McMaster University

Software & Mechatronics Capstone

eBin

System Design

Anna Wei 400031943
Alan Yin 400007807
Ditong Liu 400009984
Huajie Zhu 001403438
Junni Pan 400024523
Zhihao Yang 400013899

Last compiled on March 13, 2020

Contents

List of Tables

List of Figures

1 Revisions

Date	Revision#	Authors	Comments
Jan.10 2020	Revision 0	Anna Wei Alan Yin Ditong Liu Huajie Zhu Junni Pan Zhihao Yang	Initial draft of system design
Mar.13 2020	Revision 1	Anna Wei Alan Yin Ditong Liu Huajie Zhu Junni Pan Zhihao Yang	

Table 1: eBin Table of Revision

2 Purpose

2.1 Project Overview

Waste segregation is legally required in most countries. Sorting or arranging how people throw away their garbage is an integral part of the whole recycling process since waste management companies need to sort everything out first before loading it into recycling machinery to start the recycling process. Effective segregation of wastes means fewer recyclable materials goes to landfills or be incinerated which helps the environment to achieve sustainable development. Most places in campus have different kinds of garbage bins. Students and stuffs throw away tons of garbage every day, but it is doubtful whether the garbage is disposed in the correct bin. Imaging when a class is over, students crowd out to catch the next class and they walk pass the garbage bins trying to throw away their garbage. The students should have enough time to find the corresponding bin and dispose it correctly. For the types of garbage bins on campus, there are mainly four types: organics, paper, bottles and cans and the other. Since organics only appear in certain dining area (cafeteria, kitchen, etc.) and it is especially hard to handle and clean, we do not consider sorting it. And Papers and bottles and cans can be combined as mixed recyclable. There leaves only two type of garbage, mixed recyclable and the other. These two are also the most commonly seen garbage sorting bins in most places. To provide a more convenient way for users to sort and dispose their garbage, we introduce our eBin. When user throw garbage into eBin, it will determine the type of garbage using image recognition and ideally, displace it on a screen so that the user will know which type the garbage belongs to. Once the type is recognized, it will dispose it into the corresponding sub-bins.

2.2 The Purpose of The Project

The main purpose of eBin is to provide schools and universities with a better alternative to existing garbage bins. Current problems with garbage bins (and garbage sorting in general) on campus include cost of maintenance, inconvenience to the user, and, most importantly, an inaccuracy in the garbage sorting itself with leads to an increased amount of time and resources required from waste management facilities to further sort the garbage. eBin aims to eliminate all the above problems through the use of its image recognition

technology to automatically sort garbage. Firstly, the automated garbage-sorting technique incorporated into the device will allow for the eBin to be more clean than traditional garbage bins. This means that eBin does not need to be cleaned as frequently as existing garbage bins, thus cutting back on maintenance cost. eBin also allows a much more convenient approach for garbage disposal to users – instead of having to figure out which bin to throw their garbage into, users now need only to place their piece of garbage on top of the eBin's sorting platform and image recognition will do the rest. This also leads to the final point – with the elimination of human error in the garbage sorting process, only a minimal amount of time and resources will be required from waste management facilities to sort garbage. There is even the possibility that through the use of eBin, garbage collected from these bins can completely bypass sorting altogether and proceed directly to waste treatment.

2.3 Naming Conventions and Terminology

Term	Description
eBin	eBin is our garbage sorting machine
sub-bin	sub-bins are the two small bins embedded in the large bin
SOC	SOC stand for system on chip

Table 2: Terminologies and corresponding descriptions

3 Project Scope

3.1 Overview

The system design document contains information about the breakdown of the components of the system. The document will include a context diagram relating to how our components behave, a component diagram that shows the detail of the interactions between components, lists of system variables and constants with interpretation and the general overview of system behaviour. Next, each component will be broken down and described in terms of input/out, behaviour, timing constraints/performance requirements and

initialization procedures. At the end, this document will detail normal operating states as well as undesired events of our system and how they will be handled.

3.2 Constraints

3.2.1 Development Constraints

The project is developed in two parts. One is software. The other is hardware. In the software, we use python to develop the garbage recognition. In the hardware, we use stepper motor to control the plate.

3.2.2 Operation Constraints

Both garbage recognition and the control of plate are operated by raspberry pi, which is a tiny desktop computer.

3.2.3 Schedule Constraints

The project must be finished before April on 2020, the development of process will follow the schedule of deliverable.

3.2.4 Budget Constraints

The maximum budget of our project is 750 Canadian dollars.

4 Context diagram

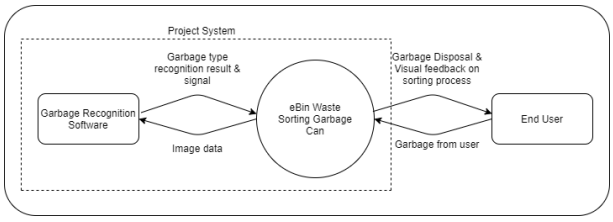

Figure 1: Context Diagram

5 Component Diagram

5.1 Software Component Diagram

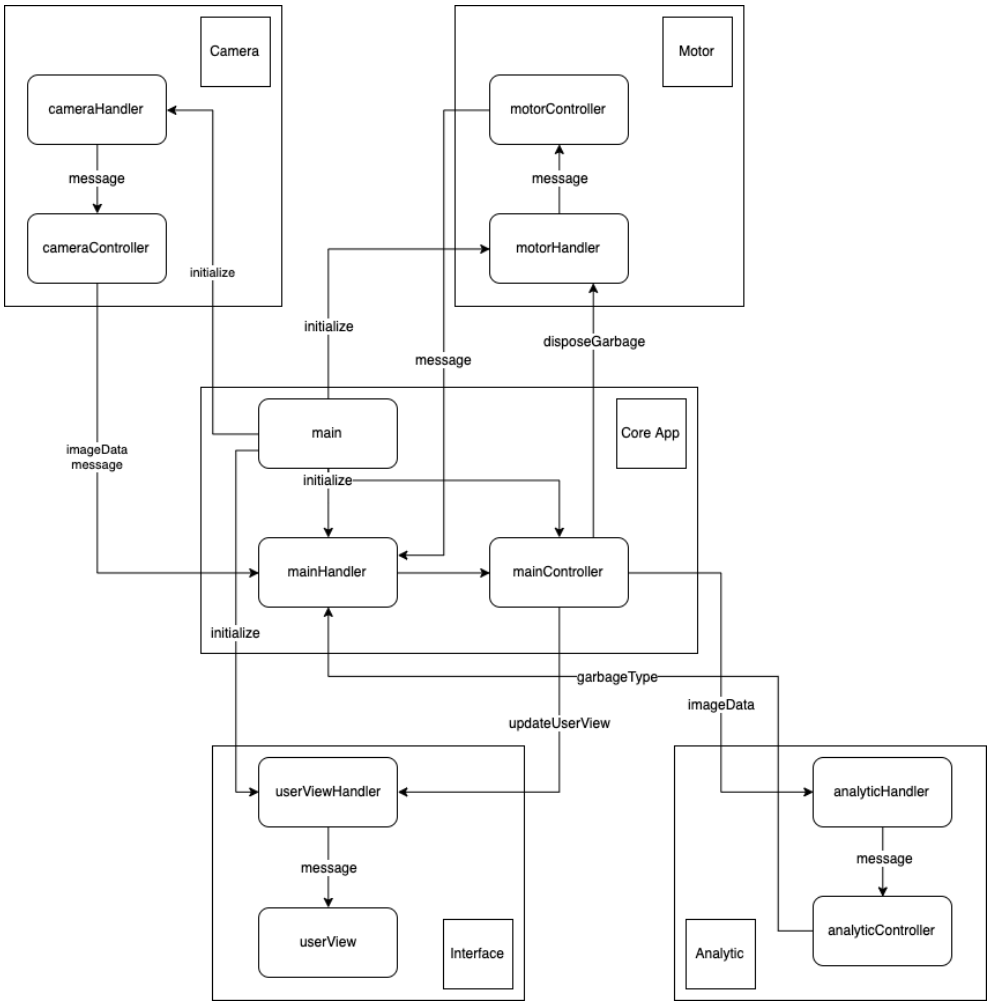

Figure 2: Software Component Diagram

5.2 Hardware Component Diagram

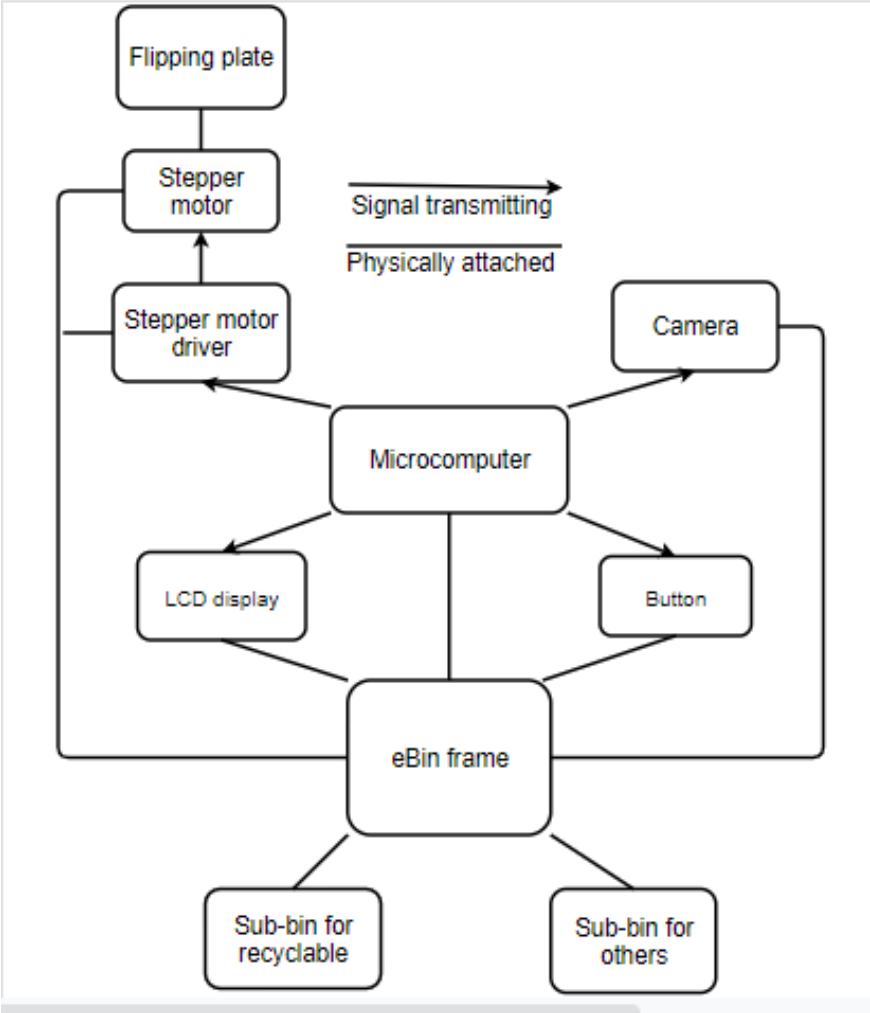

Figure 3: Hardware Component Diagram

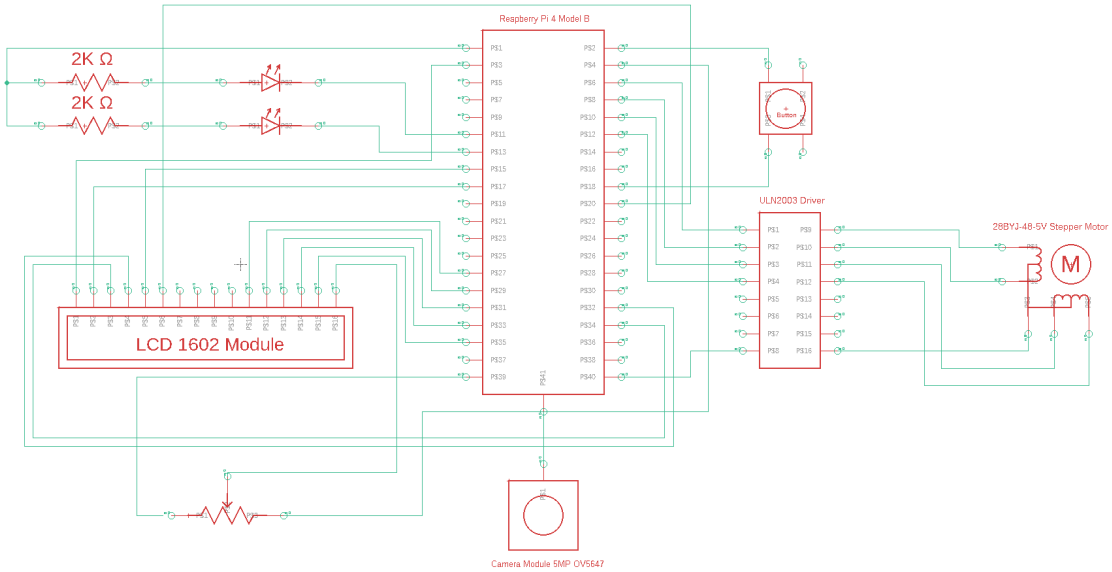

Figure 4: Electrical schematic

6 Monitored and controlled variables

6.1 Monitored Variables

6.2 Controlled Variables

7 Constants

- GARBAGE_PROCESS_TIME= 5000ms
  - How long it takes to identify garbage, display garbage type and dispose garbage into bin.
- GARBAGE_RECOGINTION_TIME= 3000ms
  - How long it takes to identify garbage.
- OBJECT_TYPE_DISPLAY_TIME= 2000ms

Name	Type	Range	Units	Physical Interpretation
object_detected	Boolean	[0, 1]	N/A	If object to recognize is detected
object_type	String	Various	N/A	Type of object with highest possibility based on the results of analysis
garbage_type	String	Recyclable or Others	N/A	Type of garbage based on object classification
output_percentage	Percent	[0, 100]	%	Estimated accuracy of result represented in percentage
processing_status	String	Various	N/A	The status of current processing stage

Table 3: Monitored Variables for eBin

Name	Type	Range	Units	Physical Interpretation
input_info	Image	N/A	N/A	The input image taken from built-in camera
hinge_default_degree	Degree	[0, 360]	Degree	The initial degree of hinge
hinge_rotation_direction	Boolean	[0, 1]	N/A	Desired hinge rotation direction
hinge_rotation_degree	Degree	[0, 180]	Degree	Desired hinge rotation degree

Table 4: Controlled Variables for eBin

  - How long it displays the object type.

- GARBAGE_TYPE_DISPLAY_TIME= 2000ms
  - How long it displays the garbage type.
- GARBAGE_DISPOSE_TIME= 2000ms
  - How long it takes to dispose garbage into bin.
- RESET_TIME= 2000ms
  - How long it takes to reset the system.
- GARBAGE_CLASSIFICATION_NUMBER= 2
  - How many types of identified garbage can dispose into bin.
- MAX_CONCURRENT_USERS= 1 user
  - How many concurrent users our ebin will support.
- TOTAL_BIN_CAPACITY= 40L
  - How much garbage can dispose into bin.
- SUB_BIN_CAPACITY= 20L
  - There are two sub bins in the whole bin. This is the capacity of every one sub bin.
- GARBAGE_PLATE_HOLDING_CAPACITY= 2KG
  - How much the garbage identification plate can hold when the garbages are put on it.

8 Behaviour overview

The behavior of eBin consists of four major steps: detecting, sorting, displaying result and dropping garbage. For the first step, after the image is collected, it is sent to the server with other information of the garbage. Then a well-trained machine learning model will determine the type of garbage based on the information, and the result will be displayed on a screen asking the user if the result is correct (this is a optimal function, might not be implemented at the end). The dropping action is finally performed by an actuator, dropping the garbage into the corresponding sub-bin.

9 Software Components

9.1 Camera Component

9.1.1 Camera Handler Component

- **Input/Output:** The camera handler component receives initialization call from main component and outputs capture signal to camera controller.
- **Behaviour Description:** The camera handler component should receive initialization call from main component and send monitoring instructions to camera controller component
- **Timing Constraints/Performance Requirements:** The camera handler component should be able to receive and send the message in real time
- **Initialization:** The camera handler component will be initialized by main component

9.1.2 Camera Controller Component

- **Input/Output:** The camera controller component will take the instructions from camera handler component as input, and take the image data as output

- **Behaviour Description:** The camera controller component should be able to capture the object image once the object was put on the garbage plate
- **Timing Constraints/Performance Requirements:** The camera controller component will be able to detect the object and capture a image immediately
- **Initialization:** The camera controller component will receive a message from camera handler component and start to monitor the garbage plate

9.2 Core App Component

9.2.1 Main Component

The main component starts the program and initializes all the handler components

9.2.2 Main Handler Component

- **Input/Output:** The inputs for main handler component include the image data and message from camera controller component, the message from motor controller component and the garbage type from analytic controller component. The outputs for main handler component include messages to main controller component
- **Behaviour Description:** The main handler component will receive image data from camera controller component and send it to main controller component. Also it will receive garbage type after analytic and dispose of the garbage
- **Timing Constraints/Performance Requirements:** The main handler component will receive data in real time and send instructions to the main controller component immediately
- **Initialization:** The main handler component will be initialized after receive the call from main component

9.2.3 Main Controller Component

- **Input/Output:** The main controller component will take the messages from main handler component as input. And the main controller component will send instructions to the corresponding handlers including user view handler, analytic handler and motor handler
- **Behaviour Description:** The main controller component will send updateUserView to the user view handler component, send imageData to analytic handler component and send disposeGarbage to motor handler component
- **Timing Constraints/Performance Requirements:** The main controller component will receive data in real time and send instructions to the corresponding handler component immediately
- **Initialization:** The main controller component will be initialized after receive the call from main component

9.3 Interface Component

9.3.1 User View Handler Component

- **Input/Output:** The user view Handler will receive the input from main Controller and output message to user view component
- **Behaviour Description:** The user view handler component should receive updateUserView from main controller component and send update instructions to user view component
- **Timing Constraints/Performance Requirements:** The user view handler component will be able to receive and send immediately
- **Initialization:** The user view handler component will be initialized by main component

9.3.2 User View Component

- **Input/Output:** The user view component will take the instructions from user view handler component as input, and display the garbage data as output

- **Behaviour Description:** The user view component should be able to display the object type and garbage type to user
- **Timing Constraints/Performance Requirements:** The user view component should able to display the object type and garbage type within 2000ms when it is initialized
- **Initialization:** The user view will display an initial interface

9.4 Motor Component

9.4.1 Motor Handler Component

- **Input/Output:** Motor handler takes dispose call with garbage type from main handler and sends function call with processed data to motor controller.
- **Behaviour Description:** Motor handler waits for function calls from main handler after initialization. It processes the garbage type into motor rotation parameters and calls motor controller component.
- **Timing Constraints/Performance Requirements:** Motor handler should not stop running to successfully receive function calls.
- **Initialization:** The motor handler component is initialized with direction and angle by main component to listen for calls.

9.4.2 Motor Controller Component

- **Input/Output:** Motor controller component takes function calls from motor handler, and sends rotation signal to motor.
- **Behaviour Description:** Motor controller generates hardware readable signals from parameters in function calls.
- **Timing Constraints/Performance Requirements:** Motor controller should listen to motor handler in real time to successfully receive signals.
- **Initialization:** Initialized by motor handler with initial direction and angle.

9.5 Analytic Component

9.5.1 Analytic Handler Component

- **Input/Output:** Analytic handler takes processed image data from main controller, and passes function call with image data to analytic controller.
- **Behaviour Description:** Analytic handler resizes the input image into 300*300 to match training dataset.
- **Timing Constraints/Performance Requirements:** Analytic handler should not stop running to successfully receive function calls.
- **Initialization:** Analytic handler is initialized by main component to listen to function calls.

9.5.2 Analytic Controller Component

- **Input/Output:** Analytic controller receives processed image from analytic handler, and returns calculated garbage type to main handler.
- **Behaviour Description:** Analytic controller runs image recognition mechanism and translates raw json output into the final calculated garbage type.
- **Timing Constraints/Performance Requirements:** The controller should return image recognition result within the limit specified in Software Requirement Specification document.

10 Hardware Component

- **Microcomputer** - Raspberry Pi 4 Model B with 1.5GHz 64-bit quad-core ARMv8 CPU (4GB RAM)
- **eBin frame**: The outer shell that contains and protects all other less robust hardware components
- **Sub-bins**: The two smaller bins that actually holding the garbage disposed, they are embedded at the bottom of the eBin frame

- **Flipping plate**: A plate placed in the middle of the bin that is drived by a stepper motor. It temporarily holds the garbage for the camera to take pictures and and is able to rotate in different angles
- **Shaft and coupling - URBEST RC Helicopter Round Rod 100mm x 5mm HSS Ground Shaft, ARQQ Aluminum Alloy Joint Connector, 5mm to 8mm Flexible Shaft Coupling**: Shaft and coupling which used to connect the motor and the flipping board.=
- **Raspberry Pi 4 Model B**: Raspberry Pi board that connects to all the electrical components. It receives pictures and processes the software algorithm on them and then sends signals to the stepper motor driver
- **Stepper motor - 28BYJ-48 Stepper Motor**: A stepper motor mounted on the back of the eBin frame that drives the flipping plate
- **Stepper motor driver - ULN2003 Driver Board**: Stepper motor driver board that connects stepper motor and raspberry pi
- **Camera - LABISTS Raspberry Pi Camera 1080P Video Camera Module 5MP OV5647 Sensor** : Camera module mounted on the side of the frame that takes picture of the garbage held on the plate and send the picture back to raspberry pi
- **LCD Screen - LCD 1602 Module**: An LCD screen outputs the result of the garbage type, as well as a confidence level of the final result. Based on the result shown on the screen, the user then can monitor if the garbage will be disposed into the correct bin or not.
- **Button**: A simple push button can capture user's input. It is pressed only when the user think the result is wrong.
- **LED**: 2 always-on LEDs are used to illuminate the garbage placed on the flipping plate.

11 Normal Operations

1. Once a user place the garbage on the plate, the garbage can be detected and a signal is transmitted to the microcomputer and start the process.

2. A photo is shot by a camera once the garbage is detected, and the photo is sent to the microcomputer which can decide the type of this garbage.

3. Then based on the type of garbage, a signal will be transmitted to the motor which controls the plate.

4. The motor will flip clockwise or counterclockwise, to make sure the garbage is finally drop into the corresponding bin.

12 Undesired Event Handling

1. In the case of power shortage, the system should have the ability to restore itself to the initial stage.

2. If the garbage recognition mechanism returns an ambiguous result, the system should treat the object as "Others".

3. To prevent the over weighted object, instructions should be provided on the interface.

4. The opening of eBin garbage can is limited to prevent oversized object.

13 System Components That Are Likely To Change

1. Number of bins used may be changed depending on how many types of garbage we want to sort, for now it is two – one for recyclable, one for the others.

2. Accuracy of image recognition may be improved once we have created the machine learning model, we will teach it to accurately distinguish most common garbage.

3. SoC may be changed based on this STRS, as some of the microcomputer is not capable of handling all the required process fast enough. For now, we are considering Raspberry Pi as our microcomputer, it may also be degraded to some other product for saving the coverall cost.

4. The power of hinge motor and the material of flipping plate may be changed based on the maximum weight of garbage we want to handle. Research is required to find out the heaviest garbage that is likely to be thrown.

5. Size/shape/colour etc. to achieve a good-looking exterior.

6. Location of the bin might also be changed base on future research, library, lecture hall are the designated location for now.

14 System Components That Are Not Likely To Change

1. The approach of garbage sorting is not likely to change – we intent to create a machine learning model by TensorFlow, train it well enough to sort different types of garbage.

15 References

N/A