

ApalaPramanik Lab1 CSCE 892

Apala Pramanik

January 2024

1 Lab 1: First Experience with PlumGeek Ringo

Setting up the PlumGeek Ringo was a fairly simple process. All the steps were mentioned in the website documentation. I started with installing Arduino ide, followed by installing the Ringo libraries. The download link was available on the website and needs to be unzipped and added to the Arduino libraries. Further, as instructed in the lab assignment, I went ahead and installed FreeRTOS and included that in Arduino libraries. Once I was done with that, I went ahead and modified the Ringo base sketch (Ringo_Base_Sketch_Rev06_02.ino) with some insights from the /examples/Blink_AnalogRead.ino file to code some functionalities in Ringo.

I implemented a basic functionality on Ringo. The Arduino code defines a program that runs on a system using the FreeRTOS (Real-Time Operating System) library.

The setup function initializes serial communication and creates two tasks, “*TaskChirp()*” and “*TaskMove()*”, each associated with specific functionalities of the Ringo robot. “*TaskChirp()*” is defined to execute actions such as playing a boredom sound (PlayBoredom), and displaying a random color on the LEDs (eyes of ringo). After performing these actions, it introduces a delay of 2000 milliseconds. “*TaskMove()*” is defined to run both the motors at a speed of 100 units each, display a random colored LED (eyes), and play an excited sound (PlayExcited).

The main loop function is left empty, as the multitasking functionality is handled by the FreeRTOS tasks created in the setup function. These tasks allow the Ringo robot to perform multiple actions concurrently, enhancing its overall functionality and responsiveness.

Here is the link to the video: Watch Ringo being Moody!