AM 41 BORG – RECONFIGURABLE ROBOTS

Submitted by
Tan Han Yong
U006222B11

Department of
Mechanical Engineering

In partial fulfilment of the
requirements for the Degree of
Bachelor of Engineering
National University of Singapore

Session 2003/04
SUMMARY

The project objective of **Borg – Reconfigurable Robots** was to design, build and test a system of multiple robotic units (Borgs). These Borgs will first be put into an original configuration, which they must recognize and move together as an entire unit. The entire unit must also react to a light source and move in that direction. Then on encountering a narrow channel, the whole unit then has to reconfigure to form up behind the robot in the through path. They must then recognize this new formation and move together as one unit through this path. These required robot behaviors are described in detail in the report.

From the research papers studied during the literature survey, it could be gathered that the reconfigurable robotic research placed emphasis on the mechanical attachment between the modular units, some even having Infrared Red sensors to guide the module-to-module attachments. However, there were no sensors to detect environmental obstacles, hence the robotic structures lacked the ability to react to the environmental obstacles and reconfigure. Thus, the challenge of this project was to develop an algorithm that could allow the entire robotic structure to reconfigure autonomously when triggered by environmental obstacles. The approach to the project was segmented into 3 different phases namely design, build and test phases.

In the design phase, an algorithm was developed for the reconfiguration process. The approach was novel as it utilized a combination of Light Emitting Diodes (LEDs) and Light sensors on the Borgs to differentiate their positions in the structure. This was not attempted in the reconfigurable researches studied for this
project. This was achieved through a unique square design of the Borg, with identical sides for ease of reconfiguration. Despite the identical faces of the robots, when 2 Borgs face each other, the LEDs of one would be pointing at the light sensors of the other. This unique design allows a form of information transfer through lighting of the LEDs and sensing of light through light sensors.

The building phase saw the fabrication of the Borg from designs drafted using Solidworks CAD program. The Borg has 3 different layers. The special sensory layer that houses the LEDs and light and sonar sensors, the Brainstem layer and the servo layer with a unique wheelbase design developed specifically for this project. The wheelbase enabled the Borg to move straighter than previous robot designs by turning 2 wheels with 1 servo. Experiments were then carried out to calibrate the sensors so as to achieve the desired results in actual hardware testing.

The wireless LAN capabilities of the iPAQ palm pilot provided a platform to establish a User Datagram Protocol for wireless communication between robotic units. Much work was also done on Winsock programming specifications.

The reconfiguration algorithm was then tested in phases, from 2 units to an eventual 4 units. The testing phase started from the second week of December 2003 to the end of March 2004. During this time, the Borgs were given different initial configurations to test the robustness of the reconfiguration algorithm. The algorithm held up successfully for different initial configurations of up to 4 Borg units. The testing phase was concluded with the successful reconfiguration of 4 Borg units through a narrow channel with an example of the reconfiguration process shown in chapter 4 of this report.
ACKNOWLEDGEMENT

The author wishes to express sincere appreciation for the assistances given by the following persons in carrying out the work successfully:

Associate Professor Gerard Leng (National University of Singapore), supervisor of the project, for his patience, advice and guidance in the project.

Mr Cheng Chee Kong (Postgraduate in CoSy Lab) for his assistance and patience and guidance in the project.

Mr Ng Wee Kiat (Postgraduate in Cosy Lab) for his guidance in this project.

Mr Low Yee Leong (Postgraduate in CoSy Lab) for his guidance in this project.

Toh Kwang Chern and Vincent Tan (fellow undergraduates in CoSy Lab) for their help and support during the entire course of this project.

The technical staff of the Dynamics Laboratory namely: Ms Amy, Ms Priscilla, Mr Ahmad and Mr Cheng. For their equipment and technical support in the project.
# TABLE OF CONTENT

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUMMARY</td>
<td>I</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>III</td>
</tr>
<tr>
<td>TABLE OF CONTENT</td>
<td>IV</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>VII</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>IX</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>LITERATURE SURVEY</td>
<td>3</td>
</tr>
<tr>
<td>CHAPTER 1: RECONFIGURATION ALGORITHM DESIGN</td>
<td>6</td>
</tr>
<tr>
<td>1.1 First Configuration Phase</td>
<td>8</td>
</tr>
<tr>
<td>1.1.1 Strength Accounting Phase</td>
<td>8</td>
</tr>
<tr>
<td>1.1.2 Borg First Configuration Phase</td>
<td>9</td>
</tr>
<tr>
<td>1.2 Navigation Phase</td>
<td>10</td>
</tr>
<tr>
<td>1.2.1 Target Acquisition Process</td>
<td>11</td>
</tr>
<tr>
<td>1.2.2 Formation Control Process</td>
<td>12</td>
</tr>
<tr>
<td>1.2.3 Obstruction to Unit Process</td>
<td>14</td>
</tr>
<tr>
<td>1.3 Reconfiguration Phase</td>
<td>15</td>
</tr>
<tr>
<td>1.3.1 Center, Left and Right Accounting Process</td>
<td>15</td>
</tr>
<tr>
<td>1.3.2 The Actual Reconfiguration Process</td>
<td>19</td>
</tr>
<tr>
<td>1.3.3 New Formation Recognition Process</td>
<td>20</td>
</tr>
<tr>
<td>CHAPTER 2: TESTING GROUND SET UP, EXPERIMENTS AND WIRELESS NETWORKING</td>
<td>21</td>
</tr>
<tr>
<td>2.1 Layout of the Testing Ground</td>
<td>21</td>
</tr>
<tr>
<td>2.2 Experiments conducted on Devantech SRF08 Range Finder</td>
<td>21</td>
</tr>
<tr>
<td>2.2.1 Sonar Ranging Experiment</td>
<td>21</td>
</tr>
<tr>
<td>2.2.2 Light Intensity Experiment</td>
<td>23</td>
</tr>
</tbody>
</table>
# Table of Content

2.3 Wireless Local Area Network ........................................... 24

2.3.1 User Datagram Protocol ........................................... 25

2.3.2 Addressing and Internet Protocol .............................. 25

**CHAPTER 3: BORG DESIGN AND ARCHITECTURE** ............... 27

3.1 What is a Borg? ......................................................... 27

3.2 Borg Architecture and Design ...................................... 28

3.2.1 Sensory layer ...................................................... 29

3.2.2 Brainstem layer .................................................. 30

3.2.3 Servo layer ...................................................... 31

3.2.3.1 Lifting Servo Part ........................................... 31

3.2.3.2 Rotation Servo Part ........................................ 32

3.2.3.3 Wheel Base .................................................. 34

3.2.4 Motion of the Servo layer ....................................... 35

3.3 Final product: The Borg ............................................. 36

**CHAPTER 4: ALGORITHM IMPLEMENTATION** ...................... 37

4.1 Reconfiguration Algorithm 1st Phase: 1st Configuration ....... 38

4.2 Reconfiguration Algorithm 2nd Phase: Navigation ............ 38

4.3 Reconfiguration Algorithm 2nd Phase: Reconfiguration ....... 39

4.3.1 Center, Left and Right Accounting Process ................. 40

4.3.2 Actual Reconfiguration Process ................................ 40

4.3.3 New Formation Recognition Process ......................... 42

**CONCLUSION** ............................................................ 43

**RECOMMENDATION** ................................................... 45

**REFERENCES** ........................................................... 45

**APPENDIX A: SONAR SENSOR EXPERIMENT SETUP** ........ 47
# Table of Content

<table>
<thead>
<tr>
<th>Appendix B: Experiment Setup for Borg-to-Borg Proximity</th>
<th>Range</th>
<th>....</th>
<th>48</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix C: Light Intensity Experiment Setup</td>
<td>....</td>
<td>49</td>
<td></td>
</tr>
<tr>
<td>Appendix D: Specifications</td>
<td>....</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>Appendix E: Component Price List / Expenditure List</td>
<td>....</td>
<td>51</td>
<td></td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1.1: Overview of Problem ........................................... 6
Figure 1.2: Flowchart for overall Reconfiguration Algorithm .......... 7
Figure 1.3: Strength Count Process ........................................ 8
Figure 1.4: First Configuration Recognition .............................. 9
Figure 1.5: Flowchart Showing Logic Line of Navigation Process ....... 11
Figure 1.6: Formation Control for Borg following ........................ 13
Figure 1.7: Formation Control for Borg at the sides ....................... 14
Figure 1.8: Flowchart showing Logic Line of Center Borg Classification ... 15
Figure 1.9: Center Borg identification in Accounting Phase ............. 16
Figure 1.10: Flowchart Showing Logic Line of Right Borg Classification .. 17
Figure 1.11: Right Borg identification in Accounting Phase ............. 17
Figure 1.12: Center, Left, Right accounting Process for different configurations .......... 18
Figure 1.13: Right Borgs in Actual Reconfiguration ...................... 19
Figure 1.14: Left Borgs in place and New Formation is formed .......... 20
Figure 2.1: Setup of the Testing Ground .................................. 21
Figure 3.1: The Acroname PPRK ............................................ 27
Figure 3.2: Brainstem GP 1.0 Module ....................................... 27
Figure 3.3: Devantech SRF08 Range Finder .............................. 28
Figure 3.4: HS-322 HD Servo ............................................... 28
Figure 3.5: CAD Drawing of Sonar Layer .................................. 29
Figure 3.6: The Sonar Layer ................................................ 30
Figure 3.7: Sonar Sensor and LEDs ......................................... 30
Figure 3.8: Light Sensor and LED placement .............................. 30
Figure 3.9: Brainstem Layer ................................................ 31
Figure 3.10: CAD Drawing of Lifting Servo Layer ......................... 31
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 3.11</td>
<td>Lifting Servo Part</td>
<td>32</td>
</tr>
<tr>
<td>Figure 3.12</td>
<td>Ball Caster</td>
<td>32</td>
</tr>
<tr>
<td>Figure 3.13</td>
<td>CAD Drawing of Rotation Servo Layer</td>
<td>33</td>
</tr>
<tr>
<td>Figure 3.14</td>
<td>Rotation Servo Part</td>
<td>33</td>
</tr>
<tr>
<td>Figure 3.15</td>
<td>Linkage Mechanism</td>
<td>33</td>
</tr>
<tr>
<td>Figure 3.16</td>
<td>CAD Drawing of Wheelbase</td>
<td>34</td>
</tr>
<tr>
<td>Figure 3.17</td>
<td>Wheelbase</td>
<td>34</td>
</tr>
<tr>
<td>Figure 3.18</td>
<td>Actuation Sequence of Change of Direction</td>
<td>35</td>
</tr>
<tr>
<td>Figure 3.19</td>
<td>Final Borg Design</td>
<td>36</td>
</tr>
<tr>
<td>Figure 3.20</td>
<td>Final Borg Design in CAD Drawing</td>
<td>36</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>First Configuration Recognition</td>
<td>37</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>Target Acquisition, Formation Control and Obstruction to Unit</td>
<td>38</td>
</tr>
<tr>
<td>Figure 4.3</td>
<td>Center, Left, Right Accounting Process</td>
<td>40</td>
</tr>
<tr>
<td>Figure 4.4</td>
<td>Actual Reconfiguration Process</td>
<td>41</td>
</tr>
<tr>
<td>Figure 4.5</td>
<td>Moving Together in New Formation</td>
<td>42</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table 2.1: Proximity Range Values ........................................... 21
Table 2.2: Proximity Range values using a Borg unit as an obstacle ...... 22
Table 2.3: Light Intensity Values at 5, 6, 7 and 8cm away from LED .... 23
Table 2.4: Subnet Mask and IP addresses of IPAQs. ....................... 25
INTRODUCTION

Objective
The aim of this project is to have multiple simple robotic units to move together in a given formation. On acquiring a target, which is a light source, the entire unit will move towards the target. They will then be required to reconfigure to a new formation that will enable them to move across a channel that is the width of one robotic unit.

The Problem
Firstly, the multiple robotic units (Borgs) have to move together as a group. In doing so, the Borgs must have a formation control algorithm that will keep each other close in formation. This cannot be achieved unless the Borgs are aware of whether there are Borgs beside itself. Hence, there has to be a process that will determine the initial formation that the Borgs are in before they can proceed to move off as an entire unit. Once moving in formation as a unit, they are required to acquire the target, which is a light source. Once the target has been acquired, the entire unit will change the direction of travel and move towards the target. This change in direction of travel has to be communicated to all the Borgs, by the Borg(s) that detected the light source and hence, a form of wireless communication between them has to be established.

The unit will then encounter a narrow channel that is the width of 1 Borg. At this point, the entire unit will have to stop despite the fact that some of the individual Borg units are not obstructed (the Borg in the through path). Hence, a message has
to be sent to the entire unit again to make all the Borgs stop together, further emphasizing the need for wireless communication. The entire unit must then reconfigure themselves and form up behind the Borg that is in the through path. In doing so, the entire unit must be able to differentiate between the Borg in the through path, and which Borgs are to the right and left of the path. Once that is accomplished, the Borgs on the left and right will have to move behind the center Borg to form up the line. On forming the line, the entire Borg unit will have to undergo another process similar to the initial configuration recognition. The Borgs will have to once again be aware of the new formation that they are in before they can move off as an entire unit across the narrow channel.

**Scope**

This report documents the work that is performed on the project. The first chapter describes in detail the thought process of the reconfiguration algorithm. Chapter 2 will describe the setup of the testing ground, the wireless communication protocol employed and the experiments conducted to determine the characteristics of the various sensors followed by chapter which looks into the Borg design and architecture, the sensors used and components from the Acroname Palm Pilot Robot Kit that are used in the Borg design. Chapter 4 will then show the actual implementation of the reconfiguration algorithm on hardware. Finally, the report is concluded and recommendations made for future work on the project. A list of references and appendixes will be included at the end. Next, the work on reconfigurable robots done by other researchers will be discussed.
LITERATURE SURVEY

The concept of reconfigurable robots has gained popularity over the last few years and much research has gone into the investigation of the physical aspect of reconfiguring robots. Most research on reconfigurable robots places emphasis on the individual robotic modules that can be manually put together to form a larger unit of a different structure. Mark Yim et al. [1] developed his Modular reconfigurable robots called the Polybot, which is made from multiple robotic modules. The multiple modules can be put together to form structures like a snake like structure or a hexapod structure. Andres Castano et al. [2] built a similar robotic module called the Conro (CONfigurable RObot), which attempts similar functions as the polybot from Yim et al. [1]. Andres Castano et al. [2] however explored the possibility of having a CMOS digital camera to be incorporated onto each individual robotic module as another sensor or as an alternate means of control. Kasper Stoy et al. [3] worked more on the formulation of the locomotion control of the different structures that the Conro robots could form.

Most of the research done is mainly on the hardware of the individual robotic modules, which differs from the main aim of this project. The robotic units presented in [2] have 2 servos, one for yaw and the other for pitch and IR sensors mounted on the connector plates on the 2 extreme ends. One end has connecting pins and the other end has sockets that fit the pins. This is to enable the docking of 2 robotic modules. However, the robots have to be manually reconfigured to another shape and have no interaction with the environment to actuate the change in configuration. This is similar to the robot that is being developed in [1]. The
robotic modules in [1] have a similar design when compared to the Conro robots. However, the polybots do have a distinct difference as they do have the ability to self reconfigure and form different structures automatically. This seems to be similar to what this project sets out to achieve, but the polybots in [1] cannot initiate the change of structure autonomously from environmental factors, which is the main objective of this project.

The need for an environmental trigger factor for reconfiguration and autonomous reconfiguration requires that the robots communicate with one another and pass on the information that there is obstruction to the entire unit. There is not much literature on this aspect considering that reconfigurable robot research focuses on the individual modules and their physical movement as a unit. However, there is much to learn from research on cooperative systems between multiple robotic units. One such literature is “Robot Teams”, Tucker Balch and Lynne E. Parker [4]. The concept of a unit of multiple robotic modules reacting to environmental factors is similar to that mentioned in [4] on multiagent and multirobot systems. In [4], there are 4 classifications of multiagent systems, Homogenous Communicating, Homogenous non-Communicating, Heterogeneous Communicating and Heterogeneous non-Communicating, of which the Heterogeneous communicating systems was particular applicable. In that concept, the communicating agents are different from one another, in terms of different goals and actions. Thus, a same message sent to all of them could trigger off different process at different times. This is very similar to that required by a reconfigurable system where by the positions of the individual units differentiate
them from each other despite the otherwise identical physical aspects. In [6], studies were done into coordination and collaboration between wasps, which help them to build and configure structures and make them a coherent lattice for movement. This gave insight into how collaboration could be applied on the reconfigurable system.

Since most of the research on reconfigurable systems is concentrated on the hardware aspect of the individual module, there is room for development of a reconfigurable system that can make use of environmental factors to act as a trigger for the autonomous reconfiguration process.
CHAPTER 1: RECONFIGURATION ALGORITHM DESIGN

The Borgs are required to move together as one unit, and reconfigure themselves into a line when they meet with a narrow channel that is only the width of one Borg unit. They therefore have to form up behind the Borg that is in the through path. The Figure 1.1 below shows what the Borgs are to achieve.

The Borg design used to illustrate the algorithm in this chapter would be square shaped; this is to facilitate the explaining of the thought process of the algorithm. The actual design of the Borg will be described in detail in Chapter 3.

Figure 1.1: Overview of Problem
The approach to the objective is to segment the reconfiguration to 3 different phases. The first phase is the First configuration phase, followed by the navigation phase then lastly the reconfiguration phase. Before the details of the algorithm are discussed, the project definitions are listed out as follows:

I. The robotic units are constrained to only move in a square or rectangular environment.

II. The obstacles will not have any angular extrusions

III. The path that the robotic units have to reconfigure to is at least the width of one robotic unit.

IV. Any formation that the robotic units are laid out in must have at least one side facing another robotic unit.

The overall reconfiguration algorithm can be described by the flow chart in Figure 1.2. In the flow chart, the dotted arrows are non-decisional process and will be carried out without the need for feedback.

Figure 1.2: Flowchart for overall Reconfiguration Algorithm
1.1 First Configuration Phase

In this phase, the program will first account for the total strength of the Borgs in action. It will then establish the initial Borg formation.

1.1.1 Strength Accounting Phase

The first process here is the Strength Count. This process will establish the total number of Borgs in action. As shown in Figure 1.3, the controller will broadcast an initiating message to all the Borgs that are in action. This message initiates the strength count process for the Borgs.

![Diagram of Strength Count Process]

On receiving the Strength Count message, the Borgs will each broadcast a strength message; all the Borgs in action will receive this message.

By counting the total number of messages that each Borg receives, the total strength of the Borgs will be 1 more of that number. Employing this method, the program does not require manual input of total strength of Borgs in action.
1.1.2 Borg First Configuration Phase

The next process of the First configuration is to establish the first Borg formation, which is the shape that the Borgs are initially put into and supposed to maintain. If each Borg were to be square shaped, then by mounting light sensors and Light Emitting Diodes (LEDs) on all its corners, the first configuration that they are put in could be determined by firstly having all the Borgs switch on all the LEDs on all its sides, and then by making use of the light sensors to sense for any light source on any of it’s sides. Hence, any light detected by the Borgs on any of it’s side would mean that that side has another Borg beside it. See Figure 1.4.

In the case shown in Figure 1.4, Borg 2 in the green circle senses light coming from 2 of it’s sides due to the LEDs of Borg 1 and Borg 4 thus it will be aware that it has 2 Borgs beside it. This information will be stored and used later in the navigation phase and the reconfiguration process. Once this information is obtained for all the Borgs, the process will switch off all the LEDs and then proceed to the navigation phase of the reconfiguration.
1.2 Navigation Phase

In this phase, the Borgs will then move together as a unit. The navigation phase of the process has 3 key processes working in the background as the unit moves. They are as follows.

I. Target Acquisition Process:

   The Borgs must activate the photo sensors in all directions and sense for any light source at this point. The light source is the target and the Borgs will then move towards the target as a unit.

II. Formation Control Process:

   In moving as a unit, the Borgs must stay together, not necessarily keeping the original configuration, but at least not allow any one Borg to be left way out of the formation.

III. Obstruction to Unit Process:

   The Borgs will have to stop the moment any one of the Borgs encounter an obstacle to its path. If at this stage, the target is acquired, this will then trigger the reconfiguration phase.

The following Figure 1.5 shows a flowchart that describes the logic line of the navigation phase of the program.
1.2.1 Target Acquisition Process

The target is a light source. During navigation, the Borgs will constantly take in readings for all the light sensors. The moment any of the Borgs senses light from
any direction, the entire unit will then change the direction of travel to that
direction. Hence, the target is acquired once light is sensed in any direction.
The Borgs will only reconfigure if target is acquired. If no target is acquired, and
the Borg unit meets with an obstacle, it will not reconfigure itself, it will only stop
and wait to acquire the target.

1.2.2 Formation Control Process
To move as a unit, the borgs must try to stay close to one another during
navigation. The information as to which Borgs is beside another has been
determined during the first configuration phase. The information is different for
each Borg and is stored locally in each Borg.

With reference to Figure 1.6, the Borgs are traveling in the direction as assigned.
Borg 1 has the information that Borg 2 is behind it relative to the direction of
travel. It would then move together as seen in part 1 of the diagram. The rear
sonar sensors of Borg 1 would be activated to take readings of the distance
between itself and Borg 2. Once the distance exceeds the assigned proximity
range value, Borg 1 would stop moving as shown in the transition from part 2 to
part 3 of the diagram. This is so that Borg 2 would be able to catch up with Borg 1
before it falls out of formation; hence Borg 1 is required to stop. Once the distance
Borg 1 and Borg 2 is within range, Borg 1 would start moving again.
This formation control algorithm is also applied to formations that have Borgs beside one another. With reference to Figure 1.7, Borg 1 and 2 are to travel in the direction as shown. Borg 1 and 2 both have information that they are beside one another. As seen in part 2 of the diagram, the moment Borg 2 moves ahead of Borg 1 and hence falls out of formation, the corner sonar sensor of Borg 2 will give a reading that exceeds the proximity range. This would cause Borg 2 to stop and wait for Borg 1 to catch up. Once Borg 1 catches up, Borg 2 will resume its movement in the original direction.
1.2.3 Obstruction to Unit Process

The obstruction to unit process will trigger the start of the reconfiguration process and stop the navigation process. This will only occur when the entire Borg unit meets with any obstacles in the direction of travel and target has been acquired. Generally, the setup of the testing area will have a channel the width of one Borg unit. Hence, the other Borg units when moving together as one, and after acquiring the target, will encounter a wall and hence an obstacle and only one Borg unit will sense the channel. All other Borgs will thus have to form up behind the Borg that senses the channel.
1.3 Reconfiguration Phase

At this point, the Borg unit as a whole would have acquired a target and met with an obstacle to the target, this would trigger the reconfiguration phase of the program. The reconfiguration phase is broken down to a 3 different process:

I. The Center, Left and Right Accounting Process
II. The Actual Reconfiguration Process
III. The New Formation Recognition Process

1.3.1 Center, Left and Right Accounting Process

In this process, the entire unit first starts to differentiate which Borgs are the center Borgs that represent the through path that all the other Borgs have to form up behind. The following flowchart will show the logic process.

![Flowchart showing Logic Line of Center Borg Classification]

Figure 1.8: Flowchart showing Logic Line of Center Borg Classification
With reference to Figure 1.9, the setup consists of 4 Borg units in a certain initial configuration and a channel that is the width of one Borg unit. To identify which Borg(s) represents the through path and to form up behind, the condition that determines this is the Borg that has sonar sensor readings indicating no obstruction in the direction of travel, which would be Borg number 2.

Once this has been established, Borg 2 will light up its rear LEDs (relative to the direction of travel). Meanwhile, all the other Borgs will have to sense for light in the front direction and should any one of them sense light, they would in turn too light up their rear LEDs. In this way, the information would be propagated down the entire Borg formation. Only if there were light in the front, then the Borg would be a center Borg. This is graphically represented in Figure 1.9. These Borgs will then be identified as the center Borgs and will not move during the actual reconfiguration process. The center phase is in a timed loop and will move out of the center phase loop and proceed to the right phase loop after a certain number of
runs. The next phase is the right phase; the following flowchart will show the logic process of this phase.

![Flowchart Showing Logic Line of Right Borg Classification](image1)

After the center Borgs have been identified, the next phase would be to identify the left and right Borgs. Firstly, the right Borgs will be identified. In this phase, the center Borgs will light up its right LEDs. See Figure 1.11.

![Right Borg identification in Accounting Phase](image2)
The rest of the Borgs that are not center Borgs will be programmed to sense for light on all its sides. And so since the center Borgs lighted up the right LEDs, any Borgs that sense light in this phase will be the right Borgs. With reference to Figure 1.11 above, Borg 3 is the right Borg and lights up all but its left LEDs.

Once this phase is done and both the right and center Borgs have been determined, the center Borgs will then switch on their left LEDs while the right Borgs will switch off all theirs. The left Borgs will then be determined the same way as the right Borgs.

This approach to differentiate the center, left and right Borgs works for more than the above shown design. Below are a few other shapes that would work using the above-mentioned method.

Figure 1.12: Center, Left, Right accounting Process for different configurations
1.3.2 The Actual Reconfiguration Process

Once the center, left and right Borgs are determined, the actual reconfiguration process starts. In this process, the center Borgs will not move at all but switch on all their LEDs. The process will first start with the right Borgs forming up behind the center Borgs first. With reference to Figure 1.13, the right Borg, Borg 3 will move back till the left sonar sensors senses that there is no obstruction.

![Figure 1.13: Right Borgs in Actual Reconfiguration](image)

Once both left sonar sensors senses that there is no obstruction, Borg 3 will then move left till both sonar and photo sensors in front gives reading indicating proximity and light respectively. Borg 3 will then light up all it’s LEDs, indicating it is in the through path. The Left Borg (Borg 1) will execute the same movements as the Right Borg as seen in Figure 1.14.
1.3.3 New Formation Recognition Process

Once Borg 1 is in position, a new formation is formed and a process similar to the first configuration phase is performed at this point. See Figure 1.14.

![Diagram](image)

**Figure 1.14: Left Borgs in place and New Formation is formed**

The new formation is recognized and the Borgs will then switch off all its LEDs. At this point, the Reconfiguration Process is complete, the program will then switch back into the Navigation Phase and proceed to move through the channel in a line and still keeping the New Formation.
CHAPTER 2: TESTING GROUND SET UP, EXPERIMENTS AND WIRELESS NETWORKING

2.1 Design of the Testing Ground.

The requirement of the testing ground setup is relatively simple. It would consist of 2 rooms, both having dimensions of 2.5m by 2.2m, and a narrow channel the width of a Borg. The dimensions of the room are such so that a maximum of 4 Borgs can assume any initial configuration to thoroughly test the reconfiguration algorithm. The figure below shows the testing room layout.

![Setup of the Testing Ground](image)

Figure 2.1: Setup of the Testing Ground

2.2 Experiments Conducted on the Devantech SRF08 Range Finder

2.2.1 Sonar Ranging Experiment

Due to the complexity of the codes that are written for the reconfiguration process, the processing speed of the microprocessor is greatly slowed down. This often results in the Borgs colliding with the obstacle that is in front of them despite the fact that the program written caters for collision avoidance.
Hence, an experiment was set up to determine the proximity range value in cm that should be set in the program which includes the error due to the affected processing speed. This value should enable the Borg to stop within the range of 3-5 cm before colliding with an obstacle. To do this, the speed of the Borg is set to a constant of 4cm/s; the Borg is then set to move towards a wall made of 4mm thick plywood. This experiment is then conducted on all 4 Borgs and on all 4 sides.

The results obtained are tabulated in Table 2.1

**Table 2.1: Proximity Range Values**

<table>
<thead>
<tr>
<th>Borg NO.</th>
<th>Side of Borg tested with 4mm thick plywood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Front</td>
</tr>
<tr>
<td>1</td>
<td>15 cm</td>
</tr>
<tr>
<td>2</td>
<td>13 cm</td>
</tr>
<tr>
<td>3</td>
<td>12 cm</td>
</tr>
<tr>
<td>4</td>
<td>12 cm</td>
</tr>
</tbody>
</table>

The above results are tested with the 4mm thick plywood because the walls of the testing ground are built with that material. As seen from the table, the highest value required stopping the Borg moving at 4cm/s within a range of 3-5 cm from the plywood wall before collision is 15cm. Hence, that value will be used as the proximity range value in the program. Please refer to Appendix A for the Experiment setup.

However, besides using the wall, the Borgs will also have also have to prevent collision with another Borg as they will be moving close to one another. Due to the large number of readings that will result if all the sides of each Borg are to be tested against the rest of the other Borgs, only one Borg will be tested with all
others since they are similar in construction. The Borgs are then set to move
towards another Borg at 4cm/s and to stop within the range of 3-5cm. The value
that results from that experiment will then be taken as the proximity value for the
borg-to-borg collision avoidance

**Table 2.2: Proximity Range values using a Borg unit as an obstacle**

<table>
<thead>
<tr>
<th>Borg NO.</th>
<th>Front</th>
<th>Back</th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8 cm</td>
<td>9 cm</td>
<td>10 cm</td>
<td>6 cm</td>
</tr>
<tr>
<td>3</td>
<td>10 cm</td>
<td>9 cm</td>
<td>10 cm</td>
<td>8 cm</td>
</tr>
<tr>
<td>4</td>
<td>10 cm</td>
<td>9 cm</td>
<td>9 cm</td>
<td>8 cm</td>
</tr>
</tbody>
</table>

As seen from Table 2.2, the highest value required stopping the Borg moving at
4cm/s within a range of 3- 5 cm from another Borg before collision is 10cm.
Hence, that value will be used as the proximity range value in the program. Please
refer to Appendix B for the experiment set up.

**2.2.2 Light Intensity Experiment**

The photo sensor on the SRF08 Ranger will also be required to run the
reconfiguration process. Hence, the intensity value readings in the program should
also be determined experimentally. The Borgs used Light Emitting Diodes
(LEDs) as means to identify each other. Hence, the experiment would need to test
the value of the intensity reading on the photo sensor when an LED is shining
right onto the photo sensor but at different distances.

The Borgs should not move more than 3-5 cm away from one another and thus,
the experiment would take readings from 5 to 8 cm, should there be any error on
the keeping close algorithm and the Borgs end up more than 5 cm from one another. The results are shown in Table 2.3

**Table 2.3: Light Intensity Values at 5, 6, 7 and 8cm away from LED**

<table>
<thead>
<tr>
<th>Photo Sensor NO.</th>
<th>Distance LEDs are away from the sensor / cm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>2</td>
<td>81</td>
</tr>
<tr>
<td>3</td>
<td>79</td>
</tr>
<tr>
<td>4</td>
<td>79</td>
</tr>
<tr>
<td>5</td>
<td>83</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
</tr>
<tr>
<td>7</td>
<td>80</td>
</tr>
<tr>
<td>8</td>
<td>85</td>
</tr>
<tr>
<td>9</td>
<td>80</td>
</tr>
<tr>
<td>10</td>
<td>81</td>
</tr>
<tr>
<td>11</td>
<td>81</td>
</tr>
<tr>
<td>12</td>
<td>84</td>
</tr>
<tr>
<td>13</td>
<td>80</td>
</tr>
<tr>
<td>14</td>
<td>80</td>
</tr>
<tr>
<td>15</td>
<td>83</td>
</tr>
<tr>
<td>16</td>
<td>79</td>
</tr>
</tbody>
</table>

16 Photo sensors are tested and there is some consistency in the values, hence, only 16 sensors are tested in the experiment. As can be seen from the table, at 8cm, the lowest reading value is 31. Hence, the light reading minimum is 31 and hence, this value would be used in the reconfiguration process. Please refer to Appendix C for the experiment setup.

**2.3 Wireless Local Area Network**

The Borgs need to communicate between each other, hence there has to be a form of networking between them. The IPAQ that is used to link up with the Brainstem GP 1.0 has an internal Wireless Local Area Network (LAN) card and could be utilized to send messages between the Borgs. Winsock specification is used for
the networking programming codes. Winsock is chosen mainly because it is WinCE compatible, which is essential as the IPAQ Operating System runs on the WinCE platform. Also, the necessary libraries and headers can be compiled in C++ Language and could be compiled together with the main program code in Embedded Microsoft Visual C++.

### 2.3.1 User Datagram Protocol (UDP)

The protocol used in the wireless LAN network setup is the User Datagram Protocol (UDP). This protocol uses similar information transport methods as the Transmission Control Protocol (TCP); however, the difference here is that UDP does not require a server client relationship in the network. Hence, any message sent out by any one user goes out to all the other users including itself. This protocol is particularly simple to implement, as there is no need to have an external router to act as a server.

### 2.3.2 Addressing and Internet Protocol

Most computers can be assigned an Internet Protocol address (IP address). This address is represented by a 32-bit quantity. Hence, when a user wants to communicate with another user on a network, the most important information is the IP address of the other user. The IPAQ palm pilot that is being used in this project could also be assigned an IP address, enabling them to send messages across to one another. As the IPAQs user UDP protocol, there is no need to setup a server/client relationship. There is however a need to identify the network that all the users are on, this requires another quantity called the subnet mask. This is a similar 32-bit quantity as the IP; just that it is solely for network identification.
Thus, all the IPAQ Palm pilots are given the setting as shown in Table 2.4, which defines the entire network and the users on them. This information is then incorporated into the C++ program code, enabling the Borgs to communicate using the wireless LAN.

Table 2.4: Subnet Mask and IP addresses of IPAQs.

<table>
<thead>
<tr>
<th></th>
<th>IP address (User Identification)</th>
<th>Subnet Mask (Network identification)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPAQ No. 17</td>
<td>192.168.1.17</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>IPAQ No. 25</td>
<td>192.168.1.25</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>IPAQ No. 18</td>
<td>192.168.1.18</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>IPAQ No. 20</td>
<td>192.168.1.20</td>
<td>255.255.255.0</td>
</tr>
</tbody>
</table>
CHAPTER 3: BORG DESIGN AND ARCHITECTURE

3.1 What is a Borg?

The Borg is the individual robotic unit in the entire experiment of reconfiguration. It is designed solely for this project of reconfigurable robots and most of its parts are taken from another robot, the Acroname Palm Pilot Robot Kit (PPRK) robot. See Figure 3.1

At the heart of the Borg is the Brainstem GP 1.0 Module. See Figure 3.2 (originally on the PPRK). It has a 40MHz PISC processor chip which is capable of 9000 instruction execution per second. It has a RS 232 serial port, which provides the interface between the processor and the palm pilot. The onboard 1Mbit IIC port provides the interface between the Devantech SRF08 Range Finder (sonar and light sensor) and the chip. The Brainstem also provides 4 high-resolution servo outputs that are used to control the servos used in the Borg design. It also has 5 digital I/O lines, which can be configured to provide a 3V output DC signal that is used to power the LEDs that are mounted on the Borg.
The parts that are taken from the PPRK unit are the Brainstem GP 1.0 and the modified R29 MX-400 servo. Additional hardware had to be purchased to facilitate the experiment, mainly the Devantech SRF08 Range finder, See Figure 3.3. The SRF08 is both a sonar range finder that is capable of sensing distances ranging from 6m to 3cm and a photocell sensor, which can detect different white light intensity. This combination of sensors is extremely useful to the reconfiguration process and its purposes will be explained in detail in Chapter 2, Algorithm design. Also, 2 more HS-322 HD Servo had to be purchased to provide the lifting and rotation component of the Servo Layer. See Figure 3.4.

3.2 Borg Architecture and Design

The reconfiguration algorithm works on a square shaped Borg design, which has both sonar and light sensors mounted on the corners on each side. LEDs must also be mounted on each side for Borg-to-Borg identification. The Borg should also be identical on all 4 sides for ease of docking of one Borg to another during the reconfiguration process. In view of these various specifications required, a square shaped Borg design was adopted. The Borg consists of 3 different layers. Each
layer is square in shape hence the entire Borg unit is shaped like a cube. The first layer is the sensory layer of the Borg, where the sonar and light sensors are mounted on. Also, there are rows of 7 Light Emitting Diodes (LEDs) mounted beside each sonar and light sensor. The second layer is the Brainstem layer. This is the layer where Brainstem GP 1.0 chip is mounted on. The switch and the batteries can all be found on this layer. The last layer of the Borg unit is the servo layer. This is the layer that provides the locomotion of the Borg unit. The Borg designs were drafted out in Solidworks CAD Software before any fabrication took place.

### 3.2.1 Sensory Layer

The design of the sensory layer is such that there is symmetry on all sides of the Borg. By having such a design, the Borgs do not have to rotate its orientation in order to form up behind another Borg. The Figure 3.5 shows the layout of the sensory layer in CAD Drawing and Figure 3.6 shows the actual sonar sensor layer.

![Figure 3.5: CAD Drawing of Sonar Layer](image)
As can be seen, the sensory layer has 2 sonar and light sensors on each side, one on the top and another at the bottom. The LEDs are mounted 7 in a row and under each sonar and light sensor, See Figure 3.7. With this arrangement, each side is identical to one another.

The Borg has identical sides but when it is beside another Borg, the LEDs are pointing straight at the light sensors. The main aim of the LEDs is for Borg-to-Borg identification and information transfer. Hence, this design serves these purposes perfectly as can be seen from Figure 3.8.

### 3.2.2 The Brainstem Layer

The Brainstem chip is located near the center of the brainstem layer. This is due to the numerous connections from the brainstem chip to the different components of
the Borg. The batteries are located at the sides so that they can be removed easily and the switch is located at the corner for ease of switching on.

3.2.3 The Servo Layer

The servo layer itself consists of 3 different parts. The first part is the Lifting servo part, second, the Rotation servo part and last being the wheelbase part. The reason for such an elaborate design is because it allows the Borg to move relatively straighter than the Acroname PPRK design.

3.2.3.1 Lifting Servo Part

The lifting servo part is the layer that the lifting servo is mounted on; it consists of the servo that lifts up the wheelbase and rotation servo layer. Figure 3.10 shows the CAD drawing and Figure 3.11 shows the actual Lifting servo layer fabricated.
This is also the layer that the ball casters are mounted on. The ball caster consists of a ball bearing cast at the bottom of the casing, which gives vertical support and minimal horizontal resistance to dragging See Figure 3.12. When the lifting servo is activated, the entire locomotion base of the Borg is lifted up and the ball casters provide the necessary support.

### 3.2.3.2 Rotation Servo Part

This rotation servo rotates the wheelbase of the Borg to achieve the change of direction in locomotion of the Borg by turning the entire wheelbase to different pre-assigned direction. To ensure vertical lifting with minimal side deviation, there are 4 vertical support pillars on the 4 corners of the rotation servo layer, which guides the lifting action. See Figure 3.13 for the CAD Drawing and Figure 3.14 for the actual fabricated component.
Aluminum L- Frames are fitted onto the rotation layer and linked to the lifting servo by a supporting beam through both the L-frames. Retrofitted servo components then connect the beam to the lifting servo to provide the lifting action. The actual mechanism can be seen in the following Figure 3.15.
3.2.3.3 The Wheelbase Part

The last part of the servo layer is the wheelbase. As seen in Figure 3.16 and 3.17, it consists of a driven axle and a slave axle.

Figure 3.16: CAD Drawing of Wheelbase

Figure 3.17: Wheelbase

As straight movement would be useful in the algorithm of the Borg reconfiguration and locomotion, the design is such that one modified servo will drive 2 wheels (driven axle) so as to move as straight as possible. The other 2 slave wheels are there to increase frictional resistance to forces that might cause the Borg to deviate from the original straight path.
3.2.4 Motion of the Servo Layer

Figure 3.18 describes in sequence how the Borg actuates a change in direction of travel.

The wheelbase in contact with the ground.

Wheelbase is lifted up by the Lifting Servo

Wheelbase is set in contact with the ground again to propel the Borg in new direction of travel

Rotation servo rotates the wheelbase 90° clockwise to change direction of travel

Figure 3.18: Actuation Sequence of Change of Direction
3.3 Final Product: The Borg

The following Figure 3.19 shows the final Borg design with all the different layers put together.

![Figure 3.19: Final Borg Design](image)

![Figure 3.20: Final Borg Design in CAD Drawing](image)
CHAPTER 4: ALGORITHM IMPLEMENTATION

In this chapter, the implementation of the algorithm on actual hardware would be presented with an example of a 4 Borg initial configuration. The design definition is stated again here:

I. The robotic units are constrained to only move in a square or rectangular environment.

II. The obstacles will not have any angular extrusions

III. The path that the robotic units have to reconfigure to is at least the width of one robotic unit.

IV. Any formation that the robotic units are laid out in must have at least one side facing another robotic unit.

4.1 Reconfiguration Algorithm 1st Phase: 1st Configuration

The first configuration phase of the reconfiguration process accounts for the strength of all the Borgs that are in action. There is no manual input of this information and it is achieved through wireless communication of the Borgs. After which, the Borgs will light up all their LEDs for first configuration recognition. The process is shown in the following Figure 4.1

Figure 4.1: First Configuration Recognition
There are 2 problems that this method faces. Firstly, some of the LED mounts are not precisely facing the light sensor of the other Borg. Hence, this sometimes causes the Borg to fail to recognize that there is another Borg beside it and this would cause the formation to fail when the Borgs move off as a unit. Also, there were a few occasions that a Borg reports more than the correct number of Borgs beside it. This could be due to the light reflecting from the Borg surface to another sensor that is not supposed to sense any light, thus causing the incorrect assumption that there is another Borg on that side.

4.2 Reconfiguration Algorithm 2nd Phase: Navigation

In the Navigation phase of the reconfiguration algorithm, the main objectives of the Borgs are target acquisition (light source), formation control and stopping when there is obstruction to the entire unit. Figure 4.2 shows the Borg formation moving in a group, acquiring a target and stopping at obstacle met.

Figure 4.2: Target Acquisition, Formation Control and Obstruction to Unit
The problem faced in this part of the algorithm is that sometimes, more than 1 Borg would detect the target. This then results in more than 1 stop all message sent to all the other Borgs, which would cause the entire unit to stop for a little while longer than it should have. The solution to this problem is to have more than 1 wireless message-receiving loop in 1 run of the navigation part of the algorithm. This makes message receiving more efficient and reduce the stopping time of the entire unit when light is detected.

Another slight problem is that the entire unit cannot move together smoothly, there will always be one Borg moving out of formation and leading to a whole chain of jagged movements by the entire unit. Setting all the Borgs to move at the same speed has minimized this problem.

4.3 Algorithm 3rd Phase: Reconfiguration

The reconfiguration phase of the algorithm has 1 main objective, getting all the Borgs to form a line behind the Borg that is in the through path. This is achieved in 3 processes.
4.3.1 Center, Left, Right Accounting Process

Figure 4.3 shows the Center, Left and Right Borg identification Process.

4.3.2 Actual Reconfiguration Process

In this process, the Borgs are made to form up behind the Borg that is in the through path. It will start with the default, which is the right Borgs, and then it will proceed with the left Borgs. Figure 4.4 shows the actual reconfiguration process.
The problem faced here is similar to that in the Center, Left and Right Accounting Process, alignment problems. However, moving straight here is also very important and this occasionally causes problems in the reconfiguration process.
4.3.3 New Formation Recognition Process

This process is very similar to the 1st configuration phase. It requires the Borgs to recognize the new formation that they are now in and move off together as a unit in that formation. Figure 4.5 shows them with all the LEDs off and then moving off as a unit across the narrow channel.

![Figure 4.5: Moving Together in New Formation](image)

The difficulty in this process is that after the actual reconfiguration process, the Borgs are sometimes poorly aligned. This occasionally causes the Borgs to not recognize that there is another Borg behind or in front of it because it cannot sense the light coming from the other Borg’s LEDs. However, by slowing down the speed of the Borg movements, this problem has been minimized.
CONCLUSION

From the studies that were conducted during the literature survey, it could be observed that the focus of the reconfigurable system, at least among the research that were studied in the literature survey, was more on the mechanical attachment between the modular units. Hence, given the project objective, research on autonomous and cooperative systems gave more insight into the basic requirements of the required autonomous reconfigurable system. However, the challenging aspect of this project is that a formation needs to be held and then changed to suit the environmental constrain imposed on it. This project objective required an original approach to the problem and the algorithm presented in this report has successfully accomplished this.

The use of LEDs and light sensors for positional recognition is a novel and effective approach to differentiate the robotic units that were in the through path and those that were beside the through path during the reconfiguration process. However, the use of light sensors meant that calibration was required as most of the time, sensors readings varies from sensor to sensor. Hence the appropriate values to be used in the program that would translate to sensible actions in hardware had to be obtained through experimentation. Experiments were also carried out on the sonar sensors, which eventually enabled the sonar sensors to help keep the formation of the entire unit. As moving straight played an important role in the reconfiguration algorithm, a unique wheelbase had to be specially designed and fabricated just to achieve moving straight.
In the reconfiguration algorithm that was implemented on this system, communication was very important and hence the amount of work put into establishing a wireless network among the robotic units that are in action. However, the implementation of wireless communication into the program codes proved to be most difficult. Thus, much effort had to be put into the study of Winsock programming to incorporate the wireless messaging codes in C++ into the program codes using Embedded Visual Basic C++ software.

Testing of the reconfiguration system started out with only 2 units and then proceeded to 3 and eventually to 4 units. The progressive testing of the system is very time consuming as testing was carried out from mid December 2003 to end of March 2004. However, this way of testing the system helped to expose the areas that were overlooked when developing the reconfiguration algorithm. With the repeated testing of the system especially when it came to 4 robotic units, the reconfiguration algorithm was time and time again fine tuned by learning from the many failed attempts to reconfigure. Eventually, the process of repeated testing greatly strengthened the reconfiguration algorithm, making it more robust. Hence, the project objective of autonomous reconfiguration triggered by environmental constrains was finally achieved by taking advantage of the novel robotic unit design, the wireless communication that was implemented into the program, the calibration of the sensors and the repeated testing of the unique reconfiguration algorithm on the actual hardware.
RECOMMENDATION

There are 2 areas that the author recommends for improvement.

Most of the research done on the reconfigurable robots focuses on the connection between the modules, but as can be seen from this report; the Borgs are not connected to one another. The sensors that are mounted on the corners of each side hold their formation but not physically. This makes the design easy to work with but the disadvantage is that the formation is not held tight. This occasionally causes problems in the reconfiguration process. More work could be done in the future in this aspect of hardware, a simple connection between the robotic units could hold the formation tighter.

The Devantech SRF08 Range Finders has an incorporated light sensor. This light sensor often gave inconsistent results even between sensors despite being in the same conditions. Perhaps a more accurate photocell sensor could be used here to avoid the many problems that were faced during the 1st configuration and reconfiguration phase due to the insensitivity of the light sensors.
REFERENCES


APPENDIX A : SONAR SENSOR EXPERIMENT SET UP

4 mm plywood

Experimental Set up
APPENDIX B: EXPERIMENT SETUP FOR BORG-TO-BORG PROXIMITY RANGE

Experiment Set up
APPENDIX C: LIGHT INTENSITY EXPERIMENT SETUP

Experiment Setup
# APPENDIX D : SPECIFICATIONS

<table>
<thead>
<tr>
<th>Physical Characteristic</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>0.21 m</td>
</tr>
<tr>
<td>Width</td>
<td>0.21 m</td>
</tr>
<tr>
<td>Height (body)</td>
<td>0.32 m</td>
</tr>
<tr>
<td>Weight</td>
<td>1.5 kg</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Power</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Brainstem Logic Voltage</td>
<td>6.0 V D.C</td>
</tr>
<tr>
<td>Servo Voltage</td>
<td>6.0 V D.C</td>
</tr>
<tr>
<td>Run time</td>
<td>2 – 3 Hrs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sensors</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Devantech SRF08 Range Finder</td>
<td>8 total, 2 on each side.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Electronics</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>Acroname Brainstem GP 1.0</td>
</tr>
<tr>
<td>Servo inputs</td>
<td>4</td>
</tr>
<tr>
<td>Sensor Input</td>
<td>1 II c Bus Port</td>
</tr>
<tr>
<td>Digital I/O</td>
<td>5</td>
</tr>
<tr>
<td>A/D</td>
<td>5</td>
</tr>
<tr>
<td>Serial Input</td>
<td>1 RS-232 serial</td>
</tr>
<tr>
<td>Servo Power Input</td>
<td>1</td>
</tr>
<tr>
<td>Logic Power Input</td>
<td>1</td>
</tr>
<tr>
<td>Power switches</td>
<td>1 main.</td>
</tr>
</tbody>
</table>
APPENDIX E: COMPONENT PRICE LIST / EXPENDITURE LIST

Components Provided by CoSy Lab (per robot)

<table>
<thead>
<tr>
<th>Part no</th>
<th>Component Description</th>
<th>Quantity</th>
<th>Cost/part</th>
<th>Total Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Brainstem GP 1.0</td>
<td>1</td>
<td>$133</td>
<td>$133</td>
</tr>
<tr>
<td>2</td>
<td>Modified Servo R29 MX 400</td>
<td>1</td>
<td>$20</td>
<td>$20</td>
</tr>
<tr>
<td>3</td>
<td>Devantech SRF08 Range Finder</td>
<td>8</td>
<td>$85.00</td>
<td>$680</td>
</tr>
</tbody>
</table>

Components Purchased for Fabrication of Borg (per robot)

<table>
<thead>
<tr>
<th>Part No.</th>
<th>Component Description</th>
<th>Quantity</th>
<th>Cost/part</th>
<th>Total Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Acrylic Sheet</td>
<td>2</td>
<td>$8.50</td>
<td>$17.00</td>
</tr>
<tr>
<td>2</td>
<td>Ball Caster</td>
<td>4</td>
<td>$4.00</td>
<td>$16.00</td>
</tr>
<tr>
<td>3</td>
<td>Servo to Wheel Horn attachments</td>
<td>4</td>
<td>$2.80</td>
<td>$11.20</td>
</tr>
<tr>
<td>4</td>
<td>Gears for wheelbase</td>
<td>2</td>
<td>$4.50</td>
<td>$9.00</td>
</tr>
<tr>
<td>5</td>
<td>Rubber Tyres (pair)</td>
<td>2</td>
<td>$8.00</td>
<td>$16.00</td>
</tr>
<tr>
<td>6</td>
<td>HI Tec Hs 322 HD Servo</td>
<td>2</td>
<td>$18.00</td>
<td>$36.00</td>
</tr>
<tr>
<td>7</td>
<td>Rechargeable Batteries</td>
<td>8</td>
<td>$5.00</td>
<td>$40.00</td>
</tr>
<tr>
<td>8</td>
<td>Spacers</td>
<td>82</td>
<td>$0.25</td>
<td>$20.50</td>
</tr>
<tr>
<td>9</td>
<td>Miscellaneous parts</td>
<td></td>
<td>$10.00</td>
<td>$10.00</td>
</tr>
</tbody>
</table>

TOTAL COST OF COMPONENTS PURCHASED FOR FABRICATION: $175.70

TOTAL COST OF EACH ROBOT (inclusive of parts provided by lab): $1,008.70

Total Number of Borgs Fabricated: 4