University of Illinois at
Urbana-Champaign

Door Access Tracker

Patrick Connelly (prc2), Ben Wasicki (wasicki2),

and John Scholl (johnts2)

ECE 445 Proposal
Team #41
TA: Chi Zhang
February 13, 2020

1. Introduction

1.1 Objective:

Many areas of day-to-day life involve the opening and closing of a door. We believe that better information on the state of a door can improve one's quality of life. For example, one could monitor a door as a security measure, such as a front door, a liquor closet, or a medicine cabinet. Alternatively, some doors may also have a tendency of getting stuck open. In this case, knowing that the door was not closed properly may be good information to have. In addition, knowing when the mailbox has been accessed could be time saving, especially for someone who has mobility problems because they would not need to check the mailbox unnecessarily.

Our proposed solution is the Door Access Tracker. This tracker would consist of a sensor to detect the state of a door, a microcontroller, a wifi card, a cloud server, and an android app. This would be a portable device that would be adhered to a door. The primary functionality involves the user getting an update on their phone via an application when the state of the door is changed. In order to make this product more versatile, we would allow for different configurations on when to send notifications. For example, a consumer may want to know the instant a medicine cabinet or liquor cabinet is opened; however, they may only care about a door's state if it were to be left open for a specific amount of time before being closed.

1.2 Background:

There are many situations in which the monitoring of a door or cabinet may be useful. From a security perspective, knowing when an area is accessed could be extremely useful information, especially for knowing when something has been tampered with. From a convenience perspective, putting this device on something such as a mailbox would let someone know when they should go to check for mail. Finally, from an energy-savings perspective, this product could let a person know when a door or window is left ajar, leading to heat loss in the winter and air conditioning loss in the summer. A key issue that needs to be addressed as well is that a user may want a different notification or set of notifications for different situations. For example, they may want to know immediately when a door state is changed, they may only want to know when a door is opened, or they may only want to know if a door is left open for a certain amount of time.

1.3 Physical Design:

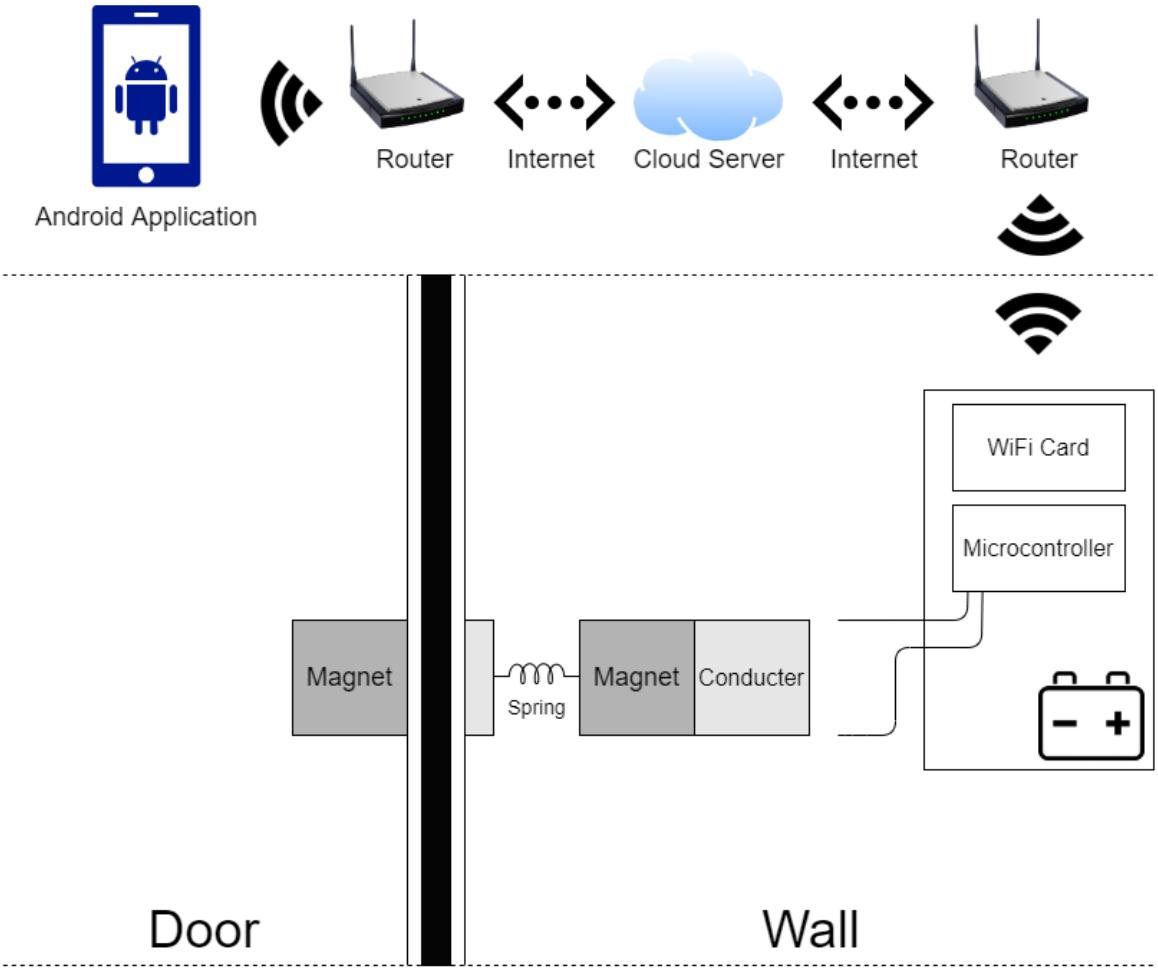

Figure 1. Physical Design

1.4 High-level Requirements:

The following are the three most important qualities our project must exhibit in order to be successful:

- The door sensor sends a different signal based on the state of the door (i.e., open or closed).
- The system controller takes the signal from the door status sensor and updates the back-end server with the door's state.
- The back-end server can send the Android application an update based on the information it receives from the system controller and the current configuration set.

2. Design

2.1 Block Diagram:

A modular representation of the core components of our design is presented in the block diagram below (*Figure 2*). The Sensor Module design provides power to the Control Module when the door is open and breaks the circuit when the door is closed. This allows for the microcontroller to know the state of the door and limits power consumption. In the Control Module, the microcontroller will process the signal from the Sensor Module and allow the WiFi card to relay the data to the back-end server. From there, the back-end server and the mobile application in the Software Module send data back and forth to present the user with the requested information.

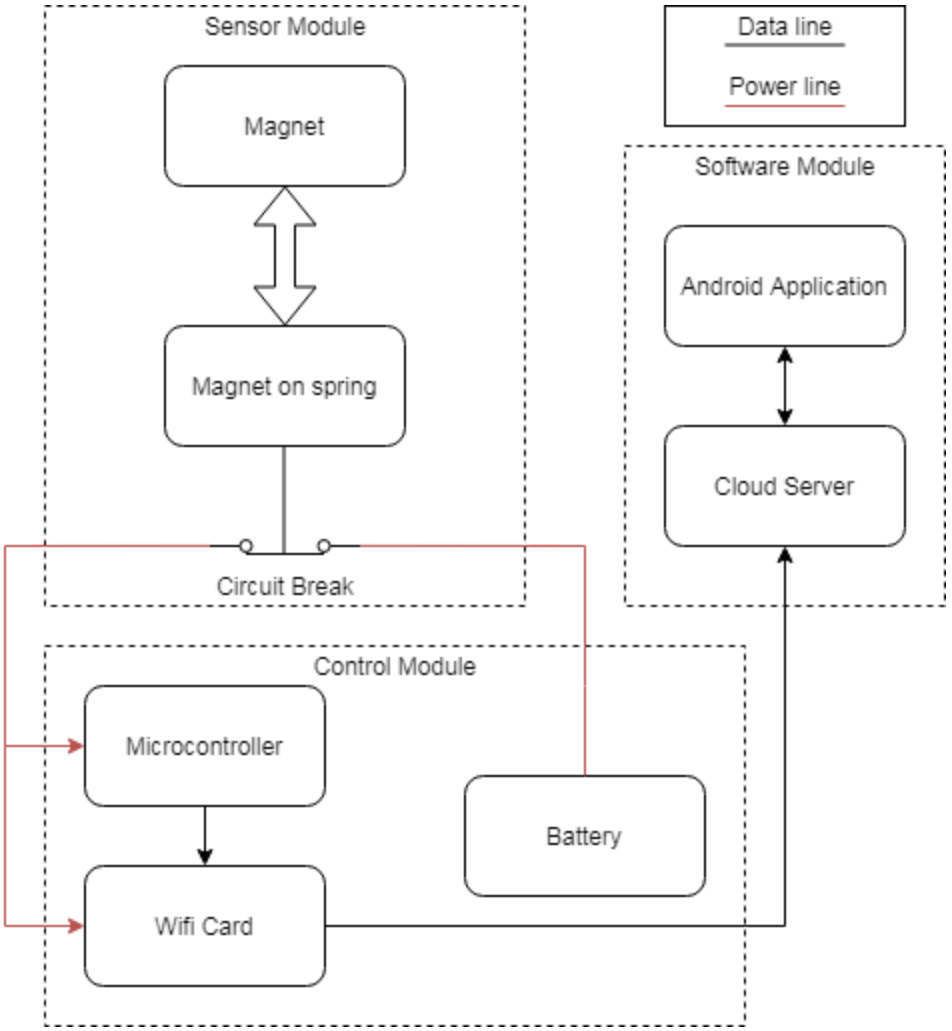

Figure 2. Block Diagram

2.2 Functional Overview:

The following are brief descriptions of the function of each module in *Figure 2*:

- **Sensor Module**
    - This is a two-piece system for tracking the state of the door. One magnet will be mounted on the door itself, and another will be on the frame of the door, allowing for the magnets to act on each other only when the door is closed. The magnet on the frame would be attached to a spring and a conductive component. Close proximity of the magnets would cause the spring to compress and pull the conductive component, opening a circuit and changing the current output signal to the Control Module. Likewise, opening the door would cause the spring to decompress, letting the conductive component complete the circuit and sending a different signal to the Control Module.

- **Control Module**
    - This is the main computing device, consisting of a battery, a microcontroller, and a WiFi card. The primary function of this module is receiving a signal from the Sensor Module, processing it, and communicating data to the Software Module. The microcontroller is responsible for taking input from the Sensor Module and transferring it to the WiFi card in a convenient format. The WiFi card will open a connection with the Cloud Server in the Software Module, communicating data once for each time the state of the door is changed. The operation time and power drain of these components will be low so as to maximize battery life.

- **Software Module**

  The Software Module will receive the door state from the WiFi card and use user-defined settings to send notifications to a user's phone via an Android application. It will consist of the following two components:.
    - **Cloud Server**

      The Cloud Server is where the door state will be received from the WiFi card as a struct {int serial_number, int door_state}. Additionally, this is where android application will both register itself to various physical door tracker components via serial number, and also set its notification configurations. When a state update is received, the server will notify the corresponding applications based on their specific configuration settings. These settings will be per-application.

      The notification configurations will include:
        - Which state changes to send a notification for?
            - (open, close, both)
        - How much delay before notifying the user?
            - (i.e. notify user if door is open for more than 1 minute without being closed)

        - Should notification be repeated?
            - (i.e. notify user every 30 seconds once door is opened until door is closed)
        - How much history should be stored?
            - (I.e. store last 30 state changes with timestamps)
    - **Android Application**
      The android application is the front-end part of the software module. From here, a user is able to register itself with the Cloud Server to receive notifications for a specific control-module (specified by serial number) and update configurations for each individual control-module. When the application receives an update from the Cloud Server, it will notify the user immediately.

2.3 Block Requirements:

- **Sensor Module**
    - Description
        - The Sensor Module is responsible for the first high level requirement, as it will track the door state and communicate it to the Control Module. This communication is done by completing the circuit exposed by the Control Module with a conductive component when the door is open.
    - Requirements
        - The magnetic force is great enough such that the spring is compressed by at least 1mm upon closure of the door.
        - Opening the door allows the spring to extend such that the conductive component is making full contact with the microcontroller input.
- **Control Module**
    - Description
        - The Control Module contributes to the second high level requirement, as it is responsible for ensuring that the Cloud Server in the Software Module has received the state change of the door.
    - Requirements
        - When the input to the microcontroller changes, the output from the microcontroller also changes.
        - When the input to the WiFi card changes, the WiFi card must open a connection with the Software Module and send exactly one data packet.
- **Software Module**
    - Description
        - The Software Module contributes to the second and third high level requirement. It connects to the WiFi card of the Control Module and deals with taking a state update and notifying the user.
    - Requirements
        - When a door state change is received, be able to notify the user based on the configurations set.

- **Cloud Server**
  - Description

    The cloud server contributes to the second and third high level requirement. It contains most of the computations and memory that will be needed. As such, it will take requests from the user via the android application and from the control module, be able to set configurations, store history, and send updates to the android application based on the set configurations.
  - Requirements
    - Server will be able to receive state changes from at least 1 Control Module (independent serial number)
    - Server will be able to complete registration of a specific application with a specific serial number.
    - Server will be able to send updates to at least 1 application after a state change is received from a control-module.
    - Server will be able to use configurations specified by an application to deliver updates to said application as set by configurations.
    - Server will keep a history of door state changes with corresponding timestamps based on the number specified by the application.
    - Server will be able to send state change history to an application upon request.

- **Android Application**
  - Description
    - The android application contributes to the third high level requirement. It will act as an interface for the user to register itself with a serial number of a control-module, update configurations for different control-modules, request state change history from the cloud server, and receive updates from the cloud server.
  - Requirements
    - The application will have an interface from which registration, configurations, and history requests can be input for communication with the cloud server.
    - Application will be able to register itself with the cloud server based on a specific serial number corresponding to a control-module.
    - Application will be able to notify the user when it receives an update from the cloud server.
    - Application will be able to retrieve and display door state history upon user request.

2.4 Risk Analysis:

The block that poses the greatest risk to the successful completion of our project is the control module. Specifically, transfering the sensor input to the WiFi card and setting up the WiFi card so that it can successfully communicate with the server is the most technically complex part. We do not have previous working with WiFi cards. However, we have prior experience with building network communication protocols, so we anticipate being able to solve this problem with further research. Additionally, we have not written an Android application before. We do, however, have plenty of experience with software development in general, and there are many available resources for learning how to develop an Android application.

3. Ethics and Safety

As the developers of this project, we believe it is important that we produce a safe, reliable, and efficient product to our user. We commit ourselves to holding a high degree of professional conduct in accordance with both the IEEE and ACM Code of Ethics. We will avoid ethical breaches by following all device specifications, working in our respective areas of competence, and clearly stating proper operating procedure (ACM 2.6). At the same time, we acknowledge that our device could be misused; therefore, we will take all necessary precautions to prevent any harmful modes of operation.

In accordance with the ACM Code of Ethics, this project will pose no risk to the user or community under standard operations. Given that our project monitors when a door is opened and closed, it could pose a safety risk to the user if the data is compromised. We will ensure that all wireless protocols are followed, and communications will be secure. The data gathered by our sensor will be the sole property of the intended user of the device (ACM 2.9). All software will follow accepted community standards.

Following the IEEE Code of Ethics, we have decided it is important to make our design as energy efficient as possible to minimize waste. As designers, it is our responsibility to limit the environmental impact of our device. We have implemented a circuit break when the door is closed to ensure power is only consumed when necessary. This will limit the amount of waste associated with battery replacements.

In addition, we will ensure there is no exposed wiring or electrical components in our design to minimize the risk of electrical shock. Similarly we will ensure all components are operating within their respective operating regions to reduce the risk of a short or fire hazard.

4. References

[1] “IEEE Code of Ethics,” *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 12-Feb-2020].

[2] “ACM Code of Ethics and Professional Conduct,” *Association for Computing Machinery*. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 12-Feb-2020].