Session 1554

Control Systems in Designing and Programming a Robotic Ant

Paul D. Schreuders, Naz Azadi, Lauri Bennett, Saba Choudhary,

Jason Congdon, Mike Ratino

University of Maryland, College Park

Abstract

Control Systems are employed in robotics to successfully achieve a desired task. The main objective of this project was to challenge the Biological Systems Control class to learn and implement control techniques in the designing and programming of a robotic ant. The class was presented with three different challenges and each group applied its own design and programming approach. Legos were used to design the ant and then RoboLab software was used to program it to successfully meet the challenge.

Introduction

Control Systems

A control system is a collection of interconnected components that can be made to achieve a desired response in the face of external disturbances. The study of control systems is an integral component of the engineering curriculum for the design and analysis of high performance robotic systems. Control systems can be of two types: open-loop or closed-loop. In an open-loop system, the response of the system is determined only by the controlling inputs. On the other hand, a closed-loop system operates on a self-regulatory mechanism by employing feedback to the control system. Feedback can be of two types: positive and negative. When feedback is employed, the output from the system is sensed and fed back to the input where it can regulate the input variable accordingly (Khoo, 2000).

Robotics

A robot is defined as a reprogrammable manipulator designed to move materials, parts, tools, or specialized devices. This multifunctional machine can be engineered and programmed to perform a variety of tasks in industry as automotive devices. In this way, humans themselves do not have to perform the tasks, but can easily regulate the robots to achieve the desired goal.

Robotics is applied to many fields including biological systems. For example, mechanical robots that can act as limbs can be employed in biological systems as arms and legs. Current research at various institutes, including the Johns Hopkins University, show that robotics can be employed for the design of medical systems such as surgical instrumentation.

Objectives

Due to the ongoing technological advances in the field, it is essential to have a sound engineering knowledge of control systems that can be employed in the field of robotics. This effort leads to the objective of this project.

The Biological Systems Control class was prompted to approach the challenge of creating a robotic ant and programming open-loop or closed-loop systems to successfully perform various tasks. The kit that was provided in order to meet the challenge included a RCX block, which is a Lego microcomputer that is programmed using RoboLab Software, Legos and various sensors including a pressure sensor, light sensor, and thermal sensor.

The Challenge

The project consisted of three challenges:

Challenge 1: Ants are always seen running along random paths. The purpose of the first challenge was to program the robotic ant to run around a diamond-shaped area twice, stop at a black finish line, and then present a victory dance. (See figure 1 for a sketch of Obstacle 1).

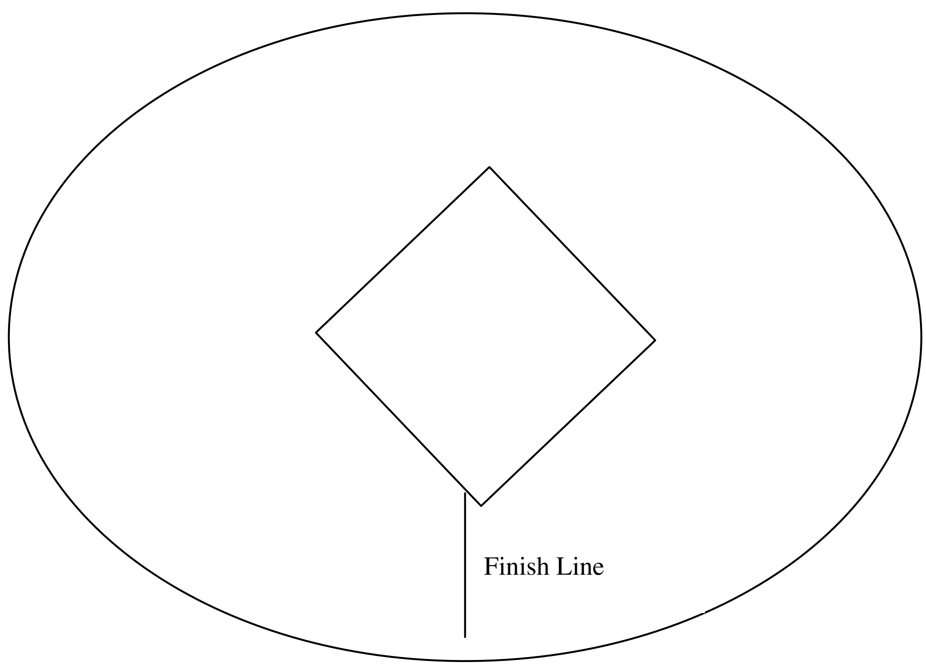


Figure 1: Obstacle for Challenge 1

Challenge 2: As ants follow their random paths, they may encounter food. When this occurs, the ant will pick up the food, and carry it straight back to its home. For the second challenge, the robotic ant had to be programmed to travel along a defined path to find food and then take it back to its home. The path was defined using three tapes of different colors. The middle was blue with yellow on one side and black on the other side. (See Figure 2 for a sketch of Obstacle 2).

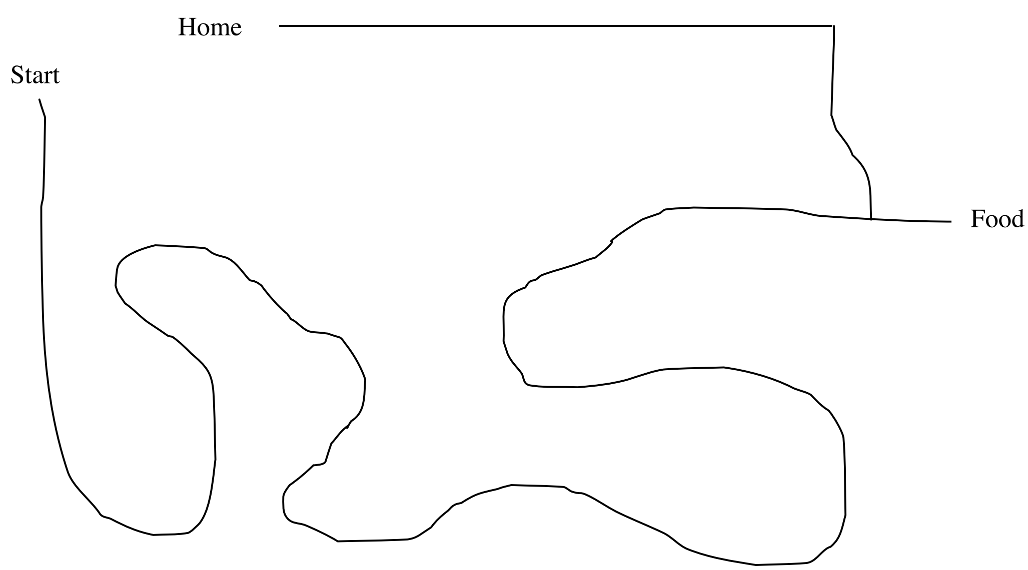


Figure 2: Obstacle for Challenge 2

Challenge 3: A worker ant has the job of searching for food for the care of the queen and protecting the queen from harsh high temperatures. The robotic ant had to be programmed to find randomly placed food, represented by red LEDs (light-emitting diodes) in a dark room. To add to the challenge, the pathway to the LEDs was obstructed by wooden blocks. After collecting as much of the food as it could find in five minutes, the ant had to find the heat lamp and control the temperature radiating to it by moving backward if the temperature reached 27°C. (See Figure 3 for a sketch of Obstacle 3).

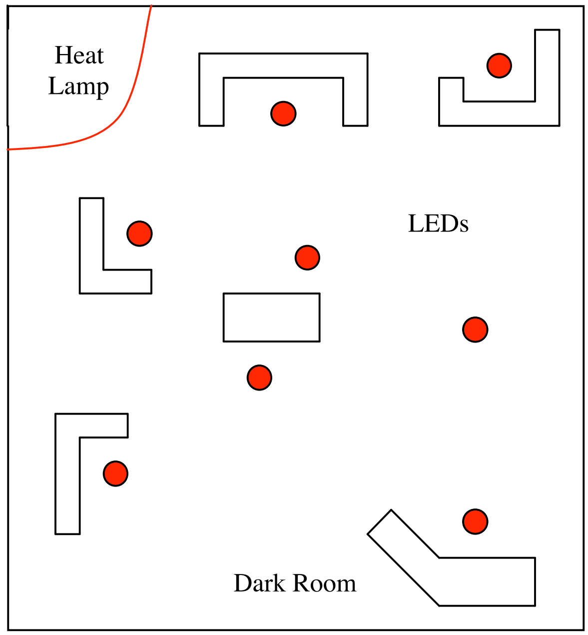


Figure 3: Obstacle for Challenge 3

Challenge #1 Approach

Summary

The challenge of controlling an ant made of Legos to travel around a defined diamond-shaped area was approached by programming a control system using Robolab. An open-loop control program was constructed to manipulate the ant and have it run the race while playing a series of musical numbers. The ant was designed to race down the track in a circular path, which was achieved by angling the front wheel. After completing two successful laps around the racetrack, the ant was prompted to stop at the black finishing line by sensing a light intensity difference of 5 units. After being still for 10 seconds, the ant was programmed to perform a victory dance for 30 seconds to further entertain the audience.

Design Approach

Many design possibilities were explored before coming to a decision on the best one for successfully completing the challenge. Initially, we made the attempt to program the ant so that it would move straight and turn 90 degrees at each corner of the diamond. The ant was designed with two large front-wheel drivers and two small wheels in the back. However, this attempt was not successful due to inconsistencies in the turn angle. Hence, it was decided to redesign the ant so that the large driver wheels were in the back with one small wheel in the front attached at an angle as to make the ant steer in a circular fashion. This design was chosen for its consistency and was found to be successful.

Solution Description

To achieve our goals, an open-loop control program in Robolab was used to manipulate the ant. (See Appendix 1 for the program used to meet Challenge 1). It consisted of a series of timed commands to activate the two driving motors in specific directions and powers, to activate the light reflectance sensor, and to play music. The music was created using the piano player and was activated by adding a subroutine to the loop. The program began with a short 4-second bit of music, and then entered the first series of commands to control the ant around the circle while a new music subroutine played. Because the front wheel of our ant was angled, we set the motor to run in the forward direction at full power for 35 seconds (a time factor that was decided on after several trial runs to make sure that the light sensor was not activated prematurely to stop the ant any other dark tiles). The light sensor command was added to stop the ant when its reflectance varied more than 5 units (i.e. at the black finish line). Once the ant stopped, it was held still for 10 seconds while a music subroutine played 10 seconds of the Jeopardy theme music. Lastly, a series of timed commands were added in parallel to another music subroutine totaling 30 seconds, during which the ant presented its victory dance featuring the moonwalk.

Solution Discussion

We are proud to say that our ant ran the race the way it was planned to do so. It went around the racetrack twice and presented its victory dance without fail. However, it was not the fastest amongst its competitors, which is the one point that must be worked on.

Planned Design Changes

The ant was designed with a pulley system that moved its legs. The pulley system was ultimately attached to the driving wheels, which can have a fatal result if the wheels were damaged from the forces acting on it by the pulleys. For this reason, the design should be changed by attaching the pulley system to another rotating device that will not hinder the driving wheels' performance. This may also help in achieving a faster traveling ant since the hindering forces due to the pulleys will be removed from the driving wheels.

Challenge #2 Approach

Summary

The challenge of controlling an ant made of Legos to travel around a defined path to find food and take it back home was approached by programming a control system using Robolab. The ant was designed so that the front wheels were attached to a motor via a series of gears that steered the ant. Specifically, the ant was programmed so that it was capable of following a random path, picking up "food" (a metal plate, which was located on a wood block situated at the end of the curved track), triggering a touch sensor, playing a song of joy, and returning to the point of origin along a second, straighter path. The path consisted of three different colored tapes: black, blue, and yellow. Each tape provided a different reflectance value to a light sensor carried on the front of the ant, which played a significant role in the program for keeping the ant on the defined path.

Design Approach

We devised a new design featuring rack-and-pinion steering for this competition. One motor was geared-down and used to turn the front wheel, which steered the ant. The second motor provided translational velocity by turning one rear, center wheel that propelled the ant forward and backward. Two smaller, rear side wheels (similar to training wheels on a bicycle) were used to stabilize the ant. The side wheels were raised slightly to allow it to turn freely as necessary.

The initial design for this challenge used two wheels in back that were both driven by the rear motor. This original design also featured rack-and-pinion steering, but proved to be a poor design for this challenge due of the resistance created by the rear wheels during turns. When turning, the wheels continued to rotate at the same speed despite the turn radius being wider for the wheel on the outside of the turn. The drag provided unacceptably inconsistent turn angles, which caused the ant to be unable to consistently complete the course.

Solution Description

The ant stays on the track by means of a light sensor. Through a series of programmed control commands (see Appendix 2 for the program used to meet Challenge 2), the ant turns right when the sensor reads a reflectance value less than a threshold of 27. The black tape offers a reflectance of around 25. The ant turns left when the reflectance is greater than a threshold of 38 (the reflectance of the yellow tape). The two motors run in a successive motion, never simultaneously, keeping the ant from driving off the track while the front wheel turns.

The pressure sensor is triggered when the ant runs into the wooden block holding the food. Triggering the pressure sensor sends the ant's program out of the turn-drive loop and initiates a song along with a short series of defined motions. The pressure sensor is located inboard and is pressed by a rod that extends out to the front of the ant. A wheel located at the anterior tip of the rod allows the rod to be moved from a wide range of contact angles. Also located on the front of the ant is a flat magnet (approximately 3" x 2") used to grab the food. The magnet is attached with double-sided adhesive tape to a swiveling arm. The swiveling motion of the arm allows for flush contact with the food at varying angles.

Upon completion of the program, trial and error was used to set up the timing and degree of turn for the front wheel with respect to the drive power of the rear wheel. The extreme case of making sharp turns was considered. A balance of turn angle, duration of the front wheel pulses of a set duration, and power for the rear wheel was determined. Each turn of the wheel is followed by a forward pulse of the rear wheel, projecting the ant forward in minute increments. The front wheel continues to turn as long as it is on either the black or yellow tape. As the front wheel becomes more perpendicular to the ant (due to successive turns), the rear wheel must overcome more friction to push the ant through the turn. This is the extreme case that limited the speed of the ant and degree of turn.

Solution Discussion

Aardvark had moving legs for Challenge 1. The legs created a tremendous amount of strain on the motor moving the legs. It was decided that for Challenge 2 the moving legs would be scrapped. The immobile legs were added to the final Challenge 2 design, as was Aardvark's driver and his "A" flag (characteristic of Design 1). Also slight modifications were made to the body of the ant for structural improvements.

The program for this Challenge was very different than that for the first Challenge, due in part to the updated steering mechanism. As with Challenge 1, the ant played a fanfare before starting the run. While-loops for the sensor were used, as were light sensor-based nested branches. Task splits were used to play the Oscar Meyer bologna theme while the ant traveled home with his food (a bologna sandwich).

The Challenge 2 design worked reasonably well. However, two improvements would have made Challenge 2 even more successful. First, upon getting to the food, Aardvark picked-up the food, backed-up, and played a short refrain from Beethoven's Ode to Joy. This was costly time-wise and should have been removed for the initial, successful run. The second improvement would have been to make the back-up motion, and turn after reaching the food more robust. While the design worked perfectly on the test track, the entry angle into the stretch before the food was longer on the Challenge track. The ant reached the food at an unexpected angle during the Challenge, and when backing up (with the front wheel still angled), the ant turned away from the homestretch. As a result, the predetermined left turn and forward motion did not successfully turn the ant into the track that lead home. Two of three runs were unsuccessful. A more robust program could be designed to allow the ant to traverse the backup under varying conditions and track angles.

Planned Design Changes

At this point, we did not anticipate making major changes for Challenge 3, but planned to act accordingly when the details for the challenge were provided. Depending on the demands the ant may need to overcome for the next challenge, some structural modifications may be needed.

Challenge #3 Approach

Summary

The challenge of controlling an ant made of Legos to travel in a defined area, searching for LEDs for five minutes, and then searching for a bright heat lamp and maintaining a constant temperature for five minutes was approached by programming a closed-loop control system using Robolab. The ant was designed so that the two large front wheels were each attached to a motor, and one single small wheel was placed centered in the back for support (which was not attached to a motor). The front portion of the ant had a crossbar made of Legos that was there to be used for when the ant ran into an object. The rod was spring-loaded and was placed in front of two pressure sensors, one placed on each lateral side of the ant, which were used to redirect the ant when it ran into an object. A light sensor was placed in the front of the ant at a height that was the exact height of the LEDs and also from where it could sense the heat lamp.

Design Approach

The design that was created came about through testing many different possibilities. The ant was created so that it could run into walls and other obstacles in its path and not get stuck or fall apart. The front crossbar was made to be longer than the actual body of the ant in order to make sure the ant would avoid getting stuck in any situation. Two pressure sensors were evenly spaced behind the crossbar to make sure that if the rod was hit on the extreme ends that the pressure sensor would activate the program and back the ant up and go a different direction. This worked much more efficiently than using a single pressure sensor that was placed in the middle of the crossbar. The rear wheel was able to slide easily which made turning much easier. The design that was created was the most functional in the fact that it would not get stuck, turned well, and was very durable. All of these attributes were needed in able to make the ant successfully find the most LEDs in five minutes ant then find the heating lamp.

Solution Description

To achieve the goals of knocking over the most LEDs in five minutes and accurately maintaining a temperature for five minutes, a closed-loop control system was programmed (see Appendix 3 for the program used to meet Challenge 3). The ant was initially attempting to find the LEDs by driving away from darkness. The program did not work because the light given off by each individual LED was not enough for the light sensor to get a large change in light, unless it was directly in front of the light. Therefore, a random pattern was created that was designed to allow the ant to travel around the entire given area in a five minute time period, this would optimize the ability for the ant to knock over the most LEDs. After five minutes the program was designed to switch over to allow the ant to search for the bright light of the heating lamp, and stop once it was directly under the light based on the brightness reading under the light. The ant then would stay under the light until it reached a temperature greater than 27ºC it would then back out of the light, and then once it dropped below 27ºC it pulled back into the light. This maintained the temperature at a constant value.

Solution Discussion

The program worked with varied success. Initially it was attempted to search for each LED and this was not effective due to the light given off by each light. The random pattern was effective, but did not ensure that all the LEDs would be found. The other portion that caused problems was that the pressure sensor was programmed so that it would work no matter what portion of the program was running. If the pressure sensor was pressed, the ant would jump out of the random movement loop and back up and turn. This caused a problem in stopping the random search after five minutes and starting the search for the heat lamp. Every time the pressure sensor was pressed it would reset the timer. Once the timer stopped the random first search the second part of the program was initiated. This program searched for the heat lamp by rotating until it read a light value greater than 30. The ant would then move forward searching for light, each time it would get darker the ant would turn until it found a lighter area, which was represented by a higher value. Once it finally reached the heat lamp it would read a value and stop. Overall this took some time to reach the lamp because of objects in the path to the lamp, which caused the pressure sensor to turn the object right. The program seemed to be the most direct and successful technique to achieve the goals of the competition.

Planned Design Changes

There will be no further changes made since this was the last competition. There could be improvements to the design, but they would vary on the given competitions.

Other Methods Employed

Challenge 1

Other methods that were attempted to meet challenge 1 by other groups of the class included:

1. Steering the ant so that it makes turns around the corners of the diamond. However, the variability of friction on the floor greatly decreased the turning consistency. As an attempt to counteract the problem, plastic wheels without rubber were used to decrease friction and allow for easier turning.

2. Different sized wheels were placed on the sides of the ant. The small wheel was placed on the side of the wheel facing the diamond, while the larger wheel was on the other side. This allowed the ant to travel around in a circular path with a defined radius.

Challenge 2

Other groups attempted challenge 2 by using different motors for each driving wheel and then varying the power of the wheels in order to turn the ant. This was efficient and successful when the ant traveled at high powers. However, at low motor powers, the ant was not time-efficient.

Challenge 3

Most other groups also took the random-path approach in meeting this challenge. However, every group's "random-path" varied due to the variations in how these wandering motions were programmed. Some groups were successful in finding seven out of the eight LEDs while others were successful in finding about half the LEDs. This variability occurred due to the degree of wandering in the ant's random-path. The more the ant wandered around the room, the more chance of finding more LEDs.

One notable difference in some of the other groups was the use of a rotational sensor to bring the ant back and change its course once it hit an obstacle in its way. Once and obstacle was hit, the rotational sensor would activate a series of commands in the program to follow a different path. This is analogous to how we used a pressure sensor to achieve the same goal of changing the ant's course when it ran into an object.

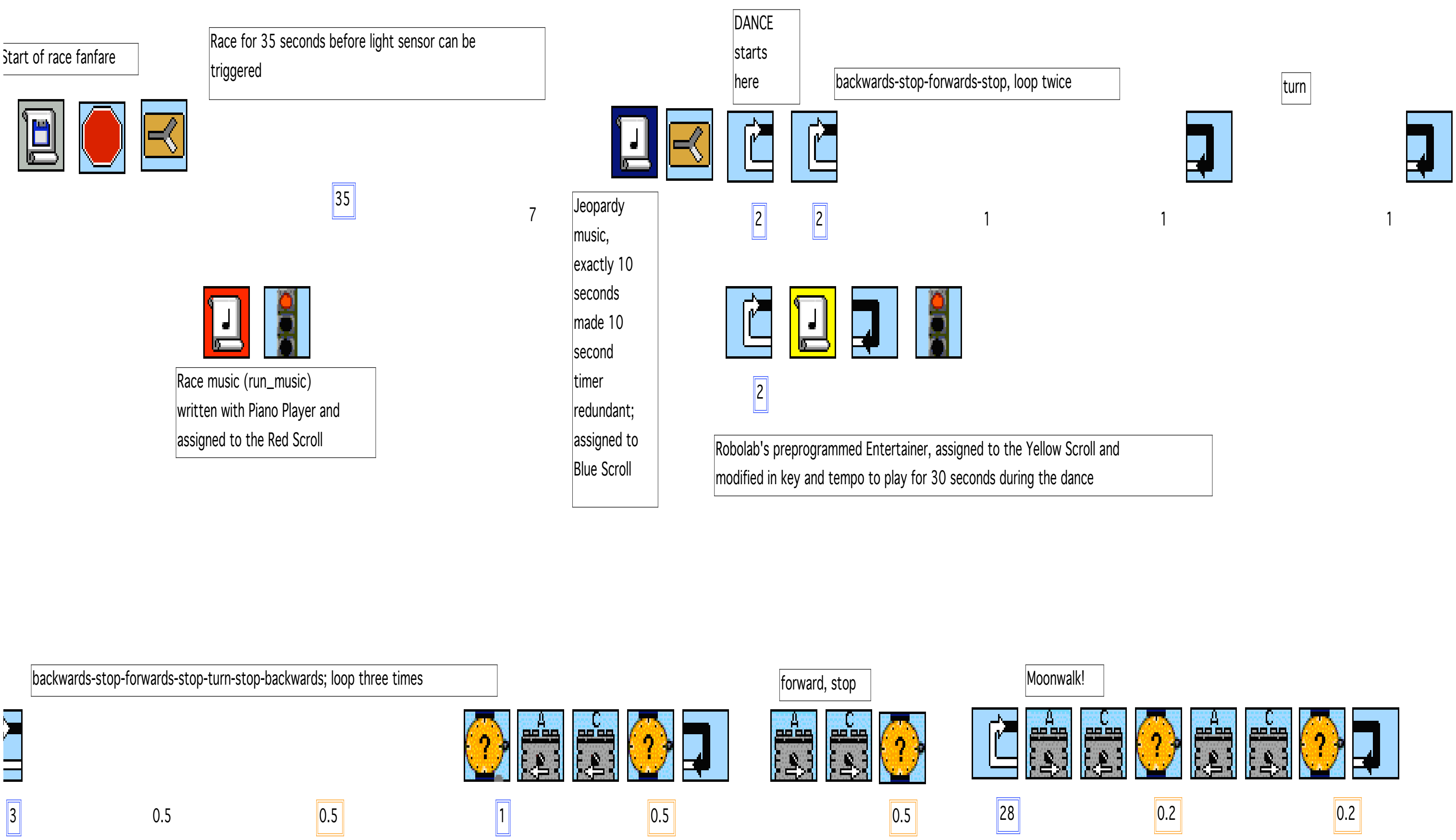
Conclusion

This project and its various challenges provided means of attaining a sound engineering knowledge of control systems. We were successfully able to employ control systems in meeting three challenges that consisted of designing and programming a robotic ant to perform defined tasks. Additionally, this project brought us to the realization that simple things such as Legos could be turned into complex engineering tools using programming software and hardware.

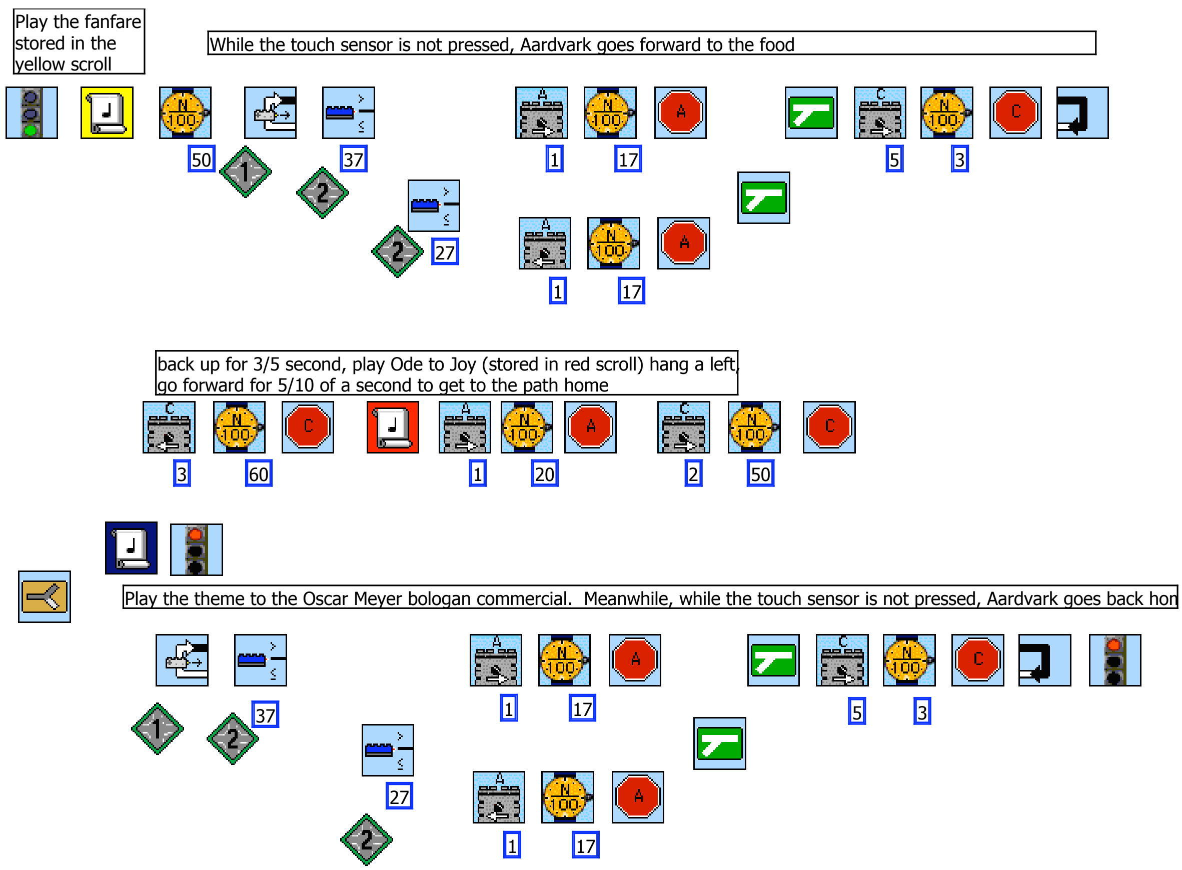
References

Khoo, Michael. 2000. Physiological Controls Systems: Analysis, Simulation, and Estimation. New York: IEEE Press.

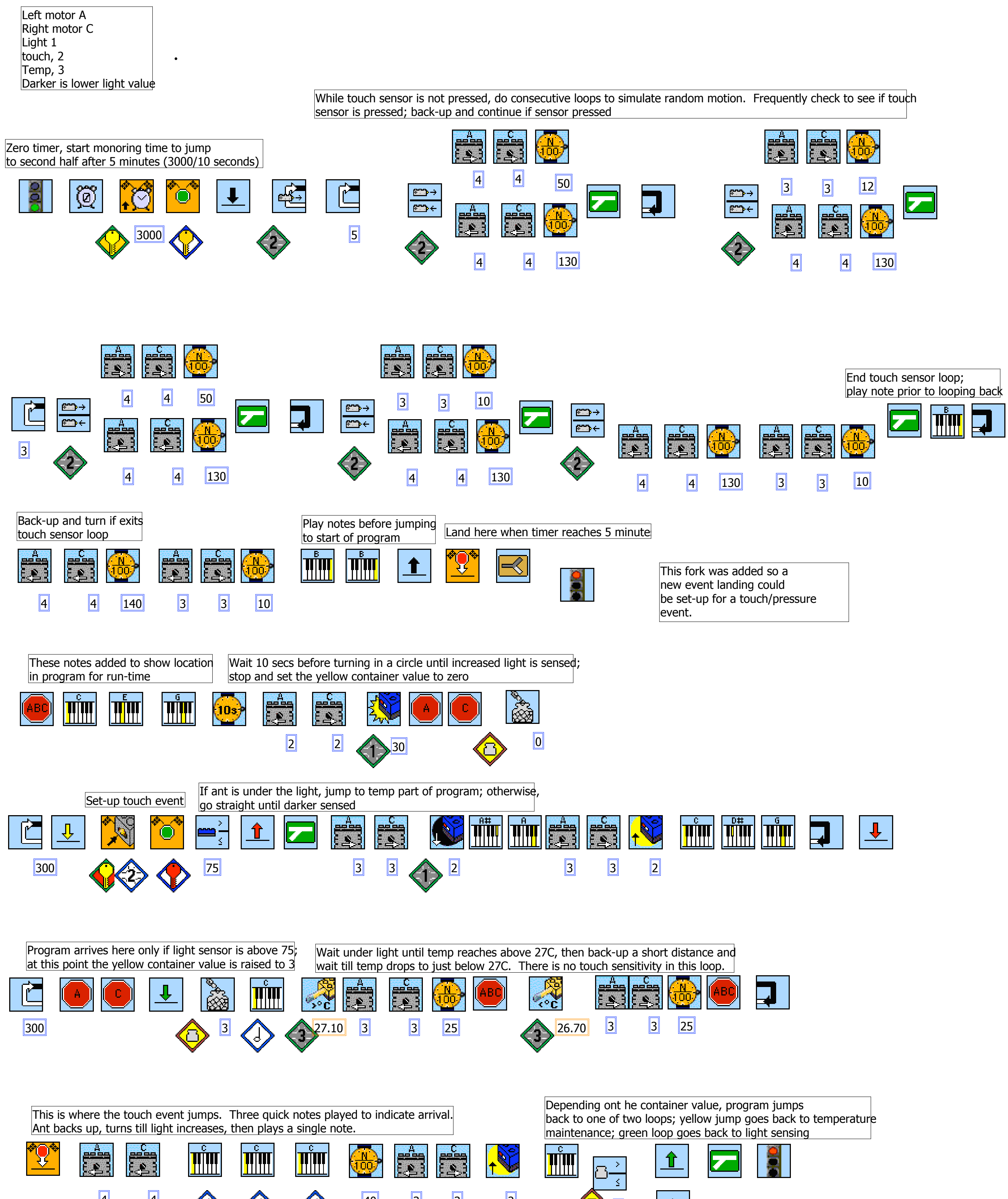
Appendix 1: Program used to meet Challenge 1



Appendix 2: Program used to meet Challenge 2



Appendix 3: Program used to meet Challenge 3



PAUL SCHREUDERS
Paul Schreuders is an Assistant Professor of Biological Resources Engineering at the University of Maryland. He teaches Biological Systems Controls and Biological Responses to Environmental Stimuli to seniors and has developed a collection of fast food toys that is the envy of his students. His research interests include cryopreservation and the structure and function of bacterial biofilms.

NAZ AZADI, LAURI BENNETT, SABA CHOUDHARY, JASON CONGDON, MIKE RATINO
Naz Azadi, Lauri Bennett, Saba Choudhary, Jason Congdon, Mike Ratino are undergraduate students in the Department of Biological Resources Engineering at the University of Maryland. They were one of the teams of students that developed the first iteration of the Ant Projects.