Self-Logging Backpack

Harry Arakkal, Suhaas Yerapathi

arakkal2, sny2

Introduction

Objective

Ensuring that your backpack has the necessary items for the day is often handled by constant rechecking or checklists that must be constantly reviewed and updated. This results in regular occasions where one forgets to bring something they need, or leaving behind an important belonging. Instead of putting the pressure on the user, we want the backpack to handle the responsibility. We propose a self-logging backpack to update the traditional carryall with Internet-of-Things capabilities.

This backpack will track the items it holds using RFID sensors and tags on the items. The state of the backpack will be reflected in a mobile app that will show the item and where in the backpack it is. The app will also integrate with the user's calendar, and ask for what items to associate with each event. Then, the app will notify users if they are missing on of the necessary items before an event begins. The app will also allow users to tag essential items, which must always be in the backpack. If the user moves from a location without one of the essential items, the user will be notified of its absence.

Background

Students regularly forget items, as they attempt to minimize the weight they carry in backpacks at any time. This requires constant inserting and removing of items, as each class often has its own notebook and related texts. Similarly, some classes may require a laptop, while other classes ban them. For total assurance, a minority of students keep everything they use for school in their backpack always. Obviously, this is very inefficient, as student's schedules often have large gaps between classes. As an optimization, students take only what they need for the next class. However, this opens the door for user error - for example, a needed notebook is missing. Another issue is leaving critical items behind like a laptop or charger. These are often always in a

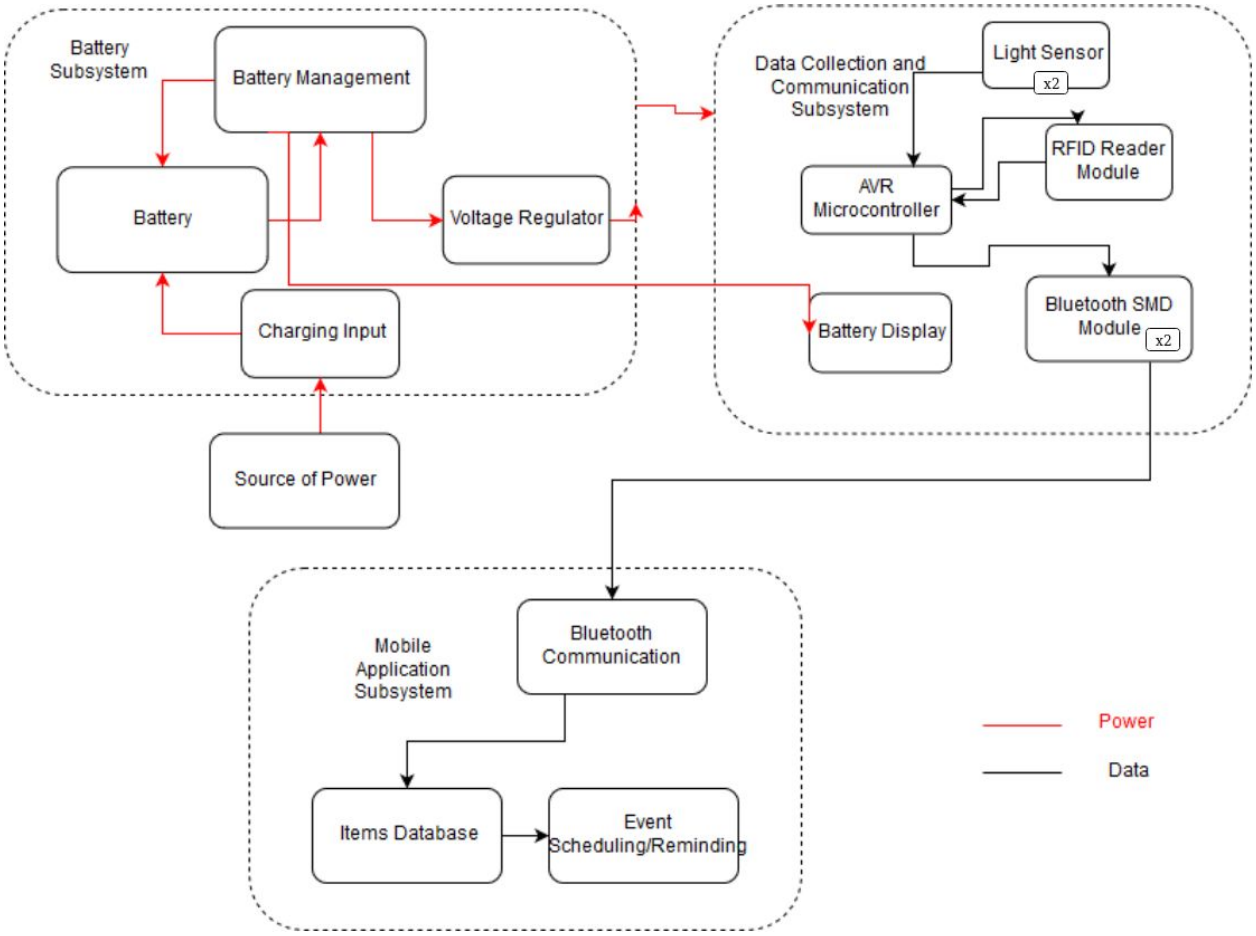
person’s backpack, so this system can also be used to notify the user when an important item is left behind, by tracking the user in addition to the contents of the backpack.

High-level Requirements

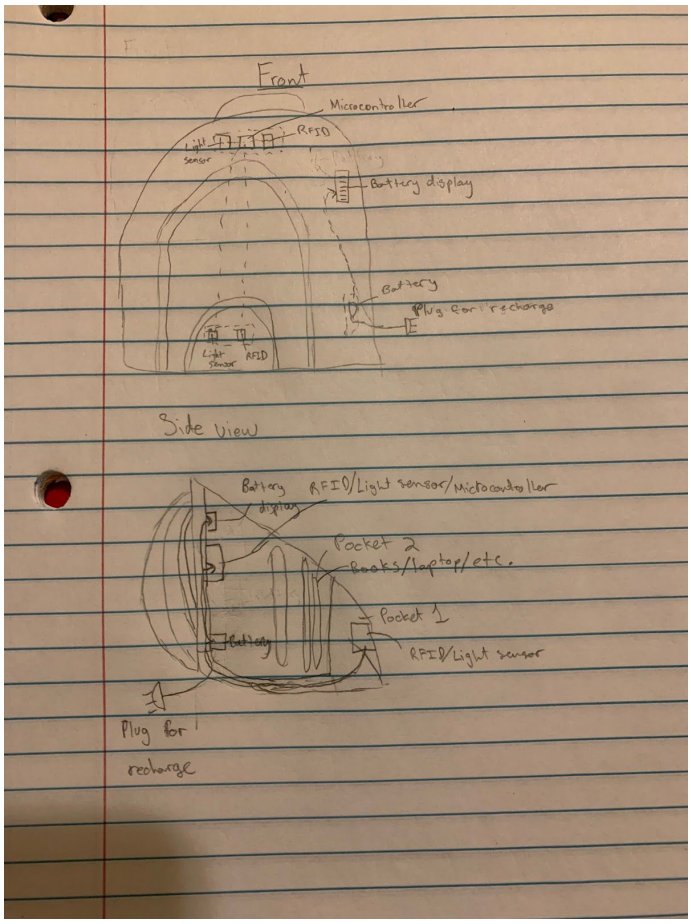
- The system should be able to recognize items using RFID and display them in the app more often than not.
- The system should not sense an RFID tag multiple times in a single action.
- Battery should last a reasonable amount of time

Design

Block Diagram



Physical Design



Functional Overview

In our system, there are 3 subsystems: the battery subsystem, the data collection and communication subsystem, and the mobile application subsystem.

In the battery subsystem, a charging module gets input from some source of power, which then powers the battery submodule. The battery submodule sends power to the battery management submodule for power management, which then returns to the battery for times such as standby. The battery management submodule also sends power to the battery display in the data collection and communication submodule so it can show how much batter is left, as well as a voltage regulator in the battery subsystem for output to everything else in the data collection and communication subsystem.

In the data collection and communication subsystem, a light sensor reads light data and sends it to an AVR microcontroller submodule, because the subsystem should only send information when enough light is detected (backpack is open). This

microcontroller can tell an RFID submodule that it should be reading and sending data back to the microcontroller. Once the microcontroller receives data and is ready to send it, it sends it to the bluetooth SMD submodule. This submodule then sends the read data to the mobile application subsystem.

In the mobile application subsystem, there will be a software submodule that will read data sent from the bluetooth SMD submodule in the data collection and communication subsystem. It will then send this data to the items database software submodule, so it can log the data (items read going in and out of the backpack). This will then send data to the software submodule that processes the information in the database and gives it to the user (reminders, etc.).

Requirements and Verifications

Block Requirements

Charging Input - Should be able to charge the battery fully in reasonable time, i.e. overnight
Battery - Should hold enough charge for 2 days of operation
Battery Management - Should make sure battery has charge, sends charge left to display
Voltage Regulator - Should output 3.3V to rest of system
Battery Display - Should not consume so much power that battery becomes inefficient
Light sensor - should correctly identify when backpack is open enough to fit a decently sized item in (a book)
AVR Microcontroller - Should correctly get input from light sensor and RFID and send information to bluetooth submodule, should stay on and not consume too much power (tearing down and remaking bluetooth stack should not happen frequently).
RFID Reader - should rarely log same thing twice when putting items in backpack, should properly send scanned information to AVR microcontroller when item is scanned.
Bluetooth submodule - Should properly receive data from AVR microcontroller, should send data to mobile application, should not drop packets frequently.
Bluetooth Communication on Mobile Application - should properly read all information coming in and process the information to figure out item, in/out, etc. at a decent rate such that if many items are scanned very fast, it can handle it
Items Database - should hold all items that are tagged, and show items that are in the backpack

Event Scheduling/Reminding - Should properly remind user if item is missing in backpack in a reasonable amount of time (before it is not possible to place item in backpack anymore, about 1 minute)

Supporting Material

- Circuit Schematics
- Simulations
- Calculations
- Measurements
- Flow charts
- Mechanical diagrams

Tolerance Analysis

Cost and Schedule

Cost Analysis

- Labor
- Parts
- Grand Total

Schedule

2/21 - Finish Design Document
2/28 - Make adjustments to design document based on feedback and purchase components
3/7 - Assemble battery subsystem and test
3/14 - Test RFID and Bluetooth outside subsystem and begin programming microcontroller
3/28 - Assemble data collection and communication subsystem and test
4/4 - Complete mobile app and test communication with system
4/11 - Finishing touches, ensure everything works together so the project is complete and ready to demo

Ethics and Safety

In our research, we could not find any relevant products that exist, so there is no risk of infringing on another's intellectual property. The closest resemblances of an electronic backpack on the market usually have something to do with security such that a backpack and its data are safe inside of it, which is not related to our project. Relating to the hardware, the PCB and the battery system should be engineered so that they never reach temperatures that risk danger to the backpack, its contents, or the user. In addition, the total system should be relatively light, so that the benefits of the tracking are not outweighed by the physical weight of the additions. Most ethical issues in our project relate to the software. The data from the user’s backpack should be kept local to the user’s device to avoid issues. The information stored on the phone should be securely stored to prevent external apps from collecting the data. In addition, any location data must only be used to notify the user of forgotten items, and never exported from the device.