**Team 5:**

Simon DeMartini - [simond13@uw.edu](mailto:simond13@uw.edu)

Devin Durham - [d1durham@uw.edu](mailto:d1durham@uw.edu)

Amrit Puri - [amrit.puri27@gmail.com](mailto:amrit.puri27@gmail.com)

**Meetings/Agreements**

Meet Tues or Thursday after class sometime

Lock Code 2947 in Lockers for Team 5

**Initial Questions:**

Communication between bricks? We are all open to using NXT robots to ease bluetooth issues.

**Initial Brainstorm**

*5/9*

*- Use multiple tasks or plan architecture better than challenge 2.*

*- For gradient detection we could have the light sensors farther apart to detect whether it is heading in the right direction.*

**Major distributions of work**

Simon - Integrate different tasks and define the control in code, research Bluetooth

Devin - Make tasks, and assist with building the robot

Amrit - Lead the build of the physical, and help with making tasks as needed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Robot Challenge Journal** Challenge No.: 3  Team Name: Team 5    Members’ Names: Simon DeMartini, Devin Durham, Amrit Puri | | | | |
| **Date** | **Seq. #** | **Name(s)** | **Hypothesis/Behavior** | **Description/Results** |
| 5/9 | 1 | Devin | Initial outline. discussed how to organize the events and tasks. decided on some key things (like using the nxt). Highlighted some major hurdles (brick to brick communication) |  |
| 5/11 | 2 | Simon | Refine outline diagram and show it to Fowler. | Created the diagram on LucidChart, and made it cleaner with a complete priority list. Fowler approves and recommended we document how we want to divide the tasks and sensors reading between brain bricks.  <https://drive.google.com/open?id=0BxUoE3_DHH-9RDZwem94a0VTM3M> |
| 5/11 | 3 | Simon | Attempt to upgrade to RobotC firmware on NXT devices so we can test out bluetooth | The first NXT we tried got stuck in firmware update mode, and continuously clicking. We are unable to proceed until we fix this. |
| 5/16 | 4 | Simon | Professor fowler has fixed the firmware issue, and we are now able to test the bluetooth communication between bricks with RobotC | I followed this tutorial <http://robotsquare.com/2012/03/21/tutorial-robotc-bluetooth/> to just get something working and start to learn the capabilities of the bluetooth.  BT has the ability to send either one value, or three values at once.  Sometimes the NXT bricks do not want to connect via USB after using the other one. Restart the brick to fix this or select the brick via Communication Link Setup in RobotC |
| 5/16 | 5 | Amrit | Begin planning robot hardware | nxtprograms.com/projects/2.html |
| 5/16 | 6 | Simon | Create a test program to see how fast I can send BT messages from one brick to another. | It looks like we can have a delay of 450ms between sending packets with minimal packet loss. Maybe we can just assume packet loss will happen and send as fast as possible. |
| 5/18 | 7 | Devin | Worked on designing and outlining the task structure for the code. We found two methods call SuspendTask() and ResumeTask(). We hope to use this as a way of decoupling our behaviors into separate tasks. | Works as expected. This will allow us to dynamically change between tasks and quickly resume or switch behaviors as needed. this also helps for development of the code because of the low dependency. |
| 5/18 | 8 | Devin | Amrit worked on building a solid base for the robot. | We found putting the bricks side by side is too wide. Amrit change the design by placing the bricks right on top of each other to create a better center of gravity. |
| 5/18 | 9 | Simon | Create a test program to see how well the Robot can use suspendTask() and resumeTask() to control which task is running | It worked with being able to control the text on the screen. This is a good sign |
| 5/23 | 13 | Simon | Make the other brick send the parsed sonar data and light sensor data for the fear via bluetooth | Worked correctly and easy to setup with bluetooth. |
| 5/23 | 10 | Simon | Attempt to transfer the previous task switching code to something that will use control the motors separately in each task | We were unable to switch out of the robot’s Wander task for some unknown reason even though the display was correctly switching modes |
| 5/25 | 11 | Simon | Somehow RobotC decided the task switching code should work fine today | Probably RobotC, the Computer or NXT bricks all needed a reboot or something. |
| 5/25 | 12 | Simon | Test the task switching with just Escape and Wander | Correct switching behavior (although the Escape task is not entirely complete yet -- it takes over control correctly) |
| 5/25 | 13 | Simon | Create a third Death task for further task switching tests | The robot switched to death mode but the sound never stopped playing |
| 5/25 | 14 | Devin | Worked on gradient follow task. Looked at challange 2 code for inspiration. | Wrote new code for this gradient follow task. a little more complex than challenge two so i had to add in a lot more if else statements. also incorporated a timer to ensure correct direction and to detect when going in the wrong direction |
| 5/25 | 15 | Simon | Attempt to fix the Death problem by making it a simple function that plays the sound, stops the motors, and kills all running tasks. | This worked. |
| 5/25 | 16 | Amrit | Attempt to create bumper response task. | The code had to be refactored for NXT, but it was a simple conversion. |
| 5/25 | 17 | Amrit | Attempt to create approach task. | The code had to be refactored for NXT and took extensive testing to tune properly. |
| 5/25 | 18 | Simon | Refactor how the Sensor tasks determines what state should be running. | After some minor tweaking works correctly. |
| 5/25 | 19 | Simon/Devin | Integrate correct escape code and test timers | Correct behavior after some minor tweaking. |
| 5/25 | 20 | Devin | built the Feeding task. Looked at gradiant follow and fear task for inspiration. | Kinda used the same style of code the fear task and gradient. had to implement another timer to keep track of the energy level. lso made a helper function for simon to keep track (decrement) of energy outside of feeding task. Feeding task includes if the robot goes outside of patch, wandering in patch, and incrementing energy level. |
| 5/25 | 21 | Devin | Amrit designed and built the robot. it looks awesome and perfectly meets the sensors needs |  |
| 5/30 | 22 | Simon, Devin, Amrit | All tasks are complete individually, so we need to work on integration and testing. |  |
| 5/30 | 23 | Devin | integrated approach into the main branch. | we had to slightly adjust the threshold values to ensure bumpers would not be activated in approach |
| 5/30 | 24 | Devin | integrated gradient follow code | had to do a lot of threshold sensor adjustments for received values. we actually discovered that our light sensors received very different values so we had to change the threshold to meet both. |
| 5/30 | 25 | Devin | Integrated feeding code | because this was the very last behavior, integration was a lot more difficult. we had to tweak the values for everything just right to get it to work. |
| 5/30 | 26 | Devin | we began to walk through and test all of our behaviors to fine tune any actions | gradient follow and patch feeding became some of our biggest problems. this required setting the right flags and getting the turning and timing just right |
| 5/30 | 27 | Devin | Demo 1 with Alan Fowler | Double bumper issues  Patch feeding issues don't display wonder |
| 5/30 | 28 | Devin | fixed bumper and patch wonder | Quickly fixed bumpers. gradient follow was resolved by changing the turnaround time to 75% to allow changing of direction while in feed mode |
| 5/30 | 29 | Devin | Demo 2 | Problems noted:  -Spin in one direction  -Not gradient following  -Not drunken sailor walk to gradient |
| 6/1 | 30 | Simon | Extra Credit Demo for having the motors separated between bricks | Despite issue with finding a battery that was not dead, we were able to successfully pass the extra credit demo |

**Diagrams**



