Design Specifications

- My solution must show the environmental issues with the Pacific Garbage Patch, highlighting the importance of recycling.
- My solution must be finished by June, 17, 2015.
- My solution must be a physical product that relates to recycling.
- My solution must be aimed towards students and faculty at ACS Hillingdon.
- My solution must be finished to a high quality.
- When making my solution, it must be made form only creative commons / copyright free media.
- My solution must be made with the resources I have access to, or can buy if necessary.
- My solution must be displayed at school.

Design Ideas

1. Monster made of plastic bottles and bags.
2. Armor made of recycled PCBs, bottles and wrappers
3. Book with recycled PCB cover
4. Short film about two different types of drink containers. Parody of Romeo and Juliet.
5. Interactive iPhone app written in Swift demonstrating the importance of recycling properly. The app will also allow users to keep track of the products they use and give them rewards for properly disposing of waste and helping the environment.
6. Smart Recycling bin. A recycling bin with a Particle Photon that allows people to how many bottles and cans have been put into the bin. It will track the amount of, plastic bottles, glass bottles and aluminum cans. I will create a Github repository with a website that will allow people to see the scores of the different types of drink container, as well as vote for what charity the bottle deposit money will go towards.

Target Market

Students:

- Use plastic daily.
- Want to recycle but don't find it interesting.
- Think that technology is interesting, especially internet-enabled devices.
- Use the internet daily.

Faculty:

- Want to increase their rate of recycling.
- Want to monitor their recycling progress, as well of keep track of their waste.

Evaluation

For this project I decided to make my last design idea. I created a Wi-Fi enabled, cloud connected recycling bin. I also created a website which gives information about my product and receives live statistics from the bin, located at http://nrobinson2000.github.io/smart-bin/index.html. I named my product Smart Bin, as it is an intelligent recycling bin that keeps track of how many bottles have been put into it. It also keeps track of five different types of drink containers / bottles have been put inside it. It accepts the following containers:

1. Glass bottles (such as coca cola and sparkling water)
2. Plastic bottles (any carbonated soda or water bottle)
3. Aluminum cans (any soda can)
4. Plastic cups (from the water coolers)
5. Milk cartons (from the refrigerator in the cafeteria).

I created Smart Bin in order to collect data from students and faculty, and get statistics on what is the most popular type of drink. Smart Bin could be used by the school cafeteria to see what is the most widely consumed drink without having to go through the recycling.

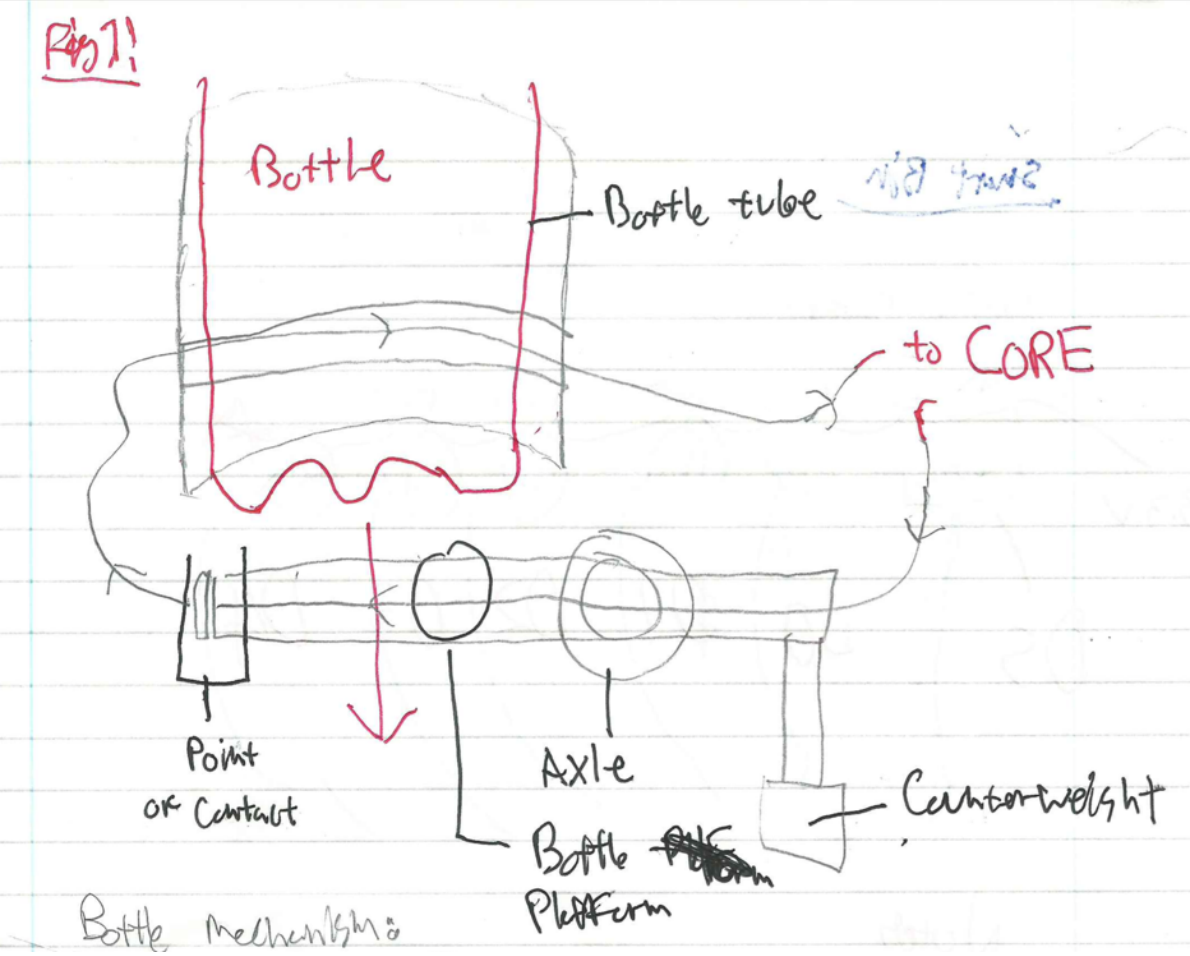
Smart Bin is easy to use. Just throw in your drink bottle and press the button for your type of drink. The score for that drink type is automatically incremented, and the change can be seen on the Smart Bin website. (http://nrobinson2000.github.io/smart-bin/stats.html)

The firmware that I wrote and the website I created is hosted on Github. GitHub is a web-based Git repository hosting service, which offers all of the distributed revision control and source code management (SCM) functionality of Git as well as adding its own features. Unlike Git, which is strictly a command-line tool, GitHub provides a web-based graphical interface and desktop as well as mobile integration. It also provides access control and several collaboration features such as wikis, task management, and bug tracking and feature requests for every project. (https://en.wikipedia.org/wiki/GitHub)

There are three parts of Smart Bin - the hardware, the firmware, and the software. The “hardware” is the actual bin itself, which has a special lid that includes the Particle Core and some circuitry. Whenever a bottle, can or cup is put into the bin and its button is pressed, the number of that type of drink is updated on the Core. The code running on the Core is called firmware. This is the C++ code written for the Core, compiled and flashed, that runs while the Core is powered from a wall socket or its 9 volt battery.

Building Smart Bin was an interesting challenge. I first needed to create a plan for how I would build it. I knew that I would have to use my Particle Core as the controller because my Photons have not yet arrived. I needed to get a bin and the required electrical components. After making my list of parts, I ventured off to Robert Dyas on the high street. I wanted to find a bin that would be large enough to hold quite few bottles, but not so large that it would be too hard to transport or be too expensive. I was able to find a 50L kitchen rubbish bin that suited my needs perfectly.

After purchasing the bin, I put it into my large Ikea bag and got onto a bus going towards Maplin, an electronics store. I had been planning to make my design have five individual holes, one for each of the drink types, each with its own automatic bottle mechanism.



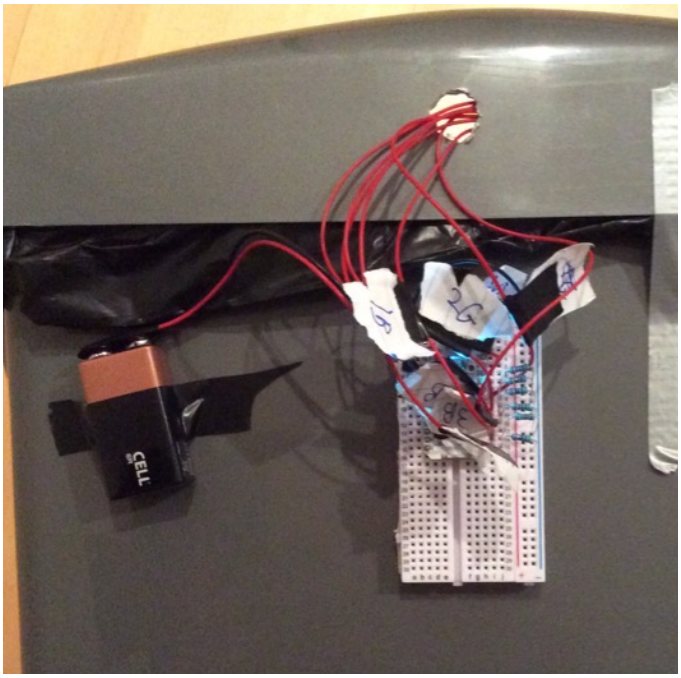
In Maplin, I realized that my design for the bottle mechanism would be too complex for the amount of time that I had to make my product. I had planned to create my bottle mechanism using several pieces of wood and plastic, opening and closing a circuit when a bottle goes through. This seemed like a simple idea, but as I was in Maplin searching for parts the complexity started to overwhelm me. I realized that I would have to go to B&Q to buy some wood and plastic, make my own conductive pads and design my own counterweight system. In my head while I was in Maplin, I had to redesign the base mechanics of the bottle mechanism. I decided to use five buttons, plus an additional locking button switch to handle the bottle counting. (It can addd or subtract.) Instead of someone having to put a bottle into a certain hole and having it getting possibly stuck, they could just press a button that corresponded to the drink type they were putting into the bin. This simplification changed my project. By using momentary push switches with a locking button, I could gather data more accurately with a much simpler design.

The image on the right is a top view of the finished Smart Bin (without the bag liner). Notice there are six buttons. One square black one, one green one, three black ones, and another green one. The first button, the black square one, controls the mode. When it is pushed down, the bin is put into addition mode. Each time a button is pressed, the tally for that bottle count is incremented. When the button is popped back up, the bin goes into subtraction mode, so that when a button is pressed the value of that drink type is decremented.

From left to right, the circular buttons go: glass bottles, plastic bottles, aluminum cans, plastic drinking cups, and plastic milk cartons. I used green and black buttons because they were the only colors of that type of button that were in stock. I could not make the buttons all the same color because Maplin only had three black round buttons and three green round buttons. I was intending to make each type of container have its own color, but this was not possible as there were only

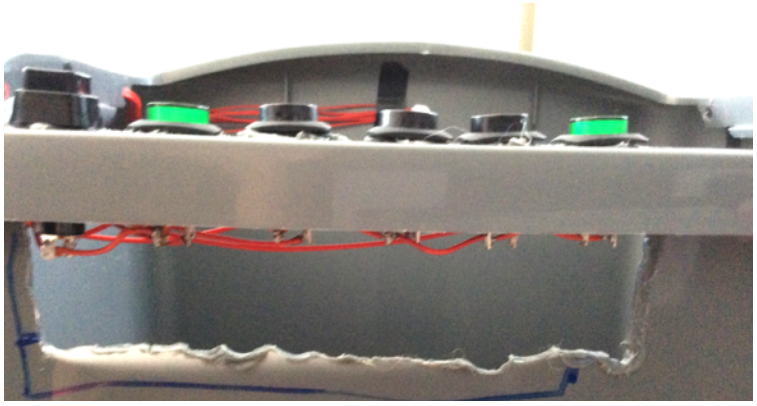
two colors of buttons available at the time. (If I made it again, I would like to get five different colors for the five container types, as I think it would help show that each is for a different type.) I then decided to use two of the green buttons to indicate the start and end of the input button range. I put the three black buttons in the center to make it symmetrical.

Here is the front of the bin. I put labels next to the buttons, indicating which button is which number. I made a document with five different images of the different types of drinks acceptable in Smart Bin, and the number button they are associated with. I also labeled the mode button, making it clear that it is not one of the bottle buttons. I also printed out the QR code to my website so that mobile users can scan it using their school iPads or phones and be able to see the stats on the website. This enables people to interact with the bin and see proof that their bottle has been taken into account.

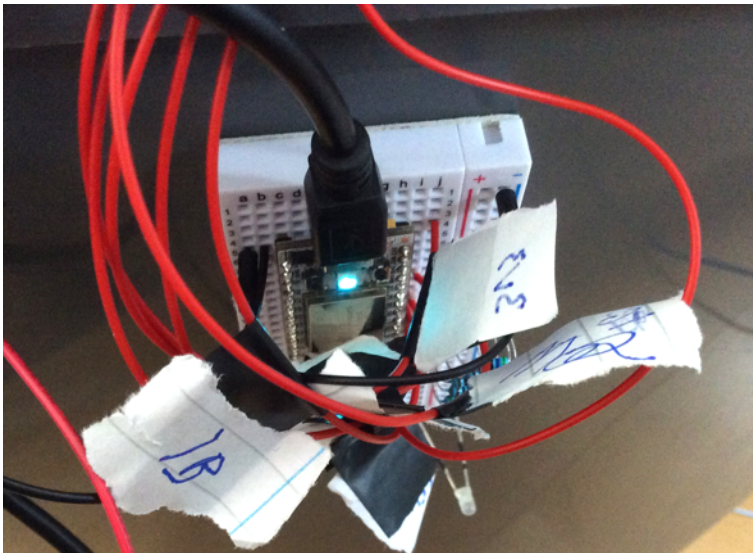


Here is the back of the bin. It has my Particle Core, the breadboard, the 9 volt battery and the wires connecting the six buttons on the front of the bin.

This is a picture taken from the inside of the bin. Coming across the top, you can see the red wires going under the lid of the bin, over the bag liner. In the previous picture, you can see how they come through the back of the bin.



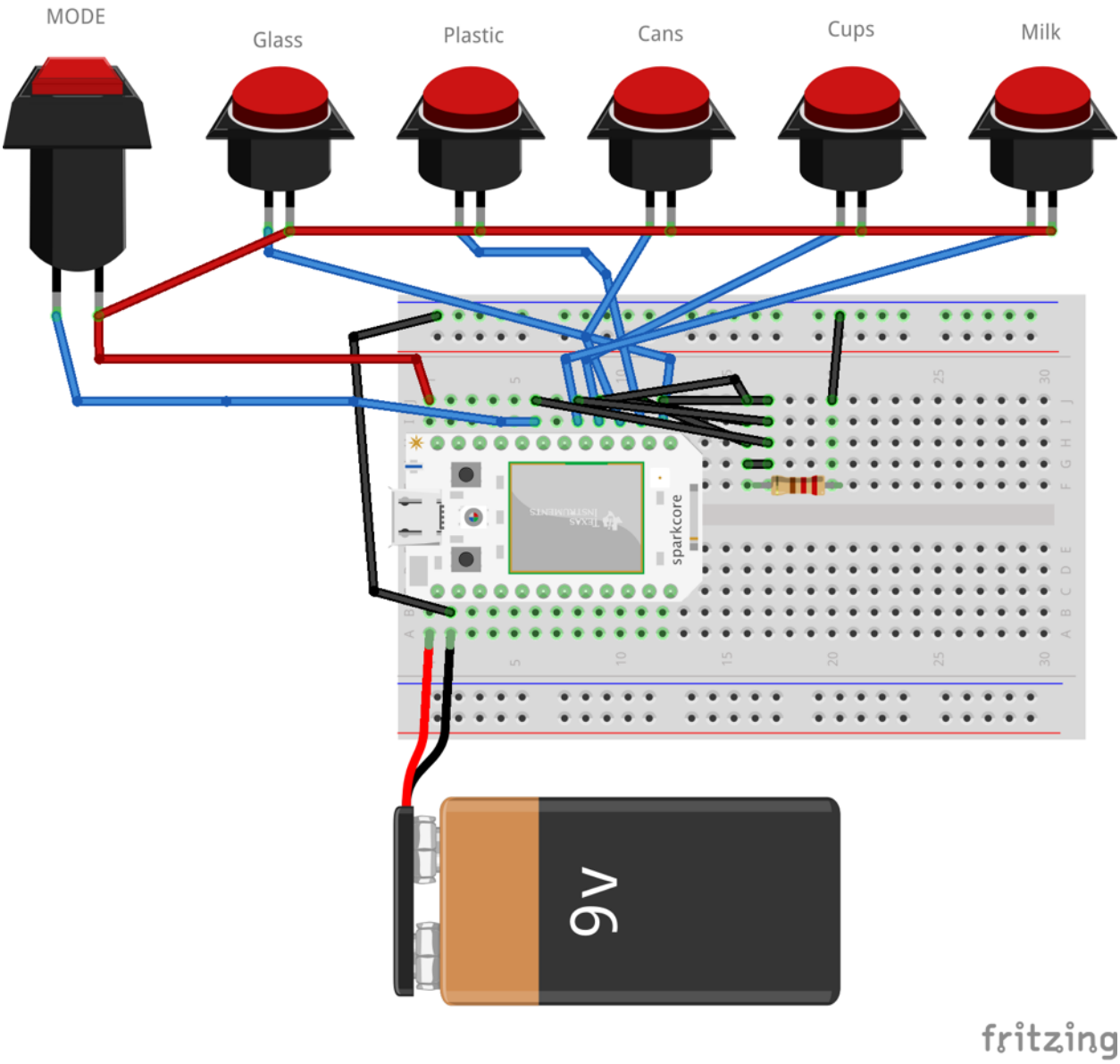
This is the front of the bin before it was finished being assembled. You can see where I had to solder the red wires to the buttons.



Here is a close up of the Core as I am debugging it. The light is breathing cyan, indicating that it is connected to the Particle Cloud, running its code, accessible through the Particle API.

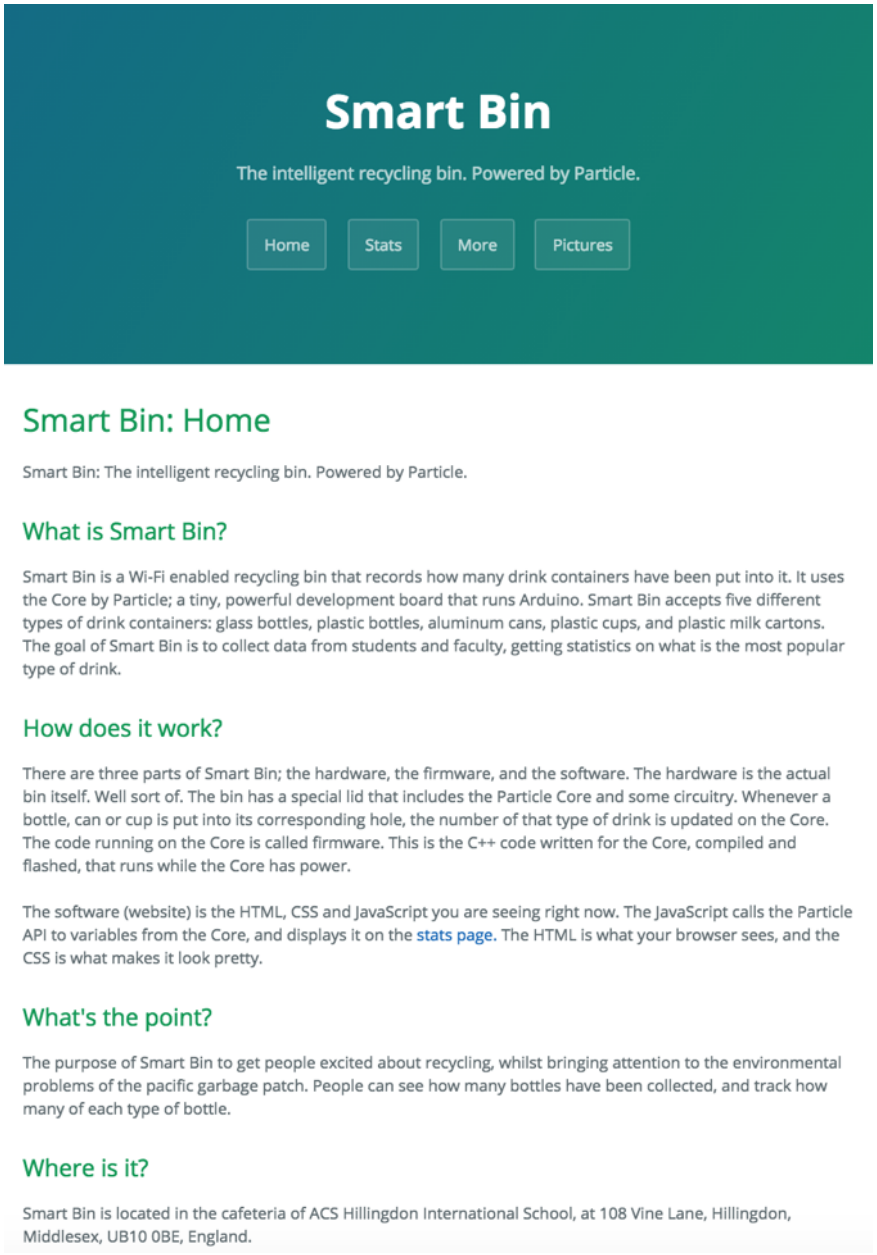
Note:

This image may look messy. That's because it is. I had to make paper labels for each of the wires so that I know which button goes into which pin on the Core.



Here is a 2D representation of the Smart Bin circuit, created with Fritzing, one of my favorite pieces of free open-source software.

I enjoyed creating my product. I think that it's a good concept, but that it could be improved. First off all, I would have liked to create a more secure, fail-proof mechanism for counting the bottles. Right now, the bin is open and anyone can skew the count by pressing the buttons when they have not placed anything in the bin or by putting something in the bin and not pressing the correct button. This is a prototype of a product. I do not think that this is a finished product that could be used every day, but I think that it shows how a Wi-Fi enabled recycling bin can be made. I would have liked to have taken more pictures, and a few videos during the creation process, but I did not have my phone out in my work shed to take videos of me melting holes through plastic with my soldering iron. Plastic is not an easy material to work with. It can not be cut like wood or folded like paper or cardboard. It can only be melted. In my shed, I was able to melt almost perfect holes for the buttons with my soldering iron. I was able to secure them in place and solder the circuit easily.



Programming the Core was frustrating at times I am quite familiar with programming, and C++ is one of my strongest coding languages, especially when writing Arduino, the code that runs on the Core. There were a few times when I was sure that I had soldered the buttons right, but could not get my code to run the way I had intended. Through sophisticated debugging, I was able to find the solution to my problem and make my code function exactly as I wanted.

All my code and documents for this project are hosted on Github, and so is the website. To create the website, I used HTML and JavaScript, as well as CSS. On the website there is information about this project as well as information about Particle and recycling. On the website, people can see the live stats from the bin as well as rate the bin and website.

In conclusion, I think that I came up with a good design for this project, and that it helps motivate people to recycle by making their decision to recycle recordable and trackable. Also, when they use the QR code to go to the website, they can learn even more about recycling. I was intending on having a poll on the website that would allow people to vote for which charity the bottle deposit money would go towards, but I found out that there is no bottle deposit in the UK. (In most US states, it is five cents per can or bottle, and this encourages recycling.)

If I were to make this product again, I would try to make a better bottle counting mechanism, as well as a companion iPhone app that would teach people how to properly recycle, and reward them for reducing the amount of waste they create each day. If I had more time to develop the website, I would consider adding ads in order to generate income and have people vote on which charity to donate the money to.