ENGR 325 Final Lab Design Project

Raspberry Pi Controller for Conveyor Belt Sensors

Robert Akinie, Heon Soo Park

ENGR 325-L

December 8, 2020

Project Overview

The overall project that this team is working on is our senior design project, which is the development of a self-sorting recycling system using a conveyor belt system and machine learning. This team is taking a part of our senior design project, and working on it as our lab project for ENGR 325, that is, double dipping.

The initial project proposal that this team tried to work on was on the control of the conveyor belt using a Raspberry Pi development kit.The team did not have much progress in this, due to these reasons: the conveyor belt required a large amount of voltage to run, 48 V, as compared to the Raspberry Pi’s 3.3 V maximum output voltage. As such, the Raspberry Pi board could essentially not have been able to run it. The team later acquired a power supply box, but its configuration required the research and installation of PLC software, and assembling, one which the team decided was beyond the scope of this project, given the short time. Another reason was the processor speed of the development kit, which would not have been efficient for running the system.

The team, therefore, decided to work with the independent sensors that were part of the conveyor belt system. The independent sensors would detect any material that has been introduced into the conveyor belt system, and have the system act on it based on such information. The feature that is new to our team would be controlling the sensors using a Raspberry Pi.

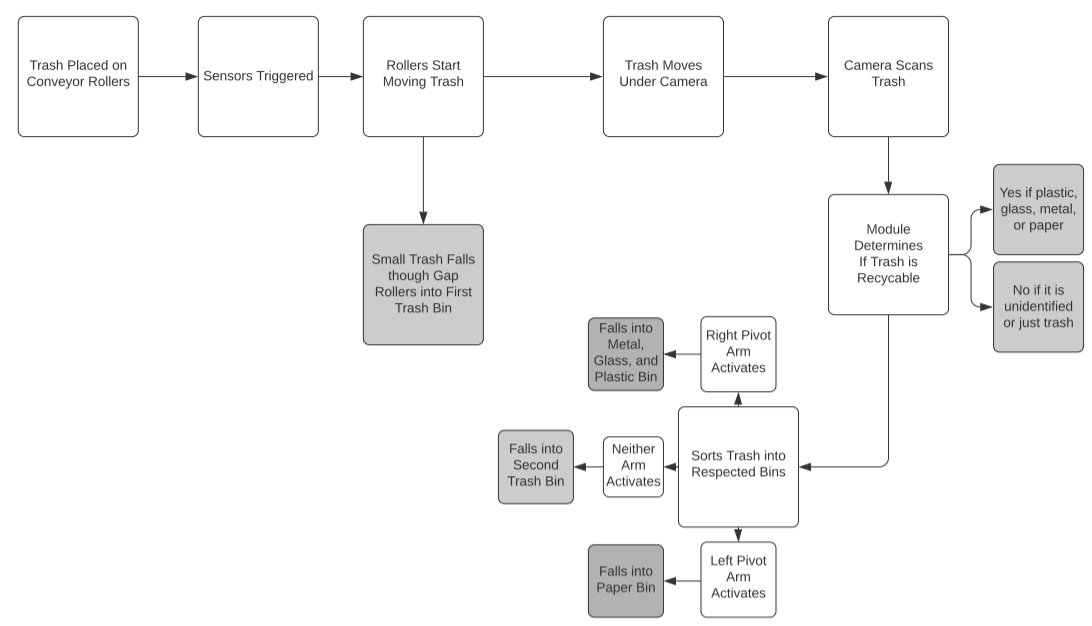
Objectives

The purpose of this lab project is to work on a development board with a new application. Working on this development board includes the utilization of some features that the team is new to which was the usage of GPIO pins in the Raspberry Pi. As such, the team decided to employ the use of sensors in our senior design project; sensors that would detect the presence of material that has to either be recycled or not, by the system.

System Architecture

Figure 1 below shows the total system architecture of the system. This project is focused on the sensor triggering portion of the system. The sensors in this project are photoelectric sensors, coupled with reflectors, whose job is to detect the presence of material in the system. The information is then signaled to the processing unit to start the conveyor belt rollers, as well as activate the camera sensors that would tell what kind of material has been introduced. The sensors work by emitting infrared light unto the reflectors, back to the sensor, in essence, completing a “circuit.” Any obstruction in the infrared light breaks the circuit, which indicates the presence of a material.

The system that the team set up was by connecting the Raspberry Pi board to the sensors via a breadboard and the board’s GPIO pins. The team also added LED functionality to the system, whose output would tell which of the sensors detected a material’s presence. Figures 2 and 3 below show the setup of the system associated with this project. The Raspberry Pi then controls the input and output of the system, utilizing code that can be found in the Appendix.

 Figure 1. System Architecture

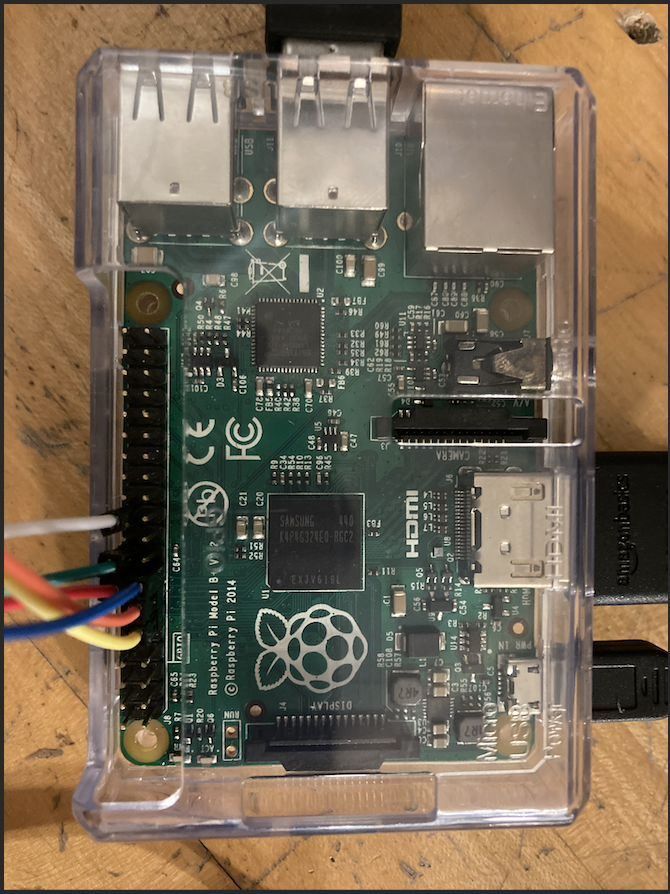
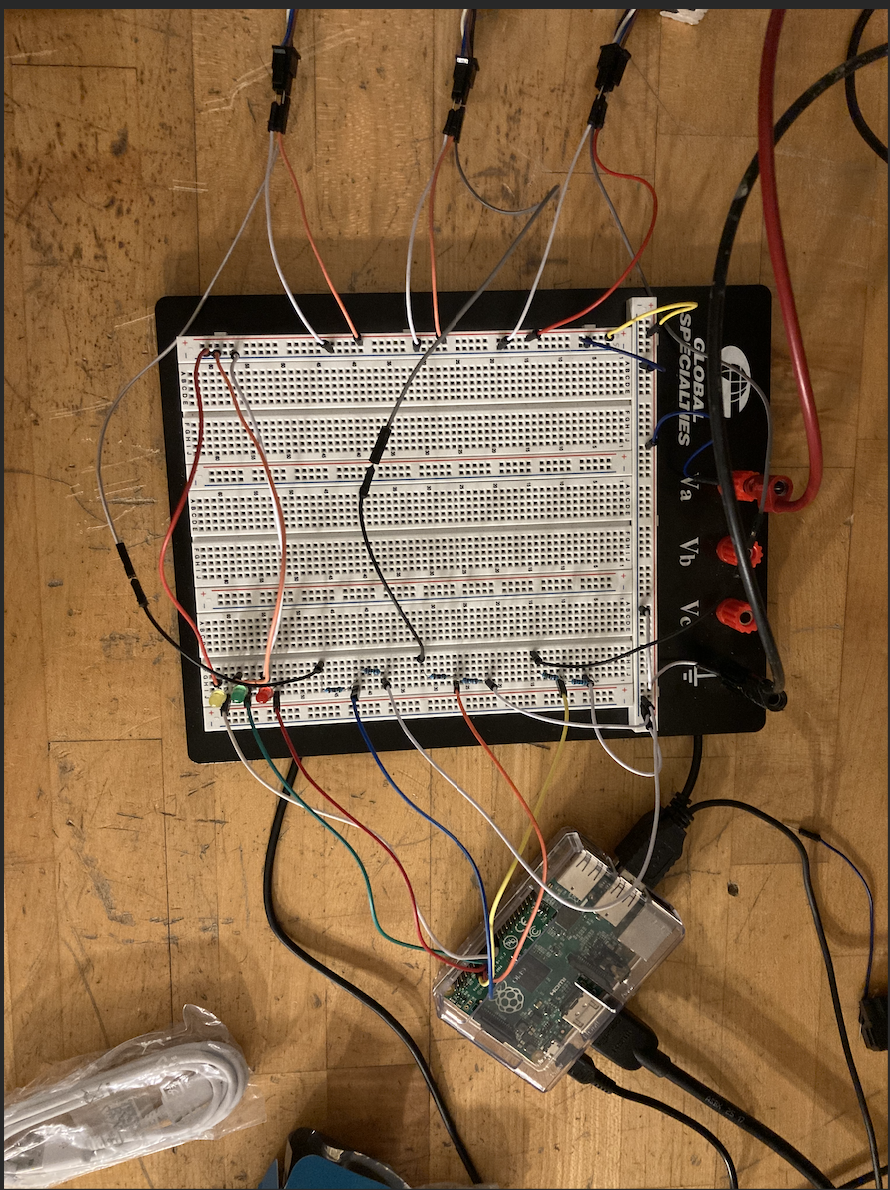


Figure 2 and 3 setup of the system

Equipment and Resources

The equipment used for this project are a Raspberry Pi Development Kit, Lab Monitor, Breadboard, Resistors, Photoelectric sensors, power supply, as well as other peripherals like the keyboard and mouse. Some of the resources used were the Python Development kit documentation, GPIO pin assignments, as well as python forums that described the workings of the GPIO pins.

Lessons learned while doing the project.

The lesson that we learned while doing this project was that working with industrial equipment required a lot of knowledge of the equipment itself but also the utilization of multiple different concepts that we learned in class. At the start, we struggled to get the sensors to turn on due to the lack of information given to us about the sensor itself. We had to dig through the web on what the different wires could be meant for and had to also take the sensor apart to see if there was any information we could gain from it. We were able to learn that the sensor required 24 volts to be turned on and that it was a PNP sensor, which meant that the sensor sourced it’s input voltage to the output wire unless the sensor detected something. After learning about our sensors and getting them turned on, we were able to pin them up into the raspberry pi, which was done by using a voltage divider in order to reduce the voltage to 3.3V (which is what the raspberry pi operates in).

Appendix

Python Code

import RPi.GPIO as GPIO

import time

while True:

GPIO.cleanup() #reset the pin states

GPIO.setmode(GPIO.BOARD)

#sensor 1

GPIO.setup(11,GPIO.IN) #input

GPIO.setup(22,GPIO.OUT) #output

#sensor 2

GPIO.setup(13,GPIO.IN) #input

GPIO.setup(18,GPIO.OUT) #output

#sensor 3

GPIO.setup(15,GPIO.IN) #input

GPIO.setup(16,GPIO.OUT) #output

#initial states of output pins to 0

GPIO.output(22,0) #sensor 1

GPIO.output(18,0) #sensor 2

GPIO.output(16,0) #sensor 3

#initialize sensor input pins

state1 = GPIO.input(11) #sensor 1

state2 = GPIO.input(13) #sensor 2

state3 = GPIO.input(15) #sensor 3

#sensor 1 to light up if sensor triggered

#logical statement for sensor triggers

if state1 == 0:

GPIO.output(22,1)

else:

GPIO.output(22,0)

#sensor 2

if state2 == 0:

GPIO.output(18,1)

else:

GPIO.output(18,0)

#sensor 3

if state3 == 0:

GPIO.output(16,1)

else:

GPIO.output(16,0)