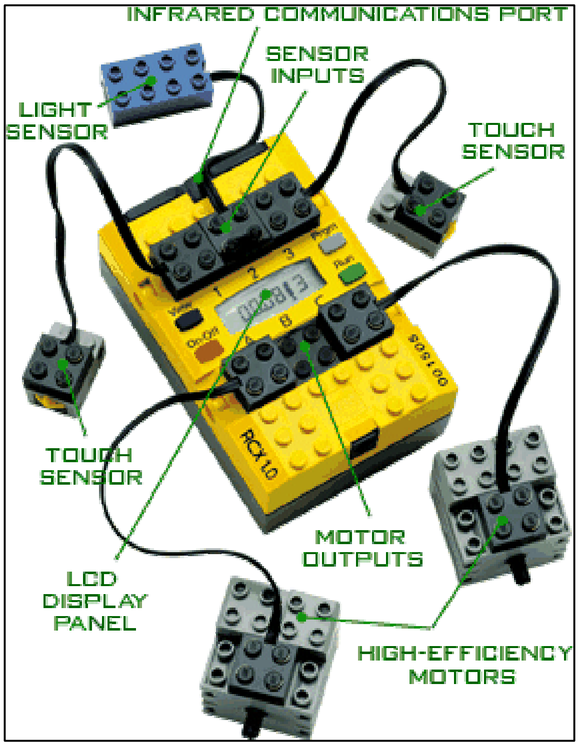
Motors

The motors used, are standard LEGO 9 volt motors. The speed of the motors was varied, by setting the output voltage between zero and nine volts.

Lego

The main chassis and mechanics of the robot have to be mainly LEGO parts. The LEGO parts are very easily put together to make quite strong robots. Because LEGO was used to build the robots is means design changes are very simple as it comes apart as easily as it is put together. The LEGO comes with a vast variety of different types of cogs, links and bricks, which allows quite complex mechanical designs to be built.

RCX - Bricks



The brain of the robot is the LEGO RCX brick, which contains a Hitachi Single Chip Micro-computer, which has 16K of ROM and 512 bytes of RAM and runs at 16 Mhz. The RCX also has a 32K RAM chip. The brick has three outputs, which can be used to drive motors and three inputs. These inputs are set-up in code for each type of sensor. The brick also has an Infrared port for communication. Code is downloaded from the PC, where the code is complied, into the brick via the IR connection. The IR port is also used to send and receive messages between two different RCX bricks. This is how the two robots can 'talk' to each other. The brick can hold a total of five different programs. More technical information can be seen at the following URL: http://www.crynwr.com/lego-robotics/

Not Quite C

The development tool used to program the RCX brick is called Not Quite C (NQC)[1], and from the name quite similar to C programming language. The code is written and compiled using NQC and then is downloaded into the RCX brick using the IR port. Up to eight tasks can be run concurrently and the brick handles the multi-tasking itself. Each task is declared in code by simply [**task** TaskName { *// execution code…* }], and then started using: [**start** TaskName;].

1 NQC is free software released under the Mozilla Public License (MPL). Available to download from: http://www.enteract.com/~dbaum/nqc/index.html

4. Design Strategy

The strategy decided upon was a two robot team in which the two robots can send messages to each other to co-operate when required. The first robot ‘Sweeper’, had two jobs. The first was to ‘sweep’ the pitch, forcing the ball from the centre of the pitch to the walls and secondly to watch any objects in the other court and follow it along the centre line to block any attacks. The second robot had the job of collecting the ball from the wall and pushing the ball across the line into the opposing court. The second robot, ‘Striker’ also had to communicate to the first robot to avoid a possible collision. The Striker robot was always given the right-of-way so that it just tells the Sweeper where it is and it’s the Sweepers job to make sure it is out of the way. The Striker continuously travelled from the centreline on the left-hand wall to the centreline on the right-hand wall, following the wall all the way. When the Striker was turning from the back wall to one of the side walls, i.e. about to go to the centreline to ‘kick’, it then tells the Sweeper it is doing so and then the Sweeper can make sure it doesn’t go near that wall. The strategy and path taken by both robots can be seen in figure 1. In this strategy the Striker continued following the same loop around the court wall regardless of whether it had the ball or not. In essence neither of the robots actually looked for the ball. The sweeping action of the Sweeper and collecting and striking action of the Striker ensured that if a ball was on the pitch in any position it should be pushed across the line in time, and hopefully within the allocated minute, provided the Striker was fast enough. This means the Striker never knew when it has the ball or not but it kicked when it got to the centreline regardless.

Sweeper

<u>Mechanical Design</u>

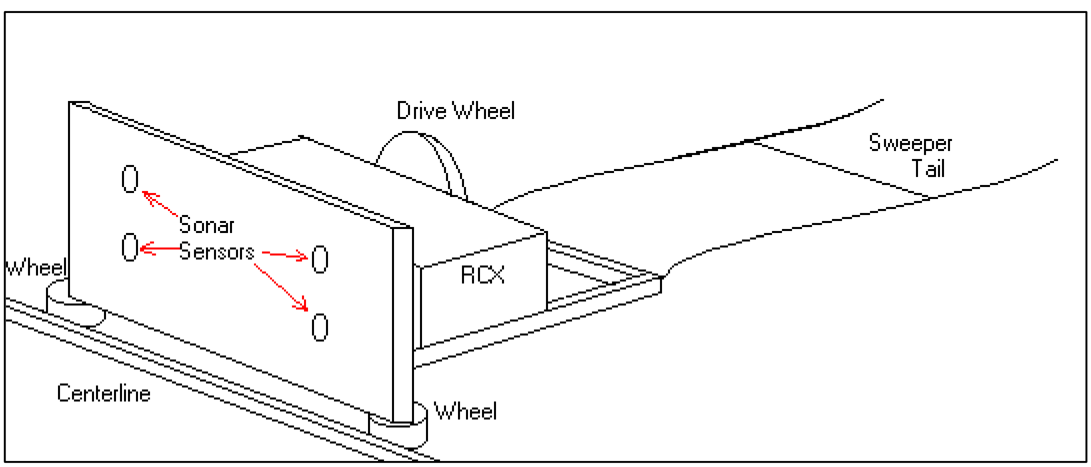


Figure 3 - Basic diagram of Sweeper design

The Sweeper robot as mentioned before has two jobs, the first was to sweep the pitch so the Sweeper robot design involves having a long tail, which extends to two thirds of the pitch. The tail folds across the top of the robot and is fixed using a small lever connected to a motor, which can be controlled by the RCX. This means that the tail can be extended at any time after the robot is turned on. The length of the pitch from the centreline to the back wall is 1.2m and as the width of the Striker robot is 40cm (the maximum allowed), this leaves 80cm for the Sweeper robot to cover. The rules of the competition states an initial 'footprint' size of 40x40cm for each robot, which is why the tail extends from the robot only after the robot has started. The second job of the Sweeper was to defend any attacks form the opposing team by creating a block in front of the centreline. The blocking face of the front of the robot was made as wide as possible, i.e. the maximum 40cm wide. There was only one driving wheel as the robot never has to turn, and only if the robot can keep tight against the centreline. The driving wheel is positioned as far away from the centreline as possible so that a turning force is applied to the robot depending on which direction it is travelling keeping the front tight against the centreline.

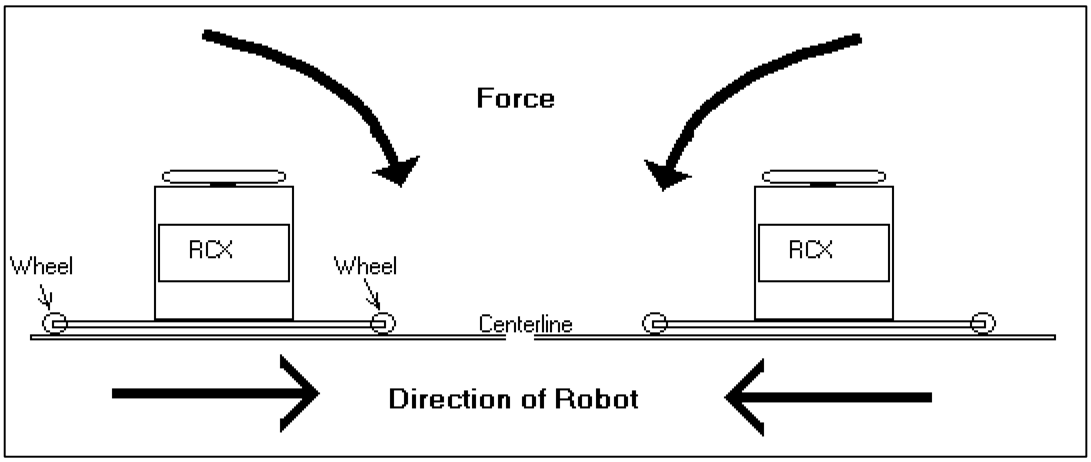


Figure 4 - Mechanical force induced when moving along the centre line

As the Robot kept tight against the centreline this produced a very efficient robot. One problem with moving robots was that sensor error accumulates quickly and with a robot using its sensors to track its position it would soon be somewhere it thinks its not. Therefore by keeping tight against the centreline the robot could not 'get lost', i.e. the rigid centreline kept the robot going in a straight line. The chassis of the robot has two RCX bricks as both robots have the same basic chassis. But in designing the robot is was realised that only one RCX brick is needed for the Sweeper robot. However the are still two bricks in the Sweeper robot as the weight of the brick is important in its design as described previously.

Sensor Design

In the Sweepers defence strategy the robot must track any objects on the opposing side. To do this sonar sensors are used and are positioned in the front of the robot facing into the opposing side. In trial and error tests the sonar sensors were positioned so that the wall wasn't detected when the robot was touching the left or right wall. For the robot to know its position, there were two types of sensors used, touch sensors, and a rotation sensor. The robot can keep track of how far it is from the wall by using the rotation sensors, which is made up using a LEGO infrared sensor **[1]**. The wheel was marked as in figure 6 so the IR sensor detects when the wheel rotates 30 degrees and can record how far the robot has moved from the wall. A turn of 30 degrees represents a move of 2.5cm by the robot. The wheel itself used as the rotation sensor is a passive wheel, as using an active wheel would induce great inaccuracy due to wheel slippage.

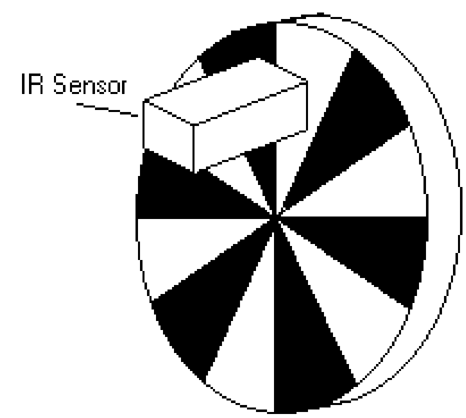


Figure 5 - Rotation sensor design with IR sensor and segmented pattern

However, due to the robot changing direction quickly following an object the robot will still not have a completely accurate knowledge of where it is so there are two touch sensors placed one on each side so that the robot knows when it has hit a wall. In such case the robot can re-calibrate itself.

Computer Control

The aim of the Sweeper robot was to watch the opposing side for any objects and then track the object using the sonar sensors. When there was no object in sight then the robot moved up and down the centreline from wall to wall. This motion up and down the centreline was the basic action of the robot until either, it saw an object so then it started following that object, or it was told by the Striker robot that it was going towards the centreline on either the left or right side. The Sweeper code uses seven tasks, each of which can be run concurrently. The software structure of the Sweeper robot was designed like a layered behaviour based architecture **[2]** where the basic behaviour was the 'Keep_Going' task, which moved the robot from left to right side. The 'track_object' task constantly watched the sonar sensors and when an object was seen on the opposing side of the pitch, the 'track_object' task took control over the robot and the 'Keep_Going' task is stopped. Then the 'Keep_From_Crash' task took over when a message was received from the Striker robot, which prevents a crash occurring.

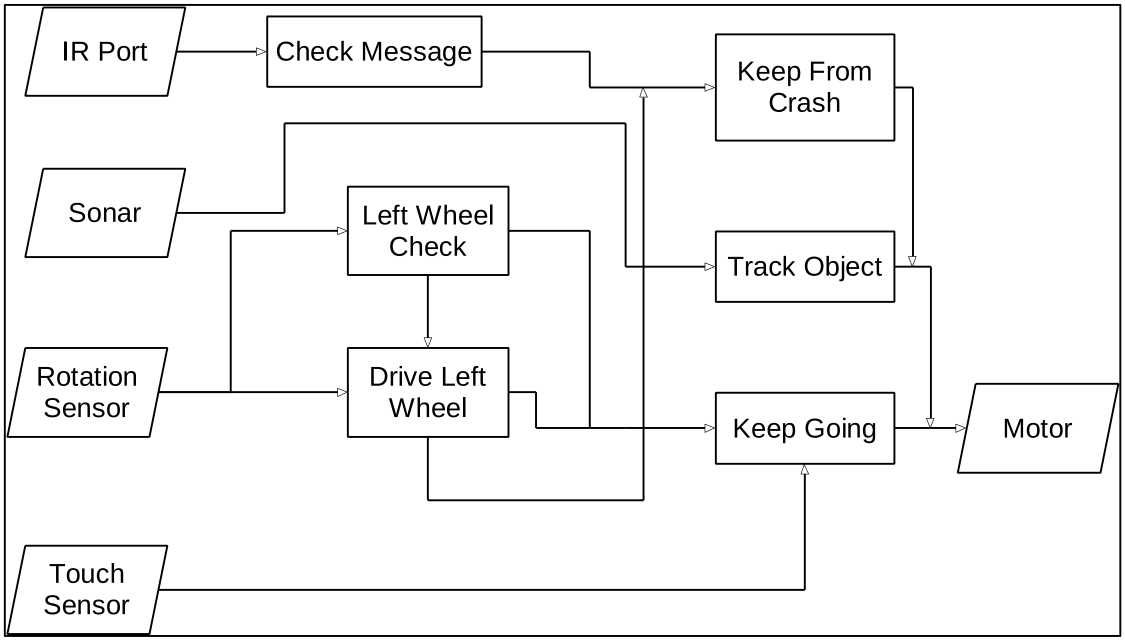


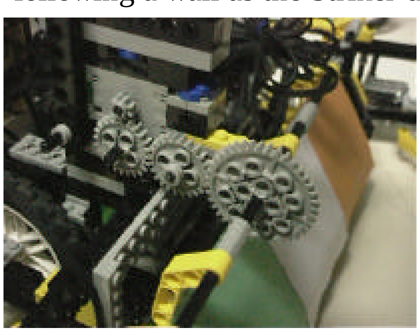
Figure 6 – Software Diagram of Sweeper Robot

The code and each of the tasks can be viewed at the following URL :
http://www.cs.tcd.ie/Derek.Cassidy/legocode.html

Striker

Mechanical Desgin

The Striker's role in the volley ball team is to go around the wall keeping tight against it to collect the ball if it is anywhere along the wall. As mentioned before one problem with designing a robot is how to keep the current position of the robot accurate. Thus by following a wall as the Striker does this problem was overcome. By following the wall, which is rigid, the robot is being constantly re-calibrated. When the Striker reaches the centreline it then pushes the ball across the line, then continues back around to the wall to kick again at the centreline. The main part of the Strikers physical design was the pushing mechanism at the front of the robot to kick the ball across the line. This is achieved using a motor attached to an arm which can rotate upwards to push the ball away from the robot. The motor had to be geared down enough to give the arm enough torque to push the ball. This was achieved through trial and error with various different cog configurations to find the best gearing. The final gear reduction used was a 5:1 reduction.



There was two arms put onto the front of the robot around the kicking mechanism to hold the ball after it has been collected. The arms had small wheels attached to the end so that the arms will run smoothly along the wall.

This picture shows the kicker activated, and also the arms are visible with the wheels attached to the ends.

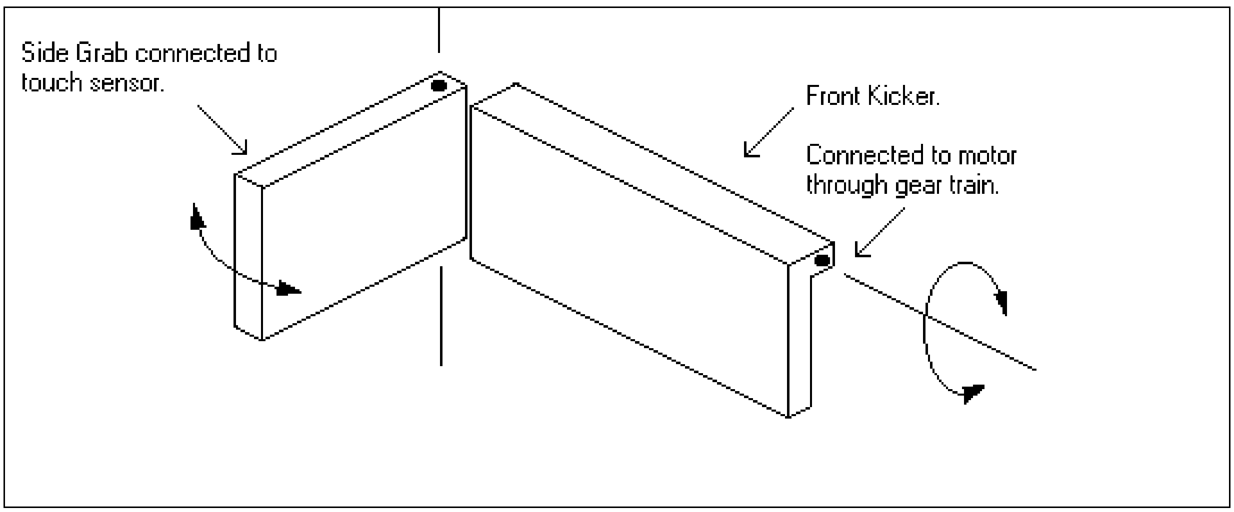


Figure 7 - Diagram of one side arm and front kicking mechanism

The robots drive mechanism itself consists of two driving wheels and a small wheel at the back only for stability.

<u>Sensor Design</u>

The path the Striker follows was around the walls so it used touch sensors on the arms to detect that it was against the wall. The side impact sensors on the Striker are integrated into the side arms, which can swivel about the end of the arm as shown in figure 7. There were also front impact sensors so that the robot knows when it comes to a wall in front so it could turn. The sensors are designed so that they also detect the centreline, which was 6mm high. The other sensors used are infrared sensors, which make up two rotation sensors as in the Sweeper robot, figure 5. Therefore a total of five sensors are used, side impact sensors (x2), front impact sensor and IR sensors (x2). The RCX bricks only have three input sensors so two of the RCX bricks were used and the two bricks had to communicate to each other via IR messages. The rotation sensors are necessary to provide information about how far the robot was turning around a corner. Timing a turn was not accurate enough as the turn could vary a lot due to batteries running down. The sensors were attached to two passive wheels so wheel slippage doesn't become a problem. The wheels were also positioned as far to the sides as possible to give a more accurate reading, i.e. the further away from the centre pivot point the greater distance the wheels will have to travel in a certain turn.

Computer Control

The software aspect of the Striker robot is more complicated as it uses two RCX bricks, which have to communicate to each other. As there is a slight delay with the time it takes for a message to be sent and received the rotation IR sensors have to be connected to the main brick which controls the driving wheels. The main brick contains all the actual calculations and controls the robots' manoeuvres. The Flowchart below describes how it functions:

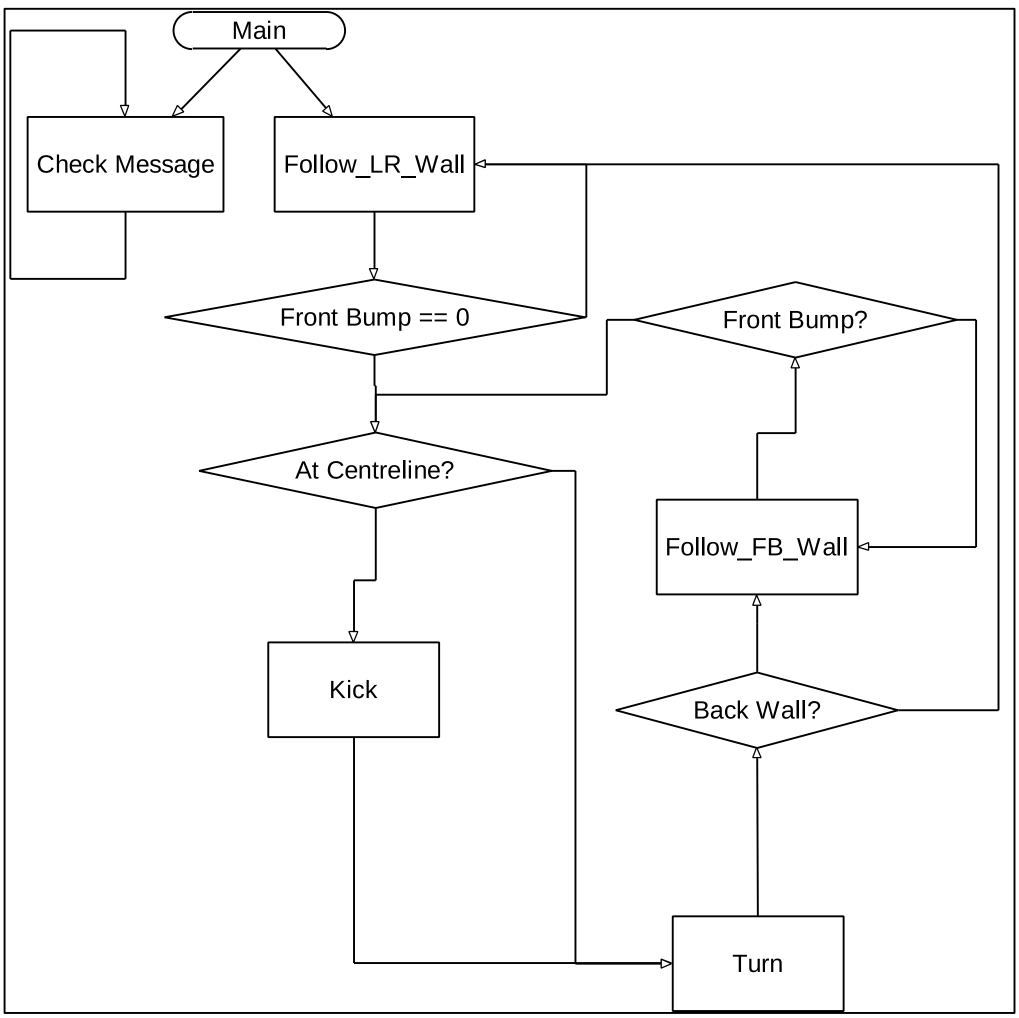


Figure 8 – Flowchart of Striker Robot

The secondary brick is used to detect the impact sensors and sends this information as a message to the other brick. There are five tasks in the secondary brick:

CheckBump_Left, CheckBump_Right and CheckBumpFront: This task continuously checks the touch sensors and sends a certain message to the main brick depending on which sensor was activated or deactivated.

Main: This task initialises the input ports and variables and starts the other tasks.

The code and each of the tasks can be viewed at the following URL :
http://www.cs.tcd.ie/Derek.Cassidy/legocode.html

5. Evaluation

Although the two robots were tested before the competition they were thoroughly tested on the competition day itself. The competition was on the 13th of September 1999 in Sheffield University as part of the British Association for Science Annual exhibition, the robots competing were from University of Reading, University of Edinburgh, University of Manchester and University College Cardiff. The competition was played as a type of league with each team playing every other team so that the robots got a thorough test against various types of robotic and strategic designs. The Sweeper and Striker design proved to be the most efficient team. One major problem with the robots was getting lost on the pitch, i.e. not knowing where it is or thinking it is somewhere it is not. Earlier designs for robot was to have the Sweeper robot only the 40 x 40cm size and covering the pitch by manoeuvring around the pitch. This proved extremely difficult to keep the robot on track, i.e. keeping straight while crossing the pitch and turning exactly the right amount. Therefore by combining this Sweeping design and a blocking type robot moving up and down the line, an efficient robot was developed. As it never had to turn it just had to keep tight against the centreline therefore making it quite difficult to 'get lost', or leave the desired path. The problem of covering the entire pitch was solved by having a tail to extend to the rest of the pitch uncovered by the Striker robot only after the robot had started thus keeping within the maximum starting 'foot print' size of 40cm x 40cm. The blocking action of the Sweeper robot proved a very useful defensive strategy. When seeing an object on the opposing court the Sweeper stayed hovering around the object and when the object was an attacking robot it effectively stopped the ball being pushed into our court most of the time. Also the Striker robot proved to be an efficient design as its' design made sure that the robot remained touching the wall at all times. It is a lot easier to keep the robot going straight by using a solid object like the wall, rather than using the sensors, which produce drift in readings as time goes on due to sensor error. This meant that during the games the robots could keep going for long periods without leaving their desired path. Another possible problem was the Sweeper and Striker colliding with each other. This is why the robots need to communicate to each other and the strategy decided was that the Striker always had the 'right of way', i.e. it is the Sweepers job to keep out of the Strikers way. When the Striker is about to turn towards the centreline from the back wall it sends a message to the Sweeper. Thus the Sweeper recognises the IR message and makes sure not to go within a certain distance of that wall.

During the games this working quite well as the robots only collided once which was at the end of the last game after using the robots for three previous games, so the batteries were run down and the Sweeper couldn't get out of the way quick enough. The two team, Sweeper – Striker combination strategy proved to be quite an effective solution attacking and defending well against all teams and ultimately winning the competition!

6. Conclusions

The goal of this project was to build a robotic volleyball team consisting of two LEGO robots and to ultimately put them to the test in a series of volleyball games against the other universities who took part. This goal was achieved as the Sweeper and Striker design, which was developed, worked very well in the competition situation. The overall design of both robots kept them as simple as possible and hence less errors could occur. This proved very efficient and they very rarely went off course.

Although the robot volleyball team working quite well there are a few aspects of the design that could be improved. One problem encountered was that when the robots batteries ran down the Sweeper couldn't get out of the Strikers way in time, which ended in a collision between the two robots. This could be avoided by making the Striker wait until the Sweeper has moved sufficiently out of the way before proceeding to the centreline to kick. This would mean by simply waiting on a message from the Sweeper after it has moved a certain distance from the wall.

Another possible problem could be if the Sweeper is defending an object somewhere in the centre of the pitch while the Striker goes around the walls, the ball could be in a position in the middle of the court where the Striker cant collect it. One possible solution would be to make the Sweeper go across the pitch at least once during each loop the Striker makes of the wall. However what if the Sweeper was defending an attack at the time, maybe by adding sensors so that the Sweeper could see whether the object it was defending was a ball or adding sensors on the Striker to see if it had the ball in the grab.

7. References

[1] Joseph L. Jones & Anita M. Flynn. Mobile Robots – Inspiration to Implementation. 1993, p 122-125.

[2] Rodney A. Brooks. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, RA-2(1):14-23, March 1986.