Final Year Project Proposal Draft

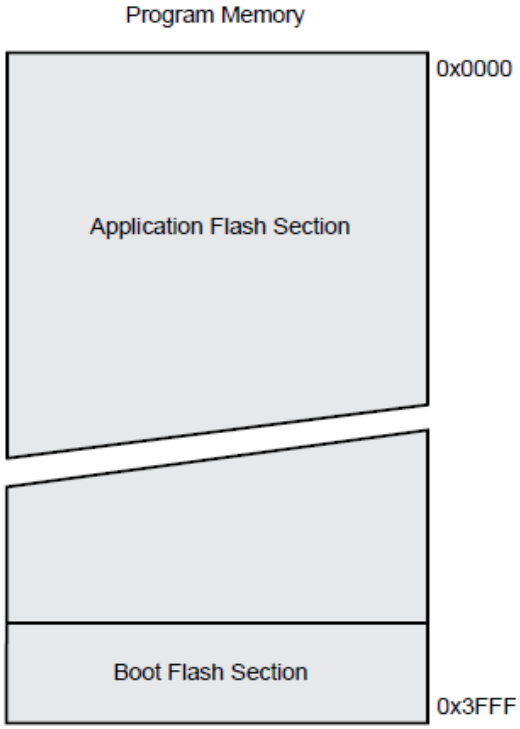
I. OTA

- **What is Over The Air Update/Upgrade/Programming?**

OTA (Over the Air) Programming is a process which allows devices to upgrade their firmware or software wirelessly without any physical access. It uses wireless technology like Wi-Fi, Bluetooth, GPRS or 4G/3G rather than wired serial communication. OTA is used to reprogram the devices like microcontrollers, cell phones, computers, set-top box etc in order to add some feature, resolve some bugs, etc.

- **How does it work?**

To understand the working, first let's get to the basics of how the codes are being stored in the non-volatile memory of the microcontroller.



Storage of Code in the Microcontroller

Every microcontroller divides the non-volatile(flash) memory into two sections namely-

1. Boot Flash Section
2. Application Flash Section

The Boot Flash section holds the **Bootloader.** A boot-loader is an application whose primary purpose is to allow a systems software to be updated without the use of specialized hardware such as a JTAG or ICSP programmer. In certain cases, it can also be the earliest point at which the integrity of an embedded system can be checked. The boot-loader manages the systems images. There are many different sizes and flavors to embedded boot-loaders. They can communicate over a variety of protocols such as USART, CAN, I2C, Ethernet, USB and the list goes on for as many protocols that exist. It is the first piece of code which runs when the device starts. During the bootloader execution, it will check if any incoming upgrade (to the program) is there from its intended ports. If so then it will go on to accept the program and flash it in the application memory section. Once done it will reset the device again. If no program is coming it will directly jump and start executing code in the Application flash section.

The Application Flash section contains the actual code which we want to run on the device.

So, a bootloader needs to be written, which can interface with the WIFI and receive the codes through it as well. A point to note here is that we don't want the trivial working of bootloaders (programming via Wired connection) to be taken up. We want to add the OTA functionality. This application comes in handy when software updates are required for the smart bin as it is not practical to remove the controller to update the code after installation.

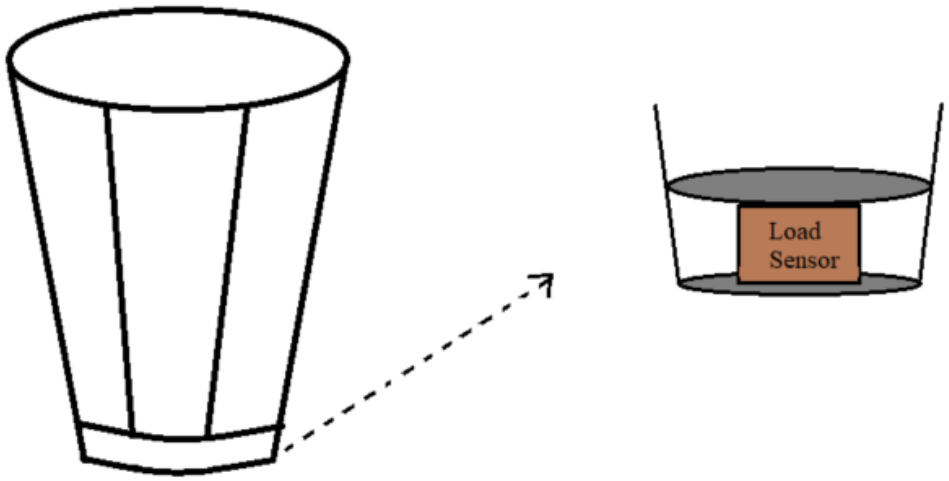
II. Sensors interface and measurements

Hardware set-up for measuring weight:

Requirements:

- Load cell
- Instrumentation amplifier

The weight sensor is to be made using a load cell and an instrumentation amplifier. This sensor will be fitted in the base of the dustbin. It will give its output to the microcontroller.



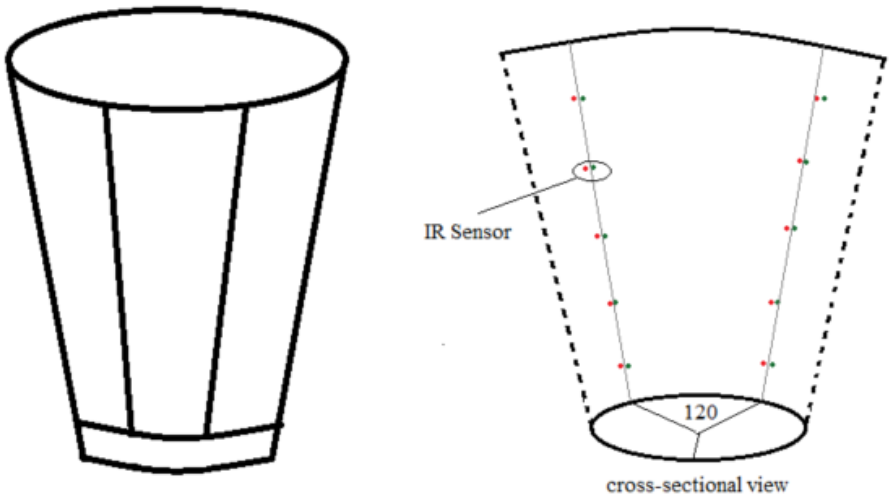
The set-up for the load cell will also depend upon the structure of the dustbin. For example, in the case of a big dustbin, one load cell will not be enough. In this case, the base will be divided into two or more sections with each section having its own load sensor. They'll then be averaged appropriately to get final estimate.

Hardware set-up for measuring height:

Requirements:

- IR transmitter-receiver array

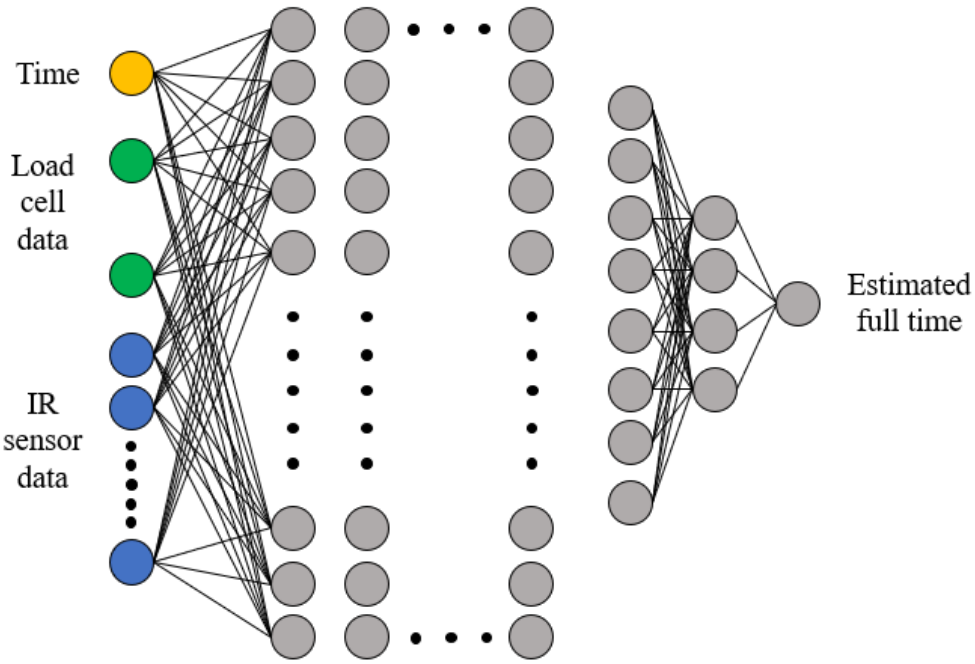
IR transmitter-receiver pair will be used as the proximity-sensor to estimate whether the dustbin is filled or not. These IR pairs will be distributed uniformly inside the dustbin to get an accurate result. The distribution of these IR sensors will be in the three lines from bottom to top. These lines are in 120 degrees with respect to the centre of the dustbin.



The output of these IR sensors will be further processed using advance data computing tool like machine learning to predict the result.

III. Adding intelligence to the system:

The smart bin contains a wide array of IR sensors to detect the amount that the dustbin has been filled. The load cell(s) give the weight of the waste inside the bin. Since the microcontroller is attached with this, the level and load can be measured with respect to time. This data can be used to train a self-supervised neural network to predict the rate of filling.



An approximate representation of the neural network architecture.

The data from these sensors can be input to the neural network as shown in the figure. Such a neural network also has the capability to learn on its own and adapt to different localities. For example, dustbin at region A can be shifted to another region B, where the filling pattern is different. Still the network will retrain itself automatically in a short amount of time to adapt to region B. The training can be done in two ways:

1. Using a online cloud platforms to deploy the neural network and integrate it with the dustbin using IoT.
2. Using the Arduino board for training and forward passes. The advantage of this method is it does not rely upon outside resources. However, deploying a neural network can become tedious. We'll try to implement this method.

The advantage of using a neural network for prediction instead of a conventional algorithm are:

1. **Robustness**. For example, if one of the IR sensor gets blocked due to waste materials stuck to it or gives erroneous results, conventional algorithms drastically fail. However, a neural network can adapt to this situation to still provide accurate results.
2. **Adaptability**. As mentioned before, when the dustbin is transferred to a different locality, it will still be capable of predicting their pattern without any changes to network.
3. **Self-learning**. Once the bin is deployed, the network starts to learn automatically from the data read by the sensors. It then stores the time series variations of the sensor data and uses the time whenever the trash is removed (when weight goes below a lower threshold) as its output. Using these input time series data and output data, the network automatically learns to analyze the pattern in data and provides accurate estimations.

Work split:

Assuming that next semester happens online, we plan to split the project into three parts and work on it individually.

1. OTA → Aditya Kumar Singh.
2. Sensor interfacing and measurements → Raj Kumar Bhagat
3. Adding intelligence to the system → Ruphan S