Robotic Maze-Mapping as a Junior-Level Design Project

Olivia Gustafson, B.S. '14, M.Eng. '15

Intro:

The goal of this design project is to create and set up a course project (CP) for the required junior class ECE 3400 Practice and Design that draws upon skills from the three required sophomore-level core courses (analog circuits, signal processing, and digital logic). The CP must be organized so that it has appropriate difficulty for junior students. Secondary goals are for the CP to have a heavy teamwork component as well as an impressive demo. The chosen CP topic is a robotic maze-mapping scenario, where a robot is let loose in a maze and wanders around to create a cohesive map of the entire area.

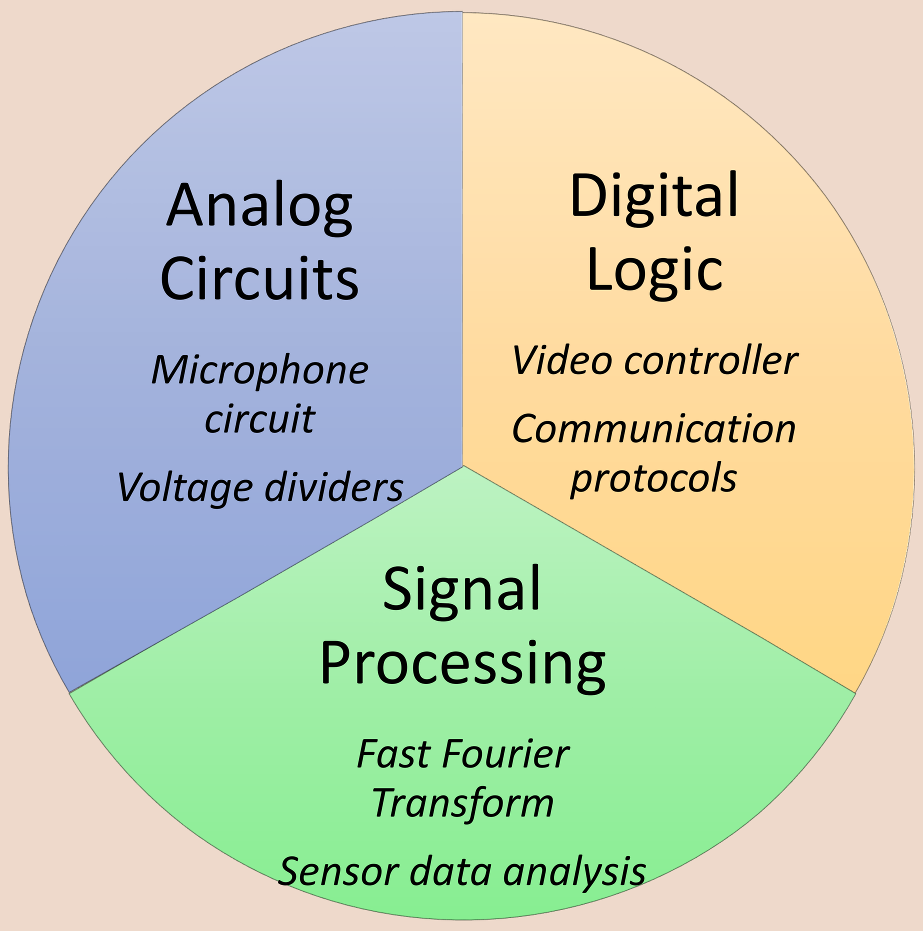
Project Description:

The students are required to create a maze-mapping robot that transmits its information wirelessly to a video controller that shows the maze as it is being mapped.

The robot's Arduino waits until it hears a buzzer of 660 Hz go off, signaling the start of the race. It then uses line sensors to follow the grid of black electrical tape to help with remembering location and traveling in straight lines. Using infrared distance sensors, the robot senses where walls are and sends that information wirelessly to another Arduino. When it is done exploring the maze, the robot stops.

The video station's Arduino receives the wireless information from the robot and sends it to an FPGA. The FPGA then converts that information into a VGA signal and sends the signal to a computer monitor. As the robot sends new information, the maze on the monitor is updated. This allows the viewer to know what the robot is seeing in real-time.

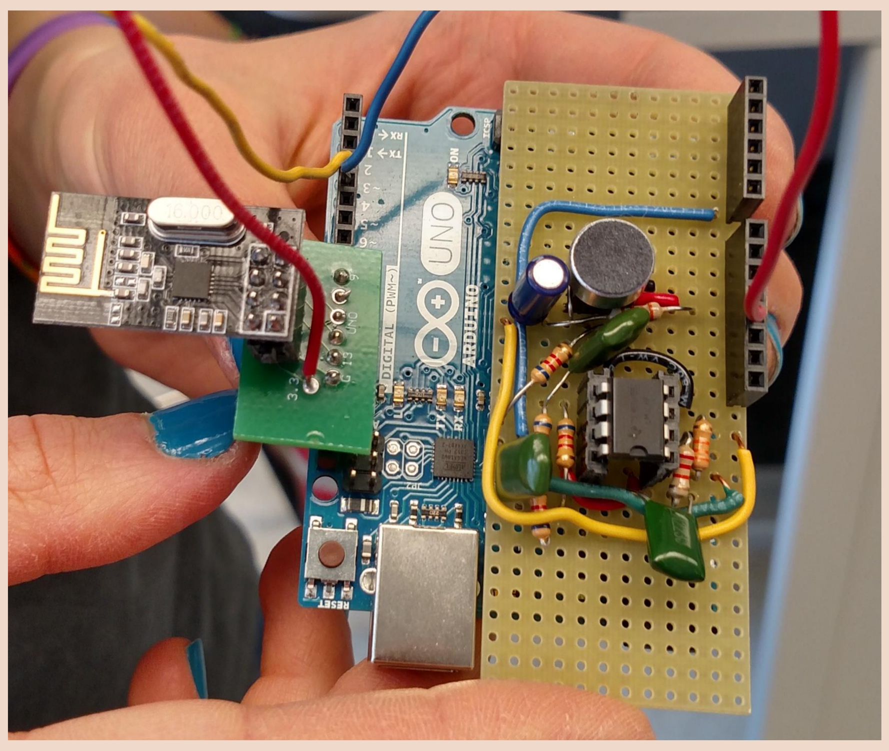
Project Breadth:



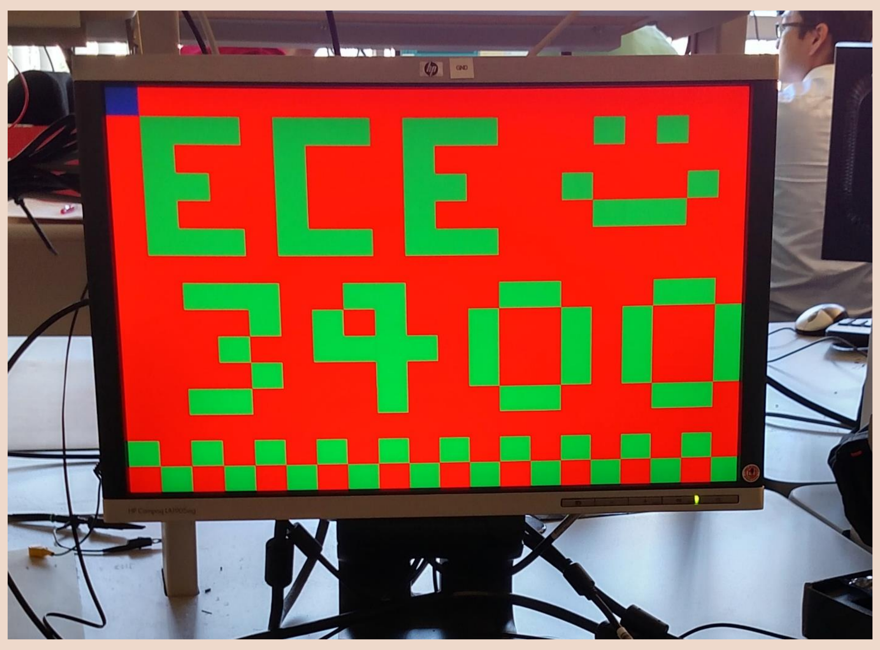
Project Setup:

For the first two-thirds of the semester, the students completed four structured labs designed to give them the building blocks for their robot. After Spring Break, they used their lab time to complete their robot and integrate the sub-systems they created earlier.

The **first lab** acquaints the students with the Arduino Uno, the microcontroller used for the robot and video station.



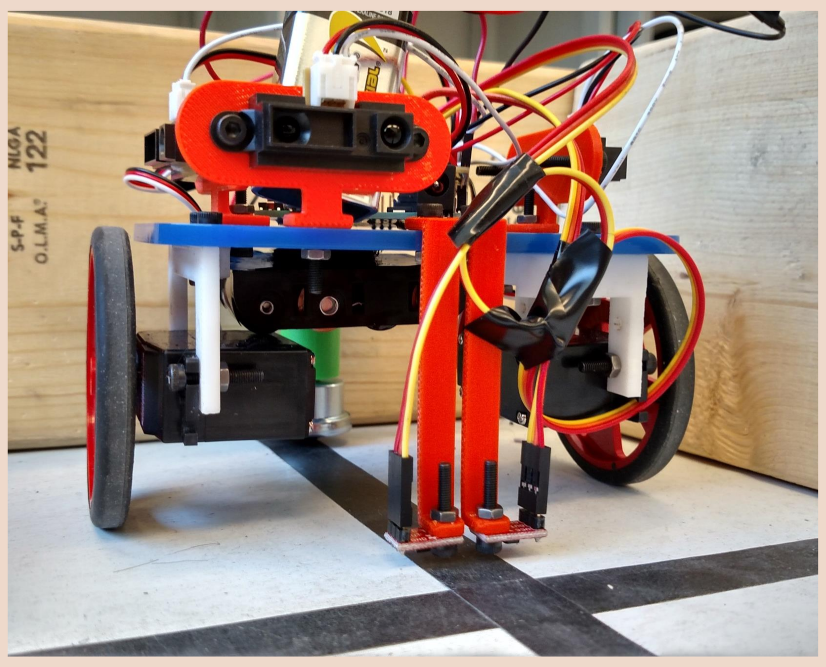
In the **second lab,** the students create the microphone circuit needed to detect the 600Hz buzzer. The circuit filters out noise, and the Arduino performs an FFT to determine the frequency heard.



The **third lab** focused on creating the video station. An FPGA converts a maze layout into a VGA signal to display on a monitor, and an Arduino sends the FPGA the maze information to display.

The **final lab** connects the two Arduinos together via wireless radios. The students determine what information to send and how to package it; then they implement their design.

Dagmar The DemoBot



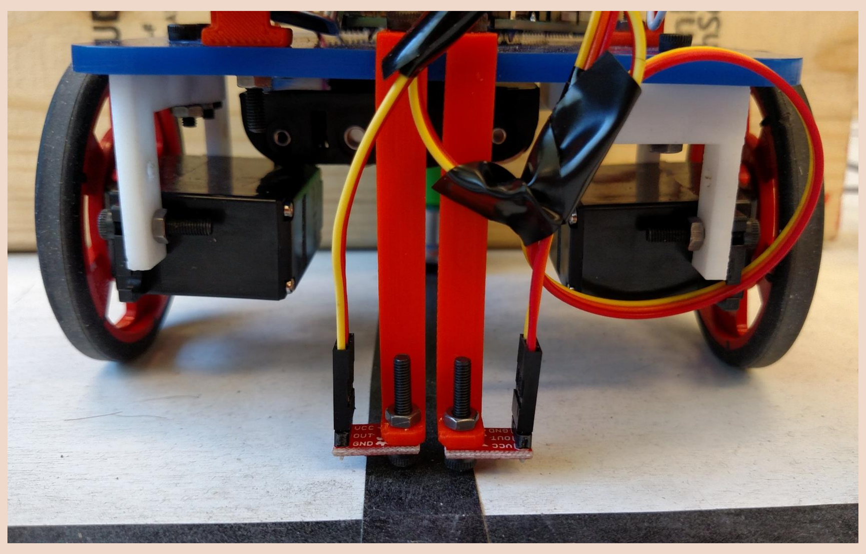
Navigation

Dagmar navigates by always following the wall to her right. Once she gets back to where she started, she checks to see if any spots in the maze haven't been mapped, then travels to the Nearest unexplored area. When Dagmar has completed navigating the maze, she stops moving around.

Wall-sensing

Dagmar has three distance sensors – one on the front and one on either side. When Dagmar reaches an intersection, she reads her distance sensors to "see" where the walls are.

Line-following

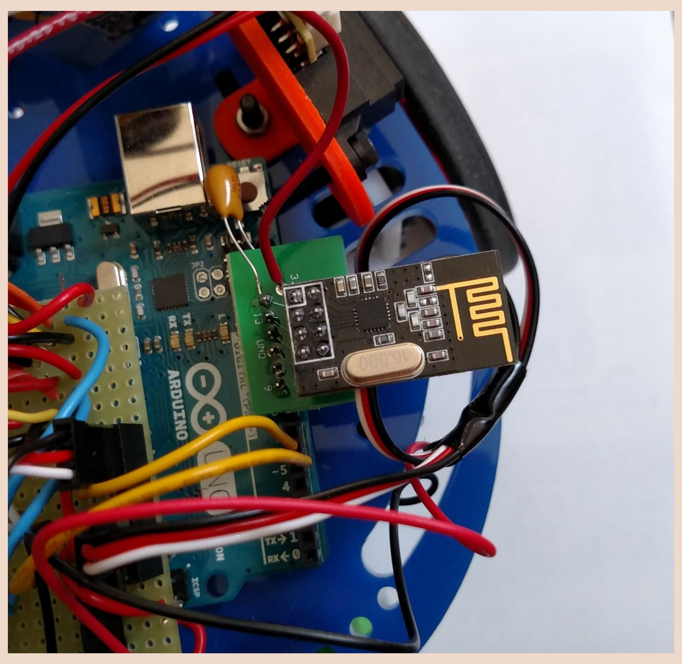


There are two line sensors (grayscale sensors) underneath the front of Dagmar's chassis. They are lined up so they straddle the black lines of electrical tape. If Dagmar drifts to one side of the line, one of the sensors hovers over the line. The outputs of the sensors are used to control the speed of each of Dagmar's wheels using a PID controller.

When Dagmar reaches an intersection, she knows because both of her line sensors read that they are on top of a line. Since they are spaced to straddle the line, this only occurs when she reaches an intersection.

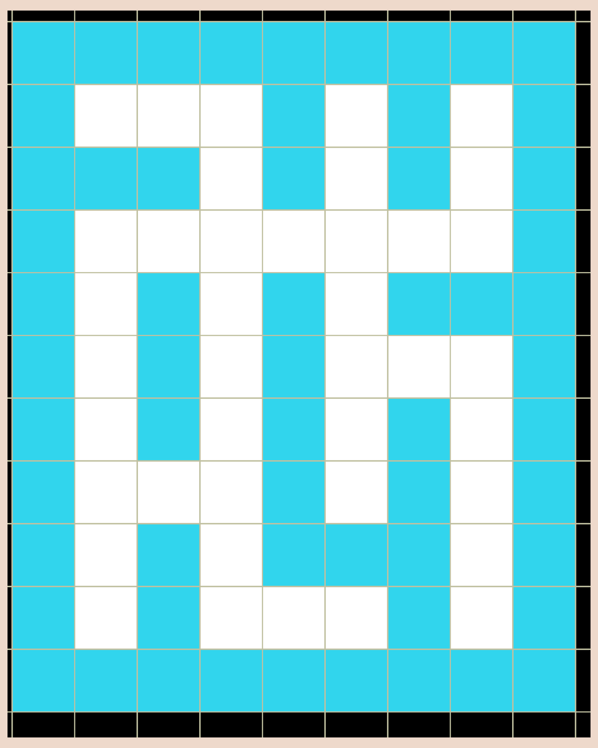
Radio Communication

Every time Dagmar reaches an intersection, she looks for walls, updates her internal knowledge of the maze, and sends that entire maze representation to the other Arduino at the video controller station. This takes four whole packets (with a maximum payload of 32 bytes).



Arduino-FPGA Communication

The maze information sent to the Arduino at the video controller station needs to be relayed to the connected FPGA to output a visual representation on a computer monitor. To do this, the Arduino needs to communicate via wires to the FPGA. In Dagmar's setup, this is done with soft-SPI, as the Nordic RF24 radios use the regular SPI pins on the Arduino.



Video Generation

The students were given a VGA driver that simply asks for the RGB pixel color of a given pixel. In Dagmar's setup, the maze is saved as a set of 99 blocks (7x11). Each block can be either an open space, a wall, unexplored, or unexplorable (blocked off). Therefore, each block can be encoded by 2 bits and then expanded into a full 8-bit RGB value. The photo to the left shows an example of what the monitor might look like after Dagmar maps an entire maze.

Microphone Circuit

Currently, Dagmar's microphone circuit is on a shelf in lab. It will hopefully be soldered to her soon. Testing the circuit in isolation was a success.