UNIVERSITY OF EAST LONDON

Automated Lawn Mower

Final Year BEng Project Report

Fei Manheche
Student Number: 0733169
Module code: EE3008
5/5/2011

This report presents a design method of a low cost automated lawn mower using a microcontroller (Arduino). It covers basic aspects such as motor control, closed loop system control, sensor analysis and location determination methods. The content of this report can be of benefit to engineering students and anyone else interested in electronics.

Acknowledgements

I would like to thank my parents, Francisco J. Manheche and Ana P. Y. S. Manheche, whom have always taught me great values and helped me reach this stage of life with a degree of success. Special thanks to my brother Francisco Manheche Junior, who has provided support in many ways. I am thankful to Uitnei A. Chamusso, who although being geographically distant from me, has helped me remain sane with long evening conversations on things other than studies. I thank Costas Ioannou (Principal Lecturer at UEL CITE), Dr. Wilton Bailey (Project Supervisor and Senior Lecturer at UEL CITE) along with all the other lecturers at the University of East London. I am equally grateful to Cativa Y. Sung, Madalena Almada, Alex King and his wife Estefania King, Valdemir Sung and his wife Yolanda Sung, and everyone else in the Manheche, Sung, Coelho and King family members.

I would also like to show appreciation to all my classmates, who have provided me with so many encouraging words. Finally but not least, I would like to show my appreciation to everyone else (close friends and those not mentioned here), who have also directly or indirectly contributed towards my accomplishment.

Fei Manheche
Student number: 0733169

Abstract

This report presents a design method of an automated lawn mower. The design described here is made such that the cost of a final product is economically viable and as efficient as other solutions that are currently available. An ultrasonic transducer is used as the distance measurement component and the relevant instructions processed by a microcontroller. The ultrasonic sensor is attached to a servo motor where, using similar techniques to those applied in radar technology the vehicle is able determine its path. Calculations and problem solutions are described for this closed loop system. This report provides content that may be of benefit to anyone attempting to create a closed loop system where many variables are involved and programming of a microcontroller is required.

Table of Contents

List of Figures

1 Introduction

This project provides a design method of an automated lawn mower, whose task is to cut grass whilst following a specified pattern with no need of user interaction. This task is expected to be made possible by using sensors to provide a microcontroller with measurement of distance. The device is expected to determine the path to follow, as shown in Figure 23, calculate its position and stop upon completion of its tasks.

The project aims to produce a machine which would relieve people from a regular task of cutting grass in their garden. Automated lawn mowers are presently available to the public, e.g. (Robomow, 1998), but the low spread of these devices is mainly due to the cost which start at £918.13 (Mowermagic, 2011). The goal of this project is therefore to produce a similar device with similar (if not better) performance, at a significantly lower cost.

The present report will provide an understanding the sensor selection process, an analysis of the microcontroller to be used and the relevant functions it has to execute, a study of the relevant motors used and a system overview. For this progress report some areas are incomplete.

1.1 Motivations

Automated lawn mowers have been made available to the general public for over 30 years (Georges 1999, p.195); its widespread or public use on the other hand has been limited mainly due to the current costs of such devices. Existing technology sell at around £899 or more (Lawn Mower Reviews, 2011) and considering the fact that the manual versions of these devices, the standard lawn mowers, sell at around £86 (Lawn Mower Reviews, 2011) although the cost of labour would need to be added to that of the equipment, the latter is still a current viable and affordable option for most consumers. There are numerous real-world benefits of having a machine that autonomously cuts grass, these include:

- Aid elderly users or those with disabilities who are unable to fulfil this task themselves
- For users with a busy schedule and rarely find time to mow, etc.

It is a device that can fit into just about everyone's lifestyle, therefore having a device that costs less, whilst accomplishing the same task as the higher end models is a great advantage in order to compete with the current market where the end consumer will benefit from.

2 Background Research

2.1 Existing Technology

Existing automated lawn mowers each have a distinct working principle, for instance the Robomow from Friendly Robotics (2011) requires the user to perform a onetime set up where the garden perimeter is set. The perimeter is set using a battery powered wire that is laid around the outer edges of the garden and any area where the robot is not to cover. Special sensors inside Robomow enable the wires to be recognized and the robot is therefore kept within the designated area. The robot travels on the garden in a systematic criss-cross pattern, as shown in Figure 21, several times from side to side to ensure that the entire area is covered and that the grass is cut from different angles (Friendly Robotics, 2011).

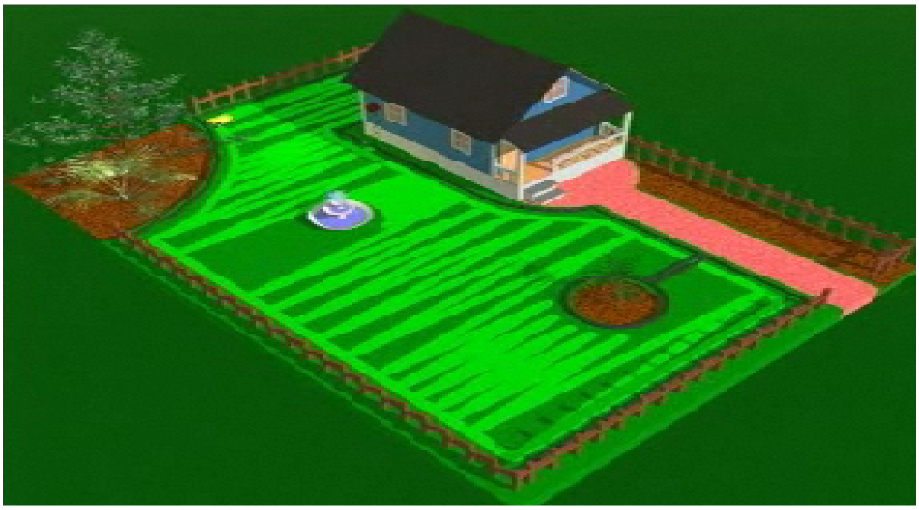


Figure 21 - Robomow criss-cross cutting pattern (Friendly Robotics, 2011)

Other technologies work around a similar principal as the Robomow, in the sense that it requires a perimeter wire to limit its cutting area. A difference between them may be added features and the cutting pattern, for instance the LawnBott and the Husqvarna both have a random operating principle (LawnBott 2011, p. 7 and 8), in the sense that they do not follow any specific cutting pattern as illustrated in Figure 22.

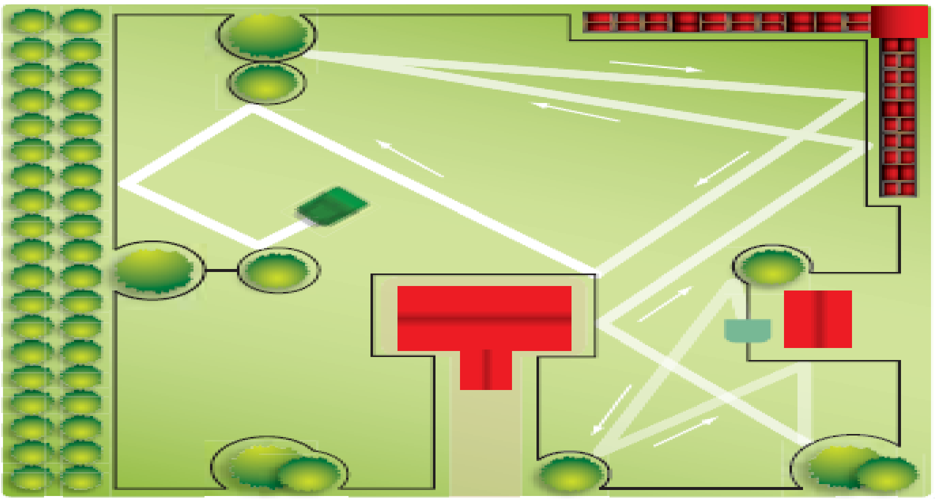


Figure 22 - LawnBott random cutting pattern (LawnBott, 2011)

2.1.1 Advantages and Disadvantages of Existing Technology

The operation method of the Robomow (described in section 2.1) although effective, is to a certain degree inefficient and contribute to a quite costly end product. The main advantages of their design include, virtually any size or shape of garden can be specified, it is flexible and it can work without requiring the user intervention at any stage, can operate at surface areas with slopes, these points can also constitute a disadvantage. The system design has the following disadvantageous aspects:

- **The size and shape of the garden –** as wires have to be placed in the garden area to be covered by the robot; the cost of the entire system may be considerably higher depending on the size. In the case of many plants being in the middle of the garden area, the setup can become cumbersome.
- **Damage or faults –** the wires are set on the grass and although the company mentions the fact that they would typically be covered by grass and become unnoticeable in a matter of 2 to 3 weeks (Friendly Robotics, 2011), the wires are prone to damage. The damage can be from any origin for example, pets can dig up and damage the wire, over time due to weather it can wear off, amongst others. A fault on the wire can provide incorrect information to the device and as consequence the device will not work as expected.
- **Cutting pattern inefficiency –** the criss-cross pattern (as shown in Figure 21) for Robomow and the random one used by LawnBott and Husqvarna (shown in Figure 22) are all not very efficient due to the fact that it is required that the robot crosses the same location more than once, consequently more energy is dissipated and the overall completion time is extended.

2.2 Design Method Considerations

It is extrapolated from existing technology (section 2.1) that in order to achieve an end product that is better than that currently available and at a lower cost, the overall design of the system described in the present report would have to operate with the following considerations:

- The system would be required to be made such that it follows a certain path and more efficient cutting pattern.

- A garden area limiting method that can be of long term, resistant life cycle and more cost effective; that is a better solution than that of the guide wire.
- Good cutting methodology
- More cost effective sensor array arrangement that would enable for obstacle and garden limit determination.
- Use of a two wheeled vehicle opposed to any other number would enable to the robot to easily navigate around the track.
- Opposed of the robot relying on additional electronics set in the garden area; it should be chosen that all electronics are onboard and that the device is able to determine all relevant parameters in this manner.

2.3 System Requirements

The system described in this report has to take into account different aspects such as motion, speed, direction of travel, maintain motion in a straight line, re-alignment in case of going off track, determination of when and where to turn, amongst others.

The robot is expected to maintain constant speed around the garden area; this speed should not be very fast as it should be able to accomplish its main purpose, which is of cutting grass. During motion it is important that the device is kept in a straight line (in the forward path) and able to turn at a 180 degrees angle at the extremity of the garden in order to continue with the same motion in the opposite direction and adjacent to the previous path taken. Figure 23 illustrates the path that the robot is to follow and the limit where it is required to turn. The robot should be able to re-align itself and continue through the path in the case of it going off track.

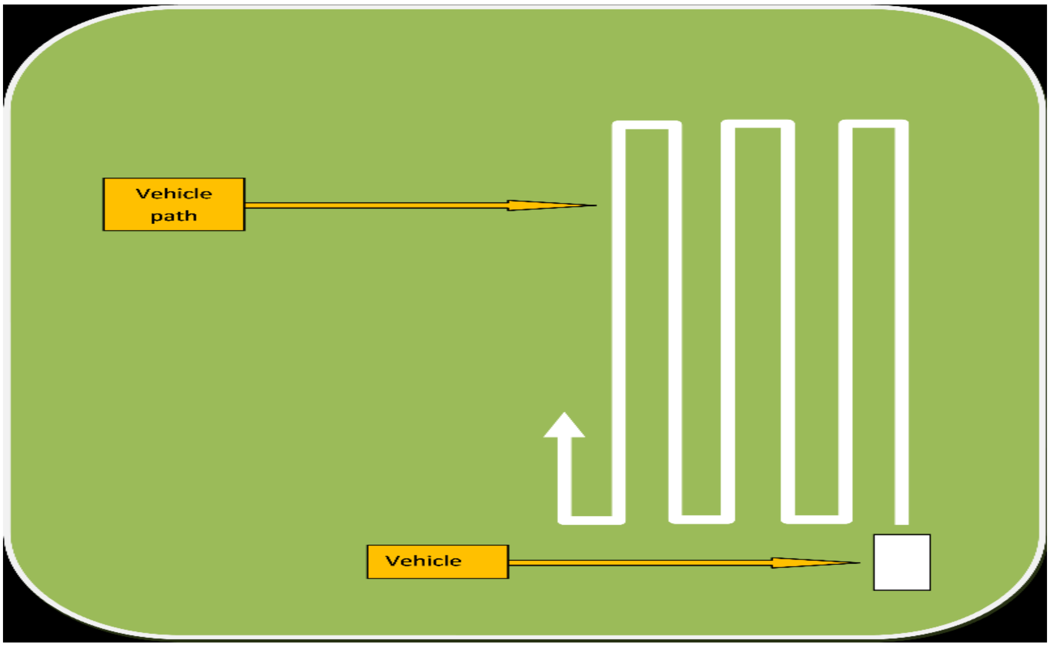


Figure 23 – Automated Grass cutter path (Bailey, W 2008)

A simple approach taken in tackling the problem is to have poles positioned in the garden area as the limits. It was chosen that the poles could be a stick independent of any electronics which could either be purchased or found in the garden area, the dimensions on the other hand is to be determined. The robot once switched on would start off looking for the poles. Once the distance from the poles is determined, the robot would then be required to establish its current location in the garden area, where it has to go in order to initiate its motion in the desired path (as shown in Figure 23) whilst mowing.

During motion of the robot, it is important that some feedback is given to the system in order to ensure that the correct path is being followed and that there are no errors in accomplishing the task, for this reason a sensor would be required. It is necessary to account for possible errors or inaccuracies that the sensor may provide under different conditions. More attention during selection process of the relevant sensor(s) is paid to its performance under different weather conditions, due to the fact that the robot is required to operate outdoors, where many disturbances are present.

The disturbances in this case can originate either from varying temperatures to the amount of sunlight present, both aspects which may impact the overall system performance and ability to fulfil its requirements.

Other aspects considered include the external design of the robot, shape and size, grass cutting blades specifications, number of motors required and application of good motor control. All of these components would enable the robot to accomplish the task of cutting grass and its mobility around the garden.

2.3.1 Specifications

It was determined that distance measuring sensors were used for this project due to the fact that they are able to provide the robot with information of its proximity to surrounding objects, hence its position can be established. With such sensors, it is possible to determine the distance from the robot to the poles.

A motor mechanism that would provide different viewing angles to the robot is required and this method was chosen as an economic alternative to having an array of sensors for the same task. The cost of having one sensor and a motion mechanism for it, to that of an array of sensors is more than halved therefore contributing to a lower cost end product.

Selection of two motors (where the wheels are attached) that would enable the vehicle motion to be controlled was made due to the fact that each wheel can be controlled independently; consequently a close to perfect 180 degree rotation can be made possible at the end of a forward motion track. This can be achieved by controlling one motor to stop, whilst the other makes a 180 degree rotation, meaning that the vehicle will be able to turn suitably adjacent to its previous motion path.

2.3.2 Vehicle Requirements

Selection of a two-wheeled vehicle was made due to the fact that it enables for the robot to be more flexible. The flexibility relates to the motion around the track such as the 180 degree rotation required for the vehicle to position itself directly adjacent to its previous track. For example if the vehicle comes to the end of a forward path and it is to change to an adjacent path. The other consideration made for the vehicle is positioning of the blades that will cut the grass, these will need to be positioned such that it can efficiently cover the garden area through one motion cycle; that is without the vehicle having to go through a motion path more than once.

Considerations are also made to the dimensions of the moving wheels of the vehicle, as the surface are covered by the wheels play an important role in ensuring that the vehicle is able to run in a straight line with being easily drifted. Wheels that cover a smaller surface area, especially with grass and sand, would normally drift to one side, whilst those with greater cover can overcome this issue and help in reducing the amount of drift.

2.3.3 Limitations

One of the limitations of the design method is the distance determination, which includes precision and maximum distance that can be determined by the sensor. As the system relies on the information captured by the sensor, its malfunction or failure may render the entire system useless.

The garden shape and dimensions also constitute a limiting factor for the system described on the present report; the reasoning behind this resides on the motion pattern chosen. For instance if the garden has rounded sides and other shapes other than part cubical, this system may leave parts of uncut grass in the garden area.

A further limitation of the system is identified when obstacles are present in the garden area, where a method is to be identified for the system to continue with normal operation by finding a way around it.

2.3.4 Different Scenarios Analysis

An analysis made for the different scenarios that can affect normal system operation includes weather, temperature, obstacle detection and avoidance and the case of the vehicle going off track. In case the wheels are drifted to a different path from its linear path, it is necessary to consider a possible solution that would enable the robot to establish that it has gone off course and have it realign itself and continue its task. A possible solution is via implementation of a closed loop, where feedback from the sensor is provided and once the values are out of range, the device would then run relevant operations to re-align.

In order for the track to be determined and its location, it was concluded that it was essential that a sensor was included in the project. Sensors are able to convert into signal form, such as instances that occur in the analogue world. The sensor would therefore enable the distances between the robot and surrounding elements to be determined. In case of failure or malfunctioning of the sensor, the system would be rendered useless, hence its importance.

2.4 Problem Solving

There are numerous problems to be solved in this project, where it is required that the device follows a set path in the garden, as shown in Figure 23 whilst cutting grass. It is important that the vehicle is maintained in the path and reposition itself in case of going off track.

The first step taken in attempting to solve the problem is to identify the most cost effective design for the automated lawn mower. The hardware specifications of the design can be at a low cost if the number of components is reduced. Many variables are taken into consideration at this stage, such as the area at which the device is to operate in, the environment aspects such as temperature amongst others. In order for the environmental aspects where the device will operate to be gathered electronically, there is a need to include a transducer of some kind, relevant circuitry that would handle the input data, and a control circuitry or method for the input data to be managed and relevant actions taken place according to specifications, a cycle of how the problem should be solved can be seen in Figure 24.

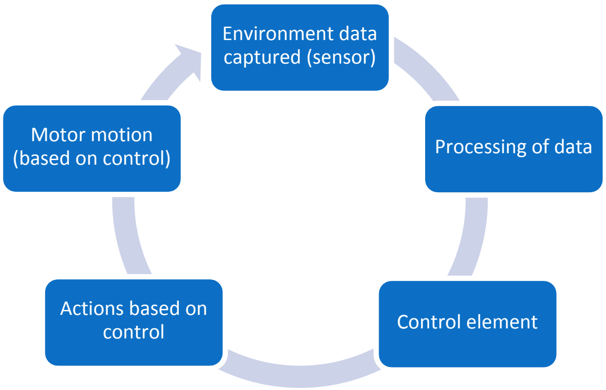


Figure 24 - Flow of problem solving elements and its dependence

For these aspects to be accomplished, research and selection of the following areas is required:

Sensors
Programming
Motors
External Design
Electronic drive Circuitry
Power supply

These areas will be discussed in more detail in later sections.

2.4.1 Distance Determination Methods

The sensors constitute one of the most important components in this project, being responsible in providing the system with information regarding to its surroundings, this external information is to be processed by additional circuitry. For this project the sensors have the main function of determining the distance between the device and objects that are set as limits in the garden. This component would also be required to determine obstacles and apply relevant control based on it, during motion. Taking by definition, a sensor measures a physical quantity and converts it into its electrical signal equivalent (Jacob 2010, p. 3).

There is an array of sensors that can be used for the purpose of determining the distance of the device from other objects. This section will discuss the various ranging sensors and also cover the important aspects that were taken into consideration when selecting the relevant sensor for this project which include cost, practicality for user to operate, maximum range and the operating characteristics under different temperatures and weather. It is important that the sensor chosen is able to provide as close to precise as possible readings. The sensors considered for this project were ones with both the transmitter and receiver in the same module, this is due to simplicity of design, implementation and total cost, and these will discussed in chapters to follow.

2.4.1.1 Sharp Infrared Rangefinder

The sharp infrared range finder is able to measure the distance of an object through a process of triangulation, where a pulse of an infra red light beam is sent to an object, the reflected signal returns at an angle, which varies depending on the distance of the object and through calculations it is possible to determine the distance (Society of Robots, 2010). The main advantages of this sensor type include accuracy, good operation in ambient light and precision. The main limitations found with this kind of sensor for the required application in this project, which is of determining the distance of the object, include:

- **Thin beam width –** the object can only be detected if placed directly in front of the sensor
- **Poor readings whilst in motion –** this aspect which may be related to beam width, means that is the device is in motion, the readings gathered may not be accurate.
- **Sunlight –** as this sensor works on the basis of lights beams, these can suffer from interference of the sun rays in bright sunny days which can affect readings.
- **Maximum and minimum range and its costs –** objects placed too close to this sensor may not be detected by it, and the maximum range at a low cost is usually around 3 meters. The average selling price for these is £8.75 (Rapid Electronics Ltd., 2011) for the cheaper versions (with an approximate 10 - 80cm range).

2.4.1.2 Laser Rangefinder

The laser range finder is able to determine the distance from its target by sending a laser beam to a target, using a sensor that can detect the beam reflected from its target and a microcontroller with an accurate timing mechanism. Essentially the distance is determined by a microcontroller through calculation of the time it takes for the sent laser beam to return, that is bounce from its target (Paul 1995, P.145). The laser range finder due to its operating principle has great advantages, such as precision, operates under sunlight with no interference, the maximum range can be of kilometres (depending on the receiving sensors range).

There are limitations to this type of sensor, mainly due to the way they function, the ones that were taken into consideration for the project were:

- **Receiver and optics –** the sensor in the receiving end is only able to detect the laser beam for a certain distance, therefore making the range dependent on the sensor, it may cost more to have higher quality sensors, hence the cost also increases depending on the range required.
- **Thin beam width –** similarly to the sharp infrared range finder (chapter 2.4.1.1), the object can only be detected if it is directly in front of the sensor, meaning that the deviation and dimensions of the target play an important role for an accurate reading.
- **Cost –** The cost of these sensors are quite high, especially for a good precision laser range finder, the cost is dependent on the cost of the receiver.

2.4.1.3 Ultrasonic Rangefinder

Ultrasonic range finders can determine its distance from a target through calculation of time taken for part of an emitted ultrasonic sound, commonly denoted as "ping" (this sound that cannot be heard by human hearing) to be reflected back to its receiver (Paul 1995, P.154). Similarly to the laser range finder (chapter 2.4.1.2), this type of sensor relies on a timing mechanism to calculate the elapsed time from send to receiving of the "ping" signal. The main advantages of the ultrasonic range finders include accuracy, maximum range, operates without any interference of sunlight and cost (for an average range of 4 meters).

The disadvantages of the ultrasonic transducers under the conditions it is expected to work in practice are as follows:

- **Temperature –** the readings vary depending on the temperature of the operating area, due to the fact that sound waves travel at different speeds at different temperatures.
- **Material structure –** material composition and reflectivity of the target can impact measurement, as materials that absorb sound waves may return no readings.
- **Maximum range –** its maximum range is dependent on its sending and receiving transducers, which in essence for the application required can be of a few meters, higher costing sensors in this range can have better range.
- **Beam width (due to the way in which sound travels) –** this aspect can be both advantageous and disadvantageous, this is because an object may not be directly in front of the sensor (but within the ultrasonic width angle) whilst being detected by the transducer which leads to not being able to accurately pinpoint the exact location of the object detected.

2.4.1.4 GPS

Global Positioning Systems was considered as a viable option due to the range being virtually unlimited (as long as it is able to receive GPS satellite signals). A GPS receiver is able to provide users with their current location on Earth by searching for and detecting the distance of the transmitting satellites and using a concept known as resection to determine the location (Ahmed 2002, p.8). The entire process of position determination may be to a certain degree more complex and will not be covered in this report, but taking into consideration the operation principle of the GPS the following advantages and disadvantages are deduced:

- **Precision and range –** a GPS itself is unable to calculate any distance, but through additional code (setting the GPS coordinates of the destination) or program it can be determined and this can theoretically be of any magnitude, which is an advantage. The precision on the other, which can be disadvantageous, is quite poor. Taking into consideration that Standard Positioning Service, which is available to the general public, has a 95% probability level (Ahmed 2002, p.10) which means that exact position can be offset by a few meters. The poor precision, especially in smaller areas, can have great impact in the task completion as expected.
- **Ease of use –** this has to be taken into consideration mainly for the end-user, who as it has been explained in chapter 1.1 can come be of any social background. A GPS system will require the user to set their gardens' limits coordinates into the robot that is to achieve

the desired task, although this may be a one-off setting, it will not be an easy procedure in their perspective. Another point taken into consideration was the area that was to be covered by the device, which may have obstacles or may have different specifications or dimensions; this would result in a requirement of added components (such as other sensors) and consequently costs.

- **Cost and other considerations –** the cost for a GPS module is relatively low taking into account their functionality, but it is still dependant on a good control segment. A GPS can function under almost any climate, temperature and sunlight conditions making it a great option but the precision and ease of use may prove it unsuitable for the desired function.

2.4.1.5 Radio Frequency Position-Location Systems

Radio Frequency (RF) position-location systems were considered for this project, there are numerous techniques described by Everett (1995, pp.395-403) that can be employed using low cost radio frequencies. One of the techniques considered, consists in using 3 RF transmitters and an RF receiver on the ground, the operation principle which would be mainly based in triangulation with the transmitted signal strengths being analysed and a control circuitry that would enable for the location of the mobile vehicle to be determined. The RF approach was abandoned mainly due to the complexity of implementation, as radio frequencies can suffer interference from other signals being transmitted, and other aspects that would require a certain degree of Communication Electronics knowledge and analysis.

This system would have advantages and disadvantages very close to those of the GPS (described in section 2.4.1.4), with the main differences being:

- **Cost –** RF positioning system can be implemented at fairly low-cost (Everett 1995, p.395)
- **Interference –** these can suffer from interference of other radio frequencies being broadcast in the area, which would result in inaccurate readings.
- **Ease of use –** it would be relatively easy for the end-user to operate such device with RF technology because the transmitters could be hard coded with a set frequency, requiring users only to place the three transmitters in the limits of the garden area to be covered.
- **Precision and range –** the maximum range, unlike the GPS, would be limited to the RF signal strength of the transmitters and receiver. Its precision on the other hand would be similar to that of the GPS, which also constitutes a disadvantage for the desired application.

2.4.1.6 Summary of distance measuring sensor selection

From the range of sensors considered (in earlier sections of this report) for implementation in the system, taking into consideration the advantages and disadvantages of each, it is possible to conclude that the ultrasonic range finder is a viable option considering its price, functionality outdoors, precision and the maximum range. A brief summary of the different sensors described previously, that could be used, and their advantages and disadvantages can be seen in Table 21. The average cost mentioned was obtained from Google Product Search for each item (Google, 2011). To note from Table 21, in the row for "*number of sensors needed*", that the abbreviations for transmitter and receiver are expressed as Tx and Rx respectively.

	Ultrasonic Range Finder	**Sharp Infrared Rangefinder**	**Laser Rangefinder**	**RF**	**GPS**
Temperature/ Weather	**3.1m for every 5°C temperature change**	Sunlight causes inaccurate readings. Works best on ambient light	Negligible	Negligible	Negligible
Solution	**Add thermometer or use code**	Use at times with low sunlight	N/A	N/A	N/A
Maximum Range	**3 to 11m at 40kHz frequency**	3m	600m to 25 KM	100m	N/A
Precision	**3cm error**	Poor	Few millimetres	1m error below 5m	*50-100m*
Average Cost	**£15 p/module**	£11 p/module	£15 plus	£5 p/module	£30 p/module
Number of Sensors needed	**1 Tx + 1 Rx**	1 Tx + 1 Rx	1 Tx + 1 Optic	3 Tx + 1 Rx	1 Rx
Limiting factors	**Maximum distance 4m. Temperature variations. Absorbent objects.**	Object colour. Sunlight. Distance measurement inaccurate	The optics. Quality of laser. Sub-nanosecond timing circuitry for time of flight. Speed of light. Object colour & sunlight.	Interference. Small change in signal wavelength	Can only be used in open field/outdoors. Speed may affect readings. User interface.

Table 21 – Sensor selection and options

2.5 System Elements

In this section of the report a description of the noteworthy parts that make up the system is made. Description is provided to the role that each element has in the system and the number of components that may be required is provided. The sub-sections include description of the motors required for both motion of the vehicle and that of the sensor, and requirements of the microcontroller chosen for the development stage.

2.5.1 Motors

In order for the robot to accomplish the required task, motion is required. Motors would therefore enable the robot to move around the garden in the predefined path. Another function that motors play in this project is to provide motion to the sensor, this was chosen as a viable approach due to the fact that motors are relatively low cost components (comparing with the cost of sensors) and using it to provide one sensor the ability to capture data that would otherwise require multiple sensors to accomplish the same, is of advantage. The aspects taken into consideration regarding motors is described in sections to follow.

2.5.1.1 D.C Motors

For the robot to be able to move around the garden and follow the required path, standard D.C motors were chosen for this project. The motor selection takes into consideration the cost, speed and power consumption. From the design method chosen it was concluded that three motors are required, where two would serve the function of driving the robot around the garden and a further motor required to move the blades that cut the grass. A motor driver would be required in order to power these motors and for control functions.

2.5.1.2 Servo Motor

The servo motor plays a fundamental role of providing motion to the ultrasonic transducer chosen in section 2.4.1, this is because its function would be to rotate at a constant speed, whilst the sensor sends signals attempting to determine objects closest to it, such that the distances of the objects in the area to be covered is established. The servo motor was chosen from a regular dc motor due to the ease and precision of control using Pulse Width Modulation (this is explained in detail in section 5.3). It is worth mentioning that use of a **stepper motor** was considered as it is also able to rotate in discrete angular increments with precise positioning and can be programmed to perform a similar function as that of a servo. The selection of the servo motor opposed to the stepper motor was mainly influenced by the number of control pins required for the latter and its relative cost.

2.5.2 Microcontroller

A microcontroller is the so called “brains” of the project, where all relevant functions and processes are taken place. A microcontroller method was chosen from that of building an analogue circuit for this specific function of our robot, mainly because of the simplicity in circuitry that also leads to reduced manufacturing costs. Microcontrollers are now available at a low cost and other benefits include ease of programming, less complexity in hardware, easy to debug in the event of errors, time and resources in building, upgradeability and to a certain degree also future proof. All of these factors were fundamental points taken in consideration when deciding on the most effective solution for the problem.

As most of the specifications such as code size, number of ports required and other elements for the project are not known, an Integrated Development Environment (denoted as IDE here forth). The **Arduino Duemilanove** was chosen as the preferred IDE for this project due to existing knowledge of the platform and for the fact that it is flexible, has sufficient memory (32KB of available memory), has both analogue and digital, input and output pins available, supports Pulse Width Modulation and interface for most basic projects. The requirements of the minimum ports configuration can be seen in Table 22.

Port Type	Number available
Analogue Input or output	5
Digital Input or output	13
Pulse Width Modulation	6
Power	2

Table 22 – Minimum IDE requirements

Using an IDE also means that the code can easily be ported to the final design (a specific microcontroller) if so chosen. The Arduino IDE has onboard an Atmel Atmega 328 microcontroller and has a USB port for communication with the computer for both programming and serial data.

The chosen IDE works around the high level C programming language, therefore knowledge of such language was a requirement for fulfilment of the programming aspect of the project.

2.5.3 External Design and Features

The external aspects of the designs were considered, as the arrangement of the wheels, the positioning of the motors and sensor, and positioning of the blades play an important role in designing a functioning system. The selection of the grass cutting blades was made based on the battery power consumption and size, it was concluded that a component with specifications of the blades used for Robomow (as seen in Figure 25) would be ideal, with rotational cutting form factor.

Figure 25 – Robomow Cutting Blade (Mowers-Online, 2011)

An efficient arrangement for the blades was established as being where the two blades are positioned at opposite ends of the vehicle, with sideways motion (with the limits being close to the wheels) applied to a shaft connected to both blades whilst blade rotation is employed, as illustrated in Figure 26. With such arrangement it was concluded that the vehicle going through a certain area of the garden once would ensure that the grass is correctly cut to length, hence overcoming the existing limitation of existing technology having to go through a certain area of the garden more than once.

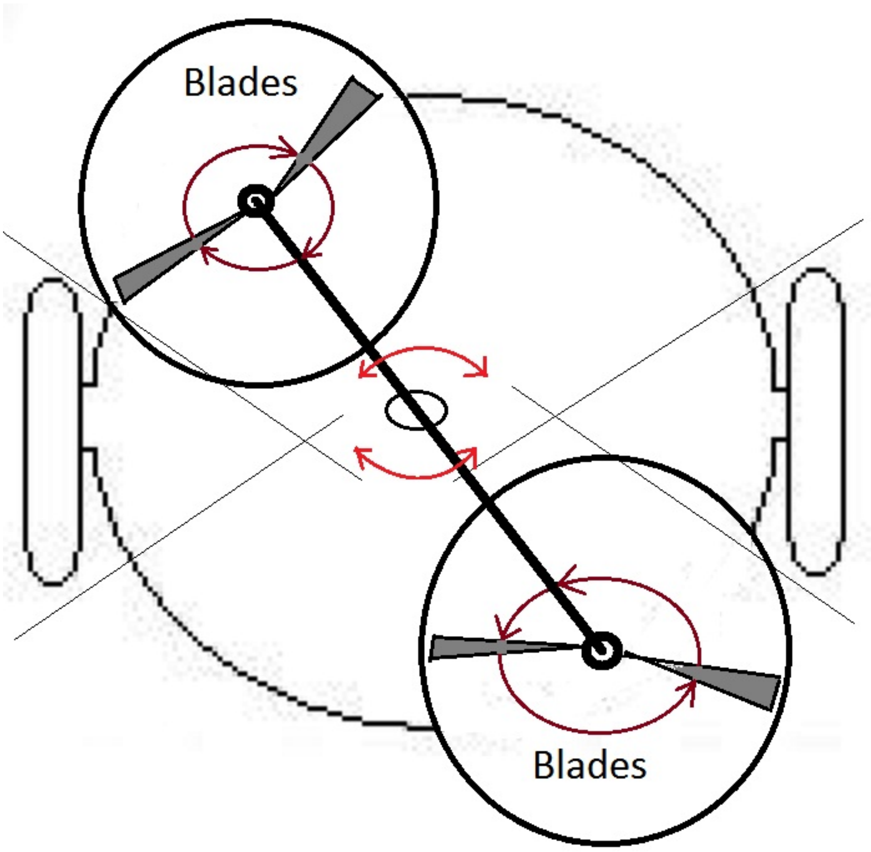


Figure 26 – Bottom view of the vehicle and cutting blades arrangement

Positioning of the servo motor and the sensor on the top of the vehicle would need to be centralized, such that the view of the objects is clear.

3 Design Philosophy

The approach chosen as the solution in building an automated lawn mower described in the report considers basic human perception when part taking in such activity, in other words it is an approach that someone would consider for the action that the robot is to achieve.

Considering that the first task someone considers before cutting grass in their garden would usually involve a visual analysis of the area. During a visual analysis the user initially determines the size of the garden and area they prefer to start to work from, the cutting pattern is also determined and the user would then be positioned at the starting point ready to start. In the case of the user being away from a chosen starting point, they would position themselves there before starting to cut grass. From this initial user action, it was then chosen that the same principle was translated as the actions performed by the robot to achieve the same task. The microcontroller can therefore be programmed with these actions in sequence in order to replicate human actions, in which case the ultrasonic sensor would serve as the robot's eyes enabling for the area to be scanned whilst establishing its parameters.

During motion in the track in a criss-cross motion pattern, it was established that a way to ensure that the vehicle is kept in a straight line is done by having a reference point in the track at which the servo and ultrasonic transducer would be positioned to. During motion the system would ensure that motion is only continued as long as it is still at range of the reference point (a pole). This action is also inspired by human action, where in order for a user mowing the garden, to ensure that they are going in a straight line would normally look at the direction at which they are travelling and ensure that keep within a set path by focusing on and comparing their actual position with that of a reference point.

4 System Design

This section provides explanation of the basic system operation and the operating principle for the automated lawn mower. The system functions and basic operations in order for the tasks assigned to it to be accomplished successfully will be discussed.

4.1 Complete System Operation Overview

The basic principle of operation for this system consists of the ultrasonic transducer being rotated by the servo motor to make a 180 degree turn, during the turn the transducer is scanning through objects and determining its proximity. A predefined function is set in order for the top three closest objects distances are stored in memory whilst it rotates. With the values of the closest objects, it is then possible to set the limits of the area that the vehicle has to cover; this is achieved through a subroutine that calculates the boundaries. This initial step is performed in order to keep the vehicle within the limit set. The next step taken is for the robot to choose one of the closest detected object as the forward path to begin following, it then begins to follow the predefined path, whilst it also checks at certain intervals that it is going in a straight line and within the desired area, and stops once the process is completed. The grass cutting process is quite trivial that is initiated as soon as the vehicle is ready to begin its motion through the predefined path, and is stopped at the end of the path. Figure 41 shows a flowchart of the described operation.

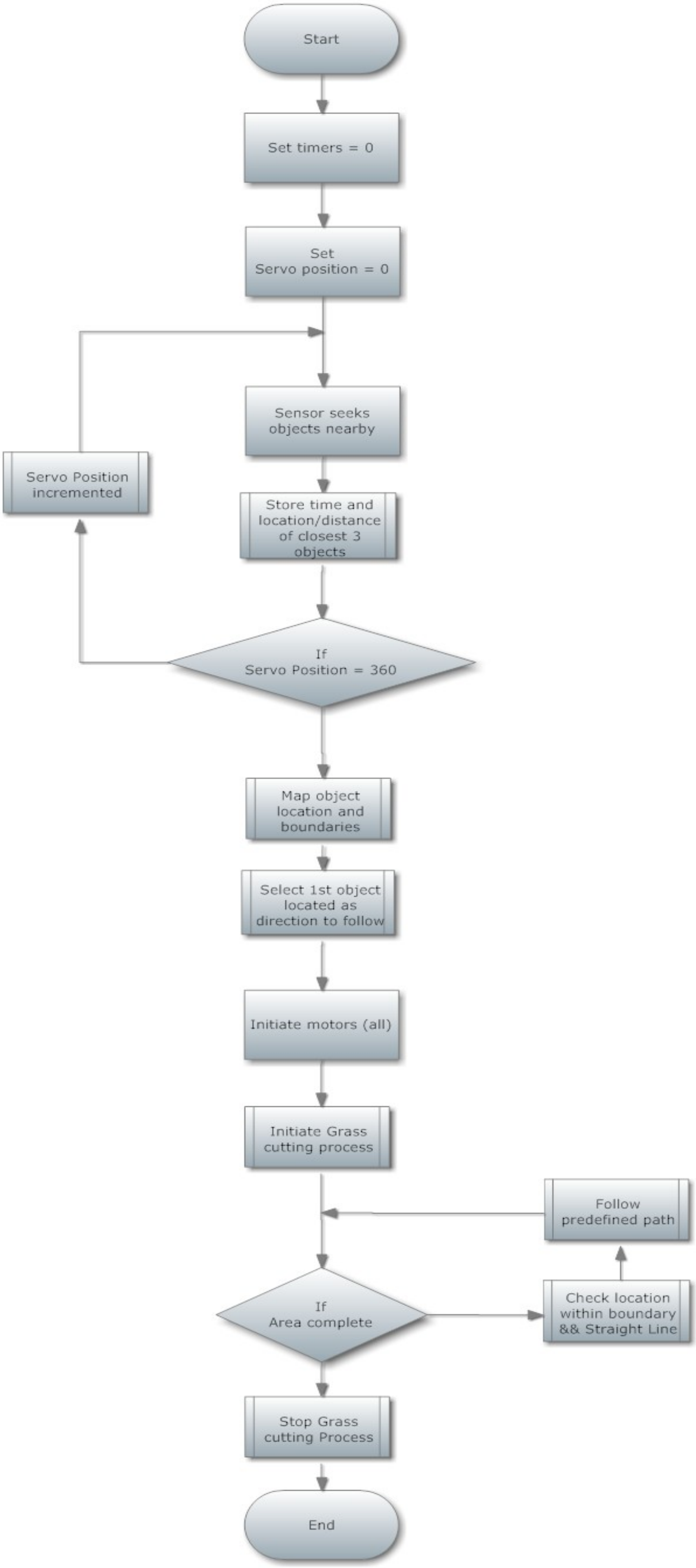


Figure 41 - Flowchart of System Overview

Initially the servo motor, where the ultrasonic transducer is attached to, scans the area for the closest objects in its range. With the points in the field determined, it is possible for the programmed microcontroller to do relevant calculations to determine the limits of the area to be covered. Before motion begins, the servo points to one of the furthest points detected, and ensures the angles match, if the case is true, it then begins motion in that direction following the pre-programmed path. Figure 42 provides illustration of this process as follows:

1. The first task at this state is for the servo motor to scan the area using the ultrasonic transducer attached to it;
2. After calculations are made for its current position to be determined, the servo positions itself to one of the closest poles and confirms that the measurement is correct using the ultrasonic transducer and some predefined angles;
3. The servo motor would then be positioned to an angle of 0 or 180 degree (where the next pole should be located), begins forward motion whilst the sensor seeks for the pole within a set proximity;
4. Once the pole is detected, motion is stopped, the sensor is positioned to the opposite angle (if 0 degrees then 180 degrees used and vice-versa) and motion is applied to the left or right wheel whilst the sensor seeks for the pole within a set proximity which determines when to stop;
5. The pole has been detected and forward motion is applied.
6. The next pole is detected by the sensor and the same process is repeated from point 3.

The set proximity described in this section is the one that is determined through calculations.

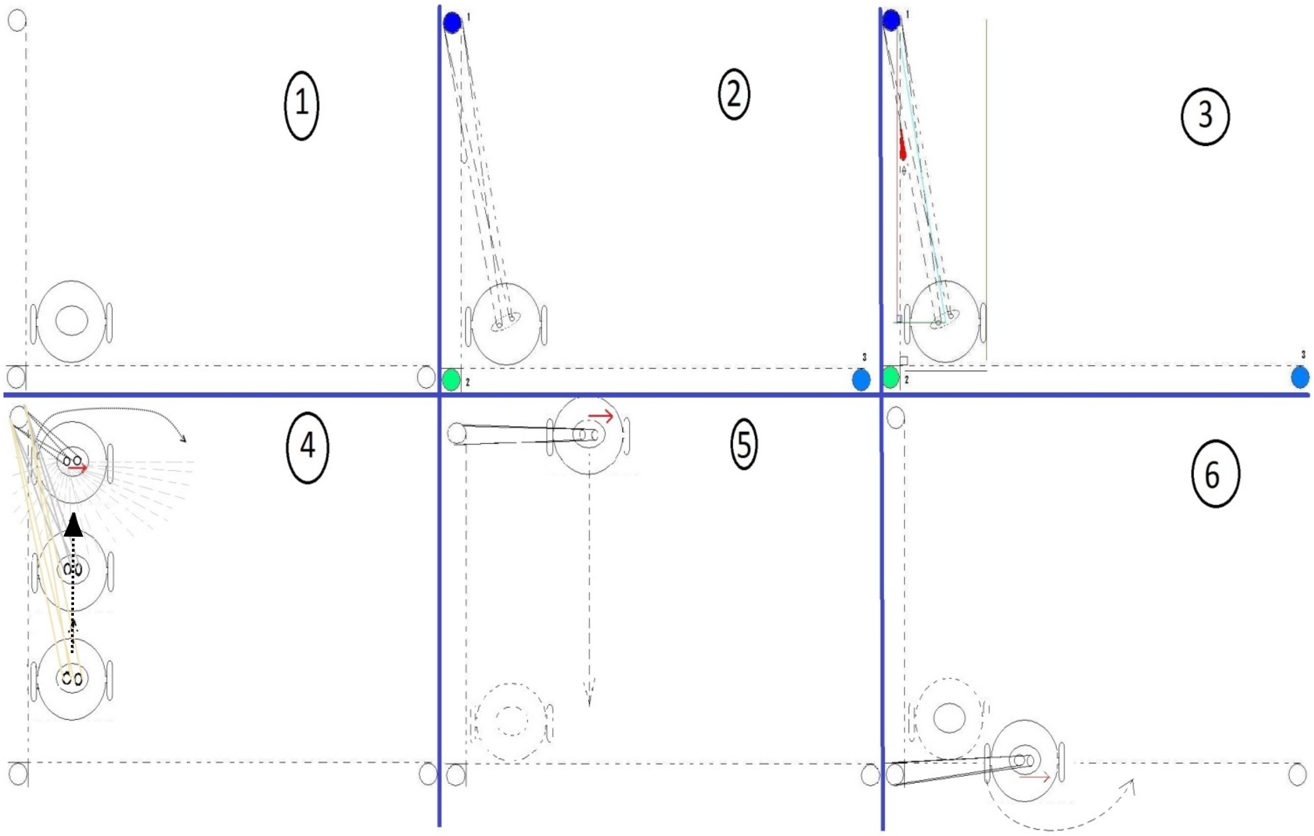


Figure 42 - Sequential overview of determination of area to cover and positioning

4.2 Determination of Pole Locations

The first task performed by the system once powered on and set for operation, is to “sweep” scan the area. Essentially the servo rotates by a degree at a time whilst the ultrasonic sensor measures the distance of any object at that degree, this is represented by the black lines in Figure 43. A value comparison of each distance measurement and the degree at which it was measured is made such that only the closest objects are determined as the limits of the garden. With knowledge of the distance and angle of the closest objects it is then possible to establish the boundaries of the area where the robot is to operate. As shown in Figure 43, the brown poles placed in front of the trees were determined as being the closest objects (represented by the red line) after the initial scan. The comparison process of the closest object is quite simple, where three values are stored in memory and the smallest distance values are stored.

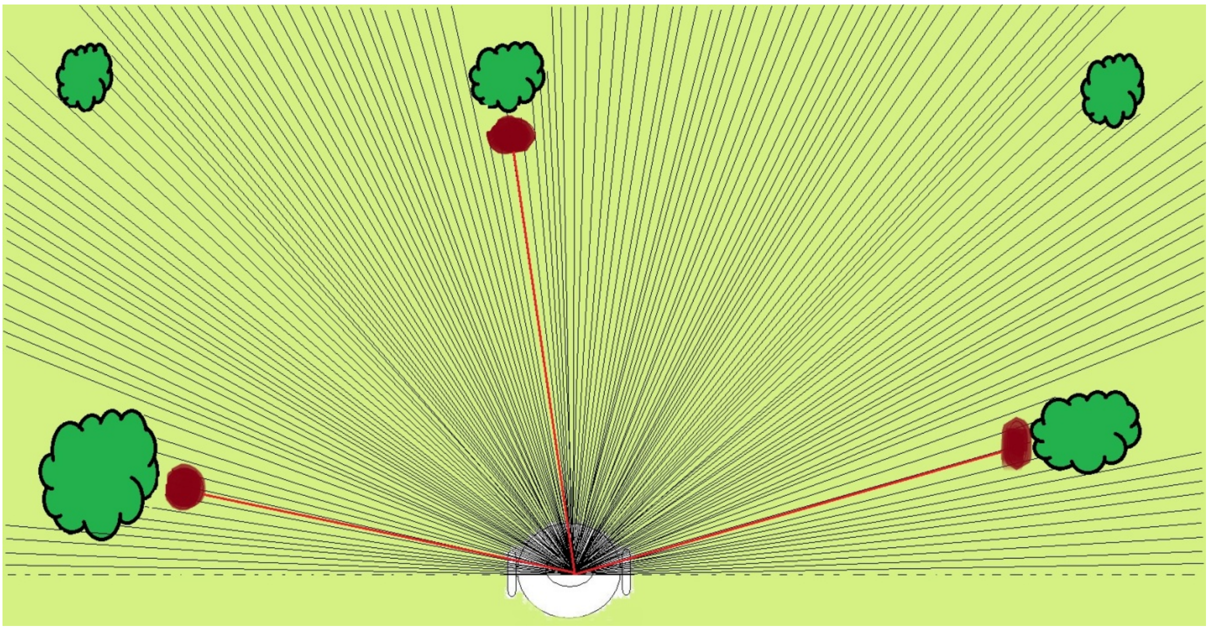


Figure 43 - Scan of area and determination of closest poles

4.3 Determination of Area to Cover

Once the location of each pole is established, it is then possible that the area to be covered is determined. Connecting the three points determined and their angles, it can be seen that they make up a triangle. The area where the robot will operate can then be assumed to be of dimensions of a square or rectangle and from a triangle it is possible to obtain a square or triangular shape. An approach taken was of comparing the distances of the sides of the triangle and selecting only the shortest ones as Xmax and Ymax, as shown in Figure 44 (the faint blue lines) and these therefore will be the maximum limits. A faint yellow line in Figure 44 represents the area that the robot will cover having established the maximum limits in either side of the triangle.

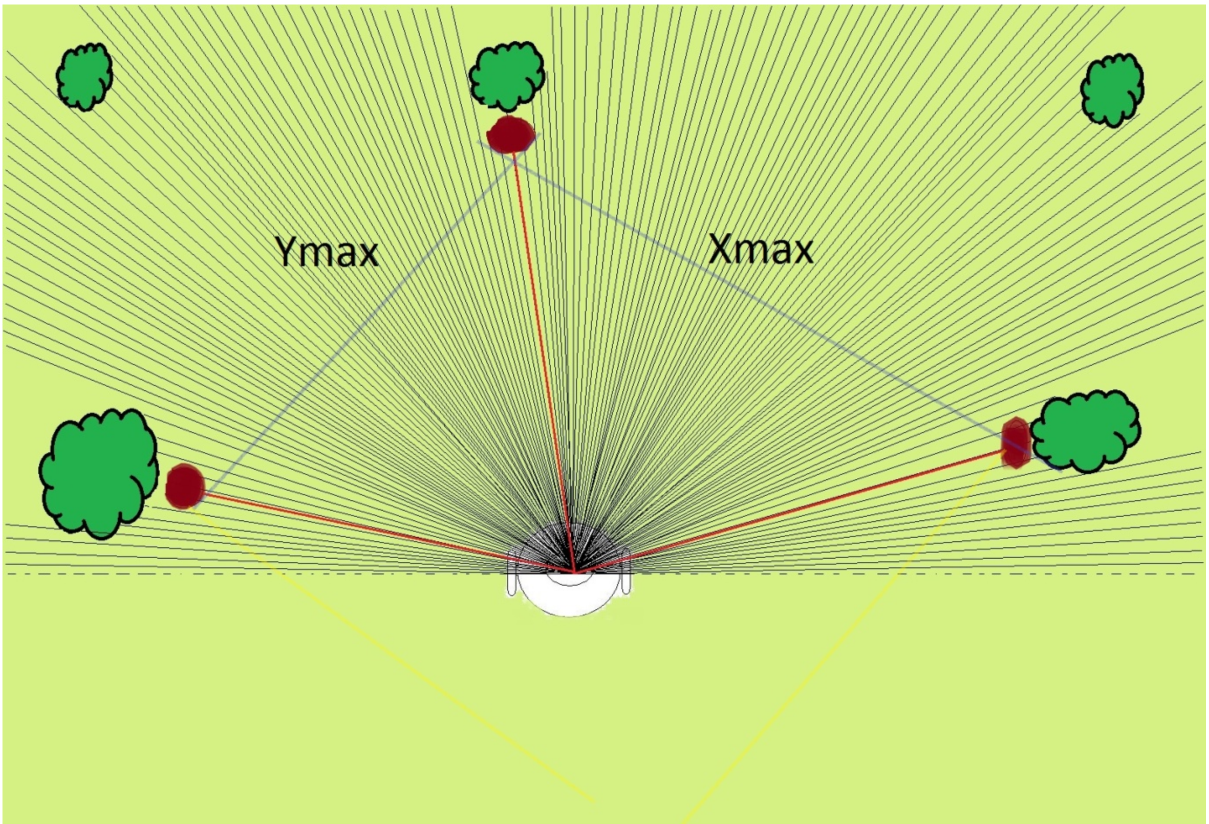


Figure 44 - Determination of Xmax and Ymax

From the determination of the boundaries in the garden, the vehicle can then position itself in one of the poles and establish the distance from an object in its sight. The selection of the pole can be arbitrary as the robot can then verify whether it faces Xmax or Ymax from the values determined previously. The motion of the robot is then established through verification of preset values that can be determined using basic trigonometry; this will be explained in the next section.

4.4 Establishing the Ideal Motion Path

With the specifications of a triangular area is determined and the maximum ranges X and Y was established, it is now possible to deduce the angles and distance from the robot to the pole. This can be determined mathematically using Pythagoras' theorems.

As the maximum range of the sensor is known as 4 meters, it is therefore known that the area to be covered is limited to dimensions meters. Also when considering the known dimensions of the robot as being of radius 14 centimetres, it is now possible to obtain Table 41 using Pythagoras' Theorem 1 shown in Equation 1. The hypotenuse would represent the distance that the ultrasonic sensor would expect to find, the X-axis distance and Y-axis distance are known depending on the vehicle's position in the garden within the poles which were detected as the limits.

Equation 1

By determining the sinusoidal angle between the Y-axis and the hypotenuse (expected pole distance) it is possible for the angle at which the servo should be positioned to be determined.

Table 41 shows an extract of the established points in the field where the servo will check for the pole and the respective angles, the full table can be found in Appendix VI – Ideal Angles Table. To note that all values in the table are rounded up. The values on the first two columns of Table 41 represents the location of the vehicle in the garden area, the first column starts with the dimension of the vehicle with respect to its mid-point (where the servo and sensor is to be placed in hardware) rounded up, that is and the same applies for the other values in that column.

Coordinates (x, y) in cm		Sine (Deg) rounded up
0.1	0.0	48
0.1	0.3	13
0.1	0.6	7
0.1	0.9	4
0.1	1.2	3
0.1	1.5	3
0.1	1.8	2
0.1	2.1	2
0.1	2.4	2
0.1	2.7	1
0.1	3.0	1
0.1	3.3	1
0.1	3.6	1
0.1	3.9	1
0.2	3.9	3
0.2	3.6	3
0.2	3.3	4
0.2	3.0	4
0.2	2.7	4
0.2	2.4	5
0.2	2.1	6
0.2	1.8	7
0.2	1.5	8
0.2	1.2	10
0.2	0.9	13
0.2	0.6	19
0.2	0.3	31
0.2	0.0	48
0.4	0.0	48
0.4	0.3	39
0.4	0.6	28

Table 41 – Determination of the expected angle and the distances

From Table 41 it is possible to obtain a plot of the trajectory travelled by the vehicle and the points at which the vehicle seeks the next pole as shown in Figure 45. To note that the figure represents the motion that the vehicle would make around the track considering the maximum range of 4 meters in both axis, if the range was anywhere within the maximum range (being 3.5, 2, 1.5 … meters) the same principle described would still apply. The servo angle is changed and the expected distance, which is that of the hypotenuse, would be sought for. Detection of the pole at the set angle would provide the system feedback that it is going in the right direction and where it is required to turn.

Figure 45 - Calculated vehicle trajectory and points where the poles are sought after

An approximate representation of how the motion path would be seen in actual implementation in the garden example described in section 4.3 is shown in Figure 46. The vehicle in this case would initiate its motion on the lower left pole (represented by the brown point in Figure 46) and the end point being at the far right pole.

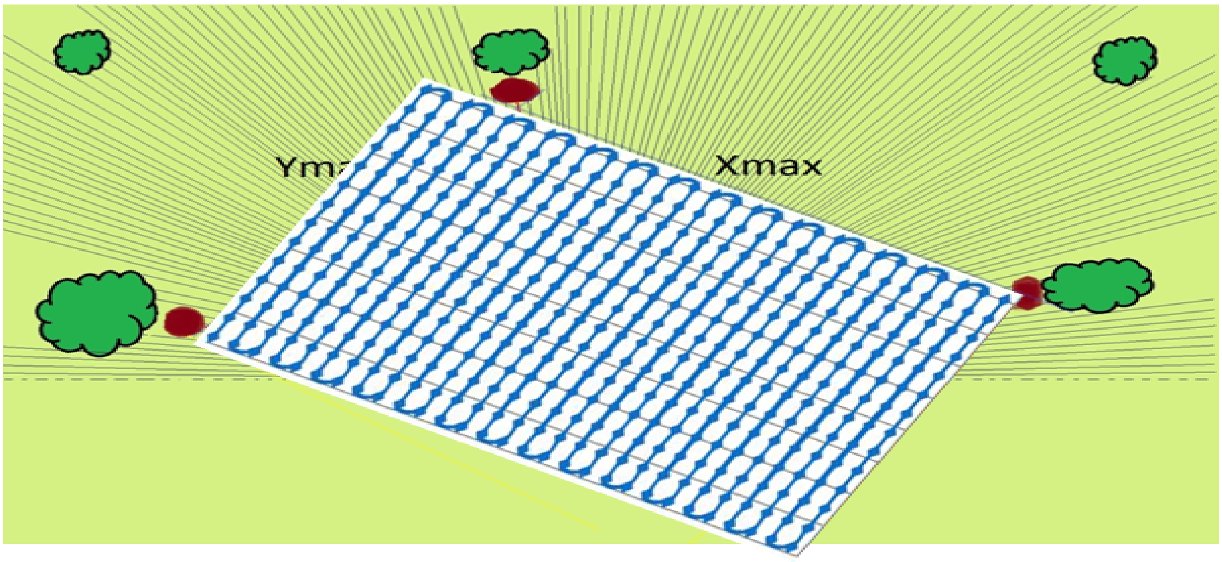


Figure 46 - Motion path in the garden according to the determined garden specifications

4.5 Filtered Data for Actual Motion

The method described in section 4.4 is for an ideal system operation where consideration of each degree of angular change is made, but in actual fact the operation of the sensor has to be considered. This consideration is with regards to the sensor measurement detection, where within a 30 degree angle an object can still be detected as being present. Therefore an arbitrary selection is made for points where the angular change is not considerable. The new points where the vehicle would check for the next pole can now be seen in Figure 47. The corresponding table that provided the plot in Figure 47 can be found in Appendix VII – Filtered Angles Data Table. It can be seen from the new data that the number of points is reduced significantly. To also note that the lines of the graph represent the centre of the vehicle, meaning that the two wheels footprint would be seen in either side of the line in actual implementation.

Figure 47 - Filtered check points

From the initial location zero of the graph, in terms of implementation, the servo is facing towards the number 4 in the Ymax (m) axis (assume that a pole is placed in place of the number). The forward path motion is made towards point 4, where it turns and the servo is then positioned to face the originating point 0.1, 0 (assuming a pole is in place of the numbers) as shown in Figure 48, where each colour lines represent the positioning of the servo in each motion path case. The positioning of the servo's angle can be clearly seen in the last motion path, where the angles are shown and represented as a1, a2, a3 and a4, the fifth angle would be approximately at the axis, hence it is not represented.

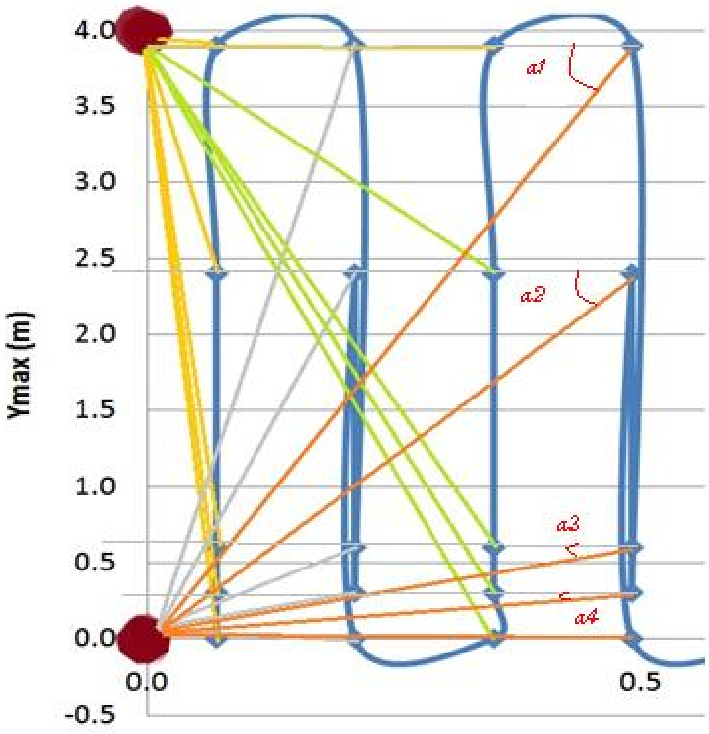


Figure 48 - Servo positioning and expected distances from each point of the graph (extract from Figure 47)

As consequence of the data filtering applied in this section of the report, there are many points at which the system will not see the pole in the respective angle, especially if Ymax is less than 4 meters, therefore it was established that a solution to this would consist of setting the servo at 0 or 180 degree angle at the end of each checkpoint. For instance if the last checkpoint is at a4 (from Figure 6-8), the servo would then be set to face either 0 or 180 degrees (depending on the hardware arrangement) and once a pole is detected by the sensor, the vehicle would then stop and turn to the next motion path.

4.6 Operation from Pre-Defined Data

With the parameters of the garden obtained (in section 4.5), it is now possible for the robot to initiate its motion. The first step is for it to move to one of the poles determined (in the case that it is not at a pole); selection of the pole is made at random. Once at the pole it then searches for another pole within the range of Xmax or Ymax. The search motion is made by first positioning the servo to the angle where the pole is located, the sensor would then begin seeking the distance to the pole (value of the hypotenuse in Table 41) and then by rotating the wheels of the vehicle in opposite directions, opposed to the servo as previously described (in section 4.2) the vehicle would only stop once the sensor obtains the valid reading. This motion method enables for the robot to be positioned facing forward and ready for motion around the track as soon as the pole is found. For instance at the starting point (coordinates (0.1, 0) in Figure 47), say the robot has already made motion to that location and in order for it to face forward on the track it would seek out Xmax or Ymax. The servo is first positioned to a value of 48 degrees from left to right, the wheels are both turned at opposite directions, such that the vehicle moves within its axis and during this motion the sensor is seeking an object at a distance of 4 meters. The motion direction should be made clockwise, as the vehicle is assumed to having made motion to the pole from any location within the area where the poles are placed and therefore would be facing the pole it travelled to. As soon as the pole is found, the vehicle motors are stopped, having the robot now facing forward as shown in Figure 49 and ready to follow the track.

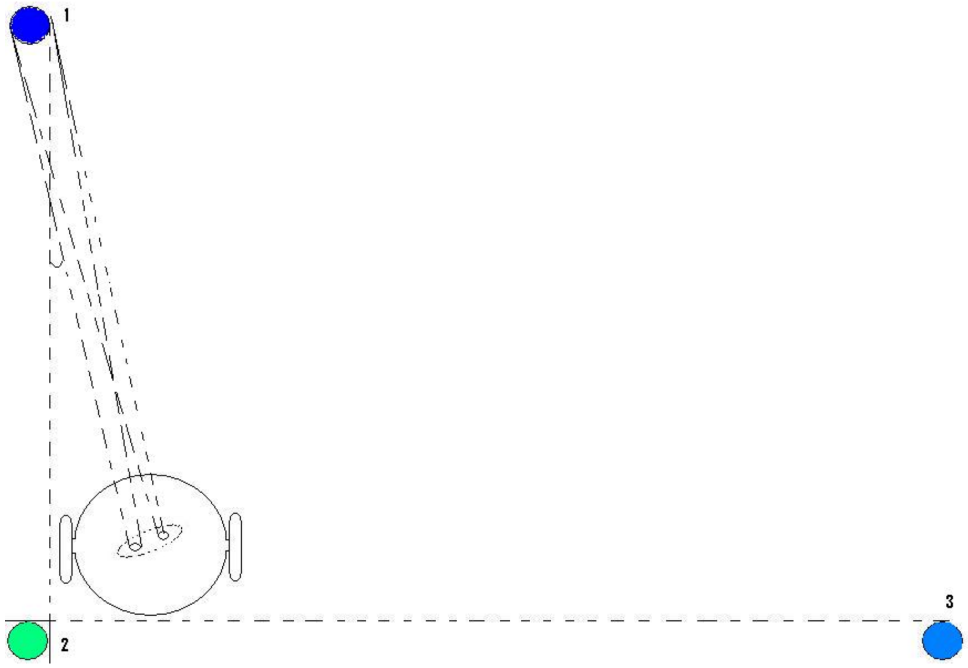


Figure 49 - The robot facing forward once the pole is detected

Once the vehicle is correctly positioned, it is now ready to follow the path. The motion at this stage takes into consideration the path specified on the graph in Figure 47, meaning that forward motion is made and at each *check point* the sensor is positioned to a new angle where the pole is looked for. As an example in the first forward motion path, the robot would go forward with its new angle set and the value of the hypotenuse used as the expected distance that the sensor has to determine as being valid in order to change the angle of the servo to the next. At the end of the first forward track the robot is required to establish that it is required to turn at a 180 degree angle, this can be accomplished by setting a maximum number of checkpoints as a limit for each path which in this case is equal to 5. Therefore when the servo has changed the angular position times, the motors are controlled to make a 180 degree turn to the next path in the opposite direction and this motion is repeated until the entire area is covered.

5 Circuit and Other Hardware Analysis

The system circuit is comprised of four main parts, being the most fundamental component the microcontroller. The other three elements of the system that are responsible for providing motion to the vehicle, motion to the sensor and external information include the ultrasonic range finder, the H-bridge motor controller and the servo motor. According to the circuit schamatics in section 5.4 of the present report, the main aspects are described in the following section.

5.1 The Motor Controller

The H-Bridge motor controller is configured such that each motor can be controlled for forward and reverse motion independently. The motors used in this project, according to the datasheet, require a minimum of 3Vdc each, meaning that the amount of voltage supplied by the Arduino IC (of 5v) would not be sufficient to power both motors. An external power supply using batteries is therefore required to supply power to the motors, this is attached to pin 16 (VCC1) of the motor controller. According to the datasheet for the H-Bridge (Appendix IV – H-Bridge (SN754410NE) Datasheet), there are three main connections (for each motor) required between the Arduino and the H-Bridge motor controller in order for the control signal to be supplied to each motor that is to be controlled:

- One which is to enable or disable the motor (Pins 1 and 9 of the SN754410NE)
- The other two are logic pins which determines the direction of rotation depending on which one is enabled (Pins 2, 7, 10 and 15 of the SN754410NE)

The other two terminals (pins 3, 6, 11 and 14) are for output, that is the signal supplied to the motors.

5.2 SRF05 Ultrasonic Rangefinder

Most of the information from the outside environment, such as objects surrounding the lawnmower, is detected by the SRF05 ultrasonic rangefinder and in this project. This ultrasonic transducer is able to output the distance of sensed objects, according to the product specifications (SRF05 – Ultra-Sonic Ranger technical specifications, *Appendix II – The Arduino IDE*), by sending an eight cycle burst of ultrasound at 40Khz (from the Arduino to the transmitter pin) and with the echo pin set high, the echo is listened for. The echo pin of the module is then set to low, once the signal is bounced back, the time it takes from the signal having been sent to the echo being received is what determines the distance of the object. This device is limited to a four meter range; therefore if no object is found within that range the returned value will be the maximum range. To start off a pulse of 10uS is sent from the transmitter. Between each sensing cycle, a 50mS wait time is required from the time that the first pulse was transmitted; this ensures that accurate readings are obtained. A timing diagram is represented in Figure 51.

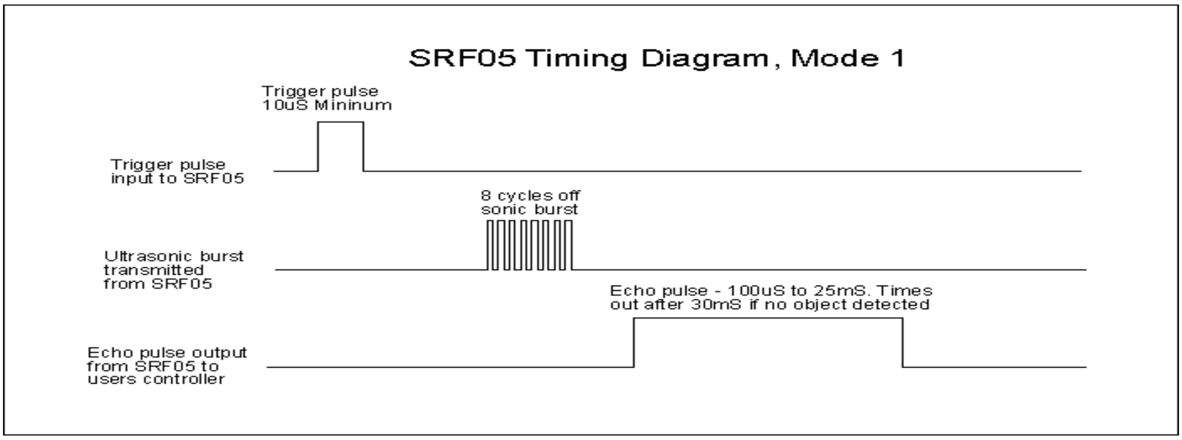


Figure 51 - Timing Diagram for the SRF05 module (SRF05 Technical Documentation, 2011)

The maximum range of this module is of 4 meters. It is noteworthy that the beam pattern of this module is conical (*Appendix II – The Arduino IDE*), the SRF05 is mounted on top of the servo motor vertically such that the signals returned are of a smaller area than when positioned horizontally, although the difference is minor.

As timing is the core method of calculating the distance, a conversion between the time of flight and echo (measured in) is obtained in centimetres by dividing it by 58.

The ultra-sonic rangefinder only requires two control pins (although it can be configured for use of one pin), one for the echo (Pin 2 of the SRF05) and the other for trigger (Pin 3 of the SRF05) and these were chosen to be connected to pins 2 and 3 (respectively) of the Arduino IDE.

5.3 Servo Motor

The servo motor selected for this project is a 180 degree one, which is controlled by one of the pins of the microcontroller (Arduino output pin 5) using pulse width modulation (PWM), which consists of turning a digital (0 and 5v) output pin of the microcontroller on and off at a certain speed such that the current perceived by the component attached to that pin is a varying one, that is any value between 0 and 5 volts depending on the pulse width. The servomechanism has an internal control circuit that enables it to be set to a particular angle, depending on the pulse width; the servo angle can be set and maintained (Monk, 2010, p.138). For instance a pulse width of 1.5 milliseconds will position the shaft of the servo motors at 90 degrees (mid-position), changing the pulse width to 1.75 milliseconds would set it to 180 degrees and 1.25 milliseconds pulse would set the shaft to 0 degrees. Figure 52 shows the angular position described, starting from the minimum pulse (1.25 ms), the pulse width is shown next to it and a representation of the position that the servo would be facing next to that; the same applies for the other angles. The microcontroller sends a pulse signal at least every 20 milliseconds to ensure that the position is maintained.

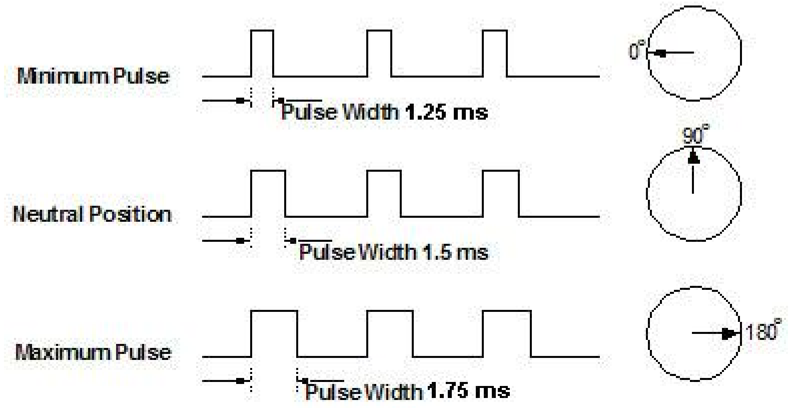
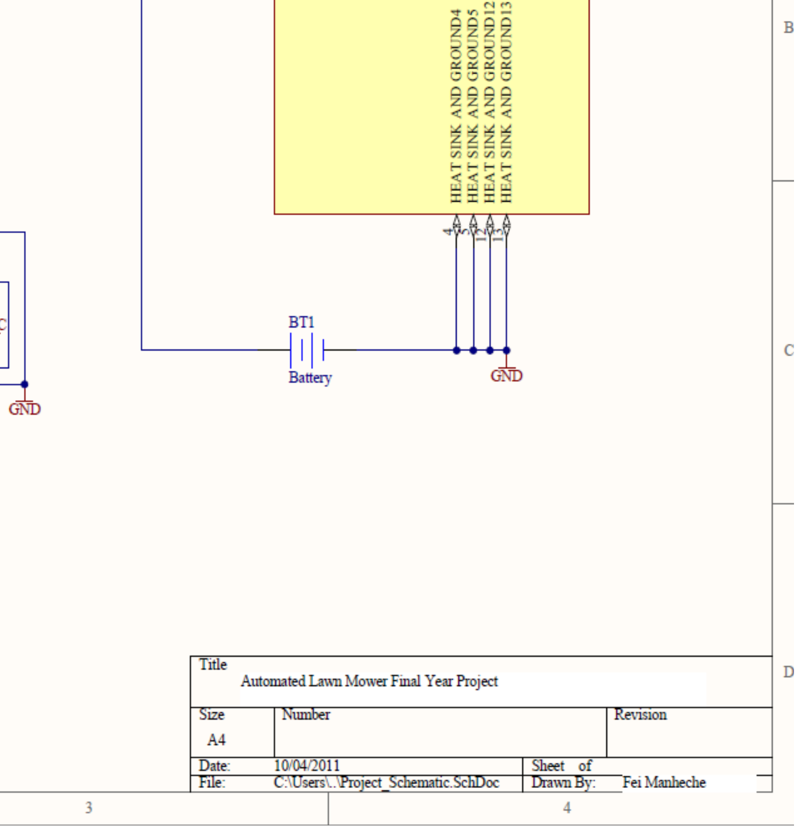


Figure 52 – Angular position maintained by the servo depending on the pulse width (Robotzone, LLC 2008)

5.4 The Circuit Schematics

According to each circuit element description and the essential considerations taken, as mentioned in sections 5.1, 5.2 and 5.3 of the present report. The circuit schematic can be seen in the following page, where the connection between the Arduino and the component is shown with a label for simplicity. A noteworthy aspect from the schematic is that all relevant VCC and GND originate from the Arduino IC, which can be powered by a battery source (9v maximum input) or via a USB connection. The battery compartment (BT1) accomodates which means that an output voltage of



5.5 Arduino Microcontroller Port Configuration

This section of the report will describe the chosen ports on the Arduino IDE and how they are configured for this project.

This project makes use of the Digital I/O ports of the Arduino; there is need for a minimum of three pulse width modulation ports, which is available on the microcontroller. Table 51 demonstrates the chosen ports and its configuration for this project. The selection of ports is to the most part made arbitrarily (out of the 13 digital ports on the Arduino); exception applies for ports that support PWM.

Interface	Arduino Address	Pin Type	Mode (I/O)
SRF05 (sensor) Echo	2	Digital	**Input**
SRF05 (sensor) Transmitter	3	Digital	Output
Servo Motor	5	Digital PWM	Output

Motor 1 Direction Port to motor controller	7	Digital	Output
Motor 2 Direction Port to motor controller	8	Digital	Output
Motor 1 PWM Port to motor controller	9	Digital PWM	Output
Motor 2 PWM Port to motor controller	10	Digital PWM	Output
Motor 1 Enable Port to motor controller	11	Digital	Output
Motor 2 Enable Port to motor controller	12	Digital	Output

Table 51 - Arduino port configuration

6 Implementation

In this section, the basic functions and operations that are to be processed by the microcontroller in order to achieve its motion in the field will be explained. The microcontroller has the task to process the data received from the sensor and be able to determine its location in the field from it. The operations that are to be carried out in their relevant orders follow.

6.1 Ultrasonic Rangefinder and Servo Motor Data

The ultrasonic transducer is able to measure the distance of objects in its sight, this component on its own is unable to provide great flexibility in capturing the data required for calculations in future of the area to be covered. Therefore the use of a 180 degree servo motor is required in order for an area scan to be made and the transducer to be pointed to any pole required, the functions that are dependent on these functions will be explained in further detail in later section of this report. The operation model used for this part of the project is mainly based on “Make a Radar Screen to Visualise Sensor Data from SRF-05” (Larry 2009), where the author designed radar using an ultrasonic transducer (the SRF05) and a servo motor which provided data that was visualized on a computer screen through some additional processing. It is noteworthy that in this section of the report the Arduino library is used.

6.1.1 Servo Motor

The servo motor chosen, as explained in *5.3* of the present report, has an advantageous ability of receiving commands using pulse width modulation to control its rotation by a degree at a time. That is to say that the servo is rotated from its initial 0 degree position in steps of 1 degree its final value of 180 degrees (1, 2, 3, 4, …, 180). During this motion, the ultrasonic transducer measures the distance of any object in front of it, this happens for all angles set and the process repeats itself until the 180 degree rotation is complete, where a reverse motion and operation is performed in order to certify that the readings are correct. A flowchart representing this process can be seen in Figure 61.

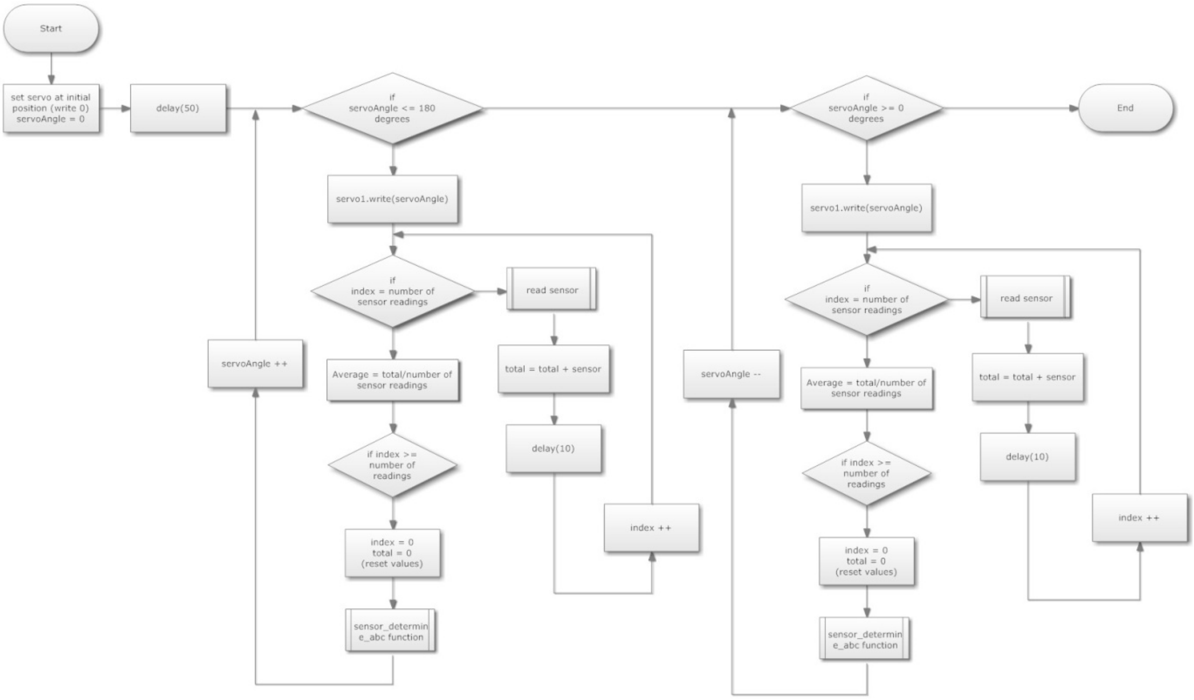


Figure 61 – Flowchart for servo positioning whilst scanning for poles

For most of these actions, use of the Arduino library is used, which provide easy control of the servo angle using PWM on the output pin of the microcontroller by allowing the programmer to write (angle) for the desired angle.

6.1.2 Ultrasonic Rangefinder

The ultrasonic rangefinder, as explained in a section 5.2 of the present report, has a transmitter that sends a sound and waits for the echo to return. For this process on the microcontroller, using a timing function to calculate the duration of the sent pulse to be received by the echo pin, then calculating this duration and converting the value into centimetres (by dividing it by 58), hence the distance of the object is determined. In order to ensure that the readings are as accurate as possible a multiple number of readings are made in each servo position and an average of all readings is taken, being this average the distance of the detected object. For example, if the servo is at position 10 degree, the ultrasonic measurements are made say five times and the sum of these readings is divided by five and this final value would then be the distance to the object. As the flowchart in Figure 62 illustrates the distance determination method employed in the microcontroller code, it can be seen that the pin that connects the Arduino to the SRF05 module is first set HIGH, a delay of 50 milliseconds is then applied before the same pin is set LOW. The receiving pin of the signal is then set to calculate the time it takes for the same pin to go high (when the transmitted signal is sensed) followed by a conversion of the pulse time into centimetres, hence providing as result the distance.

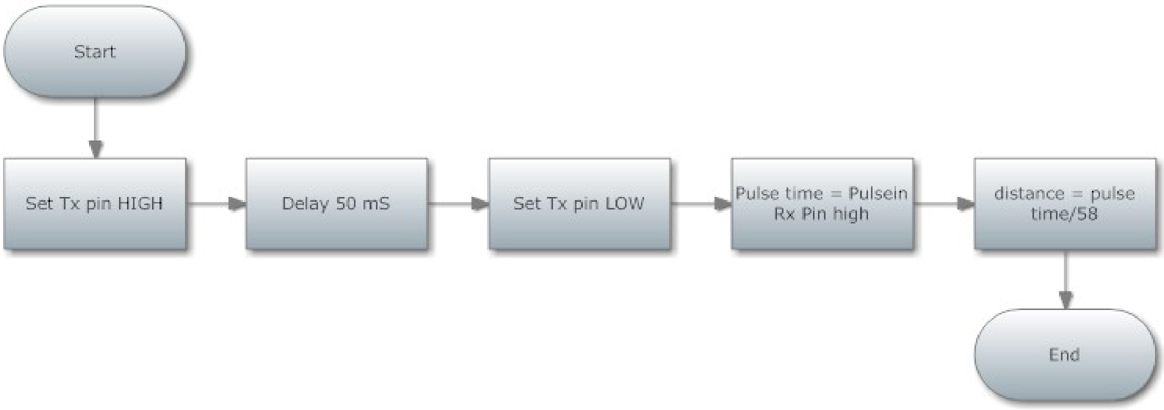


Figure 62 - Ultrasonic distance determination

6.2 Determining the Closest Objects to Store as a, b and c

The ultrasonic transducer measures the distance of an object in its path every time the servo motor rotates by pneumonia degree. During this process different distance measurements are obtained, it is required that only the closest objects detected are stored in memory, therefore conditions should be specified during each scan in order for the right data to be stored in memory locations a, b and c for future calculations. A flowchart representing the complete process is illustrated in Figure 63. Following it is the explanation of the relevant functions.

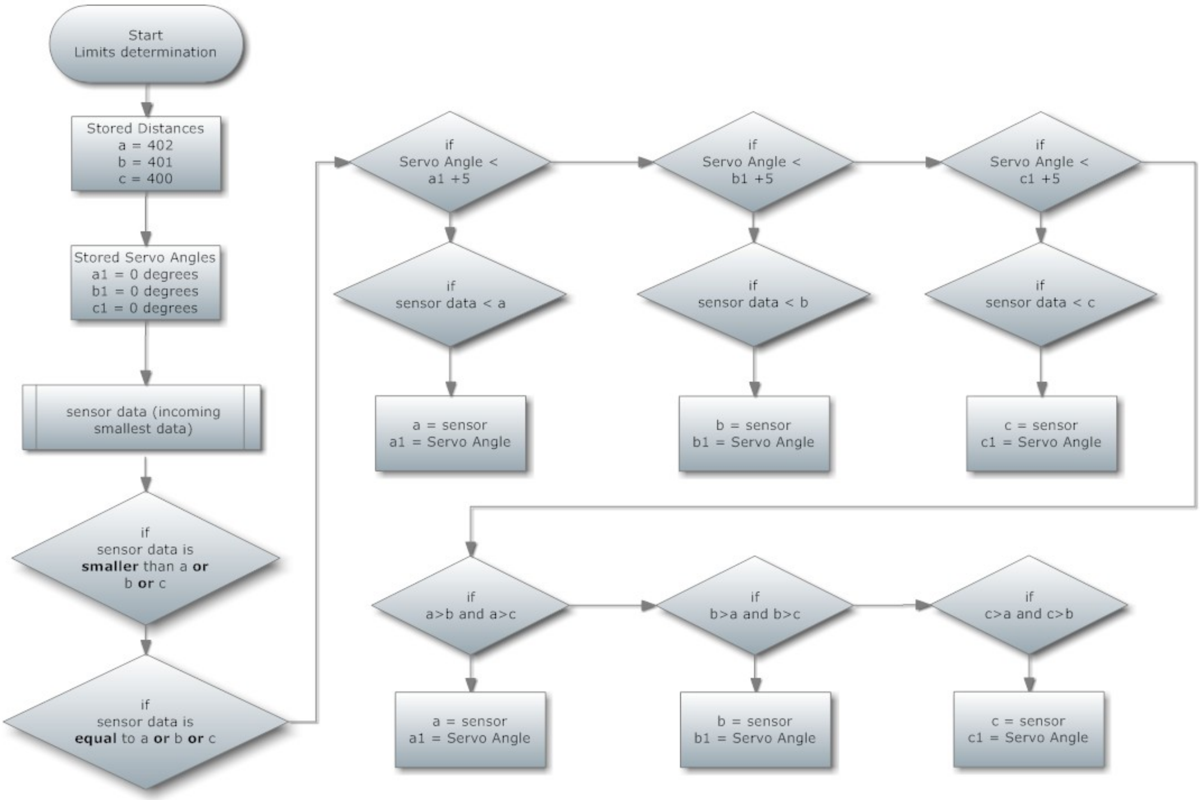


Figure 63 - Flowchart representing the function that determines the closest points a, b and c

6.2.1 First Condition

The first condition is a global comparison between the new sensed distance and the ones stored in memory locations a, b or c. The comparison takes into consideration the smallest value of distance read, being valid for the next comparison if the new reading is smaller than any of those already stored in memory locations a, b or c. As can be seen in Figure 64 the function is initialized with memory locations a, b and c equating to a value close to the maximum range of the sensor (due to programming conflicts they could not be all made equal to 400cm) and the comparison is then made when the sensor data is input. In case the reading does not meet the condition, if the value is greater than the stored values (a, b and c) then it is ignored and no further comparisons are made.

6.2.2 Closest Object Determination by Comparison of Angles between Readings

It is known that each pole is to be placed in the field at a distance no greater than four meters away, the angle between each pole placed would generally be greater than ***30*** degrees, this is because when the user selects an area for the lawnmower to work in, it will normally be as close to a square or rectangle as possible.

With this knowledge, at this point the specifications of the dimensions of the poles (size, width, length amongst others) are not considered; this is because the microprocessor can be programmed to ignore values that are the same within a set degrees scanned.

This function has one condition for when the closest distance is measured and this data is also compared with that of the angle at which the previous closest object was detected. If the angle difference between the first reading and the second is greater than ***30*** degrees, then the new distance is compared with all elements stored and replaced by it (to be explained in the next section). In the case of the angle falling within the set angle, the value of that reading is only compared with the one previously stored in any of the memory locations a, b or c, hence only replacing if a smaller distance measurement is obtained.

For example, if during the first scan, the first closest object having been detected and stored in say memory location **a**, with servo angle 45 degrees at a distance of 2 meters away, then during the second scan say at 50 degrees (servo position) and a distance of 1.9 meters is detected, then the software would only compare this value with that of memory location **a**. In this first case, this new distance and servo angle will be the new elements stored in memory location **a**. Only if the new closest distance sensed falls outside the set angle, then this new distance and its angle would be compared with all the other elements in memory, then replacing it where the distance measured is found to be greatest.

In summary, this condition ensures that the value of the closest distance measured within a set angle is only compared with the value of the previous measurement made.

This comparison is essential in order for the right data to be captured, if this function was not used, the values stored in memory locations a, b and c would at some point return equal or very close to the other, hence making the data not very useful for future calculations.

6.2.3 Condition for when Acceptable Angle Difference is met

In the case of the angles between each reading, as mentioned in section 6.2.1 being greater than **30** degrees (the set angle), a normal comparison between the values stored in memory locations a, b and c is made.

This comparison focuses on storing the closest distance measured. Therefore anytime a new distance measured is smaller than the greatest stored value in any of the memory locations a, b or c, then this new value sensed replaces the one stored.

6.3 Processing of Position

From the data received from the ultrasonic transducer, once it has scanned the field and has provided with the limits of the area it is to cover, returned values of data points **a**, **b** and **c** for the distances between the vehicle and the determined poles or closest objects are obtained and stored in memory. The respective angle between each data point is also stored during the same process. With this information, it is possible to program the microcontroller to determine the area it has to cover. As the ultrasonic transducer has a limitation of four meters maximum range, the garden area will have to be split in sections of maximum of four squared meters. In this section, the initial area (of maximum sensor limit) will be explained and the entire area to be covered will be discussed in a future section.

In order for the points to be used as the limits of operation and determination of the area to be covered, it is important to use some basic mathematical function, the cosine rule for determining the length of an unknown side of a triangle was used in this part of the project. Basic trigonometry was not used due to the unknown angles between each point. As points **a**, **b** and **c** are the distances from the vehicle and the closest objects sensed, and angles **alpha**, **beta** and **gamma** the respective angles as Figure 65 illustrates.

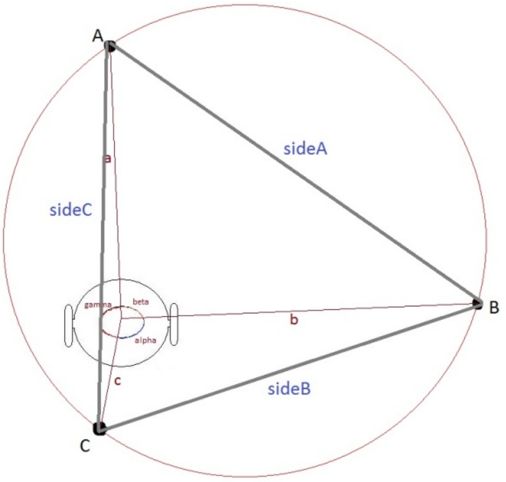


Figure 65 - Determining side A, B and C from sensed distances a, b and c

The distance between point A and B (sideA in Figure 65), B and C (sideB in Figure 65) and C and A (sideC in Figure 65) can then be determined using the *cosine rule* formula as shown in Equation 2.

Equation 2

From Equation 2, the generalised equation deduced for this project, it is noteworthy that variables such as for **sideX** can be for any of the sides sideA, sideB or sideC; the values of distanceX and distanceY can be any of the values of the distance a, b or c respectively depending on which side is to be determined. The angleXY can be any of the angles alpha, beta or gamma, depending on the side required. Equation 2 shows an example for determining the distance of sideA, a similar approach is used when programming, for the remainder sides.

Equation 3

With this information known, it is possible to program the microcontroller to perform this calculation for each side required. A flowchart representing this part of the code can be seen in Figure 66. When programming, it was chosen to have a set function that returns the relevant distance of sideX from Equation 2 and another function to replace the relevant values for that function, this was the preferred method chosen to avoid repetition.

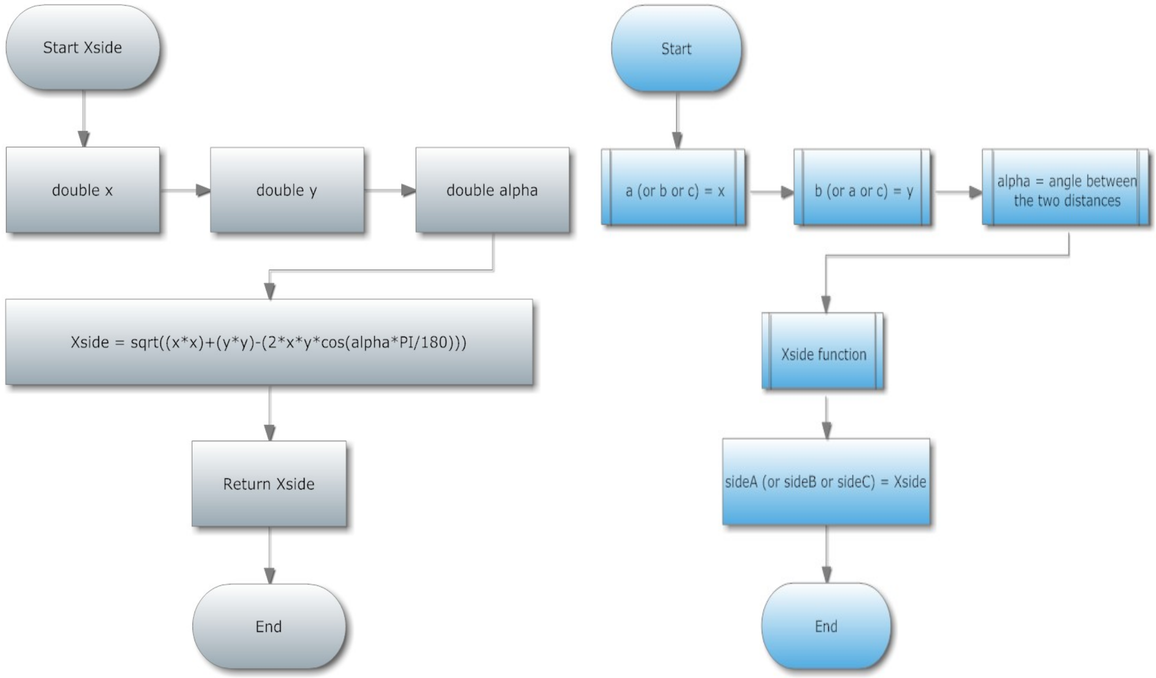


Figure 66 – Flowchart representing the code for determining each side of the triangle

The flowchart on the right (in Figure 66) applies to any of the constants a, b or c and its respective angles, where it first makes them equal to the value of x, y and alpha that are then sent to the Xside function (flowchart on the left of Figure 66) and the result is returned back to the main function (flowchart on the right) storing that new value as sideX.

6.4 Determination of Xmax and Ymax

Determination of the maximum dimensions of the area that the robot has to cover, as described in section 4.3 of the present report, can be programmed onto the microcontroller as a simple comparison between each side of the triangle. This comparison is to enable the two sides of the triangle with shortest distances to be selected as the values of Xmax and Ymax, this method was concluded from the fact that through calculations the hypotenuse of a right angle triangle has always the greatest length. Therefore selecting the two shortest values of the triangle ensures that the maximum distance of either axis is not greater than 4 meters, the values of sides selected is then chosen as Xmax and Ymax (any side can be stored as either maximum limit). Figure X shows the flowchart for the selection of the maximum sides.

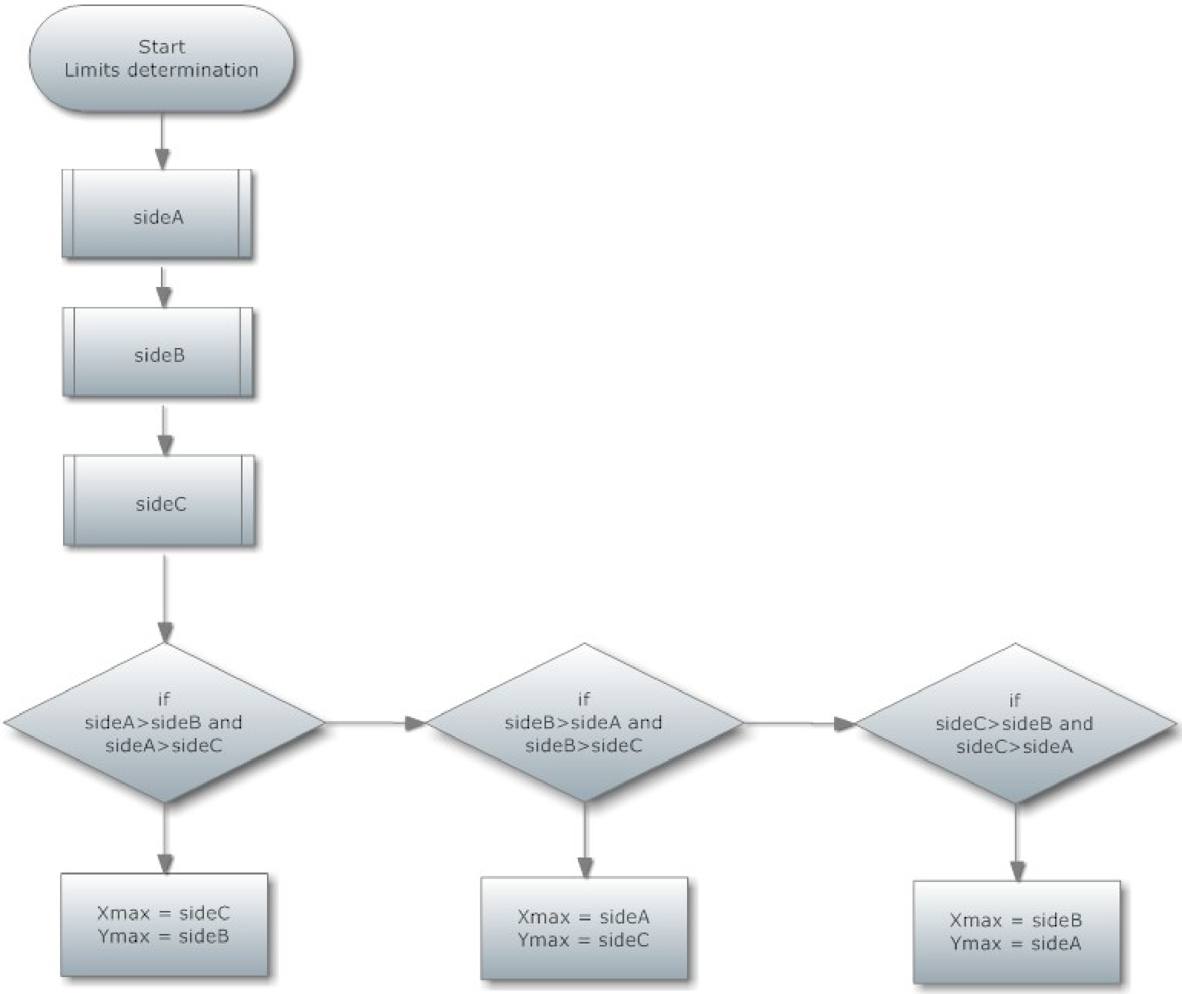


Figure 67 - Determination of Xmax and Ymax values

6.5 Pole Selection

In case the robot is initially located somewhere in the garden area other than close to a pole, it is important for it to be able to move to a pole where motion can begin. This can be achieved by first choosing a pole to go to, this selection although at random takes into consideration the closest distance to a pole detected.

The function that determines and provides motion control to the pole works as represented by the flowchart in Figure 68. The constants a, b and c represent the distances from the poles, as determined in section 6.2 of the present report. From the known values of the constants, it is possible to compare them such that the smallest value is chosen as the pole that the robot has to go to. The constant name is then stored as a new constant "dist", this done in order to enable for the value to be used on the **faceForward** and **straight2pole** functions that are for the motor control and are described in detail in sections 6.6 and 6.7.1 respectively of the present report. Before the motor functions are called the servo is first positioned to the angle where the chosen pole was determined.

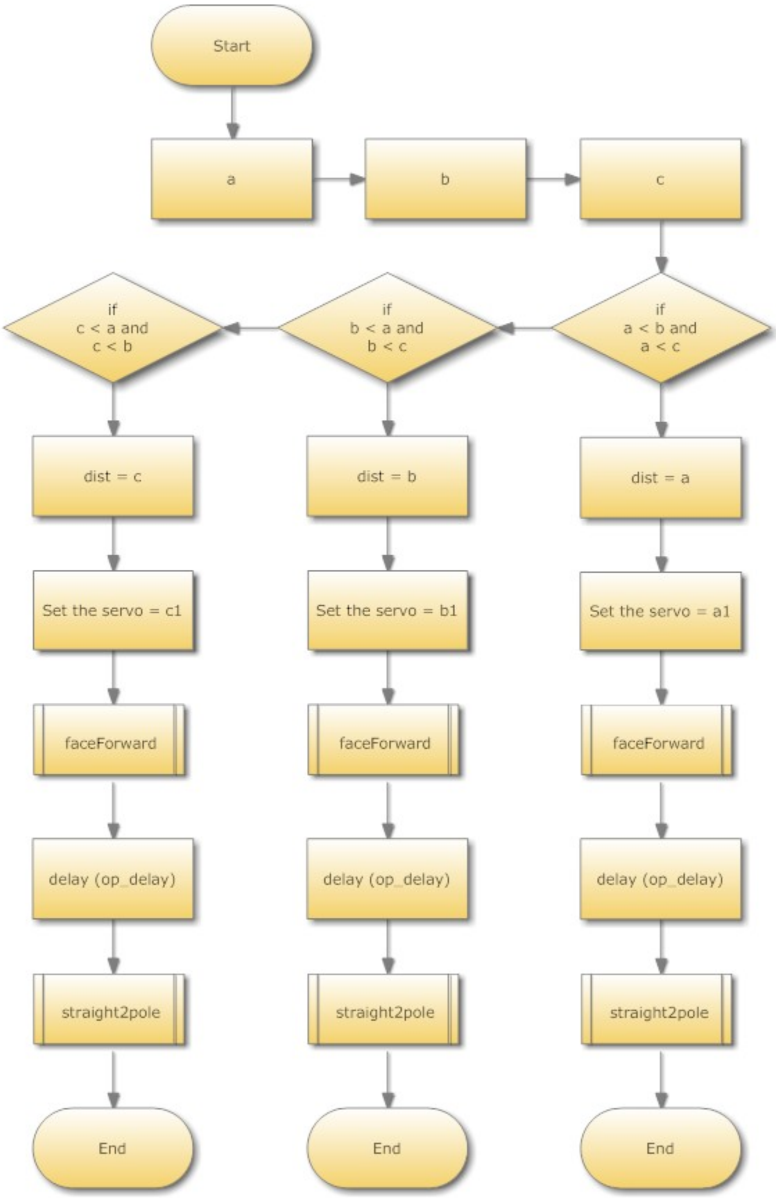


Figure 68 – Flowchart representing selection of a pole and motion to the starting point in the garden

6.6 Facing Forward

This function relates to a closed loop of the motor control, where according to values that are preset and that from the sensor reading, the robot is able to face forward. This function works mainly on a offset basis to position the robot facing the desired direction, where the servo is first positioned to the angle that the robot should be facing forward and the wheels are then controlled to turn in opposite directions, whilst the sensor reads for the expected distance to the object that serves as the reference point hence determining when the wheels motion should be stopped.

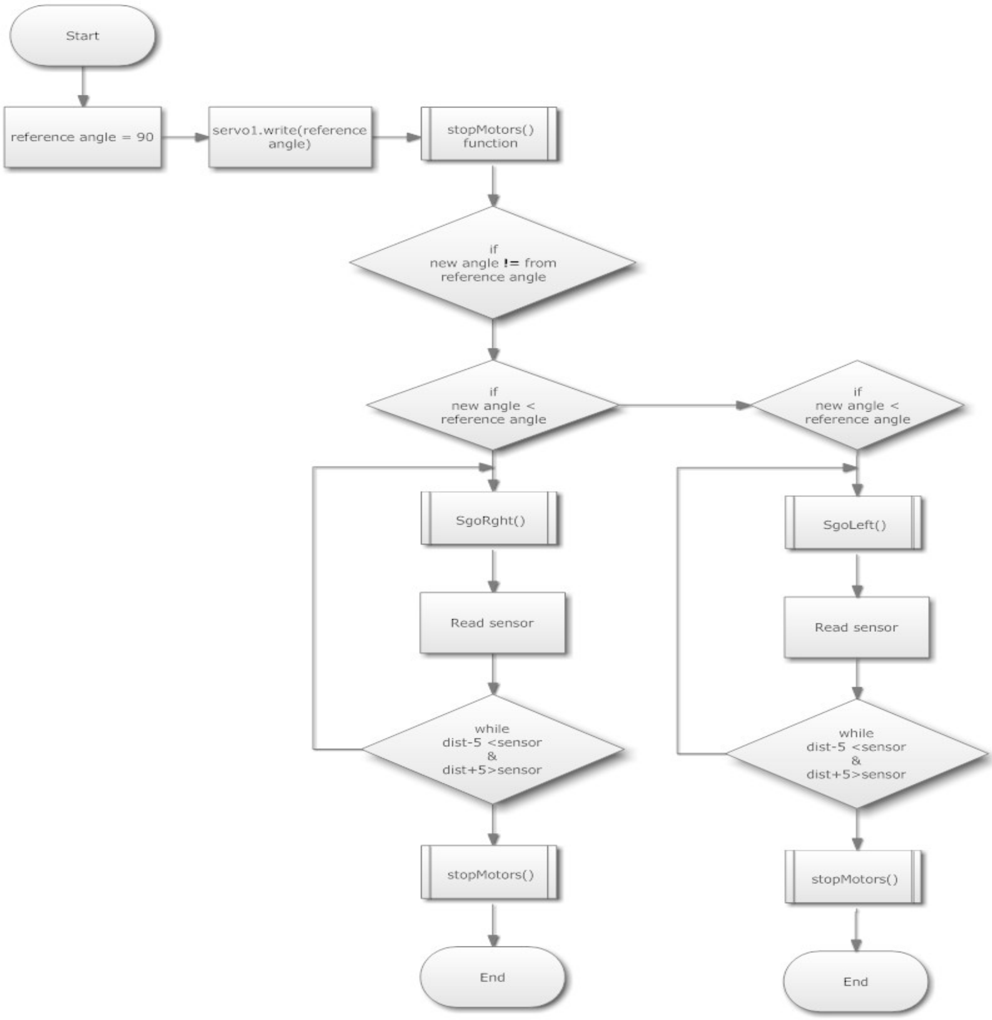


Figure 69 – Flowchart for the robot motion in order to face the required direction

Figure 6-9 represents the function that controls the motion of the wheels of the robot such that it is able to face the desired direction. The function is started with a preset reference angle, which would be for when the sensor is facing the same direction as the wheels in the forward direction. A new angle (from any function this one, such as the one described in section 6.5 of the present report) is supplied to this function such that a comparison is made to ensure that the vehicle is not already facing the desired direction, in the case that it is not then determination of the motion direction is made by choosing the closest angle distance from the reference. Once the direction of motion is established, the system calls on the motor control function (described in 6.9 of the present report) that enables the wheels to turn at opposite directions, whilst the sensor is scanning for the expected distance from the object. There is a leeway of plus and minus 5cm given for the expected distance from the object that the sensor seeks for, this is done as a precaution due to possible imprecision of the sensor or errors during motion. The vehicle only stops motion when the expected distance to the object is determined as valid meaning that the vehicle is now facing the desired direction.

6.7 Closed Loop Motion Function

A global closed loop function was made such that motions of the robot only occur whilst the condition specified is met. The closed loop in this case consists of the ultrasonic rangefinder information which is compared with a preset value that will determine when the robot should stop motion. This function relies on the parameters of motion type (forward, reverse, left, right, etc) and the value that enables for the loop to be ended (any number) to be specified by the function that will be using it, this will be described in later subsections of the present report.

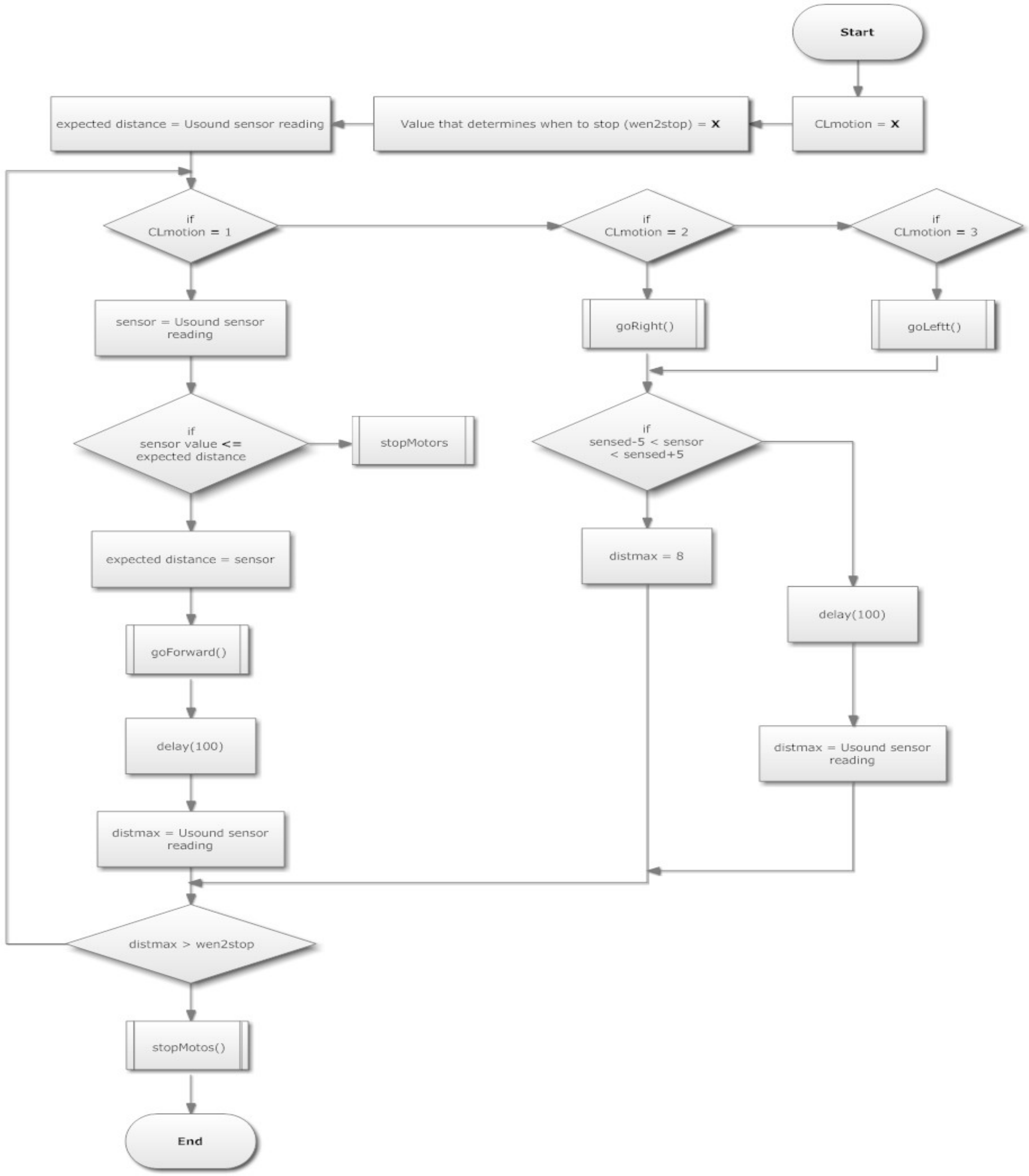


Figure 610 - Flowchart representing the closed loop motion function

From the flowchart shown in Figure 610, it can be seen that there are two values that determine how the motors should behave depending on the value obtained from the sensor reading. The way the data is input from the other functions works by setting the type of motion value followed by the distance when to stop, which is expressed in Equation 4.

Equation 5

Each condition of motion will be explained in more detail in section 6.7.1 of the present report.

6.7.1 Straight to Pole Function

The straight to pole function is used for when the vehicle is at its starting position, after having determined the poles and chosen which one to go to, it makes its motion to the chosen pole (as mentioned in section 6.5 of the present report). In this function, information is provided to the closed loop motion function () such that forward (which is expressed as 1) motor control function is activated and the limit is set as 10 cm (considering the dimensions of the vehicle and possible errors). This function, according to Equation 6 would then be made up of the following input:

Now if you analyse the flowchart shown in Figure 610, it is clear that and so it is now known that the vehicle will continue moving forward as long as the sensor reads a value greater than 10 cm. A further condition in case the vehicle goes off the straight line motion and this is accomplished by checking that the value obtained from the sensor reading is always smaller than the one determined in a previous reading. At this stage of the project, the robot would simply stop when it goes off track, but a better algorithm would be devised such that the robot repositions itself instead.

6.8 Follow Path

The follow path function is the main function of the code that enables motion around the track. The function relies on the specifications established and explained in section 4.5 of the present report. The function relies on two arrays, where the data in the **hypotenuse** and **Sine (Deg) rounded up** rows from the table in Appendix VII – Filtered Angles Data Table, are used. The data in the **hypotenuse** row are used as the angles that the expected sensor distances that it is required to read in order to determine as valid its current position, whilst the **Sine (Deg) rounded up** is used as the angle that the servo should be positioned for the next reading. This section will look into three distinct parts that play a fundamental role in ensuring that the robot is able to fulfil the requirement.

6.8.1 Determining the Array Pointer Position

The first step carried out in this function is to establish the range of operation. As the data points were filtered, it is necessary for the value of the limit determined (Xmax and Ymax) to be rounded to one of five ranges, this is done through a simple rounding up function that specifies the expected range of the ultrasonic rangefinder in relation to the table values. There are five distinct points that were chosen as the filtered data, so there are 5 ranges for the data. Furthermore it is noteworthy that this selection also determines where the array should be pointed to in the code.

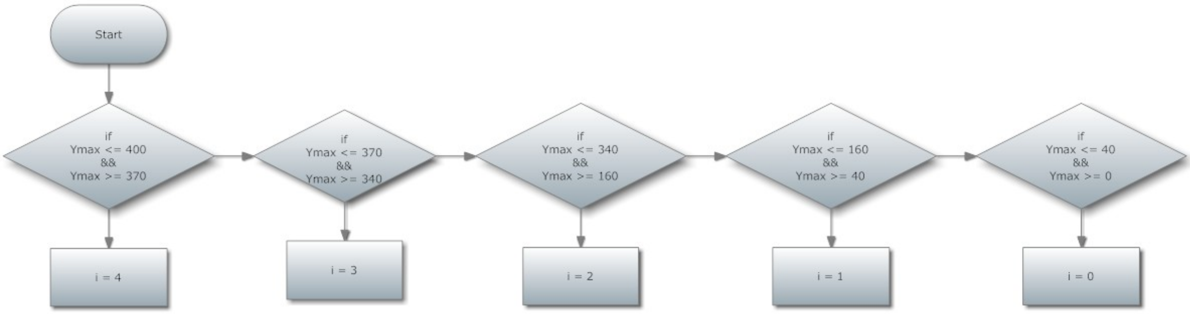


Figure 611 – Determining the range of the data and setting the pointer position

From Figure 611 it can be seen that the comparison assigns the range at which the array pointer () should be equated to. For instance if Ymax was determined as being 200, then, meaning that in the array the third element would be the first value used.

6.8.2 Closed Loop Motion Function

For the closed loop function, where it is required that the robot ensures that it is going in the right track, consists of two arrays whose data is compared with the live data from the sensors. The arrays created, using the predetermined data (as determined in section 4.5) are of size 145 and one is created for the servo angle and the other for the expected distances, they are denoted and respectively. This function is called by the Main follow path function in order to ensure that the vehicle maintains in the right path. Figure 612 illustrates how this function operates.

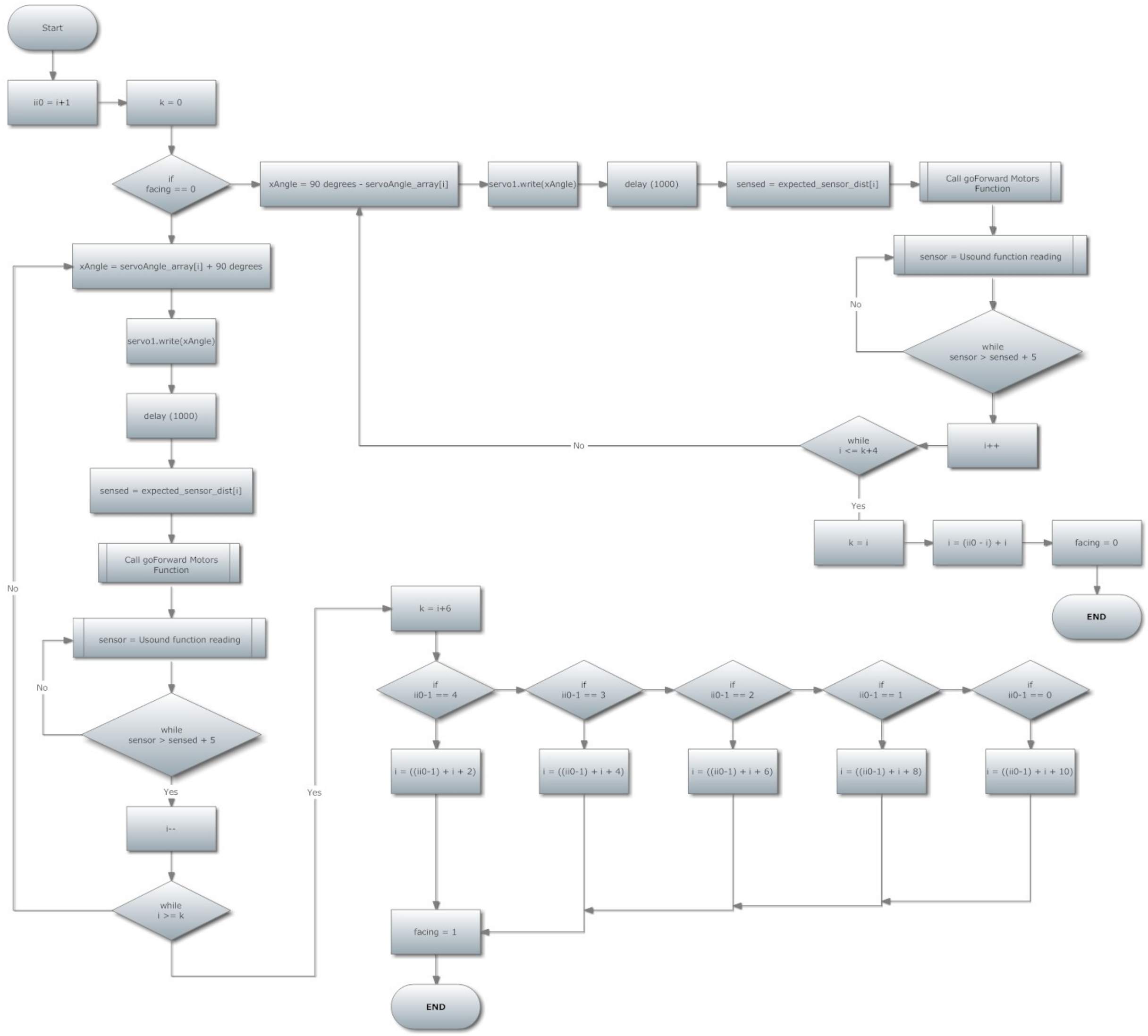


Figure 612 - Closed loop check pole during motion function flowchart

From Figure 612 it is noteworthy that the servo angle is set with an offset of degrees, this is because it was identified through hardware implementation that the servo would be positioned with 0 degrees facing the opposite side from that used during calculations. Another aspect from the flowchart is that a different task is applied according to the direction of motion and that during each motion path; the function is only run once.

6.8.3 Main Follow Path function

In the main function, the distinct elements that ensure that the robot is going in the right path include the array element pointer, servo position and the expected distance. The first thing performed in this function is to set the servo at the position where the pole is supposed to be located, according to the array pointer position which was established in the previous section (6.8.1) the value in that array position would be written as the new servo angle. A reading would be made by the ultrasonic rangefinder and if the distance from the object is determined to be equal to that in the expected sensor distance array (the same array position), then the reading is considered valid and motion can continue. A more detailed explanation of this process can be seen from the

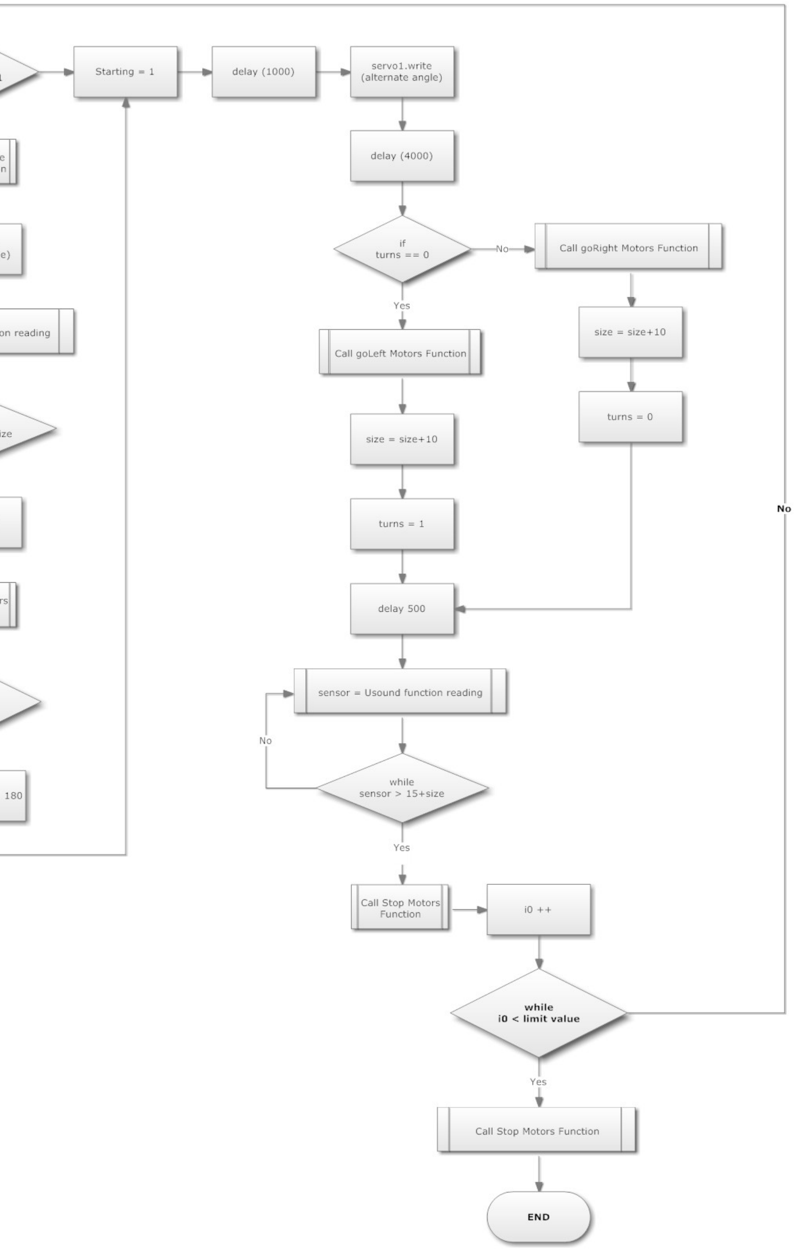


Figure 613 – Flowchart representing how motion around the garden can be achieved in the code

The flowchart in Figure 613 shows how the function that controls motion around the garden taking into consideration the reading of each array. The most relevant points from the flowchart in Figure 613 are as follows:

- The servo is initialized to the position where the pole should next be identified (from the servo data array), the sensor is then checked and the value returned is later compared with that stored on the expected sensor distance array (with a margin of acceptable reading). During this process the motors are activated for forward motion. This section is governed by the closed loop *check pole motion* function (described in section 6.8.2).
- Once the *check pole motion* function is exited, the servo is further set to either 0 or 180 degree angle and the sensor is polled until the pole is found within a set distance (size of the

vehicle’s midpoint which is 10cm and a se 15cm). The vehicle is stopped when the pole is detected at a valid distance and it is then turned.

- Once at the limit the motors are stopped and depending on the previous direction of rotation, the relevant rotation motion is applied by first setting the servo facing the opposite angle (as described in section 4.1), then polling the sensor whilst the alternate wheel motion is applied.
- The alternate wheel motion consists of a counter which switches depending on the motion used.
- The process is only ended once the last array element is reached meaning that it has covered the required area.

6.9 Motor control Functions

The motor control function provides the relevant signals to the motor controller which in turn is able to output the desired motor motion, being either forward or reverse for each motor. Once the relevant ports are correctly initialized on the microcontroller (Arduino), it is then possible to send the desired control signals. The motor control functions consist of two main components, one being responsible in adjusting the motor speed and the other for the direction.

6.9.1 Motor Velocity Function

The motor velocity function simply enables or disables the relevant output port on the microcontroller in order to accomplish the desired motion. This function is able to operate once established three components which are the velocity, the direction pin and the PWM value; these values are supplied from the functions that will call for this one. There are distinct conditions that have to be met before attempting to supply the value to the motor, one of which is to ensure that the value written to the motor controller does not exceed decimal, which in PWM terms equates to volts output. The other conditions simply determine which control signal should be sent. The flowchart in Figure 614 demonstrates the function, where the following observations are made:

- When the value for the velocity is equal to zero, both the directional and PWM control variables (which relates to the output ports) are set high, which according to the H-Bridge motor controllers’ datasheet the motor will remain stopped.
- Two further states are present where rotation is either in the forward or reverse direction (observed from the output shaft of the motor); this is controlled according to the value of velocity that is specified by the other function that calls this one.
- In order for the motor to be rotated in forward motion the value of velocity supplied should be greater than zero, the directional control output port of the microcontroller is made digital LOW (0v) and the value of velocity specified is set to the PWM output port of the motor being controlled, hence the speed is set.
- In order for the motor to be rotated in reverse, the same principle as that for forward motion is employed with the only difference being the directional output control port is in this case set digital HIGH (5v).

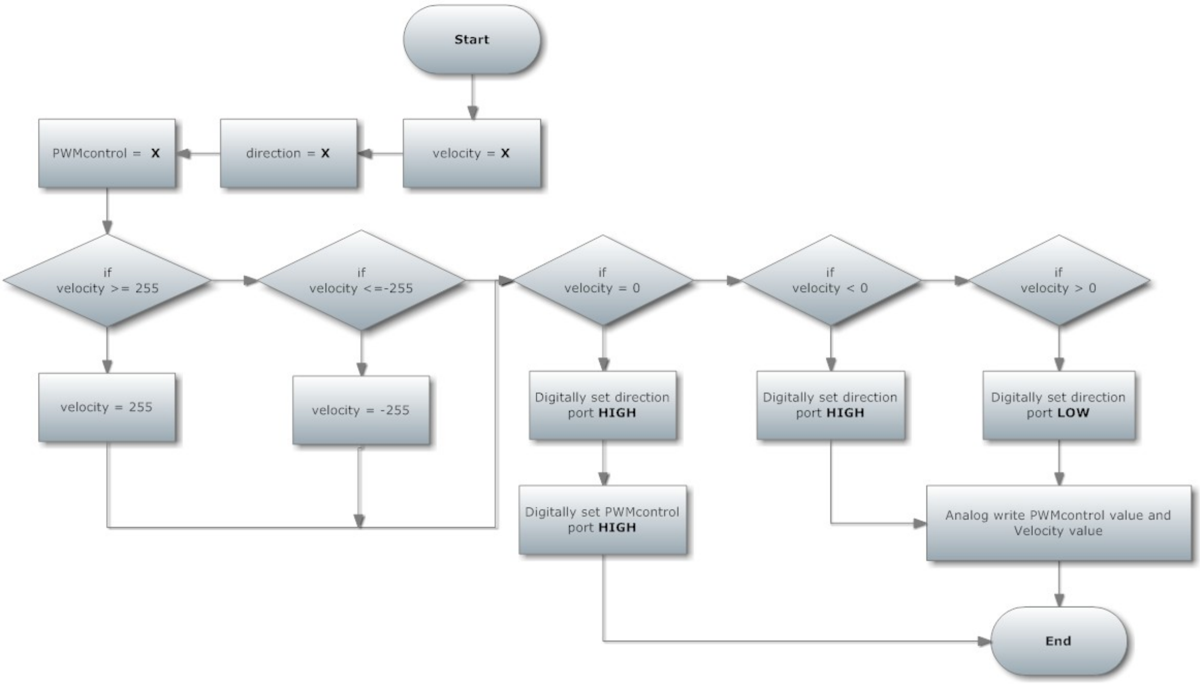


Figure 614 – Flowchart for motor speed control

For this function the only numeric value that can be varied is the velocity, which can be of any decimal value from 0 to 255. From Figure 614 the velocity, direction and PWM control variables are represented with an X is in place of the value specified to this function when called. The motor directional control function (section 6.9.2 of the present report) is the one that determines which parameters in this function should be set to which value.

6.9.2 Motor Directional Control

This function governs the direction control port and speed of motor rotation. This function is an intermediary one which depending on which vehicle motion is desired, the values are input into this one. The flowchart in Figure 615 demonstrates how the values are set for the motor velocity function mentioned in section 6.9.1 of the present report. The only variable that can be modified in this function is the velocity which can be of any decimal value from 0 to 255, the other variables specifies to the microcontroller which output port to assign the value to. Therefore from Figure 615, and specifies whether the control signal is sent to the left or right motor (numbered as 1 or 2 in code).

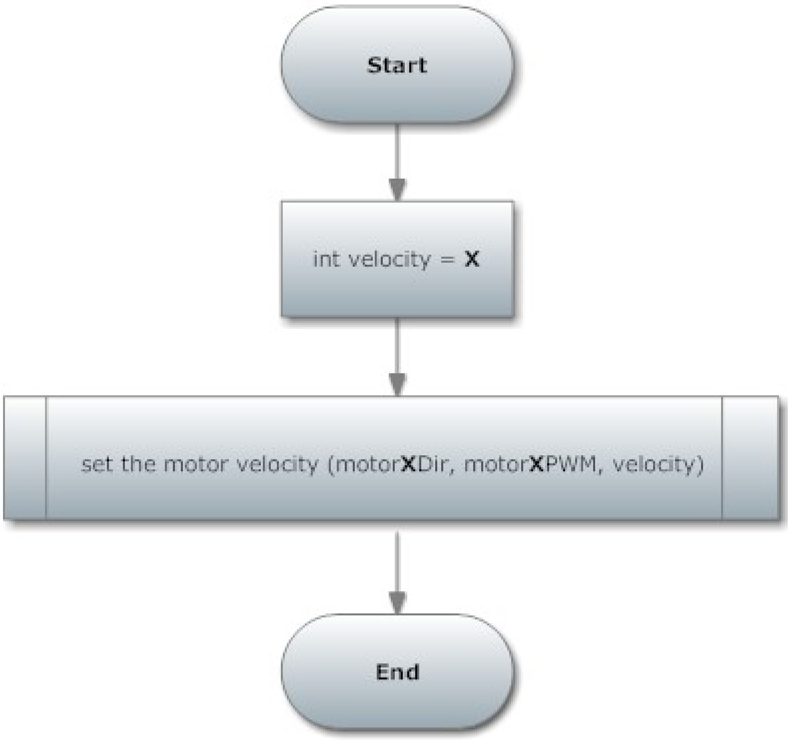


Figure 615 – Flowchart that sets the motion direction of the motors

In order to send a control signal to the motor from the sub-functions in code, it is a simple matter of inputting the value to be applied to each motor. Table 61 shows the values that can be used for the desired vehicle motion direction, meaning for example that if the vehicle is to be controlled for forward motion, then the value set to both motors is 255 (this value is for maximum speed). The relevant aspect to note from the table is that the motor values can be of any value depending on the desired output speed, but the sign (plus or minus) is the relevant aspect that determines the motor rotation.

Vehicle Motion/Action	Left Motor Value	Right Motor Value
Forward	255	255
Reverse	-255	-255
Stop Motors	0	0
Left (one wheel motion)	0	255
Right (one wheel motion)	255	0
Clockwise (both wheels opposite motion)	-255	255
Counter-clockwise (both wheels opposite motion)	255	-255

Table 61 – Motor control values that enable for the accomplishment of the desired vehicle direction motion

6.10 Main Routine

The main routine of the program puts together all functions previously described in this section of the report. Once the microcontroller is powered on, after initialization commands, the program loops in this function which calls for the other functions in a sequential manner such it is able to perform the desired task. All functions described in this section relate to the functions mentioned in this section of the report (6). A flowchart of the sequential arrangement of the functions is shown in Figure 616.

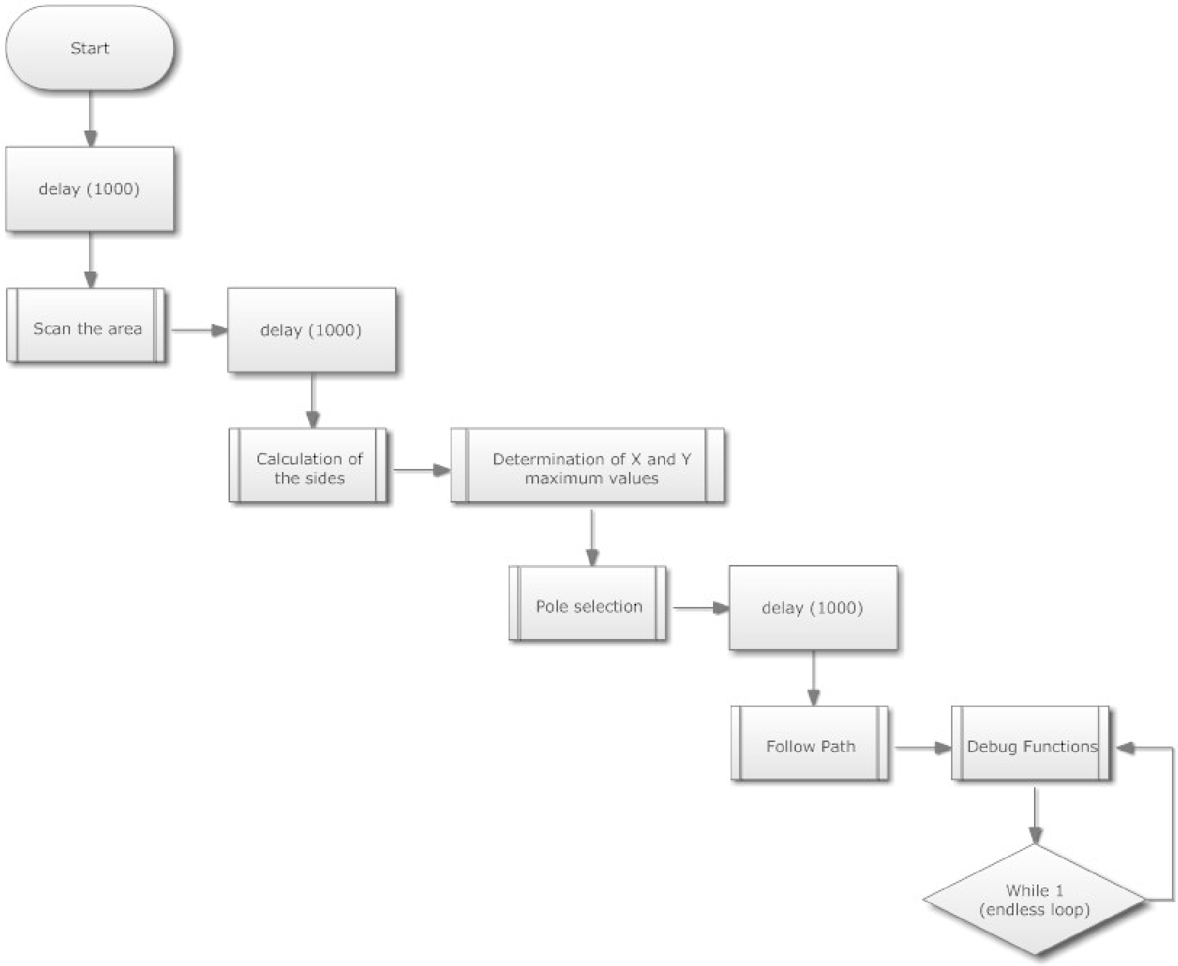


Figure 616 - Flowchart representing the main program operations

From Figure 616 the most important aspects are as follows:

- Once the system is powered on, a short delay of 1s is applied before it begins scanning the area and obtaining the parameters of the working area (as explained in sections 6.1, 6.2 and 6.3 of this report);
- After a short delay from the area scan and pole determination, system calculations for determining the distance between each of the poles detected and the maximum values of X and Y are performed (as described in sections 6.3 and 6.4);
- The system once obtained the plan parameters required for operation, it would then select what pole to follow and initiate motion to it (section 6.5);
- At the point when the vehicle reaches the pole, depending on the *Pole selection* function, it is then ready for motion around the track and this is governed by the *Follow Path* function (section 6.8). Upon its completion, the system enters a debug loop where to the user, no further action is performed.

7 Testing

Various tests were run to ensure that the system performs as expected and that the problem solving method described in the present report is valid. All tests mentioned in this section were performed indoors with average room temperature () and a 6 volt battery output for the motors, it is noteworthy that results may vary under different conditions.

The testing stage involved an initial test of each function before going on to the complete system test, this testing method enables for any bugs (errors) to be identified and corrected with much ease. The initial setup was made such that the vehicle was positioned within the three poles that serve as the limits of the garden, as a 180 degree servo was used for testing it was therefore required that the poles were positioned within that area, as illustrated in Figure 71. From the figure it can be seen that the three poles are represented by the white cylindrically shaped paper, the distance between them is of approximately 1 meter and the robot is positioned centrally at the lower portion of Figure 71, where the servo zero degrees is located to the right of the image (red arrow) and rotation is made counter-clockwise (orange arrow).

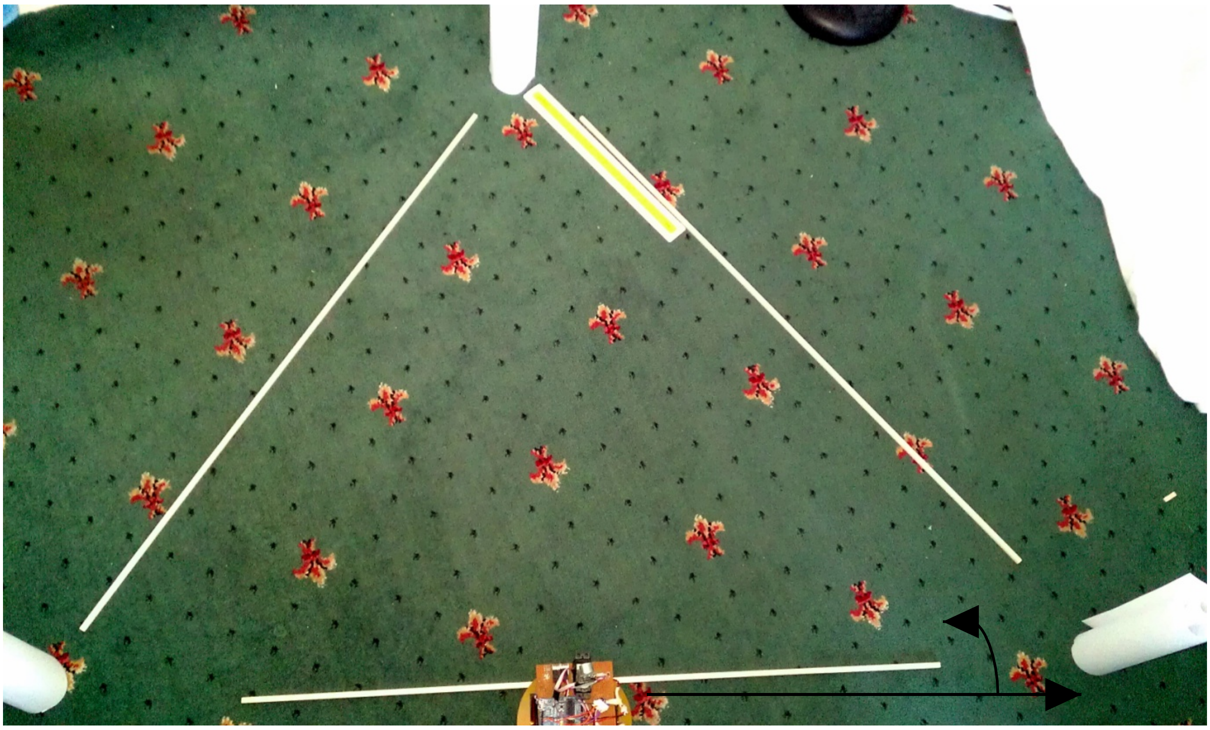


Figure 71 - Working space arrangement for testing

The first test made was for the motors, to ensure that they are able to maintain both constant forward motion and drive in a straight line. The test was first conducted using a voltmeter where for both motors 3.6 volts was measured as running across each motors' terminals.
Subsequently, numerous tests were carried out at different surfaces and it was identified that best results, that is the vehicle maintained forward motion, was when the surface was cemented or clear.
On carpeted floors after a short distance the vehicle would turn by a few degrees to the right.
The solution to this undesired outcome was identified as requiring the vehicle to have wheels that

cover a greater surface area.

These first tests are crucial in ensuring that accurate results are obtained in other tests where motion is involve. The other tests carried out required that serial connection (of 115200 baud rate) between the microcontroller and the computer was used in order to analyse the data received by the sensor.

7.1 Test Results for the Pole Determination and Calculation Outputs

This test was performed to obtain confirmation that the pole determination function is operating as expected. Using programmed serial commands (from coding process) it was possible to obtain an output of the readings as shown in Graph 1, the source table can be found in Appendix VIII – Test Data. The *Servo Angle* axis of the figure represents a complete cycle of the scan (from 0 to 180 degrees and back to 0 degrees) therefore the plot results should be mirrored about the 180 degree point.

Graph 1 – Plot of the test results for servo angle and distance measured

The results of this test were then compared with the values that were determined by the closest object determination function (described in section 6.2 of the present report) in order to establish if the correct poles were chosen by the system. The values stored by the system from the initial scan can be seen in Table 71, where the value stored in memory location A, B or C for the distance determined and the respective angle is shown.

Distance A	Angle A	Distance B	Angle B	Distance C	Angle C
78	9	102	79	89	172

Table 71 – Results from closest object determination for a, b and c storage

Comparing the calculation results between system determined and actual measurements for the distance between each pole is demonstrated in Table 72.

	Actual measurement (cm)	System Determined (cm)
Distance between first and second value	103	105.0988922119
Distance between second and third value	110	138.8351898193
Distance between third and first value	160	165.1736450195

Table 72 – Results from calculation of the distance between poles

It was concluded from both visual and data analysis that although with a small degree of error, the system is able to determine the poles at the approximate actual location as expected. The data obtained from these tests can be compared with the actual test area as shown in Figure 71.

7.2 Motion Tests

Once it is established that the system is able to determine the correct values where it is to operate, it is then possible to test the functions where motion is involved and the closed loop aspects of the system. For these tests a visual analysis was made to ensure that the system is functioning accordingly. The results of the most important functions are provided in the Table 73 below.

Test	Result	Notes
Face Forward	The servo is positioned to the correct (90 degrees) angle and opposite rotation is applied to the wheels, the vehicle stops once the pole is detected in front.	Some tests returned with the vehicle being a few degrees offset from the pole it was expected to be facing. The reason was due to the sensor's detection method (section 5.2).
Go Forward (closed loop)	The system continues motion as long as the pole is in view and stops as expected.	
Follow Path (closed loop)	The servo motor changes angles accordingly, the sensor data is obtained throughout motion and the system stops when at proximity of the pole.	The vehicle stops a few centimetres before the pole (due to the offset aspect as per face forward test).
Turning the 180 degree angle (Follow Path)	The system correctly rotates at the desired angle, stopping once at the adjacent motion path.	A small angular offset was identified in certain tests. The distance of surrounding object at proximity to the pole demonstrated to have an impact on when the system would stop motion.

Table 73 – Motion tests results

Overall it was evident from the closed loop motion tests that no modifications should be made to the working area setup after the vehicle has initiated its motion, also it was identified that objects at proximity of the detected poles did impact the system's performance and causing it to fail at certain stages. Where all of the previously conditions were met, the system performed as expected having the only issue of being at certain stages a few degrees offset.

8 Conclusions

It is concluded from this report that it is possible to build a low cost automated lawn mower with relatively better performance than the existing technologies. It is evident from this report that the sensor plays an important role in all functions for closed loop control of the system, having identified that making a comparison in code of the values obtained from actual readings with those estimated through calculations in order to accomplish the task desired. The servo motor mentioned in the report was also described as being as equally important as the sensor for two main reasons, one being the advantage of enabling one sensor to perform similarly to a system with multiple fixed sensors positioned around the vehicle and the second important aspect of such component is that is allows for angle determination of objects determined and calculations are then possible.

The three pole detection method described in this report is demonstrated to be both more economical solution and simple. The concept described shows how with no need of further electronic components to be positioned around the garden area, it is possible to determine the robots' position and actions to be performed based on the location in the garden.

The microcontroller was the key to enabling the project to come together, meaning that with the functions specified and programmed into it, the robot is able to accomplish the specified goal. Important aspects discussed about the microcontroller include the definition of the port type and the clock speed. Both aspects enable for the microcontroller to successfully communicate with the ultrasonic transducer and obtain the relevant data required for the other operations to be performed, it also enables for actions to be performed by outputting the relevant values to the devices that interface is ports. An important configuration required for accomplishment of this project, as discussed in the present report, is of pulse width modulation, which the DC motor and servo motor require for its control.

Limitations are present in this system, which include the shape and size of the garden and environmental (temperature, weather, etc.) conditions. The selection of the relevant sensor did take into consideration these aspects and a workaround was established.

The hardware and external system setup is also an aspect analysed due to the fact that proper arrangement of the components will enable for best system performance. The external setup focused mainly on the positioning of the sensor and servo motor, the position of the blades and the wheels. It was established that positioning the sensor and servo centrally on the vehicle is important in ensuring that the data captured is within the range of the calculated data, positioning of the cutting blades was established as being positioned at opposite ends of the vehicle in order to accomplish good cut of grass, and having two wheels that drive the vehicle was established as being the most effective way of enabling motion in the garden and for the 180 degree rotation required at the end of each motion path.

Overall from this project it is demonstrated that an economic automated lawn mower can be produced using a microcontroller, motors and sensors.

9 Further work

The work described in the present report focuses mainly on the vehicle motion control. From the work demonstrated for this device, the actual implementation has enabled identification of certain issues in programming that could be improved on. The improvement would be with regards to the current algorithm which currently is limited to a garden area, which would be rendered impractical in the real world.

An aspect that was not mentioned in the present report is the user interface and a better external appearance. The user interface would need to be intuitive and easy to use by all, and the external appearance should be pleasing, a graphic designer may be required for this part.

The porting of the existing project to proprietary hardware would be a step following, when the unit has passed all relevant tests and ready for mass production. As the code is written in C it means that most of the main program will still work when ported, depending on the chip architecture.

10 References

- Friendly Robotics (1995), About Robomow. [ONLINE] Available at: http://www.robomow.co.uk/about/ [Accessed: 12 December 2010]
- Mowermagic, Robomow Robotic Automatic Lawnmower. [ONLINE] Available at: http://www.mowermagic.co.uk/acatalog/RL500_Robotic_Mower.html [Accessed: 15 January 2011]
- LawnBott, 2011. *LawnBott LB3210 LB2110 manual*. [ONLINE] Available at: http://www.lawnbotts.com/pdf/LawnBott_LB3210_LB2110_manual.pdf [Accessed 27 March 2011]
- Bailey, W. 2008. *BEng 3 PROJECTS 2010-11 (DRAFT).* [ONLINE] Available at: http://homepages.uel.ac.uk/W.N.Bailey/BEng3_Projects/Images/GardenSweeper.png. [Accessed 20 October 2010]
- Georges T., 1999. *The American Lawn*. 1st ed. Princeton Architectural Press.
- Friendly Robotics (2011), *Robomow – How does it work*. [ONLINE] Available at: http://www.robomow.com/robomow/howitworks/ [Accessed: 20 March 2011]
- Lawn Mower Reviews. 2011. *Automatic Robot Lawn Mowers - Price Comparison*. [ONLINE] Available at: http://www.lawnmowersuk.org/compare/automatic/. [Accessed 27 February 2011].
- Mowers-Online. 2011. *Robomow Replacement Blades Set*. [ONLINE] Available at: http://www.mowers-online.co.uk/Lawn-Care---Garden-Care/Lawnmower-Spares/Lawnmower-Parts/Lawnmower-Blade/Robomow-Blades/Robomow-Replacement-Blades-Set.html. [Accessed 14 April 2011].
- Jacob F., 2010. *Handbook of Modern Sensors: Physics, Designs, and Applications*. 4th ed. Springer.
- Society of Robots. 2011. *How to Build a Robot Tutorial - Society of Robots*. [ONLINE] Available at: http://www.societyofrobots.com/sensors_sharpirrange.shtml. [Accessed 28 February 2011]
- Rapid Electronics Ltd. 2011. *Rapid Electronics - Electronic Components*. [ONLINE] Available at: http://www.rapidonline.com/Electronic-Components/Sensors/Proximity-Optical/Distance-measuring-sensor-digital-output/81725/kw/sharp+infrared+rangefinde. [Accessed 28 February 2011]
- Paul C., 1995. *An Introduction to Measuration and Calibration*. 1st ed. Industrial Press, Inc.
- Ahmed E., 2002. *Introduction to GPS: The Global Positioning System*. 1st ed. Artech House.
- Everett H.R., 1995. *Sensors for Mobile Robots Theory and Application*. A K Peters/CRC Press.
- Google. 2011. *Google Product Search*. [ONLINE] Available at: http://www.google.co.uk/prdhp?hl=en&tab=wf. [Accessed 02 March 2011]
- Arduino, Arduino – Homepage. [online] Available at: http://www.arduino.cc/ [Accessed: 05 August 10]
- Wharton, P. 2006. The Cosine Rule. *Essentials of Edexcel Gcse Maths.* Lonsdale: Lonsdale Revision Guides.
- SRF05 Technical Documentation. 2011. *SRF05 Technical Documentation*. [ONLINE] Available at: http://www.robot-electronics.co.uk/htm/srf05tech.htm. [Accessed 10 February 2011]
- Introduction to Pulse Width Modulation (PWM) | Netrino. 2011. *Introduction to Pulse Width Modulation (PWM) | Netrino*. [ONLINE] Available at: http://www.netrino.com/Embedded-Systems/How-To/PWM-Pulse-Width-Modulation. [Accessed 10 February 2011].

- Simon Monk, 2010. *30 Arduino Projects for the Evil Genius*. 1st ed. McGraw-Hill/TAB Electronics.
- Robotzone. LLC 2008. *How do servos work? Digital image*. [ONLINE] Available at: http://www.servocity.com/html/how_do_servos_work_.html. [Accessed 17 February 2011].
- Larry. 2009. Arduino + Processing – Make a Radar Screen to Visualise Sensor Data from SRF-05 – Part 1: Setting up the Circuit and Outputting Values. [ONLINE] Available at: http://luckylarry.co.uk/arduino-projects/arduino-processing-make-a-radar-screen-to-visualise-sensor-data-from-srf-05-part-1-setting-up-the-circuit-and-outputting-values/. [Accessed 17 February 2011].
-

11 Bibliography

The following can provide better understanding in certain aspects that were not described in much detail in the present report.

- Husqvarna. 2011. *Owner's Manuals & Parts Lookup*. [ONLINE] Available at: http://www.husqvarna.com/us/homeowner/support/download-manuals/?query=AUTOMOWER+220+AC&types=O. [Accessed 27 March 2011].
- Onwubolu G., 2005. *Mechatronics: Principles and Applications*. Butterworth-Heinemann.

12 Appendix I – The Source Code

Appendix II – The Arduino IDE

13 Appendix III – SRF05 Datasheet

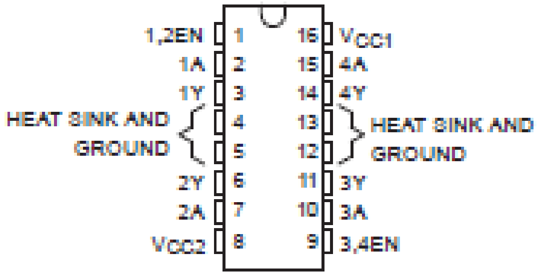
14 Appendix IV – H-Bridge (SN754410NE) Datasheet

SN754410
QUADRUPLE HALF-H DRIVER

SLRS007B – NOVEMBER 1986 – REVISED NOVEMBER 1995

- **1-A Output-Current Capability Per Driver**
- **Applications Include Half-H and Full-H Solenoid Drivers and Motor Drivers**
- **Designed for Positive-Supply Applications**
- **Wide Supply-Voltage Range of 4.5 V to 36 V**
- **TTL- and CMOS-Compatible High-Impedance Diode-Clamped Inputs**
- **Separate Input-Logic Supply**
- **Thermal Shutdown**
- **Internal ESD Protection**
- **Input Hysteresis Improves Noise Immunity**
- **3-State Outputs**
- **Minimized Power Dissipation**
- **Sink/Source Interlock Circuitry Prevents Simultaneous Conduction**
- **No Output Glitch During Power Up or Power Down**
- **Improved Functional Replacement for the SGS L293**

NE PACKAGE
(TOP VIEW)



FUNCTION TABLE
(each driver)

INPUTS†		OUTPUT
A	EN	Y
H	H	H
L	H	L
X	L	Z

H = high-level, L = low-level
X = irrelevant
Z = high-impedance (off)
† In the thermal shutdown mode, the output is in a high-impedance state regardless of the input levels.

description

The SN754410 is a quadruple high-current half-H driver designed to provide bidirectional drive currents up to 1 A at voltages from 4.5 V to 36 V. The device is designed to drive inductive loads such as relays, solenoids, dc and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications.

All inputs are compatible with TTL-and low-level CMOS logic. Each output (Y) is a complete totem-pole driver with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN. When an enable input is high, the associated drivers are enabled and their outputs become active and in phase with their inputs. When the enable input is low, those drivers are disabled and their outputs are off and in a high-impedance state. With the proper data inputs, each pair of drivers form a full-H (or bridge) reversible drive suitable for solenoid or motor applications.

A separate supply voltage (V_{CC1}) is provided for the logic input circuits to minimize device power dissipation. Supply voltage V_{CC2} is used for the output circuits.

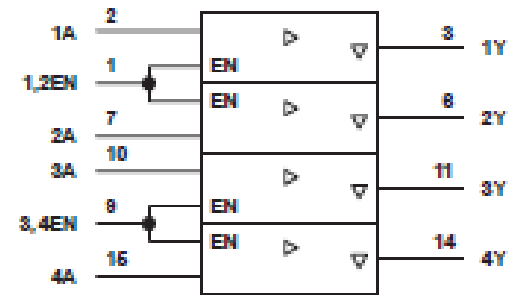
The SN754410 is designed for operation from −40°C to 85°C.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

SN754410
QUADRUPLE HALF-H DRIVER

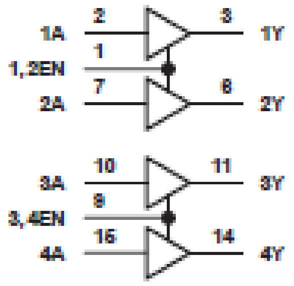
SLRS007B – NOVEMBER 1986 – REVISED NOVEMBER 1995

logic symbol†

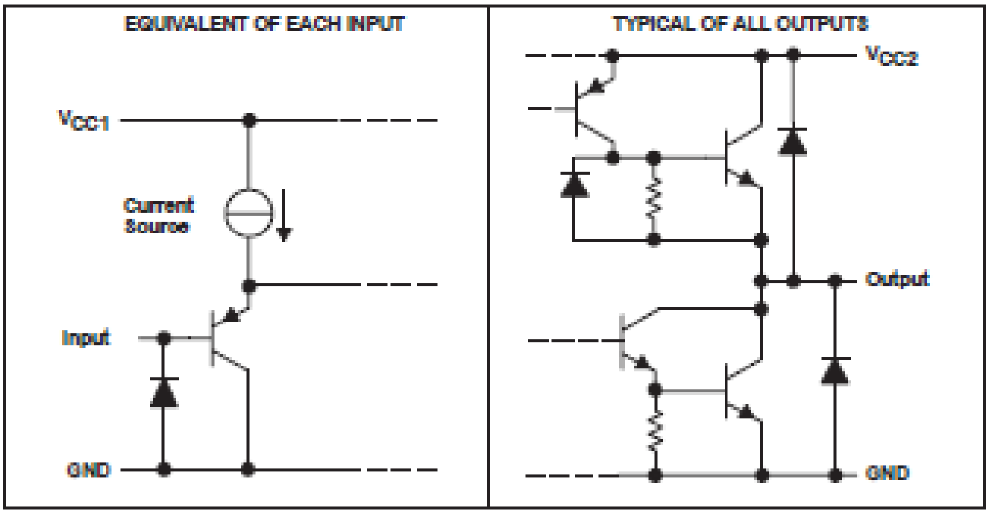


† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

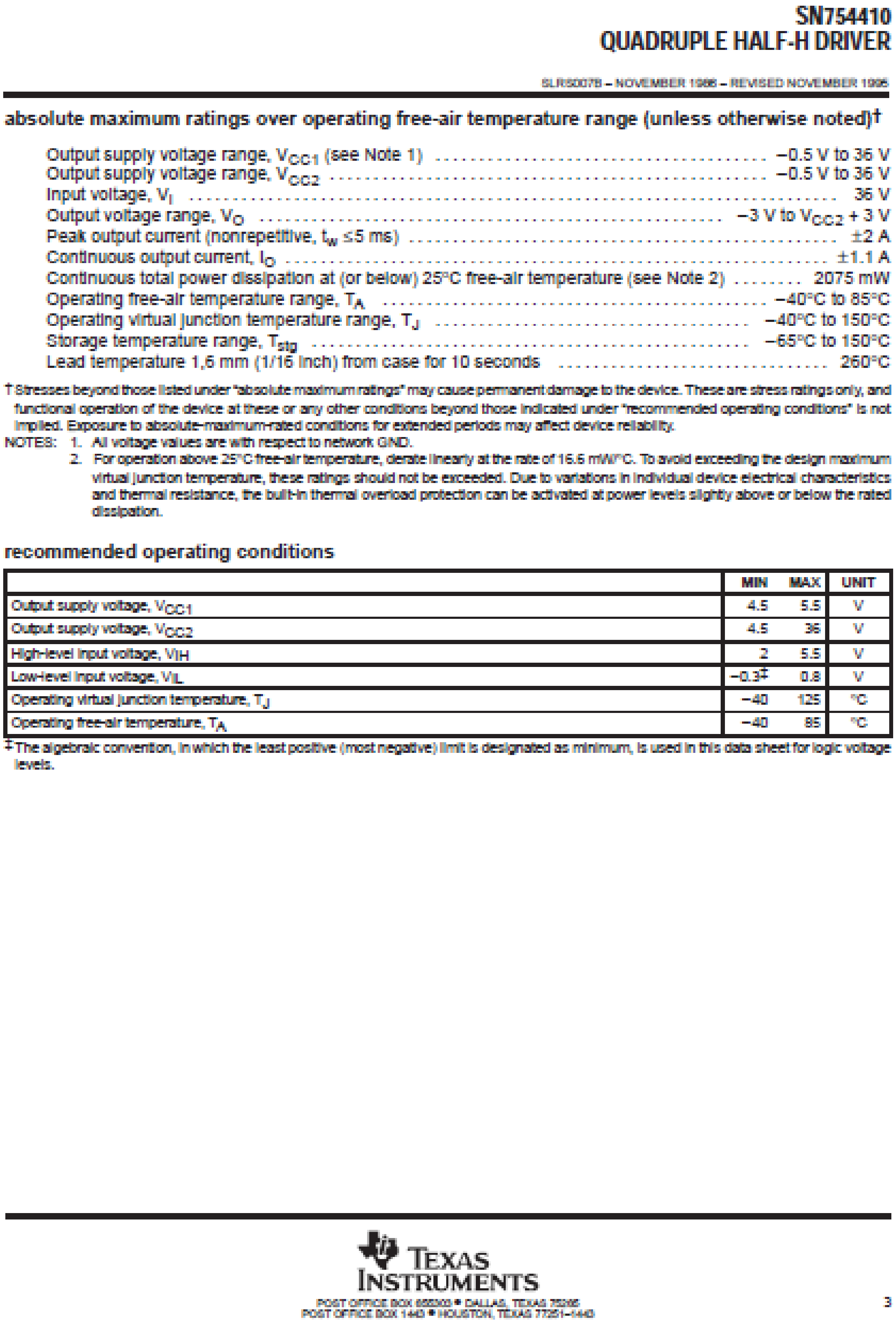
logic diagram



schematics of inputs and outputs



TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)[†]

Output supply voltage range, V_{CC1} (see Note 1)	−0.5 V to 36 V
Output supply voltage range, V_{CC2}	−0.5 V to 36 V
Input voltage, V_I	36 V
Output voltage range, V_O	−3 V to V_{CC2} + 3 V
Peak output current (nonrepetitive, $t_w \leq 5$ ms)	±2 A
Continuous output current, I_O	±1.1 A
Continuous total power dissipation at (or below) 25°C free-air temperature (see Note 2)	2075 mW
Operating free-air temperature range, T_A	−40°C to 85°C
Operating virtual junction temperature range, T_J	−40°C to 150°C
Storage temperature range, T_{stg}	−65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds	260°C

[†] Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

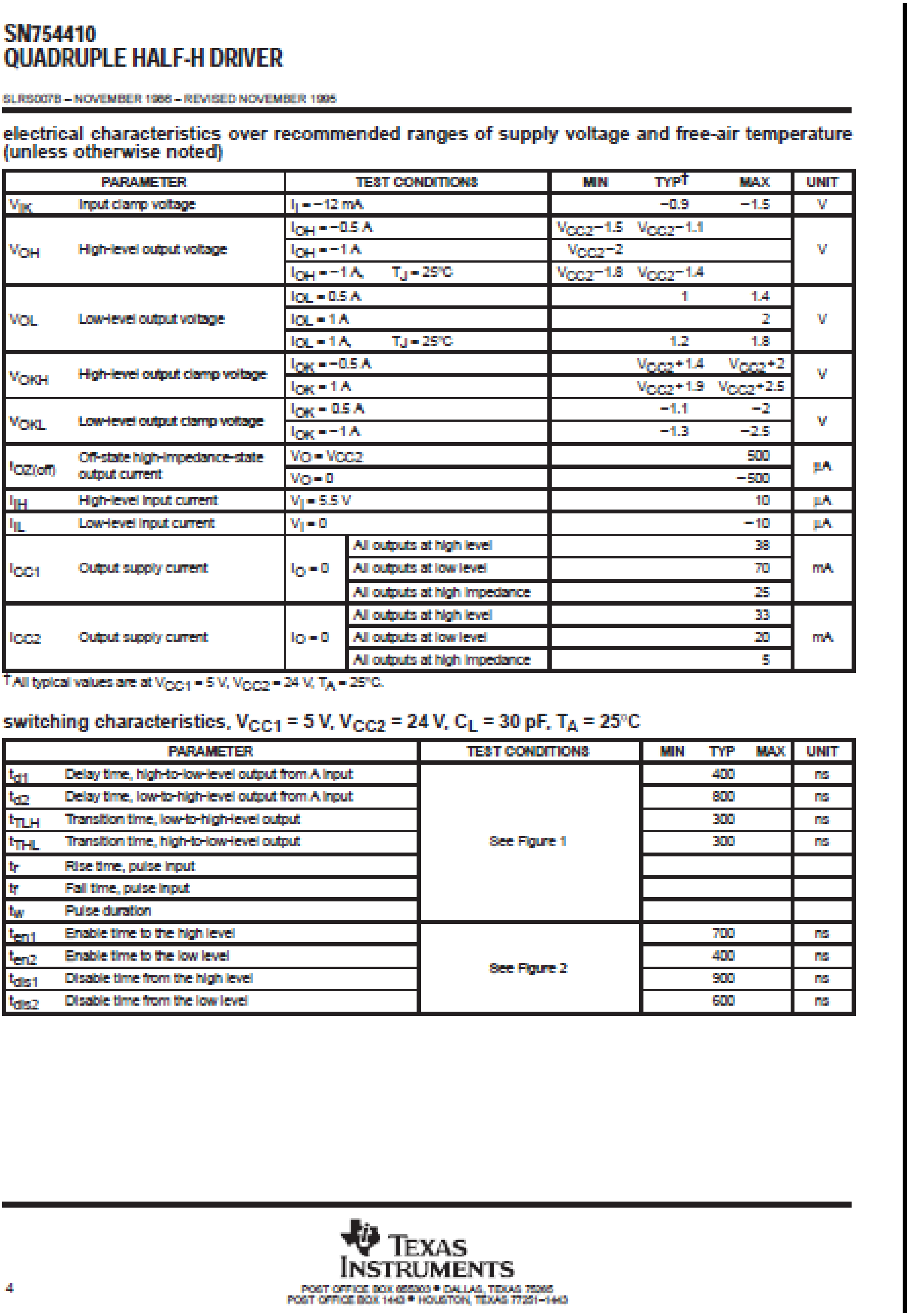
NOTES: 1. All voltage values are with respect to network GND.

2. For operation above 25°C free-air temperature, derate linearly at the rate of 16.6 mW/°C. To avoid exceeding the design maximum virtual junction temperature, these ratings should not be exceeded. Due to variations in individual device electrical characteristics and thermal resistance, the built-in thermal overload protection can be activated at power levels slightly above or below the rated dissipation.

recommended operating conditions

	MIN	MAX	UNIT
Output supply voltage, V_{CC1}	4.5	5.5	V
Output supply voltage, V_{CC2}	4.5	36	V
High-level input voltage, V_{IH}	2	5.5	V
Low-level input voltage, V_{IL}	−0.3[‡]	0.8	V
Operating virtual junction temperature, T_J	−40	125	°C
Operating free-air temperature, T_A	−40	85	°C

[‡] The algebraic convention, in which the least positive (most negative) limit is designated as minimum, is used in this data sheet for logic voltage levels.



electrical characteristics over recommended ranges of supply voltage and free-air temperature (unless otherwise noted)

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{IK}	Input clamp voltage	$I_I = -12$ mA			−0.9	−1.5	V
V_{OH}	High-level output voltage	$I_{OH} = -0.5$ A		$V_{CC2}-1.5$	$V_{CC2}-1.1$		V
		$I_{OH} = -1$ A		$V_{CC2}-2$			
		$I_{OH} = -1$ A, $T_J = 25°C$		$V_{CC2}-1.8$	$V_{CC2}-1.4$		
V_{OL}	Low-level output voltage	$I_{OL} = 0.5$ A			1	1.4	V
		$I_{OL} = 1$ A				2	
		$I_{OL} = 1$ A, $T_J = 25°C$			1.2	1.8	
V_{OKH}	High-level output clamp voltage	$I_{OK} = -0.5$ A			$V_{CC2}+1.4$	$V_{CC2}+2$	V
		$I_{OK} = 1$ A			$V_{CC2}+1.9$	$V_{CC2}+2.5$	
V_{OKL}	Low-level output clamp voltage	$I_{OK} = 0.5$ A			−1.1	−2	V
		$I_{OK} = -1$ A			−1.3	−2.5	
$I_{OZ(off)}$	Off-state high-impedance-state output current	$V_O = V_{CC2}$				500	µA
		$V_O = 0$				−500	
I_{IH}	High-level input current	$V_I = 5.5$ V				10	µA
I_{IL}	Low-level input current	$V_I = 0$				−10	µA
I_{CC1}	Output supply current	$I_O = 0$	All outputs at high level			38	mA
			All outputs at low level			70	
			All outputs at high impedance			25	
I_{CC2}	Output supply current	$I_O = 0$	All outputs at high level			33	mA
			All outputs at low level			20	
			All outputs at high impedance			5	

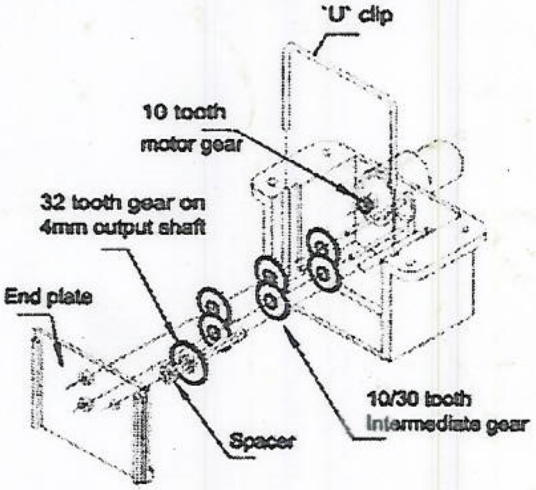
† All typical values are at $V_{CC1} = 5$ V, $V_{CC2} = 24$ V, $T_A = 25°C$.

switching characteristics, $V_{CC1} = 5$ V, $V_{CC2} = 24$ V, $C_L = 30$ pF, $T_A = 25°C$

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{d1}	Delay time, high-to-low-level output from A input	See Figure 1		400		ns
t_{d2}	Delay time, low-to-high-level output from A input			800		ns
t_{TLH}	Transition time, low-to-high-level output			300		ns
t_{THL}	Transition time, high-to-low-level output			300		ns
t_r	Rise time, pulse input					
t_f	Fall time, pulse input					
t_w	Pulse duration					
t_{en1}	Enable time to the high level	See Figure 2		700		ns
t_{en2}	Enable time to the low level			400		ns
t_{dis1}	Disable time from the high level			900		ns
t_{dis2}	Disable time from the low level			600		ns

15 Appendix V – Clearbox Motor (Datasheet)

CLEARBOX MOTOR



The clearbox motor has been designed especially for use in schools CTD and science project work.

It has a virtually unbreakable polycarbonate gearcase and tough nylon gears. The output shaft is 4 mm diameter and is suitable for Plawcotech, Meccano and Fischertechnik wheels, gears and pullys to be fitted directly to it. A Plawcotech brass adapter boss may also be fitted to this output shaft to enable wooden and plastic wheels, arms, levers and LEGO beams to be attached.

The motor is a standard MJR3L type and the simple press fit allows replacement motors to be fitted should one become damaged.

The motor will operate at voltages of between 3Vdc and 6Vdc. At 6 Vdc the motor will run free at about 6000 RPM. Many factors such as load,volts drop from the supply or wiring, or a lack of available current can affect this speed, so calculations of final drive speed can only be approximate.

The gearbox is a multi-stage spur gear type. Access to the gears is obtained by removing the metal 'U' clip from its groove and withdrawing the end plate and final drive shaft. When changing the gear ratio, spacers are provided to take up the slack on the Drive shaft. A thin spacer is already fitted to the assembled Clearbox and three thick spacers are provided loose in the packet.

The thickness of one thick spacer is equal to two thin spacers and one thin spacer is equal to one intermediate gear.

Changing the gear ratio

After removing the 'U' clip slide the End Plate off the Output shaft and away from the body of the gearbox. Withdraw the output Shaft and as many 10/30 intermediate Gears as necessary to achieve the required ratio.

Reassemble in the reverse order remembering to add spare spacers as previously outlined. For example to remove one intermediate gear you must replace the thin spacer with a thick spacer. To remove two gears one thick and one thin spacer is required and so on. With all the gears fitted the speed of the output shaft will be at its slowest. By removing gears the output speed will increase and the torque will decrease.

Calculation

The motor will run freely (without load) when supplied with 6 volts at approximately 6000 revs.

The intermediate gear has a reduction of 30 teeth to 10 teeth, i.e. a 3 : 1 ratio.

The output shaft gear has 32 teeth, i.e a 3.2 : 1 ratio.

To calculate the ratio of the whole gear train, the following formula should be used.

Ratio = (3n) x 3.2

Where n= the number of intermediate gears.

For example with 2 intermediate gears fitted

Ratio = (3^2) x 3.2
= 9 x 3.2
= 28.8 : 1

A motor speed of 6000 rpm would give a theoretical speed of 6000/28.8 rpm.

16 Appendix VI – Ideal Angles Table

Specs.		**Dimension in (m)**	**14 elements**
	Max. Distance	4	
	Vehicle Horizontal	0.14	
	Vehicle Vertical	0.14	
	Centre	0.07	
			Export to array1
			Servo Angle

Coordinates (x, y) in cm		**Sine (Deg) rounded up**
0.1	0.0	48
0.1	0.3	13
0.1	0.6	7
0.1	0.9	4
0.1	1.2	3
0.1	1.5	3
0.1	1.8	2
0.1	2.1	2
0.1	2.4	2
0.1	2.7	1
0.1	3.0	1
0.1	3.3	1
0.1	3.6	1
0.1	3.9	1
0.2	3.9	3
0.2	3.6	3
0.2	3.3	4
0.2	3.0	4
0.2	2.7	4
0.2	2.4	5
0.2	2.1	6
0.2	1.8	7
0.2	1.5	8
0.2	1.2	10
0.2	0.9	13
0.2	0.6	19
0.2	0.3	31
0.2	0.0	48
0.4	0.0	48
0.4	0.3	39
0.4	0.6	28
0.4	0.9	20
0.4	1.2	16

0.4	1.5	13
0.4	1.8	11
0.4	2.1	9
0.4	2.4	8
0.4	2.7	7
0.4	3.0	7
0.4	3.3	6
0.4	3.6	6
0.4	3.9	5
0.5	3.9	7
0.5	3.6	8
0.5	3.3	8
0.5	3.0	9
0.5	2.7	10
0.5	2.4	11
0.5	2.1	13
0.5	1.8	15
0.5	1.5	18
0.5	1.2	21
0.5	0.9	26
0.5	0.6	34
0.5	0.3	43
0.5	0.0	48
0.6	0.0	48
0.6	0.3	45
0.6	0.6	38
0.6	0.9	31
0.6	1.2	26
0.6	1.5	22
0.6	1.8	19
0.6	2.1	16
0.6	2.4	14
0.6	2.7	13
0.6	3.0	12
0.6	3.3	11
0.6	3.6	10
0.6	3.9	9
0.8	3.9	11
0.8	3.6	12
0.8	3.3	13
0.8	3.0	14
0.8	2.7	16
0.8	2.4	17

0.8	2.1	19
0.8	1.8	22
0.8	1.5	25
0.8	1.2	29
0.8	0.9	35
0.8	0.6	41
0.8	0.3	46
0.8	0.0	48
0.9	0.0	48
0.9	0.3	47
0.9	0.6	42
0.9	0.9	37
0.9	1.2	33
0.9	1.5	28
0.9	1.8	25
0.9	2.1	22
0.9	2.4	20
0.9	2.7	18
0.9	3.0	16
0.9	3.3	15
0.9	3.6	14
0.9	3.9	13
1.1	3.9	15
1.1	3.6	16
1.1	3.3	17
1.1	3.0	19
1.1	2.7	20
1.1	2.4	22
1.1	2.1	25
1.1	1.8	28
1.1	1.5	31
1.1	1.2	35
1.1	0.9	39
1.1	0.6	44
1.1	0.3	47
1.1	0.0	48
1.2	0.0	48
1.2	0.3	47
1.2	0.6	45
1.2	0.9	41
1.2	1.2	37
1.2	1.5	33
1.2	1.8	30

1.2	2.1	27
1.2	2.4	25
1.2	2.7	22
1.2	3.0	21
1.2	3.3	19
1.2	3.6	18
1.2	3.9	16
1.3	3.9	18
1.3	3.6	19
1.3	3.3	21
1.3	3.0	23
1.3	2.7	25
1.3	2.4	27
1.3	2.1	29
1.3	1.8	32
1.3	1.5	35
1.3	1.2	39
1.3	0.9	42
1.3	0.6	45
1.3	0.3	47
1.3	0.0	48
1.5	0.0	48
1.5	0.3	48
1.5	0.6	46
1.5	0.9	43
1.5	1.2	40
1.5	1.5	37
1.5	1.8	34
1.5	2.1	31
1.5	2.4	29
1.5	2.7	26
1.5	3.0	24
1.5	3.3	23
1.5	3.6	21
1.5	3.9	20
1.6	3.9	21
1.6	3.6	23
1.6	3.3	24
1.6	3.0	26
1.6	2.7	28
1.6	2.4	30
1.6	2.1	33
1.6	1.8	35

1.6	1.5	38
1.6	1.2	41
1.6	0.9	44
1.6	0.6	46
1.6	0.3	48
1.6	0.0	48
1.8	0.0	48
1.8	0.3	48
1.8	0.6	46
1.8	0.9	44
1.8	1.2	42
1.8	1.5	39
1.8	1.8	37
1.8	2.1	34
1.8	2.4	32
1.8	2.7	30
1.8	3.0	28
1.8	3.3	26
1.8	3.6	24
1.8	3.9	23
1.9	3.9	24
1.9	3.6	26
1.9	3.3	27
1.9	3.0	29
1.9	2.7	31
1.9	2.4	33
1.9	2.1	36
1.9	1.8	38
1.9	1.5	40
1.9	1.2	43
1.9	0.9	45
1.9	0.6	47
1.9	0.3	48
1.9	0.0	48
2.0	0.0	48
2.0	0.3	48
2.0	0.6	47
2.0	0.9	45
2.0	1.2	43
2.0	1.5	41
2.0	1.8	39
2.0	2.1	37
2.0	2.4	34

2.0	2.7	32
2.0	3.0	30
2.0	3.3	29
2.0	3.6	27
2.0	3.9	26
2.2	3.9	27
2.2	3.6	28
2.2	3.3	30
2.2	3.0	32
2.2	2.7	34
2.2	2.4	36
2.2	2.1	38
2.2	1.8	40
2.2	1.5	42
2.2	1.2	44
2.2	0.9	46
2.2	0.6	47
2.2	0.3	48
2.2	0.0	48
2.3	0.0	48
2.3	0.3	48
2.3	0.6	47
2.3	0.9	46
2.3	1.2	44
2.3	1.5	43
2.3	1.8	41
2.3	2.1	39
2.3	2.4	37
2.3	2.7	35
2.3	3.0	33
2.3	3.3	31
2.3	3.6	29
2.3	3.9	28
2.5	3.9	29
2.5	3.6	31
2.5	3.3	32
2.5	3.0	34
2.5	2.7	36
2.5	2.4	38
2.5	2.1	39
2.5	1.8	41
2.5	1.5	43
2.5	1.2	45

2.5	0.9	46
2.5	0.6	47
2.5	0.3	48
2.5	0.0	48
2.6	0.0	48
2.6	0.3	48
2.6	0.6	47
2.6	0.9	46
2.6	1.2	45
2.6	1.5	44
2.6	1.8	42
2.6	2.1	40
2.6	2.4	38
2.6	2.7	37
2.6	3.0	35
2.6	3.3	33
2.6	3.6	32
2.6	3.9	30
2.7	3.9	31
2.7	3.6	33
2.7	3.3	34
2.7	3.0	36
2.7	2.7	37
2.7	2.4	39
2.7	2.1	41
2.7	1.8	42
2.7	1.5	44
2.7	1.2	45
2.7	0.9	47
2.7	0.6	47
2.7	0.3	48
2.7	0.0	48
2.9	0.0	48
2.9	0.3	48
2.9	0.6	48
2.9	0.9	47
2.9	1.2	46
2.9	1.5	44
2.9	1.8	43
2.9	2.1	41
2.9	2.4	40
2.9	2.7	38
2.9	3.0	37

2.9	3.3	35
2.9	3.6	33
2.9	3.9	32
3.0	3.9	33
3.0	3.6	34
3.0	3.3	36
3.0	3.0	37
3.0	2.7	39
3.0	2.4	40
3.0	2.1	42
3.0	1.8	43
3.0	1.5	45
3.0	1.2	46
3.0	0.9	47
3.0	0.6	48
3.0	0.3	48
3.0	0.0	48
3.2	0.0	48
3.2	0.3	48
3.2	0.6	48
3.2	0.9	47
3.2	1.2	46
3.2	1.5	45
3.2	1.8	44
3.2	2.1	42
3.2	2.4	41
3.2	2.7	39
3.2	3.0	38
3.2	3.3	36
3.2	3.6	35
3.2	3.9	34
3.3	3.9	34
3.3	3.6	36
3.3	3.3	37
3.3	3.0	39
3.3	2.7	40
3.3	2.4	41
3.3	2.1	43
3.3	1.8	44
3.3	1.5	45
3.3	1.2	46
3.3	0.9	47
3.3	0.6	48

3.3	0.3	48
3.3	0.0	48
3.4	0.0	48
3.4	0.3	48
3.4	0.6	48
3.4	0.9	47
3.4	1.2	46
3.4	1.5	45
3.4	1.8	44
3.4	2.1	43
3.4	2.4	42
3.4	2.7	41
3.4	3.0	39
3.4	3.3	38
3.4	3.6	36
3.4	3.9	35
3.6	3.9	36
3.6	3.6	37
3.6	3.3	38
3.6	3.0	40
3.6	2.7	41
3.6	2.4	42
3.6	2.1	43
3.6	1.8	45
3.6	1.5	46
3.6	1.2	47
3.6	0.9	47
3.6	0.6	48
3.6	0.3	48
3.6	0.0	48
3.7	0.0	48
3.7	0.3	48
3.7	0.6	48
3.7	0.9	47
3.7	1.2	47
3.7	1.5	46
3.7	1.8	45
3.7	2.1	44
3.7	2.4	43
3.7	2.7	41
3.7	3.0	40
3.7	3.3	39
3.7	3.6	38

3.7	3.9	36
3.9	3.9	37
3.9	3.6	38
3.9	3.3	39
3.9	3.0	41
3.9	2.7	42
3.9	2.4	43
3.9	2.1	44
3.9	1.8	45
3.9	1.5	46
3.9	1.2	47
3.9	0.9	47
3.9	0.6	48
3.9	0.3	48
3.9	0.0	48
4.0	0.0	48
4.0	0.3	48
4.0	0.6	48
4.0	0.9	47
4.0	1.2	47
4.0	1.5	46
4.0	1.8	45
4.0	2.1	44
4.0	2.4	43
4.0	2.7	42
4.0	3.0	41
4.0	3.3	40
4.0	3.6	39
4.0	3.9	38

17 Appendix VII – Filtered Angles Data Table

Coordinates (x, y) in cm		Hypotenuse	Sine (Deg) rounded up	Notes
0.1	0.0	0.1	48	Starting pt
0.1	0.3	0.3	13	
0.1	0.6	0.6	7	
0.1	2.4	2.4	2	Mid-pt
0.1	3.9	3.9	1	End-pt
0.2	3.9	3.9	3	
0.2	2.4	2.4	5	
0.2	0.6	0.6	19	
0.2	0.3	0.4	31	
0.2	0.0	0.2	48	
0.4	0.0	0.4	48	
0.4	0.3	0.5	39	
0.4	0.6	0.7	28	
0.4	2.4	2.4	8	
0.4	3.9	3.9	5	
0.5	3.9	3.9	7	
0.5	2.4	2.4	11	
0.5	0.6	0.8	34	
0.5	0.3	0.6	43	
0.5	0.0	0.5	48	
0.6	0.0	0.6	48	
0.6	0.3	0.7	45	
0.6	0.6	0.9	38	
0.6	2.4	2.5	14	
0.6	3.9	4.0	9	
0.8	3.9	4.0	11	
0.8	2.4	2.5	17	
0.8	0.6	1.0	41	
0.8	0.3	0.8	46	
0.8	0.0	0.8	48	
0.9	0.0	0.9	48	
0.9	0.3	1.0	47	
0.9	0.6	1.1	42	
0.9	2.4	2.6	20	
0.9	3.9	4.0	13	
1.1	3.9	4.0	15	
1.1	2.4	2.6	22	
1.1	0.6	1.2	44	
1.1	0.3	1.1	47	
1.1	0.0	1.1	48	
1.2	0.0	1.2	48	
1.2	0.3	1.2	47	
1.2	0.6	1.3	45	
1.2	2.4	2.7	25	

1.2	3.9	4.1	16	
1.3	3.9	4.1	18	
1.3	2.4	2.7	27	
1.3	0.6	1.5	45	
1.3	0.3	1.4	47	
1.3	0.0	1.3	48	
1.5	0.0	1.5	48	
1.5	0.3	1.5	48	
1.5	0.6	1.6	46	
1.5	2.4	2.8	29	
1.5	3.9	4.2	20	
1.6	3.9	4.2	21	
1.6	2.4	2.9	30	
1.6	0.6	1.7	46	
1.6	0.3	1.6	48	
1.6	0.0	1.6	48	
1.8	0.0	1.8	48	
1.8	0.3	1.8	48	
1.8	0.6	1.9	46	
1.8	2.4	3.0	32	
1.8	3.9	4.3	23	
1.9	3.9	4.3	24	
1.9	2.4	3.1	33	
1.9	0.6	2.0	47	
1.9	0.3	1.9	48	
1.9	0.0	1.9	48	
2.0	0.0	2.0	48	
2.0	0.3	2.1	48	
2.0	0.6	2.1	47	
2.0	2.4	3.1	34	
2.0	3.9	4.4	26	
2.2	3.9	4.5	27	
2.2	2.4	3.2	36	
2.2	0.6	2.3	47	
2.2	0.3	2.2	48	
2.2	0.0	2.2	48	
2.3	0.0	2.3	48	
2.3	0.3	2.3	48	
2.3	0.6	2.4	47	
2.3	2.4	3.3	37	
2.3	3.9	4.5	28	
2.5	3.9	4.6	29	
2.5	2.4	3.4	38	
2.5	0.6	2.5	47	
2.5	0.3	2.5	48	

2.5	0.0	2.5	48	
2.6	0.0	2.6	48	
2.6	0.3	2.6	48	
2.6	0.6	2.7	47	
2.6	2.4	3.5	38	
2.6	3.9	4.7	30	
2.7	3.9	4.8	31	
2.7	2.4	3.6	39	
2.7	0.6	2.8	47	
2.7	0.3	2.7	48	
2.7	0.0	2.7	48	
2.9	0.0	2.9	48	
2.9	0.3	2.9	48	
2.9	0.6	2.9	48	
2.9	2.4	3.7	40	
2.9	3.9	4.8	32	
3.0	3.9	4.9	33	
3.0	2.4	3.8	40	
3.0	0.6	3.1	48	
3.0	0.3	3.0	48	
3.0	0.0	3.0	48	
3.2	0.0	3.2	48	
3.2	0.3	3.2	48	
3.2	0.6	3.2	48	
3.2	2.4	4.0	41	
3.2	3.9	5.0	34	
3.3	3.9	5.1	34	
3.3	2.4	4.1	41	
3.3	0.6	3.3	48	
3.3	0.3	3.3	48	
3.3	0.0	3.3	48	
3.4	0.0	3.4	48	
3.4	0.3	3.4	48	
3.4	0.6	3.5	48	
3.4	2.4	4.2	42	
3.4	3.9	5.2	35	
3.6	3.9	5.3	36	
3.6	2.4	4.3	42	
3.6	0.6	3.6	48	
3.6	0.3	3.6	48	
3.6	0.0	3.6	48	
3.7	0.0	3.7	48	
3.7	0.3	3.7	48	
3.7	0.6	3.8	48	
3.7	2.4	4.4	43	

3.7	3.9	5.4	36	
3.9	3.9	5.5	37	
3.9	2.4	4.5	43	
3.9	0.6	3.9	48	
3.9	0.3	3.9	48	
3.9	0.0	3.9	48	
4.0	0.0	4.0	48	
4.0	0.3	4.0	48	
4.0	0.6	4.0	48	
4.0	2.4	4.7	43	
4.0	3.9	5.6	38	

18 Appendix VIII – Test Data

Results for Sensor and Servo data test

Sensor	Servo Angle
0	103
1	105
2	99
3	79
4	79
5	79
6	79
7	79
8	79
9	78
10	78
11	78
12	78
13	78
14	78
15	78
16	78
17	78
18	78
19	78
20	78
21	78
22	78
23	79
24	79
25	80
26	81
27	90
28	104
29	103
30	103
31	104
32	105
33	104
34	105
35	104
36	102
37	103
38	103
39	103

40	103
41	104
42	99
43	104
44	105
45	106
46	105
47	105
48	105
49	105
50	104
51	105
52	105
53	105
54	105
55	106
56	106
57	111
58	117
59	112
60	111
61	104
62	104
63	103
64	103
65	103
66	103
67	103
68	103
69	103
70	103
71	103
72	103
73	103
74	103
75	103
76	103
77	103
78	103
79	102
80	103
81	102
82	102

83	103
84	102
85	102
86	102
87	103
88	102
89	102
90	102
91	102
92	102
93	102
94	103
95	102
96	103
97	103
98	103
99	103
100	103
101	103
102	103
103	103
104	103
105	103
106	103
107	103
108	103
109	103
110	104
111	104
112	104
113	105
114	105
115	105
116	113
117	106
118	117
119	124
120	116
121	123
122	123
123	123
124	123
125	116

126	123
127	116
128	123
129	116
130	123
131	116
132	99
133	91
134	98
135	91
136	91
137	91
138	91
139	91
140	91
141	90
142	91
143	91
144	90
145	91
146	91
147	91
148	91
149	91
150	91
151	91
152	91
153	91
154	90
155	91
156	91
157	90
158	91
159	90
160	90
161	90
162	90
163	90
164	90
165	90
166	90
167	90
168	90

169	90
170	90
171	90
172	89
173	90
174	90
175	90
176	90
177	90
178	91
179	91
180	91
180	91
179	91
178	90
177	90
176	90
175	90
174	90
173	90
172	90
171	90
170	90
169	90
168	90
167	90
166	90
165	90
164	90
163	90
162	90
161	90
160	90
159	90
158	90
157	90
156	91
155	91
154	90
153	91
152	91
151	91
150	91

149	91
148	91
147	91
146	91
145	91
144	91
143	91
142	91
141	90
140	91
139	91
138	90
137	91
136	91
135	91
134	91
133	91
132	92
131	111
130	117
129	117
128	116
127	123
126	116
125	123
124	116
123	123
122	123
121	127
120	123
119	127
118	124
117	117
116	106
115	109
114	105
113	105
112	104
111	104
110	104
109	104
108	104
107	103

106	103
105	103
104	103
103	103
102	103
101	103
100	103
99	103
98	103
97	103
96	103
95	103
94	103
93	103
92	103
91	102
90	103
89	103
88	102
87	103
86	103
85	103
84	103
83	103
82	103
81	102
80	103
79	103
78	102
77	103
76	103
75	103
74	103
73	103
72	103
71	103
70	103
69	103
68	103
67	103
66	103
65	103
64	104

63	104
62	104
61	105
60	108
59	105
58	118
57	115
56	106
55	106
54	106
53	105
52	105
51	105
50	105
49	104
48	105
47	105
46	105
45	106
44	105
43	105
42	105
41	104
40	104
39	105
38	99
37	105
36	105
35	104
34	104
33	104
32	104
31	103
30	103
29	103
28	98
27	102
26	86
25	85
24	80
23	79
22	79
21	79

20	79
19	78
18	78
17	78
16	78
15	78
14	78
13	78
12	78
11	78
10	78
9	79
8	79
7	79
6	79
5	79
4	79
3	79
2	79
1	79
0	79