Internet of Things (IoT) Home Automation

RUTGERS
THE STATE UNIVERSITY OF NEW JERSEY

Student: Diego Urquiza, Amgad Armanus, Evan Arbeitman, Felix Mateo, Nirjan Thayaparan

Advisor: Prof. Zhang

Goal

- ❑ Design and build an inexpensive solution for the everyday consumer to make their homes smarter. Our technology focuses on a linux server hub running openHAB that can communicate via wifi to other smart devices. To complement the server hub we created three smart devices: Smart Surveillance Camera, Smart Lock, and Smart Outlet. These three devices are the biggest push for smart devices in 2015 with companies like Samsung and we aim on doing it cheaper.

Motivations and Objectives

- ❑ Motivations
  - A lot of expensive Smart Locks have come on the market and we wanted to design and built an affordable Smart Lock.
  - A simple Smart Outlet that can be controlled from your mobile device, can go a long way in making any electronic device smart.
- ❑ Objectives
  - Monitor activity outside your door by viewing a live feed of your surveillance camera on your mobile device and also have pictures captured, timestamped, and uploaded to your dropbox automatically when motion is detected.
  - Lock and unlock your door from your mobile device.
  - Turn on and off electronics that are plugged into the Smart Outlet from your mobile device.
  - Create a wifi compatible hub that will work with other smart devices on the market.

Research Challenges

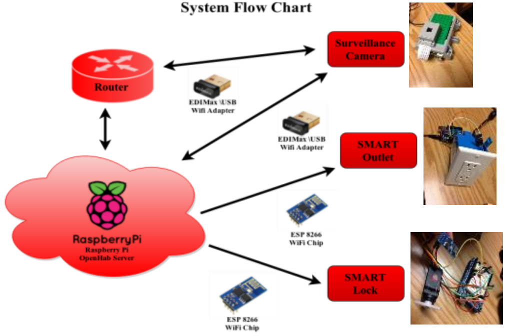
- ❑ Setting up communication between Raspberry Pi Hub and Smart Lock with nRF24L01 chips.
- ❑ Finding the most efficient, affordable, and reliable wireless communication to use for our smart devices.
- ❑ Designing a power efficient circuit with an MSP430 and sensors to extend the battery life of the Smart Lock.
- ❑ Measuring the current draw of Smart Lock and Smart Outlet and displaying statistics on openHAB.

Acknowledgement

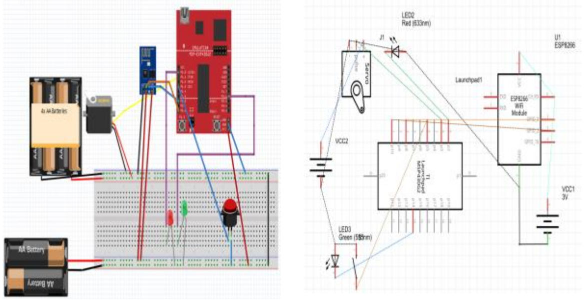
Special thanks to Prof. Yanyong Zhang and Prof. Bernhard Firner.

System Design

System Flow Chart



Circuit for Smart Lock



Results

- ❑ Using the OpenHab Andoird / iOS app, the user is able to connect to the OpenHab server on the Raspberry Pi.
- ❑ From the app, the user is able to lock and unlock their door, turn on and off the smart outlet, and view a live feed of the IP surveillance camera.
- ❑ The IP surveillance camera software continuously monitors the camera feed and once motion is detected, it captures a picture and transfers the picture to the Raspberry Pi hub. Using the Dropbox Developer API, the Raspberry Pi Hub automatically uploads the pictures to the user's Dropbox account.
- ❑ The Smart Outlet is built using a GFCI outlet, relay, ESP8266, and an Arduino Uno. The Arduino is programmed to receive signals from the ESP8266 and then send those signals to the relay box which toggles the outlet.
- ❑ Each ESP8266 WiFi chip is programmed to automatically connect to our WiFi network, receive a static IP address, set up a web server and then listen for incoming commands.

Theoretical Battery Life

Average Current Draw for the Servo Motor

$$\text{Duty cycle} = \frac{T(on)}{T(on)+T(off)} * 100$$

Assume that servo is on for 5 seconds each hour

T on = 0.083 minute

T off = 59.917 minutes

$$\text{Duty cycle} = \frac{0.083}{59.917+0.083} * 100 = 0.138\%$$

Average current draw per hour = 0.00138 * 160 mA = 0.22 mA

2200 mAh /0.22 mA = 10000 hours

10000 hours / 24 hour = 416 days

Battery Life comparison between Arduino Uno and the Ultra-Low Power MSP430 implementation of the Smart Door Lock

Arduino Uno

Average current draw of Arduino Uno = 50 mA
Average current draw of ESP8266 WiFi chip when on stand by = 0.9 mA

Using four AA batteries to provide 6 V
for the Arduino Uno
2200 mAh / 50.9 mA = 43.22 hours
43.22 hours / 24 hours = 1.8 days

MSP430

Average current draw of MSP430 = 400 uA
Average current draw of ESP8266 WiFi chip when on stand by = 0.9 mA

Using two AA batteries to provide 3 V for the MSP430 and the ESP8266
2200 mAh / 1.3 mA = 1692.3 hours
1692.3 hours / 24 hours = 70.5 days

References

[1] "OpenHAB - Empowering the Smart Home." http://openhab.org
[2] "ESP8266." - NURDspace. https://nurdspace.nl/ESP8266
[3] "Motion for Linux." http://www.lavrsen.dk/foswiki/bin/view/Motion/WebHome