Background Research

for

Machine Learning Smart Home

Prepared By:

Matthew Maynes

Cameron Blanchard

Peter Mark

Jeremy Dunsmore

Table of Contents

1. Introduction

The field of home automation systems is a young and expanding market. In recent years there has been a vast expansion into micro-computing and the internet of things. The smart home automation system is yet another project in this growing field. This report is intended to give context and background to this project.

The research in this report starts by reviewing existing systems and comparing their benefits and weaknesses. It then explores individual smart devices that exist on the market for IoT deployment. Next, the report moves into some more technical topics that explore options for implementing this system. The goal of the remaining research is to find adequate solutions to the requirements outlined in the system specification. The research will first examine the communication protocols available for IoT devices. It then examines some different computing devices available on the market for custom IoT development. Finally, the report investigates a number of different embedded sensors, controllers and actuators for actual system development.

2. Automation Systems

2.1 Background

There are many different home automation systems.In this section we are looking at existing systems so that we can better understand what technologies currently exist in the market as well ... so we can take these into consideration while designing our system.

This section will concentrate on the following characteristics:

- **Supported Communication protocols**

What different protocols do the systems support. We want to find the protocols that the other systems have decided are the most important to support.

- **Device Discovery/Setup**

How are new devices added to the automation systems network.

- **Network**

How do these systems setup the network of devices. Do they use a central server, are all devices independent etc..

- **API**

Does the system provide an API that our system could potentially use to control the devices connected to these systems. In order to control these devices we require that the following; a way to view all connected devices, a way to get notifications when a device in the system changes states, a way to change the state of a device.

- **Third Party Integrations**

Which companies have these systems decided are important to support

2.2 INSTEON

Description

INSTEON is a home automation system that allows control of the home through a phone. Insteon supports contorling devices based on a scheduled, using IFTTT or through setting up **scenes**.

A scene is configured a configuration for a room that the user sets up at a specific time or when specific environmental conditions are met. INSTEON has all of the devices remember their current states and whenever it sees those states it has the devices recreate the saved scene.

Technical Overview

Supported Communication Protocols

INSTEON created their own protocol also called INSTEON which is the communication protocol they support

Device Discovery/ Setup

Adding a new device is done through the INSTEON app. The device is automatically discovered by the system. Using the app the user selects the new device and enters the device ID after the ID is entered the device is able to be configured, naming the device adding it to a room etc..

Network

INSTEON uses a central hub to communicate between the devices and the users app.The central hub is required in order for the system to work (can't use INSTEON devices individually).

INSTEON uses a peer-to-peer network to connect the devices. All of INSTEON's devices can act as a controller to send messages, a repeater to forward messages or a responder to receive messages.

API

INSTEON provides a REST API to interact with their devices.

Feature	supported
List all devices	Y
Receive update on device state change	Y
Modify device state	Y

In order to use the API approval from INSTEON is required. Applying for an API key is done through the INSTEON website.

Limitations

In order to use the API an INSTEON Hub is required also adding new INSTEON devices to a network still require configuration through the INSTEON App.

Third Party Support

INSTEON supports devices from the following manufacturers:

- Nest
- Amazon Echo
- logitech
- first Alert
- Sonos
- MiLOCKS
- Apple HomeKit

Evaluation

The INSTEON system is very reliable and easy to setup however because they use their own communication protocol the devices that can be added is limited mostly to those manufactured by INSTEON.

Refrences

[1] Apiary, "INSTEON API · Apiary,". [Online]. Available: http://docs.insteon.apiary.io/. Accessed: Oct. 6, 2016.

[2] INSTEON®, "Home," in INSTEON, INSTEON, 2016. [Online]. Available: http://www.insteon.com/. Accessed: Oct. 6, 2016.

[3] INSTEON®, "WHITEPAPER: Compared,". [Online]. Available: http://cache.insteon.com/documentation/insteon_compared.pdf. Accessed: Oct. 6, 2016.

[5] INSTEON®, "WHITEPAPER: The Details,". [Online]. Available: http://cache.insteon.com/documentation/insteon_details.pdf. Accessed: Oct. 6, 2016.

[4] SMARTHOME®, "INSTEON home automation,". [Online]. Available: http://www.smarthome.com/sc-what-is-insteon-home-automation. Accessed: Oct. 6, 2016.

2.3 Wink

Description

Wink Hub is a hub that allows connections from many different smart devices. Wink supports control of devices using scheduling and IFTTT.

Technical Overview

6Wink is a central hub that supports most of the popular connection protocols so that users can connect and control a variety of smart devices from different vendors in one app.

Communication Protocol

Wink Hub supports the following communication protocols:

- Wifi
- Bluetooth smart (BLE)
- Z-Wave Plus
- ZigBee
- Lutron's Caseta
- Kidde

Device Discovery/Setup

Supporting many different manufacturers means that there are a variety of different ways for devices to connect to the Wink Hub. The two most common ways are:

1. Pressing a button on the Wink Hub to start transmitting a pairing signal and the new device will receive the signal and connect to the Hub.

2. The device needs to be setup in the manufacturers app once this is done the device can be added to Wink through the Wink app.

Network

The Wink system is a central hub that connects different devices. All of the different devices are controlled through the central hub.

API

Wink provides a RESTful service through the Wink hub and a secondary partner PubNub.

Feature	supported
List all devices	Y
Receive update on device state change	Y
Modify device state	Y

Third Party Integrations

Wink has support for the following manufacturers:

Nest	Philips	GE	Leviton	Rheem
Honeywell	TCP	Kidde	Kwiset	Lutron
Rachio	Bali	Amazon	Andersen	Canary
Carrier	Chamberlain	Commercial Electric	Cree	Dropcam
Ecobee	Emerson	GoControl	Hampton Bay	IHome
Leaksmart	Osram			

Evaluation

Providing support for the most popular communication protocols allows Wink to connect with almost any device a user can purchase. This makes them more attractive to consumers.

References

[1] Apiary, "Wink API · Apiary,". [Online]. Available: http://docs.winkapiv2.apiary.io/#reference/oauth/obtain-access-token. Accessed: Oct. 9, 2016.

[2] "Wink FAQ - Wink@Home Wiki," 2015. [Online]. Available: http://wiki.winkathome.net/Wink_FAQ. Accessed: Oct. 6, 2016.

[3] Wink, "A simpler smart home," Wink, 2016. [Online]. Available: http://www.wink.com/help/faq/. Accessed: Oct. 8, 2016.

2.4 SmartThings

Description

SmartThings is Samsung's home automation system. The System provides a hub that connects smart devices together. These devices are able to be controlled through the SmartThings App. The app supports controlling devices using a schedule or using IFTTT.

Technical Overview

Communication Protocols

```
- Z-wave
- Zigbee
- Wifi
```

Device Discovery/Setup

Adding a new device is done through the app. A user will click "find device" the hub will then search for new z-wave and zigbee devices. After the device is found the user adds it to a room and names the device.

Network

SmartThings uses a central Hub to connect all of the smart devices. The SmartThings app talks with the SmartThings Cloud which talks to the Hub which then controls the devices.

API

```
provides a Groovy API to create SmartApps that allow control of devices.
```

Feature	supported
List all devices	Y
Receive update on device state change	Y
Modify device state	Y

Limitations

Requires a SmartThings hub and connection to the SmartThings cloud.

Third Party Integrations

2Gig	Aeon Labs	Amazon	Belkin	Bose
Cree	ecobee	Ecolink	EcoNet Controls	Enerwave
Everspring	Fibaro	Fidure	First Alert	FortrezZ
GE	Google	Honeywell	iHome	Keen Home
Kwikset	Leak Intelligence	Leviton	LiFi Labs	Linear
Netgear	OSO Technologies	OSRAM LIGHTIFY	Philips Hue	Remotec Technology
Samsung	Samsung SmartThings	Schlage	Sengled	Skybell
Spruce	Yale	Zen		

References

[1] "Samsung SmartThings hub FAQ — SmartThings developer documentation," 2016. [Online]. Available: http://docs.smartthings.com/en/latest/sept-2015-faq.html. Accessed: Oct. 6, 2016.

[2] SmartThings, "How it works," SmartThings, 2016. [Online]. Available: https://www.smartthings.com/how-it-works. Accessed: Oct. 10, 2016.

2.5 Apple HomeKit

Description

Apple HomeKit allows users to control their smart devices using their IPad or IPhone. Apple HomeKit doesn't require any central hub however if a user wants to control devices when not at home their needs to be an Apple device in the home that the devices are connected to, this device can be an Ipad or apple TV.

Technical Overview

The smart devices are able to be scheduled and controlled in groups from the app. ex. All the lights in the living room turn off at 11:00 pm.

Communication Protocol

Apple HomeKit uses Wifi as the communication protocol,

Device Discovery/Setup

Adding new devices is done through the app. Once a device is connected to the network it can be added to the home through Apple's app, some devices require some configuration in their manufacturers apps.

Network

HomeKit uses the homes wifi network to connect devices and all devices on the network are able to communicate with one another.

API

Feature	supported
List all devices	Y
Receive update on device state change	Y
Modify device state	Y

References

[1] "Use the home app on your iPhone, iPad, and iPod touch," Apple Support, 2016. [Online]. Available: https://support.apple.com/en-ca/HT204893. Accessed: Oct. 10, 2016.

[2] Apple, "Creating homes and adding accessories," 2016. [Online]. Available: https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/HomeKitDeveloperGuide/WritingtotheHomeKitDatabase/WritingtotheHomeKitDatabase.html#//apple_ref/doc/uid/TP40015050-CH4-SW1. Accessed: Oct. 10, 2016.

[3] E. Betters, "Apple HomeKit and home app: What are they and how do they work?," Pocket-lint, 2016. [Online]. Available: http://www.pocket-lint.com/news/129922-apple-homekit-and-home-app-what-are-they-and-how-do-they-work. Accessed: Oct. 10, 2016.

2.7 Summary of Evaluation

Looking at these systems we found that we should research the Z-wave, Zigbee, Insteon and Wifi communication protocols. We also found some manufacturers that are commonly supported that we should look at, in order to potentially provide support for their devices.

3. Smart Devices

3.1 Background

What is this section in the context of this system?

3.2 Relation to System

We are looking at existing smart devices so that we can see if there are any devices that exist on the market that our system could use in order to lower the time required to build our own. Adding support for other smart devices also makes our system more appealing to the end user because they are not limited to only using devices we create but can go to a store and purchase any device that use technologies we support.

Might need to reword

3.3 WeMo

Description

WeMo is a line of smart products made by belkin. These devices are able to be controled over a wifi connection with a phone.

Technical Overview

Communication Protocol

uses wifi

API

There is a comunity supported open source python API for controlling WeMo devices `ouimeaux`

Evaluation

References

[1] B. International, "WeMo," WeMo, 2014. [Online]. Available: http://www.belkin.com/whatiswemo/. Accessed: Oct. 6, 2016.

[2] "Ouimeaux 0.8: Python package index,". [Online]. Available: https://pypi.python.org/pypi/ouimeaux. Accessed: Oct. 6, 2016.

3.4 Nest

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.5 Philips Hue

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.8 Osram

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.10 ecobee

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.11 Bose LifeStyle

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.12 Aeon Labs Remotes

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.13 Spruce Irrigation

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.14 OSO PlantLink

Description

What is this item?

Technical Overview

Technically speaking, what does this item do?

Evaluation

How does this specific item do against our criteria?

3.17 Summary of Evaluation

All of the evaluation grouped together

Resources

[1] R. Crist, "Best smart home devices of 2016," CNET, 2016. [Online]. Available: https://www.cnet.com/topics/smart-home/best-smart-home-devices/. Accessed: Oct. 6, 2016.

3.18 Conclusion

What did we decide upon? Why?

4. Communication Protocols

4.1 Background

There are many different ways to connect devices across a network. The every day example is WiFi but that is not the only option. There are different protocols available that specialize in different aspects of wireless communication. To achieve a connected smart home environment, our devices need to communicate to each other and to the system. The goal of this section is to discover communication options, and to evaluate them against each other to determine which is most appropriate for our system.

Selection Criteria

The available communication protocols will be evaluated based largely on the non-functional requirements of our system. For the protocols, this includes: ease of integration with devices, battery life of devices, range, reliability, interoperability, data transmission rate, number of concurrent connections, device discovery time, and cost.

4.2 ZigBee

Description

ZigBee provides an alternative to WiFi. The goal of ZigBee is to provide simple transmission for a low amount of data, within a limited range. The decrease in data transmission rates and range provides an increased battery life for components. A computing device can be turned into a ZigBee device with the appropriate shield, and some ZigBee devices, such as a light switch, can be purchased.

Technical Overview

ZigBee uses a mesh networking topology, as opposed to the star topology used by Wifi. A mesh topology means that every node in the network is connected. Each node transmits it's own information, as well as assisting in relaying information received from other nodes. Having every node connected allows for data to be transmitted between nodes simultaneously, and increases network stability be not relying on one central node. This increase in stability come at the cost of having potentially many redundant connections in the network. Zigbee devices use the mesh topology to send messages using message routing. This means that if the endpoint device is out of range of the initial device, intermediate devices will relay the message through the mesh until it reaches it's endpoint.

ZigBee operates within three possible frequency bands: 868-870 MHz, 902-928 MHz, and 2.4-2.4835 GHz. The lowest band only has one available channel, the middle band has ten available channels, and the highest band has 16 available channels. The respective data transfer rates are 20Kbps, 40Kbps, and 250Kbps. It should be noted that devices on different frequency bands cannot communicate with each other, and generally only devices using the 2.4 Ghz range are produced.

The low power consumption of ZigBee devices when compared to WiFi leads to large power consumption and battery life gains. A ZigBee device can last for up to ten years.

A ZigBee device can be added to the network in approximately thirty milliseconds, and 256 devices can be connected to one network in theory. However, in practice, the system performance tends to degrade at around thirty devices.

One of the major drawbacks of ZigBee is that for it to be effective, it must operate in the 2.4 GHz frequency band. This would not be an issue, except for the fact that this is the same frequency band as WiFi. This can cause interference between the two networks, resulting in packet loss for both networks. The lost packets have to be retransmitted until they are received by the intended endpoint, causing lag in both networks. ZigBee packets suffer more from this interference in practice, with the level of interference rising as the number of nodes and the amount of traffic rises.

A second potential drawback of ZigBee is the historical lack of official standard for communication protocols between devices. This resulted in in different companies having their own protocols for ZigBee device communication, and not all devices could be purchased and assumed to work together. This limited interoperability has been somewhat fixed with the introduction of the ZigBee Alliance, but could present some legacy issues.

References

[1] "What Technology?," in SMARTHOME® - Home Automation Superstore, 1995. [Online]. Available: http://www.smarthome.com/sc-what-technology. Accessed: Oct. 6, 2016.

[2] L. LABS, "Z-Wave vs. Zigbee," in Wireless Technology, Link Labs, 2015. [Online]. Available: http://www.link-labs.com/z-wave-vs-zigbee/. Accessed: Oct. 6, 2016.

[3] L. LABS, "The ZigBee vs WiFi battle for M2M communication," in IOT Networks, Link Labs, 2015. [Online]. Available: http://www.link-labs.com/zigbee-vs-wifi-802-11ah/. Accessed: Oct. 6, 2016.

[4] P. Sparrow, "Mesh Topology: Advantages and disadvantages," in I Answer 4 U. [Online]. Available: http://www.ianswer4u.com/2011/05/mesh-topology-advantages-and.html#axzz4MJR54MC7. Accessed: Oct. 6, 2016.

[5] J. Gracia Castro and Ó. Pérez Domínguez, "ZigBee: IEEE 802.15.4," in Tampere University of Engineering, 2007. [Online]. Available: https://www.cs.tut.fi/kurssit/TLT-6556/Slides/4-802.15ZigBee.pdf. Accessed: Oct. 6, 2016.

[6] W. Mardini, Y. Khamayseh, R. Jaradatand, and R. Hijjawi, "Interference Problem between ZigBee and WiFi," 2012. [Online]. Available: http://www.ipcsit.com/vol30/024-ICNCS2012-G3061.pdf. Accessed: Oct. 6, 2016.

4.3 Z-Wave

Description

Z-Wave is a very similar option to ZigBee, except it uses a proprietary radio design. This slightly limits the number of devices available for it, as chips are mostly produced by Sigma Designs. The advantage to this is that because the chips are made largely by one manufacturer, there is a high level of interoperation.

Technical Overview

One area where Z-Wave differs from ZigBee is the frequency range of operation. It operates at 908.42 MHz instead of at 2.4 GHz, which avoids the issue of conflicting with WiFi signals. In terms of device limits, it is very similar, being able to handle between 30 and 40 devices before issues start to occur. Z-Wave is similar to ZigBee in terms of device range and power consumption.

Refrences

[1] "What is Z-Wave," in SMARTHOME® - Home Automation Superstore, 1995. [Online]. Available: http://www.smarthome.com/sc-what-is-zwave-home-automation. Accessed: Oct. 6, 2016.

[2] "Z-Wave vs. Zigbee," in Link Labs, Link Labs, 2015. [Online]. Available: http://www.link-labs.com/z-wave-vs-zigbee/. Accessed: Oct. 6, 2016.

[3] L. Frenzel, "What’s the difference between ZigBee and Z-Wave?," in Electronic Design, 2012. [Online]. Available: http://electronicdesign.com/communications/what-s-difference-between-zigbee-and-z-wave. Accessed: Oct. 6, 2016.

4.4 INSTEON

Description

INSTEON is substantially different from the two above protocols.

Technical Overview

INSTEON uses a similar mesh topology as the above protocols, but it is not limited to radio frequencies. It utilizes a dual-mesh system to increase overall stability. The dual-mesh system is a combination of radio frequencies at 915 MHz (in the US), and powerline layer operating at 131.65 KHz. When the radio frequencies encounter interference, the powerline layer makes sure the message gets broadcasted to the appropriate destination. INSTEON also uses a different message delivery system compared to ZigBee and Z-Wave. Instead of sending a message from one device and routing it through other devices, it takes advantage of simulcasting. This is the process of having multiple devices broadcasting the same message, so the intended recipient gets the message faster and more reliably. This method is not feasible for high data rates, but INSTEON shares it's low data rates with ZigBee and Z-Wave. Simulcasting is also a result of the fact that an INSTEON network does not have a master/slave relationship. Every node has the ability to send and receive messages without having a controller. This makes it possible to have any number of devices in a network without being restricted by a maximum number of connections to a controlling device.

One thing INSTEON is lacking compared to ZigBee and Z-Wave is third part support for their devices. They manufacture almost all of their own devices, which leads to a limited amount of choice in terms of different types of devices designed for the same task.

A unique advantage of INSTEON is its ability to interface with devices following the X10 protocol. The X10 protocol was one of the original protocols designed to work using only powerlines. It is outdated now in terms of being a reasonable choice for a new system, as it was designed over 30 years ago. This means that there is no wireless communication involved, which is essential for a modern day smart home communication protocol. That said, there are many legacy automation systems in place which still use X10 devices. If this were the case, then INSTEON would be an ideal choice for a communication protocol.

References

[1] "WHITEPAPER: Compared," in INSTEON. [Online]. Available: http://cache.insteon.com/pdf/INSTEONCompared.pdf. Accessed: Oct. 6, 2016.

[2] "X10," in Build Your Smarthome, 2014. [Online]. Available: http://buildyoursmarthome.co/home-automation/protocols/x10/. Accessed: Oct. 8, 2016.

[3] "What home automation protocol should I choose?," in Intellihome, 2015. [Online]. Available: https://www.intellihome.be/en/kbase/INSTEON/What_home_automation_protocol_should_I_choose_-2.html. Accessed: Oct. 8, 2016.

4.5 WiFi

Description

WiFi is by far the most common communication protocol used in a home on a daily basis. Nearly every other communication protocol is compatible with WiFi devices, and they tend to be easy to install. There are several pros and cons associated with using WiFi for home automation, which are outlined below.

Technical Overview

WiFi operates using a star network topology. Every node is connected to a central server node. All communications go from the source node, through the central server node, and arrive at the destination node. This means that if the central server goes down, no messages can be passed, leading to potential stability issues. Because messages cannot be routed through intermediate nodes, the communication range must generally be much larger than in a mesh network. WiFi also boasts substantially higher data transmission rates than the previously discussed communication protocols. A WiFi (802.11b) connection can transfer data at a rate of 54 Mbps at a range of 100 meters. As stated earlier, WiFi typically operates at a frequency of 2.4 GHz.

The biggest advantage of using WiFi for home automation is the simplicity. There is virtually no limit to the number of devices that are WiFi compatible, as devices using nearly every other big communication protocol for home automation are also compatible with WiFi. It also requires no extra hardware to communicate with WiFi, which reduces the complexity of setting up hardware. Non-technical users are still generally familiar with WiFi, making the setup process easier.

An issue with using WiFi for home automation is that in terms of required range and data transfer specifications, it is overkill. WiFi is designed with high intensive data transfer in mind, which home automation systems do not take advantage of. A WiFi device needs to provide enough power to operates with these specifications, leading to a large increase in power consumption. If the device is battery powered, it will drain much more quickly than if it were using one of the above protocols.

Another issue stems directly from the prevalence of WiFi. Having a smart home network sharing a WiFi network with regular household uses can cause the network as a whole to slow, due to the bandwidth being shared across so many devices.

References

[1] "Introduction to Wi-Fi (802.11 or WiFi)," in CCM Benchmark, CCM, 2016. [Online]. Available: http://ccm.net/contents/802-introduction-to-wi-fi-802-11-or-wifi. Accessed: Oct. 8, 2016.

[2] E. Contributor, "Home Automation Protocols: A Round-Up," in Electronic House, Electronic House, 2016. [Online]. Available: https://www.electronichouse.com/smart-home/home-automation-protocols-what-technology-is-right-for-you/. Accessed: Oct. 8, 2016.

4.6 Bluetooth

Description

Bluetooth is the most similar to WiFi of the alternative options. It is fairly common in households, and non-technical users are more likely to be familiar with it than other protocols. There are two main classifications of bluetooth when it comes to home automation, both of which will be discussed below.

Technical Overview

Bluetooth operates in the 2.4 GHz frequency band, alongside WiFi and ZigBee. Bluetooth also shares the star network topology with WiFi, where there needs to be designated master and slave devices. This can lead to the same issues of interference that were discussed above. As the number of devices on the same radio frequency increases, the competition for bandwidth also increases, causing potential lag and interference. The range and data transfer rate for bluetooth ranges from 1 Mbps and 10 meters to 24 Mbps and 100 meters. All of these data transfer rates are acceptable, if not overkill, for a smarthome system. The range on the earlier versions of bluetooth is potentially very restricting. Bluetooth is somewhere between WiFi devices and ZigBee/Z-Wave devices in terms of power consumption.

There is another choice for bluetooth that addresses some of the issues above. Bluetooth version 4.0, also branded as Bluetooth Low Energy (BLE). This is a direct competitor with ZigBee and Z-Wave. The range for a BLE device is 50 meters, but BLE is able to take advantage of a mesh network topology. The maximum data transfer rate 1 Mbps in theory, but it is generally much lower than that in practice. BLE splits the 2.4 GHz channel into smaller sub-channels to help avoid interference with WiFi channels.

One of the goals of BLE is to make devices that do not require constant data transmission more efficient. It accomplishes this by not keeping connections active while there is no data being transferred. Once data needs to be transferred, it reestablishes the necessary connection, completes the transfer, and closes the connection again.

References

[1] Jim, "Bluetooth basics," in sparkfun. [Online]. Available: https://learn.sparkfun.com/tutorials/bluetooth-basics/common-versions. Accessed: Oct. 9, 2016.

[2] "Data rates using BLE," in Anaren atmosphere. [Online]. Available: https://atmosphere.anaren.com/wiki/Data_rates_using_BLE. Accessed: Oct. 9, 2016.

[3] "Bluetooth low energy," in CSR. [Online]. Available: https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=227336. Accessed: Oct. 9, 2016.

4.7 Summary of Evaluation

As stated above, the goal of this research was to pick an appropriate protocol for our system. One of the differing attibutes between the protocols is the data transfer rates, which are summerized here using the general rate used.

Communication Protocol	Data Transfer Rate
ZigBee	250 Kbps
Z-Wave	40 Kbps
INSTEON	13 Kbps
WiFi	54 Mbps
BLE	10 Kbps

The required data rate for most smart home devices is minimal, and a higher data rate demands more power. The data rate provided by WiFi is clearly overkill for our project, so it will not be the primary communication protocal used. That being said, being WiFi compatible is important to us because of its prevalence in homes, and because of the enormous amount of devices supported by WiFi.

Having as inclusive device support as possible is a key aspect to our project, as it allows a user to have whatever functionality they desire. This heavily influanced us in deciding not to use INSTEON as our primary protocol, as the backwards compatibility with X10 is not something we require.

ZigBee and Z-Wave are very similar, with the key difference in our eyes being the frequancy they operate at. Being able to avoid conflicts between the WiFi already assumed to be in the house and the home automation system is a clear benifit. Z-Wave operates in the less used 900 MHz frequency band, avoing any potential conflicts.

A mesh network topology makes more sense than a star topology for home automation. The stability offered by a mesh topology is substantial, and the amount of data transmission is low enough that the redundant connections are not costly. This made Z-Wave a more attractive option to us over Bluetooth Low Energy.

Overall, we decided that using Z-Wave as our primary communication protocol was the best choice for this project, while supporting WiFi devices as well.

4.8 Z-Wave Implementation Specifics

Any processing unit with USB support can be easily turned into a Z-Wave master, using a Z-Wave USB stick. Converting an arduino into a Z-Wave slave is not a simple. There are specialized arduino boards that have the Z-Wave protocol built in. Another option is to attach a radio frequency device to an arduino, and to implement the Z-Wave stack protocol manually. The easiest way to have Z-Wave slave devices is simply not to make them, but to buy them.

5. Computing Devices

5.1 Background

A computing devices is any electronic device that can be used to process data and send it over a smart home network. Computing devices include computers, micro-controllers and other embedded devices. Computing devices that are being examined all offer control of external hardware interfaces. This report examines computing devices that can be used for prototyping embedded control of sensors and actuators.

5.2 Relation to System

Computing devices will be used for many aspects of the smart home system. A computing device will be used for the central smart hub as well as for various other devices in the system. Different devices will be more suitable to some tasks than others. This report compares a number of these computing options with a focus on the areas of need within the system.

5.3 Arduino Uno

Description

The Arduino Uno is a small starter microcontroller that is intended for hobby projects and newcomers to embedded programming. The Uno is intended for rapid prototyping of small circuits and small embedded systems.

Technical Overview

The Uno is a microcontroller that uses the Arduino boot loader software for launching the system and executing code. The Uno is fully compatible with the Arduino SDK that provides simple access to all pins and controls on the board. The Uno comes with a 8-bit ATmega328P processor running at 16 MHz. The system comes with 32 KB of on board flash memory as well as 2 KB of SRAM and 1 KB of EEPROM. The system boot loader occupies 0.5 KB of the flash memory capacity.

The Uno comes with 14 digital pins, 6 of which can be programmed to use pulse with modulation. Included in the microcontroller is 6 analog to digital input pins. The board has a operating voltage of 5 V and will accept between 7-12 V for input pins. The maximum voltage range of the input pins are 6-20V.

The Uno is a medium sized microcontroller that is 68.6 mm long and 53.4 mm wide. It has a total weight of approximately 25 g. The Uno also includes an integrated USB-Mini port used for serial communication and as a power source. The board includes a 5 V DC power input for running the board continually.

Evaluation

The Arduino Uno offers fast and simple prototyping options for quickly building embedded circuits. This board is very useful for rapidly testing but would not be practical in a mass production system. For the smart learning system, this board would be optimal for experimentation and quick deployment.

References

[1] "ArduinoBoardUno," in Arduino, 2016. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardUno. Accessed: Oct. 6, 2016.

5.4 Arduino 101

Description

The Arduino 101 is a more sophisticated board that uses the Intel Curie processor for computing. This board leverages the Real-Time Operating System (RTOS) developed by Intel for running the board.

Technical Overview

The Arduino 101 is the most feature rich board that Arduino is offers. It comes with a full Real-Time Operating System (RTOS) that is powered by the Intel Curie. The processor is a 32-bit, 8 MHz or 16 MHz backed by 196 KB of flash memory. The RTOS is a light weight OS that only occupies 2 KB of memory to provide managed, concurrent applications.

The 101 comes with a number of additional features above other microcontrollers in the Arduino family. Beyond the standard USB-Mini serial connection, the 101 offers a 6 axis gyroscope accelerometer and integrated Bluetooth. The 101 also provides a 5 V DC power input for deployment use.

The 101 offers many of the same prototyping capabilities as the Uno with 14 digital pins, 4 of which provide pulse with modulation. The 101 also exposes 6 analog to digital pins.

Evaluation

The 101 is a much more feature rich board than the Uno, however the added features are not applicable to this project. The 101 also has a much higher price point per unit than the Uno with no real added value. This makes the Uno a more appealing candidate for rapid prototyping and simple circuit design.

References

[1] "ArduinoBoard101," in Arduino, 2016. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoard101. Accessed: Oct. 6, 2016.

5.5 Arduino Pro

Description

The Arduino Pro is a slim, no frills version of the Ardiuno Uno. This board requires more technical knowledge then the Uno or the 101. All pins exposed on the Pro require soldering to make a connection. The Pro is intended for replication of a complex circuit design with requirements for a wide range of input and output ports.

Technical Overview

The Pro is an embedded system that leverages the Arduino boot loader and a 32-bit ATmega328 processor. The Pro has two operating speeds; 8 MHz or 16 MHz. This device has an operating voltage of 5 V but has a lower power alternative that runs at 3.3 V. The Pro has the same limited memory as the Uno with only 32 KB of flash memory, 0.5 KB of which is occupied by the boot loader.

The Pro offers the same pin configuration as the 101 with 14 digital pins, 4 of which can be pulse with modulation, and 6 analog pins. The Pro also provides some more advanced I/O options with a universal asynchronous receiver / transmitter (UART), serial peripheral interface (SPI) bus, and an inter-integrated circuit (I2C) connection.

Evaluation

The Pro has many advanced features but offers little in the way of rapid prototyping. The Pro is intended for a more advanced audience than what is required for this project and is likely not a good candidate for practical applications. However, if this system was to be replicated or redistributed, the Pro would be useful for building a final product for an unmodifiable system.

References

[1] "ArduinoBoardPro," in Arduino, 2016. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardPro. Accessed: Oct. 6, 2016.

5.6 Arduino Micro

Description

The Arduino Micro is the smallest microcontroller of the Arduino family. The Micro also provides the most external ports of any Arduino making it suitable for large circuits.

Technical Overview

The Micro provides high performance embedded computing with a 8-bit, 16 MHz ATmega32U4 processor. The Micro also comes standard with the Arduino bootloader and a standard 32 KB of flash memory. The Micro has a operating voltage of 5 V and comes with a standard DC power input.

The Micro has 20 exposed digital pins, 7 of which can be pulse with modulation. These extra pins make the Micro very suitable for large complex circuits that require many inputs or outputs. The Micro also has 12 analog to digital pins. The Micro does lack in special features. It only offers serial communication over a standard USB-Micro port. There are other special features with this device.

Appropriately, the Micro is the smallest and lightest microcontroller in the Arduino family. It is only 48 mm long and 18 mm wide. The Micro also only has a weight of roughly 13 g.

Evaluation

This Arduino would be perfectly suited to a mass production environment where size and weight were valuable resources. If the learning home automation system was to be commercially produced, this could be a very valuable microcontroller. This microcontroller could be used for this system, but would take more effort for prototyping and would likely not be a suitable fit.

References

[1] "ArduinoBoardMicro," in Arduino, 2016. [Online]. Available: https://www.arduino.cc/en/Main/ArduinoBoardMicro. Accessed: Oct. 6, 2016.

5.7 Comparison of Arduinos

Operation Criteria

Criteria	Arduino Uno	Arduino 101	Arduino Pro	Arduino Micro
Operating System	None	RTOS	None	None
Processor Size	8-bit	32-bit	32-bit	8-bit
Processor Family	ATmega	Intel	ATmega	ATmega
Operating Voltage	5V	3.3V - 5V	3.3V - 5V	5V
Input Voltage	7-12V	7-12V	7-12V	7-12V
Clock Speed	16 MHz	8 MHz - 16 MHz	8 MHz - 16 MHz	16 MHz
Digital Pins	14	14	14	20
Pulse with Modulation Pins	6	4	4	7
Analog Input Pins	6	6	6	12
DC Current per Pin	20 mA	20 mA	40 mA	20 mA
Flash Memory	32 KB	196 KB	32 KB	32 KB
System Size	0.5 KB	2 KB	2 KB	4 KB

Features

Feature	Arduino Uno	Arduino 101	Arduino Pro	Arduino Micro
USB	USB-Mini	USB-Mini	USB-Micro	USB-Micro
Accelerometer	No	6-Axis Gyro	No	No
Bluetooth	No	Yes	No	No
UART	No	No	Yes	No
SPI Bus	No	No	Yes	No
I2C	No	No	Yes	No

Physical Characteristics

Dimension	Arduino Uno	Arduino 101	Arduino Pro	Arduino Micro
Length	68.6 mm	68.6 mm	52.1 mm	48 mm
Width	53.4 mm	53.4 mm	53.3 mm	18 mm
Weight	25 g	34 g	N/A	13 g

5.8 Raspberry Pi Zero

Description

The Raspberry Pi Zero is a minimal computer board that offers a full computing platform in a compact form for embedded computing. The Zero is the smallest board in the Raspberry Pi family and is ideal for medium to heavy computation with minimal footprint. The Raspberry Pi Zero is one of the only Raspberry Pi boards that competes for a real embedded computing experience.

Technical Overview

The Raspberry Pi Zero has the smallest surface area of any of the Pi's, measuring in at only 65mm long by 30mm wide. To make the board as small as possible, many of the standard Raspberry Pi features were removed. This means that the Zero has no on-board WiFi, Bluetooth or even Ethernet. Despite these losses, the board is still equip with a 32-bit 1GHz Broadcom BCM283 processor backed by 512MB of flash storage.

The Zero provides a lot of room for flexibility with 40 available GPIO pins. The combination of the Zero's computing power and general IO makes it ideal for small spaces that need a lot of power.

Evaluation

The Zero could be useful for programming devices in the system; however, while the Zero does offer a smaller physical footprint and more GPIO than the Arduino Uno, it does require more power to maintain operation. This extra power consumption does come with more performance which may be useful but likely unnecessary for the smart learning system.

Reference

[1] "Raspberry pi Zero," Raspberry Pi. [Online]. Available: https://www.raspberrypi.org/products/pi-zero/. Accessed: Oct. 10, 2016.

[2] "Raspberry Pi Zero,". [Online]. Available: https://shop.pimoroni.com/products/raspberry-pi-zero. Accessed: Oct. 10, 2016.

5.9 Raspberry Pi 1 Model A+

Description

The Raspberry Pi 1 Model A+ is the original Raspberry Pi with some performance improvements. This device is a computing board that provides desktop equivalent computing power in only a few square inches of space. The Pi 1 is the first candidate being considered for the role of the smart home learning hub.

Technical Overview

The Pi 1 is the base Raspberry Pi that is powered by a 32-bit 700MHz Arm processor and 256MB of DDR2 RAM. The Pi 1 comes with a standard HDMI output for visual output. The board also comes with a standard Ethernet port, and a single USB port for serial communication. The Pi 1 does also offer 40 GPIO pins for more embedded purposes.

Evaluation

The Raspberry Pi 1 is a good candidate for the central hub as it uses low power and provides adequate computing performance.

References

[1] "Raspberry pi 1 model A+," Raspberry Pi. [Online]. Available: https://www.raspberrypi.org/products/model-a-plus/. Accessed: Oct. 10, 2016.

[2] J. Adams, "Raspberry Pi Model B+," Mar. 07, 2014. [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/mechanical/Raspberry-Pi-B-Plus-V1.2-Mechanical-Drawing.pdf. Accessed: Oct. 10, 2016.

5.10 Raspberry Pi 2 Model B

Description

The Raspberry Pi 2 Model B is the second generation of Raspberry Pi designs. The Pi 2 uses the same design as the Pi 1 with all the same features and more performance.

Technical Overview

The Pi 2 is very similar to the Pi 1 but provides a slight faster 32-bit 900MHz Arm processor. The Pi 2 has significantly more RAM than the Pi 1 with 1GB of DDR2. The Pi 2 comes with a standard HDMI video output for monitoring it from a screen. It also is equip with an Ethernet port and 4 USB ports. The Pi 2 also provides the same 40 GPIO pin configuration as the Pi 1

Evaluation

The Pi 2 out performs the Pi 1 in all areas and is likely a better candidate for the smart hub. It uses the same amount of power but provides far more computing performance.

References

[1] "Raspberry pi 2 model B," Raspberry Pi. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-2-model-b/. Accessed: Oct. 10, 2016.

5.11 Raspberry Pi 3 Model B

Description

The Raspberry Pi 3 Model B is the most advanced Raspberry Pi available. The Pi 3 is a computer board that uses a very similar design to the other Pi Models. The Pi 3 offers more computing performance than all of its predecessors and is a very suitable candidate for the smart hub.

Technical Overview

The Raspberry Pi 3 offers massive embedded performance with a 1.2GHz 64-bit Quad-core Arm processor. Similar to the Pi 2, the Pi 3 also comes with 1GB of RAM. As with all other Pi models, the Pi 3 provides a 40 GPIO pin configuration for external devices.

The Pi 3 also comes with a number of standard options that set it apart from all other Pi models. It comes with integrated WiFi, Ethernet and Bluetooth communication interfaces. It has 4 standard USB ports and an HDMI output for visual feedback.

Evaluation

The Raspberry Pi 3 is the most advanced Raspberry Pi board available. Its extra computing power would be a good asset for heavy computation making this an ideal candidate for the central smart hub. The Pi 3 comes with many standard features including WiFi and Bluetooth communication which will make external interfacing simple with minimal investment.

References

[1] "Raspberry pi 3 model B," Raspberry Pi. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/. Accessed: Oct. 10, 2016.

[2] "Power supply - raspberry pi documentation,". [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md. Accessed: Oct. 10, 2016.

[3] "GPIO - raspberry pi documentation,". [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/gpio/README.md. Accessed: Oct. 10, 2016.

[4] "Raspbian - raspberry pi documentation,". [Online]. Available: https://www.raspberrypi.org/documentation/raspbian/. Accessed: Oct. 10, 2016.

[5] J. Adams, "Raspberry Pi 3 Model B," Jun. 10, 2015. [Online]. Available: https://www.raspberrypi.org/documentation/hardware/raspberrypi/mechanical/RPI-3B-V1_2.pdf. Accessed: Oct. 10, 2016.

5.12 Comparison of Raspberry Pi

Operation Criteria

Criteria	Raspberry Pi Zero	Raspberry Pi 1	Raspberry Pi 2	Raspberry Pi 3
Operating System	Raspbian	Raspbian	Raspbian	Raspbian
Processor Size	32-bit	32-bit	32-bit	64-bit
Processor Family	Broadcom	Arm	Arm	Arm
Operating Voltage	5V	5V	5V	5V
Input Voltage	3.3V	3.3V	3.3V	3.3V
Clock Speed	1 GHz	700 MHz	900 MHz	1.2 GHz
Digital Pins	40	40	40	40
Pulse with Modulation Pins	N/A	N/A	N/A	N/A
Analog Input Pins	N/A	N/A	N/A	N/A
DC Current per Pin	50 mA	50 mA	50 mA	50 mA
Flash Memory	512 MB	256 MB	1 GB	1 GB
System Size	1.6 GB	1.6 GB	1.6 GB	1.6 GB

Features

Feature	Raspberry Pi Zero	Raspberry Pi 1	Raspberry Pi 2	Raspberry Pi 3
USB	USB-Micro	1	4	4
HDMI	Mini-HDMI	Yes	Yes	Yes
Bluetooth	No	No	No	Yes
WiFI	No	No	No	Yes
Audio	No	3.5 mm Jack	3.5 mm Jack	3.5 mm Jack
Ethernet	No	Yes	Yes	Yes
Camera Interface	No	Yes	Yes	Yes
Display Interface	No	Yes	Yes	Yes
Micro SD	Yes	Yes	Yes	Yes

Physical Characteristics

Dimension	Raspberry Pi Zero	Raspberry Pi 1	Raspberry Pi 2	Raspberry Pi 3
Length	65 mm	85 mm	85 mm	85 mm
Width	30 mm	56 mm	56 mm	56 mm

5.13 BeagleBone

Description

The BeagleBone is another computing board that is on the edge of embedded computing. The BeagleBone is a small, feature rich, Linux system for concurrent, real-time embedded programming. This board could be used for the learning hub or even a small device that requires more computation power than a microcontroller can offer.

Technical Overview

The standard BeagleBone is very comparable to the Raspberry Pi 2 in features and performance. It is driven by a 32-bit 700MHz Arm processor with 256MB of flash memory. This computing board offers 2 USB ports and a standard Ethernet port for external communication.

The BeagleBone stands out from the Raspberry Pi with its general pin configurations, as it offers 60 GPIO pins. This makes the BeagleBone more suitable for embedded computing than the Raspberry Pi models. The BeagleBone is a Linux compatible board that comes with the Angstrom distribution.

Evaluation

The BeagleBone has a number of useful features for embedded computing but does not provide as much computation performance as the Raspberry Pi 3. The BeagleBone may be suitable for other devices in the system that require more computational power than what a traditional microcontroller can provide.

References

[1] "Bone-original,". [Online]. Available: http://beagleboard.org/bone-original. Accessed: Oct. 10, 2016.

[1] "BeagleBone Schematic," Jun. 28, 2012. [Online]. Available: https://github.com/CircuitCo/BeagleBone-RevA6/blob/master/BEAGLEBONE_REV_A6A.pdf?raw=true. Accessed: Oct. 10, 2016.

5.14 BeagleBone Black

Description

The BeagleBone Black is a more advanced version of the standard BeagleBone. It provides more features and is a higher performance system than the standard board. The BeagleBone Black is most comparable to the Raspberry Pi 3 board in terms of performance and features. The Black model still offers all of the major BeagleBone embedded features making it a good candidate for both the learning hub and other computing devices

Technical Overview

The BeagleBone Black is driven by a 32-bit 1GHz Arm processor and is backed by 512MB of RAM. It also comes with more standard communication options than is predecessor including WiFi and Bluetooth. The Black model comes standard with 2 USB connections for serial communication and an HDMI interface for video feedback.

The Black Model uses the same pin configuration as the basic BeagleBone with a few extra specialty pins. It provides 68 GPIO pins with an additional UART connection.

Evaluation

The BeagleBone Black is does have many powerful features but cannot complete with the Raspberry Pi 3 for the learning hub roll. The Black model may be useful for embedded devices that require additional computation but since it requires more power than the standard BeagleBone with minimal gain it may not be feasible.

References

[1] "BeagleBone Black,". [Online]. Available: http://beagleboard.org/black. Accessed: Oct. 10, 2016.

[2] "Beagleboard: BeagleBoneBlack,". [Online]. Available: http://elinux.org/Beagleboard:BeagleBoneBlack. Accessed: Oct. 10, 2016.

5.15 BeagleBone Green

Description

The BeagleBone Green is the best embedded option of the BeagleBone family. It is a computing board that runs a full operating system but has the most available circuit controlling pins. It is an ideal candidate for both the learning hub and smart devices.

Technical Overview

The BeagleBone Green model is powered by the same 32-bit 1GHz processor as the Black model. It also has the same 512MB of RAM available. Almost all of the core features of the Green model are the same as the Black model. The Green model stands out with its 4 additional pulse with modulation pins and 3 additional analog pins. In addition to a UART connection, the Green model provides an I2C connection.

Evaluation

The Green model shares many of the same features as the Black and is therefore not an adequate alternative for the Raspberry Pi 3 and the learning hub. However, the Green does require less energy than the Black model and is possibly a better candidate for embedded devices.

References

[1] "BeagleBone Green,". [Online]. Available: http://beagleboard.org/green. Accessed: Oct. 10, 2016.

5.16 Comparison of BeagleBone

Operation Criteria

Criteria	BeagleBone	BeagleBone Black	BeagleBone Green
Operating System	Angstrom	Debian	Debian
Processor Size	32-bit	32-bit	32-bit
Processor Family	ARM	ARM	ARM
Operating Voltage	5V	5V	5V
Input Voltage	1.8V	1.8V	1.8V
Clock Speed	700MHz	1GHz	1GHz
Digital Pins	60	68	65
Pulse with Modulation Pins	4	4	8
Analog Input Pins	4	4	7
Flash Memory	256MB	512MB	512 MB
System Size	1.8 GB	2.2GB	2.2 GB

Features

Feature	BeagleBone	BeagleBone Black	BeagleBone Green
USB	2	2	2
HDMI	No	Yes	No
Bluetooth	No	Yes	Yes
WiFI	No	Yes	Yes
Ethernet	Yes	Yes	Yes
UART	No	Yes	Yes
I2C	No	No	Yes
Micro SD	Yes	Yes	Yes

Physical Characteristics

Dimension	BeagleBone	BeagleBone Black	BeagleBone Green
Length	86 mm	86 mm	86 mm
Width	54 mm	54 mm	54 mm

5.17 Summary of Evaluation

Embedded Devices

There will be a number of applications for embedded devices in the smart home system. There may be various computing devices that are suitable for different tasks depending on the requirements. However, for the general needs of the embedded computing in this system, the Arduino Uno will likely be the most suitable candidate.

The Uno provides a rapid prototyping environment with support for a wide array of custom devices. It has sufficient computing power with minimal power consumption for the requirements of the general devices that have been identified in the system scenarios.

As an added benefit, the Arduino community offers many high quality tutorial, examples and documentation of system usage. The extensive support offered by the community is a major advantage over the other systems and will be a major asset for developing on this device.

Learning Hub

The learning hub will require a significant amount of computational performance to make decisions about home environment. For this component, full computer boards were considered as they provide more performance than the available microcontrollers. After a close examination of a number of computing devices it was determined that the Raspberry Pi 3 Model B is the most suitable option for this roll in the system.

Since this device is heavily reliant on performance, the decision was reduced to 2 candidates, the Raspberry Pi 3 and the BeagleBone Black. They both offer considerable performance but the Pi 3 offers more processing cores with a higher clock speed, larger registers and more RAM. This makes the decision simple, the Pi 3 is the better candidate.

6. Custom-Built Devices

6.1 Background

The purpose of this section is to present our investigation into the process of building various types of smart devices which might be of use for home automation. The purpose of this investigation is to provide information that the team can use to decide which devices will be included in the system to showcase its machine learning capabilities. In particular, this section will examine the feasibility of assembling such devices from basic electronic components, rather than purchasing a commercial device.

In this section, a **smart device** refers to a sensor or actuator which can be controlled over a wireless network. The process of building a smart device is expected to consist of interfacing a hardware module with a microcontroller which is capable of controlling the device and communicating over a network.

Evaluation of custom devices will focus on the following characteristics:

Expertise required

How difficult is it for a team of software engineering students to build the device?. Some complex devices will be impractical for the team to build due to the lack of hardware knowledge. Factors which will be considered in the evaluation of the devices include the availability of tutorials and development kits. Safety will also be a critical factor (Does the team have the facilities and expertise necessary to build the device safely).

Level of Effort

In order to allow the team to focus on the machine learning aspects of the system, devices which can be used to demonstrate the system must be available quickly. In order to measure this criteria, the amount of effort required to build an LED controlled through a voltage relay will be used as a baseline for estimation. The following scale will be used:

Low- Comparable to baseline (less than a day of work) High - High effort (A couple days) Extreme - Extreme effort (many days or weeks)

Quality Comparisons

Commercially available devices may offer features which could be useful to demonstrate the system, but are difficult to include in a custom-built device. It will be useful to determine whether or not the devices described by most custom-build tutorials are fully featured. In some cases, the features of a typical custom-built device will be compared against several commercial options.

Devices of Interest

This section contains a list of device types which were investigated, and a short explanation of why the devices could showcase the machine learning capabilities of the system. In general, we believe that in order to showcase the machine learning capabilities of the system, it will be important to include a wide range of sensors, as well as actuators whose desired states may be influenced by several unrelated sensors. By providing such a range of devices, we hope to demonstrate that the machine learning algorithm is capable of identifying more complex interactions between devices than a simple "If this then that" relationship.

In addition to providing a varied range of sensors, we would like to provide several sensors and actuators with non-binary inputs and outputs. The interactions between such devices may be more complex than for binary devices due to the increased number of possible states.

This section investigates some smart devices which are potentially within the team's ability to build.

- Motion Sensors

Many actions could be triggered by a person entering or leaving a house or a room. A motion sensor could potentially interact with every other device on this list. Due to the wide range of potential interactions, we expect that a motion sensor will be a good test of the machine learning component's ability to determine the relationships inputs and outputs. The motion sensor could also reveal flaws in the machine laerning algorithm. For example, the machine learning algorithm may determine that when a motion sensor is triggered outside a homeowner's bedroom, the coffee machine should be turned on. However, this behaviour would only be desired during certain hours of the day (the coffee machine should not turn on when the homeowner goes to bed, for example). It is likely that there are many such scenarios that we haven't thought of which would present challenging cases for the machine learning component to handle.

- Thermostat

The setting of a thermostat may be affected by several different factors, such as temperature, time of day, and light levels. Since thermostats are also a non-binary output device, their behaviour when controlled by the machine learning algorithms may be more varied than other devices.

- Light Sensor

A light sensor provides a non-binary input to the machine learning component. Similar to a motion sensor, the level of light in a room may be related to the behaviour of many other devices.

- Dimmer switch

A variable-voltage switch is an example of a non-binary output device. A variable-voltage switch could be used to control the brightness of lights, or the speed of a fan.

- Coffee Makers

The coffee maker could be turned on in response to several different input devices (light sensor, motion sensor)

6.2 Motion Sensor

Description

Technical Overview

This section considers devices that can be used to alert the system when a person enters or leaves an area of the home.

Sensor Technologies

This section provides a comparison of several different sensor technologies that are commonly used in motion detection.

Passive Infrared (PIR)

A passive infrared sensor detects motion using the infrared radiation (heat) from a warm body. A sensor contains two slots, each of which contains a material which is sensitive to infrared radiation. As a warm body passes in front of the sensor, a different amount of radiation is received at each slot. This difference in radiation causes a signal in the output of the device.

Use of the PIR modules encountered during this research does not require expert knowledge of circuit design; the complexity of the circuits involved in connecting the modules to a microcontroller is comparable to that of a simple LED circuit. A power source and a resistor is sufficient to get the module up and running.

Most PIR modules have 3 pins to connect to a circuit. One pin connects to ground, another to power, andthe third pin is used for output. The output consists of a single digital signal indicating when motion is detected.

Tutorials describing how to connect PIR modules to common microcontrollers, such as a Raspberry Pi or Arduino are widely available online.

The complexity of the circuit required to interface with a PIR module is comparable to the light switch circuit, consisting of a single push button used to toggle an LED. Therefore, development time required for the complete device is expected to be under 10 hours.

PIR sensor modules are reliable because they do not typically contain any moving components, other than potentiometer used for sensitivity adjustment.

PIR sensors are easy to use, the main difficulties that a user may encounter in setting up a PIR sensor are ensuring that the detectorès field of view covers the correct area, as well as potentially adjusting the sensitivity of the detector.

PIR sensors are available with a range which is suitable for a typical room (approximately 7m range).

The following tutorials explain how to interface with PIR modules:

[https://cdn-learn.adafruit.com/downloads/pdf/pir-passive-infrared-proximity-motion-sensor.pdf][https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor/]

Doppler-Effect based sensors

Several types of motion detectors use the Doppler effect to detect motion. The detector transmits A signal of a known frequency. This signal is reflected by any objects in its path and detected when it returns to the sensor. The sensor tracks frequency of the reflected wave; when a wave is reflected off of a moving object the frequency of the reflected wave differs from that of the incident wave due to the Doppler effect. This change in frequency is detected by the sensor and registered as motion.

Infrared Break-Beam

An infrared break-beam sensor consists of two physical modules separated by some distance. One side of the detector transmits a beam of infrared light which is detected by the other side.

Expertise Required - Little expertise required, circuits are simple - Tutorials widely available online

Component Availability - Components widely available

Reliability - Sensors generally have a lower range than PIR sensors, availability of detectors with a longer range may be lower - Allows for finer control over the area of detection than a PIR sensor

Useability - More difficult setup than a PIR sensor, requires the user to precisely aim the transmitted beam.

Feature Comparison

Feature	Custom	D-Link Wifi Motion Sensor	Samsung Smartthings Motion Sensor	Belkin WeMo Switch + Motion
Range	7m	8m	15m - 40m	3m
Interfaces	_	Wifi	ZigBee	Wifi
Type	PIR	PIR	PIR	PIR

Note: The custom device can potentially support multiple different communication interfaces

Custom device: https://www.adafruit.com/product/189 D-Link: http://ca.dlink.com/products/connected-home/wi-fi-motion-sensor/ Samsung: https://shop.smartthings.com/#!/products/samsung-smartthings-motion-sensor WeMo: http://www.belkin.com/au/p/P-F5Z0340-APL/

Evaluation

Criterion	Score
Expertise Required	Low
Level of Effort	Low
Quality Comparison	Equal

6.3 Variable-Voltage Switch

The characteristics of variable-voltage switches vary significantly depending on the type of device that the switch is expected to control (DC vs AC).

Controlling DC power

If the type of device being controlled requires DC power input, the amount of power supplied to the device can be controlled using pulse-width modulation. A digital output is used to create a square wave. By varying the frequency of the square wave, it is possible to simulate the application of a steady voltage between the pin's high and low voltages.

[https://www.arduino.cc/en/Tutorial/PWM]

Limitations to simple PWM

- Power available to the device is limited by the power supplied by the controlling device
- Limited to DC devices

Expertise Required

Minimal expertise is required to use PWM to control a device. The circuits required are as simple as the circuits required to power a simple LED.

Reliability

Simple PWM circuits are limited to controlling devices with low power requirements. While the technique is reliable for devices which meet the power requirements, such as LED lamps, the power requirements of most home devices means that the applicability of this technique will be severely limited. Most devices are intended to be connected to a wall socket, meaning that they are expecting AC current, with a significantly higher amount of power than is available from a microcontroller such as an Arduino.

Controlling AC power

Expertise Required

- Safety concerns: Controlling devices which expect to receive the voltages available from a wall socket means dealing with high voltages. Without any team members trained in using high voltages, building a device to control high voltages is a very serious safety risk.

- Lack of trusted tutorials: For most of the circuits considered during this research, it has been possible to find tutorials online from trusted sources, such as the official Arduino website.

Level of Effort

- When it was possible to find a tutorial online for building a circuit to control AC voltages, the circuit was found to be much more complex than the simple light switch circuit. Due to the amount of time required to fully understand these circuits, it is estimated that effort required is high.

[http://playground.arduino.cc/Main/ACPhaseControl]

Feature	**Custom**	**GE Z-Wave Plugin-in Smart Dimmer**	**Philips Hue Dimmer Switch**	**Lutron Caseta Wireless Plugin-In Lamp Dimmer**
Compatible Devices	AC devices	AC devices	Philips Hue Products	AC Devices
Interfaces	_	Z-Wave	ZigBee	Lutron Integration Protocol

Philips Hue: http://www.developers.meethue.com/documentation/how-hue-works Lutron: https://www.lutron.com/technicaldocumentlibrary/040249.pdf GE Z-Wave: http://www.zwaveproducts.com/shop/brands/ge/z-wave-plug-in-smart-dimmer-1

Evaluation

Criterion	**Score**
Expertise Required	High
Level of Effort	High
Quality Comparison	Equal

Note: There are safety concerns for custom-building this device

6.4 Light Sensor

Description

This section investigates devices which can detect the amount of ambient light in an area of the house.

Photoresistors

Light sensors generally make use of a component called a photoresistor. The resistance of a photoresistor differs depending on the amount of light incident to the component. Phtoresistors are also called photocells.

Reliability Characteristics

- Photoresistors are unsuitable for precise lighting measurements because its resistance may vary due to temperature as well as light

- Photoresistor may exhibit latency (delay between a change in light and a change in resistance)

Photodiode

Photodiodes are components which can be used similarly to photoresistors to detect light. A photodiode allows electrons to pass when there is light shining on it. The current allowed to pass is proportional to the amount of light incident on the device.

Unlike photoresistors, a photodiode is not sensitive to temperature changes and may allow for more accurate light detection. The circuit required to interface with a photodiode is of similar complexity to the circuit required to interface with a photoresistor.

Expertise Required

Minimal expertise is required to build the devices described by most guides and tutorials. The circuit is not significantly more complex than the light switch.

Level of effort

Due to the simplicity of the photoresistor circuit and the wide availability of the required components and tutorials, a custom-built light sensor is required requires a low amount of effort.

Feature Comparison

Many commercially available light sensors are included in combination devices which include several different types of sensors. Commercial solutions provide very little data about the range of light that their products can detect. For a custom built solution, the range of detectable light depends on combination of the particular photocell chosen. Assuming that both a custom device and most commercial devices are able to differentiate between daytime and evening levels of light, additional features available from commercial devices are generally related to other types of sensors

Everspring Z-Wave illumination sensor: http://www.smarthome.com/everspring-st815-z-wave-wireless-illumination-sensor-with-lcd-screen.html HomeSeer Z-Wave: http://www.smarthome.com/homeseer-hsm200-z-wave-multi-sensor.html

Evaluation

Criterion	Score
Expertise Required	Low
Level of Effort	Low
Quality Comparison	Equal

6.5 Thermostat

This section considers controlling a typical wall-mounted thermostat using a microcontroller such as an Arduino.

Expertise Required

Safety: Controlling a wall-mounted thermostat means working with mains voltage levels. Without knowledge in using high voltages, building this device is a significant safety risk

Effort Level

Due to the lack of trusted tutorials online, as well as the wealth of features that users are used to having in a thermostat, the time investment required to built a fully-featured thermostat is extreme.

Evaluation

Due to the extreme level of effort required and the safety concerns with building a thermostat, a custom-built thermostat is not feasible for this project.

6.6 Alarm Clock

There are a large variety of features available for alarm clocks, this section will focus on a simple alarm clock that beeps at a programmed time of day. The clock should have manual (button) input in order to allow the user to set alarm times so that the machine learning component can be trained. More sophisticated clocks are expected to be much too complex for the team to build.

After review of many online tutorials for building alarm clocks, a common structure for a clock circuit has been identified.

The basic alarm clock circuit generally requires the following components:

- LCD display
- Microcontroller
- Piezo Buzzer or Speaker
- Varying numbers of push buttons for manual configuration

A typical circuit has the following form:

http://fritzing.org/media/fritzing-repo/projects/a/arduino-lcd-alarm-clock/images/Untitled.png

Level of Effort

Due to the need to interface with an LCD display, and because of the large number of components, a custom build of an alarm clock is expected to be more complex than the light switch, so the level of effort is high.

Feature Comparison

Homeowners may be used to alarm clocks with nice interfaces, often in the form of an app on their phone. The custom alarm clocks investigated have significantly inferior interfaces, making use of a 4-button interface to set the time. The clocks considered here also lack features such as adjustable alarm sounds, configuring multiple alarms, and setting weekly alarm schedules. Making a custom built alarm clock as usable as commercially available clocks would require significant time investment.

[http://www.engineersgarage.com/microcontroller/8051projects/LCD-digital-alarm-clock-AT89C51-circuit] [http://www.instructables.com/id/Arduino-alarm-clock/] [http://circuitdigest.com/microcontroller-projects/arduino-alarm-clock]

6.7 Coffee Makers

Using a relay switch

A simple way to connect a coffee machine to a microcontroller is by using a voltage relay. A voltage relay is an electrically controlled switch for high power devices. A simple coffee machine that only needs to be plugged in in order to start brewing could be controlled using a simple voltage relay circuit.

Level of Effort

The same circuit which is used in the LED switch could be used to control the coffee machine, since both the coffee machine and LED can be connected to the same voltage relay, so the level of effort required is low.

Required Expertise

- Safety risks: Coffee machine are normally connected directly into a wall socket. Controlling the coffee machine using a voltage relay would therefore involve high voltage levels which are unsafe to work with without proper training

[http://www.instructables.com/id/Tweet-a-Pot-Twitter-Enabled-Coffee-Pot/]

In order to interface a microcontroller with more sophisticated coffee machines, some level of reverse engineering of the coffee machine's control circuits would be necessaary. This would greatly increase the amount of time necessary to create the device, due to the lack of electronics knowledge on the team.

Evaluation

Custom builds of both simple and fully-featured coffee makers may be infeasible due safety concerns when controlling high voltage devices.

6.8 Summary of Evaluation

Device	Level of Effort	Safety Concerns	Fully-Featured
Motion Sensor	Low	None	Yes
Light Sensor	Low	None	Yes
Dimmer Switch	Low	High Voltage	Yes
Coffee Maker	High	High Voltage	No
Thermostat	High	High Voltage	No
Alarm Clock	Low	None	No

6.9 Conclusions

It is feasible for the team to build motion sensor, light sensors, and an alarm clock if no acceptable commercial solution is available. We should avoid building dimmer switches, coffe makers, and thermostats due to the potential high-voltage work involved and the lack of electronics knowledge available.